International Journal of Data Science and Analysis
2021; 7(4): 98-108
http://www.sciencepublishinggroup.com/j/ijdsa
doi: 10.11648/j.ijdsa.20210704.11
ISSN: 2575-1883 (Print); ISSN: 2575-1891 (Online)

A Logical Clock Based Discovery of Patterns

Friedemann Schwenkreis

Business Information Systems, Baden-Wuerttemberg Cooperative State University, Stuttgart, Germany

Email address:
friedemann.schwenkreis@dhbw-stuttgart.de

To cite this article:
Friedemann Schwenkreis. A Logical Clock Based Discovery of Patterns. *International Journal of Data Science and Analysis*. Vol. 7, No. 4, 2021, pp. 98-108. doi: 10.11648/j.ijdsa.20210704.11

Received: July 29, 2021; **Accepted**: August 7, 2021; **Published**: August 11, 2021

Abstract: This paper focusses on aspects of applied data mining in the context of team handball. It presents an approach to transform the collected data of team handball matches into formats that allow the use of classification and methods to search for association rules. To be able to search for patterns at arbitrary times of matches a concept of a logical clock is introduced, which becomes an essential part of the data preparation. The applied data mining methods are described in detail using RapidMiner processes and their settings. However, the approach is independent of the used data mining tool. Based on the results of the data mining processes, the applicability of data mining techniques in the given context will be discussed. Particularly it will be shown that rule-based results have significant advantages compared to approaches using support vector machines in the given context. The results are also compared based on the logical clock which will show how patterns evolve over time in case of team handball. We will show that the overall prediction accuracy of a model is not the primary concern in the chosen application area. It is rather to discover rules which clearly help to identify the need for action. The concept of time is crucial in this context because rules are less helpful if they are detected when the game is over, and we are at the end of a slippery slope which could have been prevented long before.

Keywords: Data Science, Applied Data Mining, Classification, Co-Occurrence Grouping, Team Handball

1. Introduction

Innovative team handball coaches are looking for support by modern analysis methods to be able to make information-based decisions during team handball matches. Patterns extracted from data of past games would be a perfect basis for that. However, the patterns need to be easily recognizable and applicable for coaches in future matches and they need to be convertible into actions as early in a match as possible.

There is a significant number of publications in the area of applied data mining in the context of sports like soccer, basketball, baseball, and ice hockey. It is impossible to cover all of that work, but Schumaker et al. contains a good introduction into the field [1]. Brefeld et al. contains latest developments in the area [2]. The major difference of the work described in this paper is the sport itself, which differs significantly from the above-mentioned sports.

For instance, there are the so-called low-scoring games, like soccer and ice hockey, which are characterized by the fact, that the number of attacks is significantly greater than the number of scored goals (or points). The insights of sports in that area have only limited applicability in case of high-scoring games like team handball [3]. Another group of sports consists of games like American football or baseball which do frequently interrupt the game. Team handball belongs to the group of games with a “game continuum” that is only interrupted in case of special conditions. Thus, the insights of non-continuous games cannot be applied to team handball. Particularly not because team handball coaches have only a few interrupts to influence their team.

Basketball is the sport which is closest to team handball from the point of view of timing and scoring. A difference compared to team handball is the absence of a goal as well as a goalkeeper and the fact that there is no penalty area which must not be entered by field players. Furthermore, team handball has a completely different notion of physical fouls. Nevertheless, some concepts regarding game event recording in basketball were re-used but the concept of the GCT (see section 2.3) is unique in our approach [4].

The objective of this paper is not to optimize on prediction accuracy, but rather to help coaches with insights derived using well-known data mining methods. The optimal application of the methods in the area of team handball is the core aspect we are focusing on.

The paper describes an approach to derive helpful patterns from collected event data of team handball matches. Section 2 will introduce into the data formats that are needed to apply data mining methods and a notion of time is introduced which helps to generate patterns for different points in time of a match. Because the notion of time has a direct impact on the data preparation, details of the transformation processes are presented.

Section 3 focusses on practical aspects of data mining, introducing the tool-based solution to find useful patterns. The used mining techniques and their parameter settings will be described. Section 4 will discuss the results found and particularly the correlation with the introduced notion of time is presented. Section 5 concludes the paper with a short summary and an outlook on future work.

2. Data and Notion of Time

After multiple years of data collection in team handball, enough data has been recorded to start with pattern discovery. As introduced in previous papers, the focus of our work is to discover patterns in the context of team handball matches using classification and co-occurrence grouping. Since the two families of methods differ significantly regarding the needed input data, the data preparation concepts will be presented in the following subsections.

Table 1. *Collected attributes of games.*

Attribute	Semantics
attacks	Number of attacks
tore	Number of scored goals.
ggstore	Number of scored fast break goals.
ggsangriffe	Number of fast break attacks.
siebenmeter	Number of scored penalty goals.
fehlwuerfe	Number of attempts that missed the goal.
ballfehler	Number of ball handling errors.
regelverstoesse	Number of (offense) rule violations.
paraden	Number of saves.
blocks	Number of blocked attempts.
gelbekarten	Number of yellow cards.
zweiminuten	Number of suspensions.
penalty	Number of received penalties.
fouls	Number of sanctioned fouls.

2.1. Data for Classification

Basically, the app for recording match information collects 14 team indicator values (see Table 1) as for example the number of scored goals or the number of fast-break attacks [5]. Given the expertise from the application field we know that the success of a team varies even when the absolute number of the indicator values stays constant.

Thus, it is important to use an indicator which expresses the specific performance of a team in the context of the opponent's performance. This can be achieved using the ratio of the indicator values or the difference of these values. Since ratios are significantly more complicated to be calculated by humans compared to differences, it has been decided to use differences. This will allow coaches to easily recognize, understand and use detected patterns without the need for support by means of IT.

As a result, the collected data are transformed into a table with 14 indicator differences as attributes and one row for each participation of a team in a match. The current set of data represents 194 matches of the first German Handball Bundesliga (HBL).

2.2. Data for Co-Occurrence Grouping

Co-Occurrence Grouping, or the search for association rules, needs data in so-called transaction data format. There are two "formats" of transaction data:

1) Row-based transaction data [6]: Using a single row with two attributes to express the occurrence of an item in a transaction. One attribute holds the "transaction identifier" to group the rows and one attribute represents an "item identifier" to identify the item that "occurs" in the transaction.
2) Column-based transaction data [7]: A transaction is represented by a single row and all possible items are represented by columns. The names of the columns correspond with the identifiers of the items. The values of the item columns are of type Boolean reflecting whether an item is contained in a transaction or not. A transaction identifier is not needed in this case.

Since we use RapidMiner™ [8] to search for patterns and RapidMiner only supports column-based transaction data, the recorded data had to be transformed into column-based format. Furthermore, a value transformation is needed because the transaction format consists of Boolean values rather than numerical differences as in case of the classification data.

Previously, the differences were mapped onto Boolean values by just expressing the fact whether one team has a higher value in an indicator or not [9]. This means, that we add an attribute "has more xxx" to the transaction data for each difference attribute of the classification data. But there is a significant limitation with this one-to-one mapping: We will only get rules containing the fact that a team has a higher value of a certain indicator but not (!) if the team has a lower value. There are no rules generated with the "non-occurrence" of an item (indicated by the value false in the corresponding column of the data). Hence it is necessary to explicitly add another attribute for the "non-occurrence" for each differences attribute of the classification data. The second attribute expresses the fact that the indicator difference is less or equal to zero.

Another significant difference of the transaction data, compared to the classification data, is the fact that the difference values of the classification data can be arbitrary, while the transaction data just represents a difference of greater than 0 or less or equal zero. The application side helped by adding one level of further detail. Differences are distinguished in *significant differences* and *small differences*. Significant differences mean differences of three or more while small differences are differences of one or two.

Consequently, the differences attributes of the classification data were mapped onto four attributes of the transaction data for each attribute of the classification data representing:

1) Equal or more: the difference is 0, 1, or 2.
2) Significantly more: the difference is greater than two.
3) Less: the difference is -1 or -2.
4) Significantly less: the difference is smaller than -2.

2.3. Logical Time: The Game Clock

Rather than supporting coaches with patterns that are derived from the data at the end of matches, the objective is to provide support as early as possible during a match. However, coaches need to be able to easily detect the point in time when a pattern applies.

An application specific clock has been introduced to address this challenge [10]. It ticks whenever the nth goal is scored first (called *Goal Clock Tick n* or *GCT n*):

1) When a team scores the first goal of a match the clock ticks the first time: 1:0 or 0:1 (GCT 1).
2) Whenever a goal is scored and the maximum of the number of goals of the two teams changes, then the clock ticks again.

Examples: 1:1 – clock does not tick; 1:2 – clock ticks the second time (GCT 2); 1:3 clock ticks the third time (GCT 3); 2:3 – clock does not tick and so on and so forth.

The GCTs are very easy to detect and track for coaches. Compared to the simple number of goals, the GCT has the advantage that it can be detected much easier because it requires only to determine the maximum of two number, rather than calculating the sum of two numbers. Given that almost 55 goals are scored in average during a team handball match, this is a significant difference to soccer or ice hockey. Furthermore, there is always a leading team associated with a GCT which is not the case for the sum of goals.

It was an open question at the beginning, whether there are certain GCTs that are of "special interest". The simple evaluation of the correlation of the ownership of a GCT with the result of match has already been previously described [9]. However, this was just a first step of the evaluation which will be continued in this paper.

3. Tangible Data Mining

Rapid Miner™ has been used on behalf of the project to perform the search for patterns. Thus, some of the described approaches might not be directly applicable in case of other tools but the solution concepts will be applicable as well.

3.1. Data Preparation

The resulting data of the apps used for match recording is event data with a schema like the one used by Sportradar [11]. The data is stored in a schema of a PostgreSQL™ database using one row per recorded event. To support multiple teams while avoiding any data leakage between the teams, the data of each team are stored in a separate database (teams of different seasons are treated as different teams).

The conversion into indicator-based data as discussed in section 2, consisting of one row per match, containing one attribute value per indicator, has been implemented with a table-valued function inside the database. A key feature of this table-valued function is, that it can be parametrized with the GCT at which the indicator values are needed (always including the final result of the match as well). I.e., we can extract the classification format for arbitrary GCT values using the table-valued function.

In the design tool of RapidMiner (Rapid Miner Studio), the indicator-based data can be retrieved using standard SQL from the table-valued function. The differences of indicators are directly calculated in the query that retrieves the data. Thus, the resulting dataset is in the classification format introduced in section 2.2. This data does not contain any team identifying information anymore and can thus be shared with interested parties.

The important part of the data transformation is to have the flexibility to extract the classification format for any given GCT as well as to be able to merge the data from the databases of an arbitrary set of teams. It is not trivial because the target data extraction should be reusable for each team and the set of teams shall be extensible with minimal effort. Furthermore, it must be guaranteed that the most current data is used without having any duplicates from previous extractions.

Hence, the solution concept is a three-layered approach:

1) Top layer: clean data and initiate re-population for a specific GCT value.
2) Re-population layer: Loop over all source databases and initiate data extraction for each database.
3) Data extraction layer: Compute the query expression from the source database name and the provided GCT value, extract indicator values and differences, and store the resulting data in a so-called analysis database for the subsequent analysis step (see Figure 1).

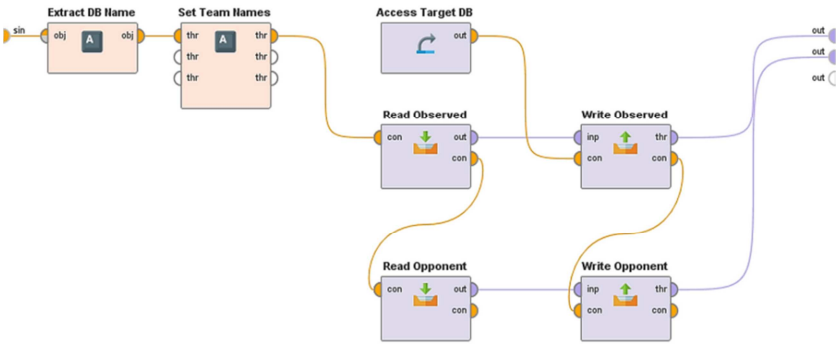

Figure 1. *The data preparation process.*

3.2. Classification

A commonality of all classification algorithms is the fact that they produce a classification model, and the quality of the model can be expressed by a confusion matrix [12]. The set of data to produce the model completes the set of important information regarding the model.

In the approach described in this paper, a set of GCTs was used to produce models with the aim to:

1) Compare the quality of the models to identify the earliest GCT with a high-quality model.
2) Compare the resulting set of rules or weights to derive the most helpful insights for coaches.

Figure 2 shows the process driving the actual classification used to produce classification models for an arbitrary set of GCTs. The GCT set of interest is explicitly defined in the "Create Example Set" operator and then used as input for the loop operator which initiates the data transformation and subsequently the model computation and the quality evaluation. The driving process produces three results: A collection of models, a collection of corresponding confusion matrices and a collection of datasets that were used to compute the models.

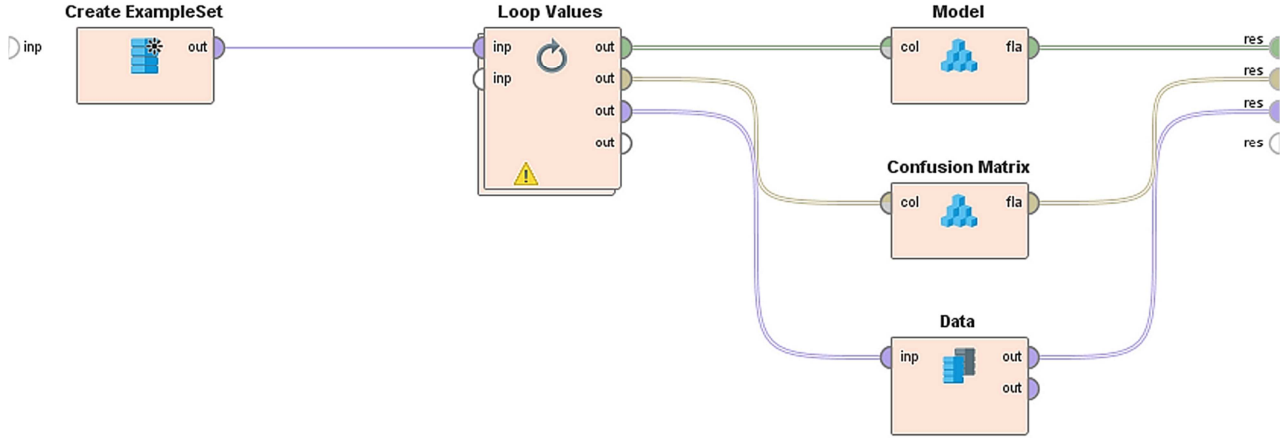

Figure 2. *The driver process.*

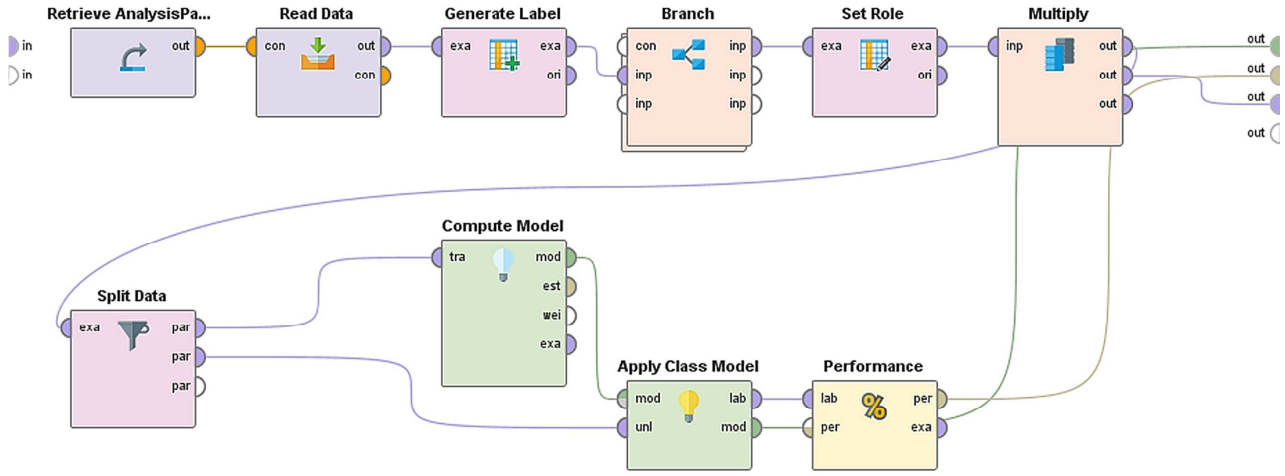

Figure 3. *The classification process.*

Figure 3 depicts the process computing the classification model and the confusion matrix. A "Generate Label" operator is used to add the attribute that is used as the class label (in the subsequent "Set Role" operator). For the results described in this paper the class label "lose" was used (Boolean value that indicates whether the goal difference is less than zero).

The "Branch" operator in the process differentiates between two cases:

1) Request to compute the model at a given GCT. In this case the difference of the number of goals at the given GCT is used as an input attribute.
2) Request to compute the model at the end of matches. Hence, the difference of the number of scored goals is not used as an input attribute, because that would dominate all other attributes (by definition).

The "Set Role" operator is needed to define the target attribute for the classification operator and the "Split Data" operator divides the input data set into a training data set and a test data set. 85% of the data have been used as training data and 15% of the data were used as test data. The used split

strategy is to generate a so-called stratified sample [13].

The specific classification operator in the process can be easily replaced to switch between different classification technologies. Two classification techniques have been identified as being most suitable in the given application context.

3.2.1. Support Vector Machines

Several classification techniques have been tested to find a classification technique that produces a prediction model with a high accuracy while having a low computation time. Furthermore, the technique should allow to summarize the model such that coaches can, at least to some extent, understand how the model works.

Support Vector Machines turned out to be highly suitable in case of predicting the "lose" attribute of the classification data [14]. After trying multiple variations of parameter settings of the model generation, a model was computed that reached an accuracy of 100% for complete matches based on the test data. Additionally, SVMs can be described using the attribute weights of the SVM function. Thus, further classification techniques, like artificial neural network-based classifiers, that lack a simple description of the model, have not been further investigated.

The results presented in the following have been computed using the following key settings:

1) SVM kernel function: radial (Rapid Miner default: dot)
2) Gamma: 0.01 (Rapid Miner default 1.0)
3) Complexity Constant: 0.50 (Rapid Miner default 0.0)
4) Max iterations: 106 (Rapid Miner default: 105)

3.2.2. Random Forests

SVMs do not help very much in terms of rules that can be extracted for coaches (or tools) to identify the need for action during a game. As previously described tree classifiers help to some extend but they are somehow limited because they derive the rules from a single root node [9].

To have the advantage of rules from tree models while not having the restriction of a single root node, the random forest technique [15] was selected as the second classification technique. The overall accuracy of a random forest model, meaning a collection of trees, is not the main focus in the described application context. It is rather to find branches with a high confidence in the trees of the model. Consequently, the complete data was used as test and training data rather than splitting the data to optimize the parameter selection, consciously ignoring potential overfitting aspects.

Furthermore, rules consisting of many parts are difficult to handle by coaches. Thus, the tree depth was pre-restricted. Testing different combinations of parameter settings has led to the following settings resulting in the best overall prediction accuracy of 89,64% using the data at the end of the matches:

1) Number of trees: 9.
2) Split criterion: Gini index.
3) Maximal tree depth of 5.
4) No pruning or pre-pruning.
5) Voting strategy: confidence vote.

3.3. Co-Occurrence Grouping

As in case of classification a driver process (see Figure 2) is used for the mining for co-occurrence groups or association rules [6]. And similarly, the data is deleted and re-populated to allow the rule mining at different GCT values.

The "inner" process to search for association rules differs significantly from the classification case (see Figure 4). Co-occurrence grouping is a descriptive method. Thus, there is no performance test operator or anything like a confusion matrix. It is rather a two-step process, first generating the so-called frequent itemsets from which the rules are then derived in a second step. For generating frequent itemsets a minimum support criterion of 8,5% has been used [6]. In the specific calculations this means that approximately 16 matches must support the set of differences expressed by the itemset. The maximum number of items was limited to 5 – again to ensure that the resulting rules can be "understood" by coaches.

The minimum confidence criterion of the rule extraction was set 85%, which is just below the confidence that has been determined for simple rules based on the early GCT 13 (see Figure 5). The subsequent rule extraction generates all possible rules from a frequent itemset.

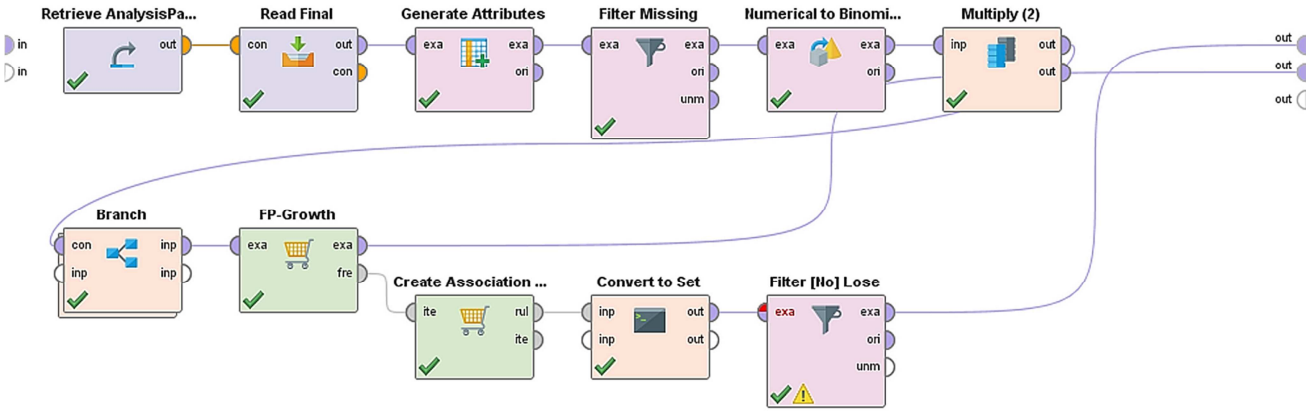

Figure 4. *The process for co-occurrence grouping.*

As depicted in Figure 4, the rules model is then converted into a set, to be able to use the filter operator. Since we are only interested in rules which have the result of the match as the consequent (to lose or not to lose), the rule extraction is followed by a filter operator which removes all rules not having a match result as the consequent of the rule. The result is a so-called example set which can be stored in tabular format and further processed using any tool for tabular data.

4. Discussion of Results

4.1. The GCT Focus

As introduced in section 3 all data mining methods were used in the context of the logical game clock. Hence, the first step was to investigate how the prediction accuracy for predicting the loss of a match changes with increasing GCTs. As introduced in [10] the prediction accuracy does not simply increase with increasing GCTs. There were two local maxima identified in the previous publication. The work presented in this paper is based on additional data and confirms the non-steady behavior.

Figure 5 depicts the confidence values and respectively accuracies of four approaches. GCT values have been tested from GCT 12 up to GCT 24. Since the number of matches not reaching the GCT increases significantly beyond GCT 24, higher values have not been investigated.

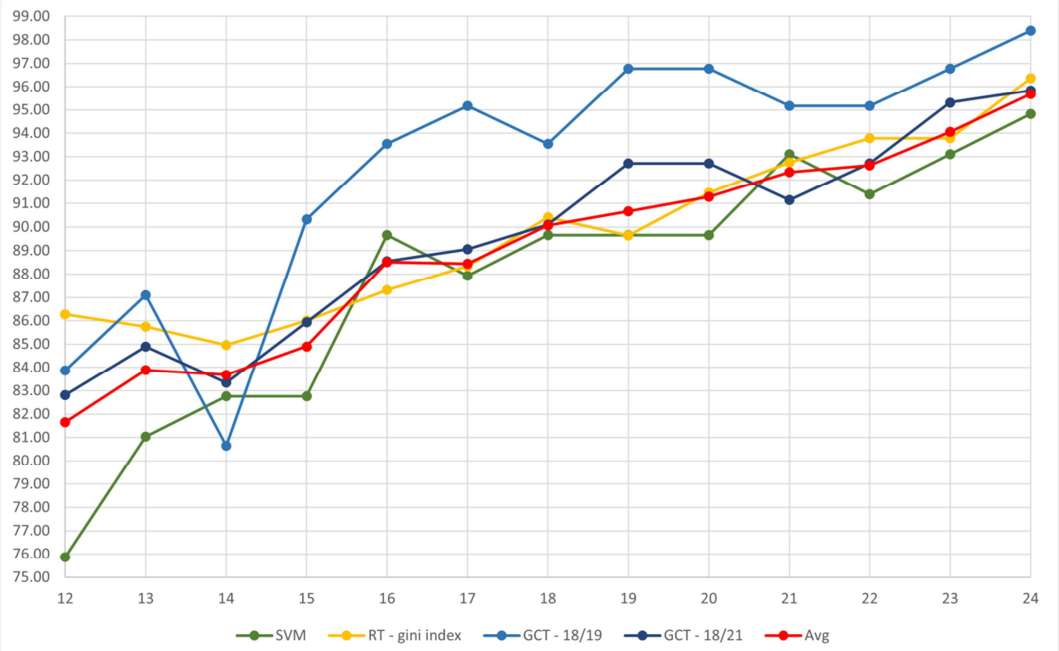

Figure 5. *Accuracy of GCTs.*

The blue curves depict the confidence value of the simple rule "The team that leads at GCT n will not lose the match". These numbers were calculated just by simple statistics for the seasons 18/19, 19/20, and 20/21 (dark blue) and just for the season 18/19 (light blue). The green curve shows the accuracy of the SVM model, and the yellow curve depicts the accuracy of the Random Forest model given the parameter settings introduced in section 3. Finally, an average confidence value has been calculated using the dark blue curve values, the SVM values and the Random Forest values, which is shown as the red curve.

The goal of the investigation is to find the early GCTs with an acceptable confidence regarding the prediction of the outcome of the match. The rules at this GCT will then be used to help coaches with their decisions. It is important to keep in mind that the earlier a certain confidence value is reached the better it is, because then coaches have more time to act.

Coaches would expect a steadily increasing confidence of the prediction of the outcome of a match the higher the GCT is. The general trend confirms this assumption, but the details contradict the hypothesis.

It is surprising that the blue curves both have a local minimum at GCT 14 and a drop at GCT 21, while having a local maximum at GCT 13. The local maximum at GCT 17 of the light blue curve was not confirmed by the longer-term observation depicted by the dark blue curve. The SVM accuracy has a local maximum at GCT 16 and a drop at GCT 17.

Based on the average confidence values (red curve) three GCTs have been selected to compare the results and extract rules:

1) GCT 13 as the "early indicator" (usually in the first half of a match) with a better average accuracy than GCT 14.
2) GCT 16 as an "intermediate indicator" (usually early in the second half of a match) being a local maximum in most curves.
3) GCT 21 as the "late" indicator (usually in the mid of the second half of match) that might still allow to act, being a local accuracy maximum of the SVM model.

These "critical" GCT values are used by coaches to verify their game plan and to detect the need for change at specific points in tim of the game.

4.2. Insights from SVM Classification

For the identified GCTs of section 4.1 as well as for the final match, SVM models have been computed given the parameter settings of section 3.2.1). the Boolean attribute "lose" has been used as the class label, which means that we want to know which attributes "contribute" to losing a match or "prevent" losing a match. The resulting weight values are depicted in Figure 6.

For each attribute a group of four bars is depicted: Each bar represents the weight value of an attribute of the SVM at the given GCT and the end of the match respectively. Each attribute represents the difference of the values of the original attribute of each team as described in section 2.1. The goal attribute (top group in Figure 6) has only three bars, because it is not used for the classification based on the final match data (yellow bars).

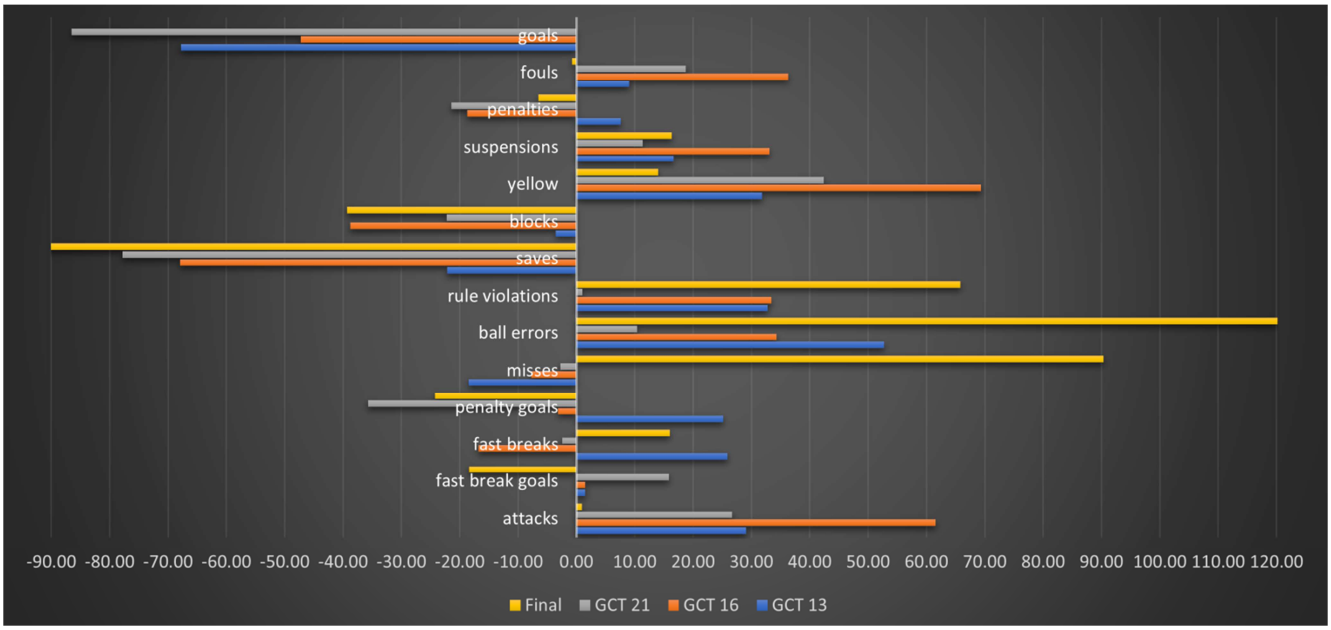

Figure 6. *SVM weights at GCTs.*

Whenever a bar grows to the right of the diagram it indicates a positive influence of an attribute on losing a match while a bar growing to the left indicates a negative influence on losing a match. Usually, groups of the same attribute are expected to have bars with the same direction, meaning that the influence of an attribute does not change fundamentally with changing GCT values. This is not the case for all attributes. There are five attributes that show significant differences.

1) While having the second highest weight regarding finally losing a match, the attribute misses is weighted negatively for all selected GCTs.
2) Penalties and penalty goals are weighted positively only for the model of GCT 13. For all other GCTs the attributes have a negative weight.
3) Fast breaks are weighted positively in case of GCT 13 and the model at the end of matches, in case of the other GCTs fast breaks are weighted negatively.
4) In case of fast break goals only the model weight at the end of matches is negative and it is positive for all GCTs.
5) There is one more weight worth mentioning: the weight of attack differences. It is relatively high for the GCTs but close to 0 in case of the model at the end of matches.

It seems that the impact of attributes changes during match time. Attributes having a positive impact become a negative impact and vice versa. While this is only somehow surprising from data science perspective, it becomes a real problem on the level of the application domain. Since the weights are interpreted as causal dependencies, it is deemed inconsistent if the principal influence changes. It cannot be negative to have more fast breaks at one GCT when it becomes negative at a later GCT.

However, the SVM weights are just a mathematical construct and the interpretation of the weights as causal strengths for a certain outcome, is by definition invalid.

4.3. Insights from Rules

4.3.1. Rules from Random Forest Models

The Random Forest model consists of 9 trees which are used to classify records based on voting. The trees of a random forest model are just like "regular" decision trees depicting the distribution of all training records. Hence, the leaves of the tree represent rules that express the path from the root decision node to the leaf. To find the interesting rules of the trees, the tree description (see Figure 7) was used and the class distribution at the leaves has been extracted.

```
attacks_diff > -0.500
|   attacks_diff > 0.500
|   |   ggstore_diff > 2.500
|   |   |   fehlwuerfe_diff > 0.500: true {false=0, true=2}
|   |   |   fehlwuerfe_diff ≤ 0.500: false {false=4, true=0}
|   |   ggstore_diff ≤ 2.500
|   |   |   ggsangriffe_diff > 3.500: false {false=1, true=0}
|   |   |   ggsangriffe_diff ≤ 3.500: true {false=5, true=68}
|   attacks_diff ≤ 0.500
|   |   ggsangriffe_diff > 1.500
|   |   |   siebenmeter_diff > -1.500: false {false=47, true=5}
|   |   |   siebenmeter_diff ≤ -1.500: false {false=4, true=4}
|   |   ggsangriffe_diff ≤ 1.500
|   |   |   tore_diff > -1.500: false {false=47, true=31}
|   |   |   tore_diff ≤ -1.500: true {false=7, true=56}
attacks_diff ≤ -0.500
|   ggsangriffe_diff > 2.500: false {false=41, true=0}
|   ggsangriffe_diff ≤ 2.500
|   |   zweiminuten_diff > -0.500: false {false=31, true=0}
|   |   zweiminuten_diff ≤ -0.500
|   |   |   paraden_diff > -1.500: false {false=27, true=4}
|   |   |   paraden_diff ≤ -1.500: true {false=0, true=2}
```

Figure 7. *Tree description.*

Additionally, the confidence and support for the derived rules were calculated as in case of association rules. However, these numbers were calculated based on the values provided in the tree model which deviate from the original data distribution due to the sampling strategy that is used internally by the Random Forest algorithm of RapidMiner.

Based on a minimum support of 8,5% and a minimum confidence of 85%, the interesting rules were identified. In essence, the predictive approach of a random forest is "misused" to derive a descriptive set of rules. Thus, overfitting effects can be ignored in this case.

The model at GCT 13 and the model at GCT 16 reveal 23 and 20 rules, respectively that satisfy the selection criteria. Rather than enumerating all the rules, only a few examples will be given. The examples will be described in the application language as we describe them to coaches:

1) The team will not lose if it has less attacks at GCT 13, equal or more blocks and not fewer yellow cards than the opponent team. Confidence: 100%, support 12.18%
2) The team will not lose, if it owns GCT 13 and has three or more fast breaks than the opponent team. Confidence: 100%, support 10.62%.
3) The similar rule at GCT 16: Confidence: 100%, support: 17.62%
4) The team will not lose if it owns GCT 16, has more saves, less than four more rule violations, and only one penalty less than the opponent. Confidence: 97.54%, support 31.61%.

There is one more rule at GCT 16 having a surprising support: The team will not lose, if it has less attacks at GCT 16 and not more than 4 rule violations, it is not more than two goals behind, and the team is just one fast break worse compared to the opponent team. The rule has a confidence of 89.47% and a support of 49.22%, meaning that it is contained in almost 50% of all observed matches.

Rules consisting of more than 4 combined conditions are very hard to understand and handle for coaches. The numeric limits of the conditions are needed to express the conditions precisely, but coaches are not able to cope with the numbers. They usually prefer a "yes/no expression" like stating the fact to have more or less of something.

4.3.2. Rules from Co-Occurrence Grouping

Since extracting rules from random forests is very costly and given the fact that the additional level of detail is not perceived as being beneficial, the search for association rules became the preferred method. Furthermore, as described in section 3.2.3) the set of rules extracted by the RapidMiner process can be converted into a dataset that can be further processed with for example RapidMiner itself or with tools like Microsoft Excel.

#P-Items	Premise	Conclusion	Support	Confidence	Lift	Laplace	Gain	Ps	GCT
3	[nm_blocks, nm_ballerrors, lm_saves]	[no_lose]	7.29%	100.00%	1.9	1.0	-0.1	0.0	13
3	[nm_blocks, m_saves, lm_tore]	[no_lose]	13.54%	100.00%	1.9	1.0	-0.1	0.1	13
3	[nm_blocks, lm_tore, lm_ggsangriffe]	[no_lose]	9.38%	100.00%	1.9	1.0	-0.1	0.0	13
3	[nm_blocks, lm_tore, lm_saves]	[no_lose]	8.07%	100.00%	1.9	1.0	-0.1	0.0	13
3	[nm_attacks, nm_ballerrors, lm_saves]	[no_lose]	8.85%	100.00%	1.9	1.0	-0.1	0.0	13
3	[nm_attacks, m_saves, lm_ggsangriffe]	[no_lose]	10.16%	100.00%	1.9	1.0	-0.1	0.0	13
3	[nm_ruleviolations, lm_tore, lm_ggsangriffe]	[no_lose]	9.38%	100.00%	1.9	1.0	-0.1	0.0	13
3	[nm_suspensions, m_ggsangriffe, lm_saves]	[no_lose]	5.21%	100.00%	1.9	1.0	-0.1	0.0	13

Figure 8. *Association rules result.*

Figure 8 shows an example section of the result computed for GCT 13. The result of GCT 13 comprises 407 rules. For GCT 16 we get 700, and for GCT 21 1,171 rules. The rule set for the end of matches consists of 195 rules which have "[lose]" or "[no_lose]" as their conclusion.

Since the method generates all possible combinations, the rules also include rules for which the premise parts have a subset relationship. Thus, when looking for interesting rules a high confidence is important but a low number of items in the premise should be the preferred starting point.

Just two examples are explained in detail to show how the rules evolve over (logical) time. The first example is the difference of the number of scored goals. At GCT 13 we find a rule expressing that "The team will lose, if it is two goals behind." with a confidence of 89,9%. The same rule at GCT 23 has a confidence of 99.3%. The rule expressing that "a team will not lose when it is in the lead", has a confidence of 85% at GCT 13 and 96,4% at GCT 23. In both cases it is an attribute that clearly indicates an increasing importance with increasing logical time regarding the outcome of matches.

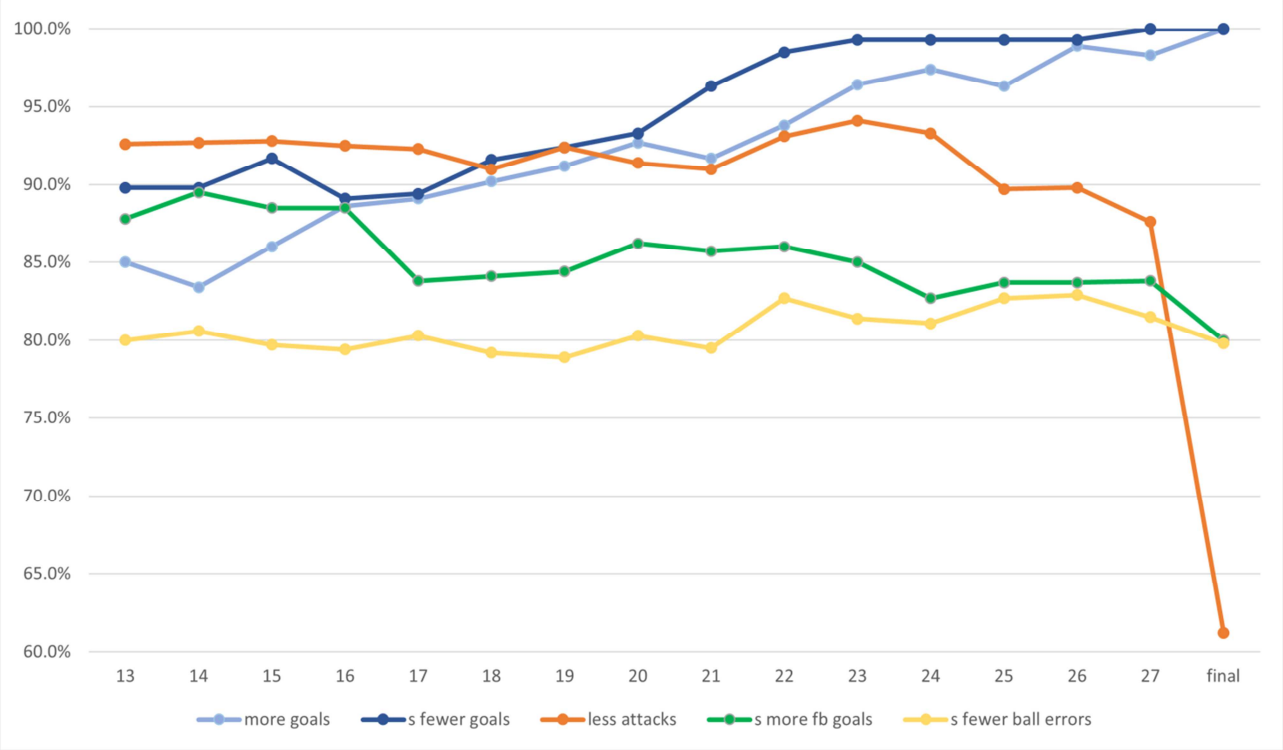

Figure 9. *Confidence by GCT.*

In Figure 9 five developments of confidence values of rules with single item premises at different GCT values have been depicted. The blue lines represent the two mentioned cases based on goals.

The orange line depicts the confidence values for the rule expressing that "a team will not lose if it has a lower number of attacks". At GCT 13 this rule has a confidence of 92.6%, decreasing to 91% at GCT 21. It finally drops dramatically to 61.2% the end of the match. Somehow similar is the development of the confidence of the rule described by: "if a team scores three or more fast break goals more than an opponent team, it will not lose" (the green line). Its confidence also starts at 87,8% at GCT 13 and decreases to 80% at the end of a match. It can be concluded that there are attributes that have a higher significance at the beginning regarding the outcome of a match which is decreasing towards the end.

A third case is depicted by the yellow line in Figure 9. It is the confidence of the rule "if a team has significantly fewer ball errors, it will not lose", The confidence remains almost at a constant level over time.

It is worth mentioning that the lift of rules [6] also differs significantly and can also help identifying interesting rules. It ranges between 1.3 and 2.17. For instance, the rule represented by the dark blue line of Figure 9 reaches a lift of 2.17 at GCT 27 showing its exceptional deviation from the statistically expected.

Again, it becomes obvious that patterns extracted by descriptive methods cannot be interpreted as causal dependencies. However, in the given application case they provide helpful guidance for coaches at certain points in time of a match. They are used to define "intermediate objectives" for the teams to reduce the likelihood of losing.

5. Conclusions

Several data mining techniques seem to be helpful to solve classification problems and usually the emphasis is on the optimization of the prediction accuracy when selecting a classification technique.

This paper focused particularly on the application scenario

of the decision support for team handball coaches. Even though the technique of support vector machines looks promising based on its prediction accuracy it cannot be directly used in the specific application scenario because the weights of the results based on the logical game clock are somehow contradicting and because the model cannot really be explained such that coaches can act on it.

The Random Forest approach solved some problems, but the extraction of rules is very costly. Furthermore, the used data to calculate the criteria to filter rules do not exactly match the distribution of the original data that was used to generate the trees. Thus, promising rules might not have a confidence as high as indicated by the model.

The search for association rules allows to find arbitrary rules that can be easily explained and used by coaches. The important step to be able to use association rules, was the mapping of the indicator value differences onto the transaction data format, needed by the search for association rules. The presented mapping in this paper is application specific and it has not been investigated whether the approach is applicable in other scenarios as well. However, it is important to recognize that the concept of differentiating small differences and significant differences directly helps finding useful rules.

Looking at the confidence of rules at different points in time of the logical clock has shown, that team handball is all but linear. The "influence" of attributes changes over time and this change of emphasis makes sense in the application context. Particularly the weights of the SVM model can be used to verify the change over time in case of the confidence of the rules.

By comparing the rules at different GCTs, coaches have now adjusted their decisions. To give some examples:

1) If you have the choice in the beginning, do not take the throw-off (to reduce the number of attacks of the team).
2) Try to prevent to be behind at GCT 13 by more than two goals. Take a timeout to adjust tactics before this happens!
3) Fast break goals are crucial! Put more emphasis on training efforts to increase the success rate of fast breaks and ensure to prevent fast breaks of the opponent team as much as possible.
4) The end of the first half and the beginning of the second half is critical. The team must be fully motivated to own GCT 17.

Particularly training efforts need adjustment. In general, there is not enough time to optimize on everything. Thus, the rules are used to spend more time on the crucial aspects.

The search for association rules has generated thousands of rules at different GCTs. The next step is to extract the most important rules which help coaches improve the team's performance, i.e., to win (or at least not to lose) more matches. We also believe that we need to collect more data to verify the rules in a broader context. However, some surprising rules at early GCTs have been successfully used already, which prove that the collection of data and the subsequent analysis are beneficial for team handball.

Acknowledgements

I would like to thank the HBL teams: "Frisch Auf! Göppingen", "HBW Balingen-Weilstetten" and "TVB Stuttgart" for their support of this work by providing their data for analysis. Furthermore, I want to thank Eckard Nothdurft and Heiko Ruess for helping to record the data, which was the basis for this work.

The publication of this article was partially funded by the ministry of science, research, and arts (MWK) of the state of Baden-Württemberg, Germany.

References

[1] R. P. Schumaker, O. K. Solieman und H. Chen, Sports Data Mining, Springer, 2010.

[2] U. Brefeld, J. Davis, J. Van Haaren und A. Zimmermann, Hrsg., Machine Learning and Data Mining for Sports Analytics, Springer, 2020.

[3] F. Goes, L. Meerhoff, M. Bueno, D. Rodrigues, F. Moura, M. Brink, M. Elferink-Gemser, A. Knobbe, S. Cunha, R. Torres und K. Lemmink, "Unlocking the potential of big data to support tactical performance analysis in professional soccer: A systematic review," *European Journal of Sport Science,* pp. 481-496, 21: 4, 2021.

[4] J. Lee, J. Lee, S. Moon, D. Nam und W. Yoo, "Basketball event recognition technique using Deterministic Finite Automata (DFA)," in *Proceedings of the 20th International Conference on Advanced Communication Technology (ICACT)*, Chuncheon, Korea (South), 2018.

[5] F. Schwenkreis, "A Three Component Approach To Support Team Handball Coaches", in 23rd Annual Congress of the European College of Sport Science, Dublin, 2018.

[6] R. Agrawal und R. Srikant, "Fast Algorithms for Mining Association Rules", in Proceedings of the 20th VLDB Conference, Santiago Chile, 1994.

[7] M. Hashler, B. Grün und K. Hornik, "arules - A Computational environment for Mining Association Rules and Frequent Item Sets", Journal of Statistical Software, Bd. 14, Nr. 15, pp. 1-25, October 2005.

[8] RapidMiner, "RapidMiner Studio", 2021. [Online]. Available: https://rapidminer.com/products/studio/. [Zugriff am April 2021].

[9] F. Schwenkreis, "Why the Concept of Shopping Baskets helps to analyze Team-Handball", in Proceedings of the 2020 International Conference on Intelligent Data Science Technologies and Applications (IDSTA), Valencia, Spain, 2020.

[10] F. Schwenkreis und E. Nothdurft, "Applied Data Science: An Approach to Explain a Complex Team Ball Game", in Proceedings of the 9th International Conference on Data Science, Technology and Applications, DATA 2020, Lieusaint, Paris, 2020.

[11] Sportradar, Handball Scout Admin (HAS) Manual, Sportradar AG, 2015.

[12] F. Provost und T. Fawcett, Data Science for Business, Sebastopol, CA: O'Reilly and Associates, 2013.

[13] J. Trost, "Statistically nonrepresentative stratified sampling: A sampling technique for qualitative studies", Qual Sociol, pp. 54-57, March 1986.

[14] I. Steinwart und A. Christmann, Support Vector Machines, New York City: Springer, 2008.

[15] T. K. Ho, "Random Decision Forests", in Proceedings of the 3rd International Conference on Document Analysis and Recognition, Montreal, 1995.