Artificial Intelligence in Medicine (2006) **37**, 19—30

ELSEVIER

ARTIFICIAL INTELLIGENCE IN MEDICINE

http://www.intl.elsevierhealth.com/journals/aiim

Semi-automatic learning of simple diagnostic scores utilizing complexity measures

Martin Atzmueller *, Joachim Baumeister, Frank Puppe

Department of Computer Science, University of Würzburg, Am Hubland, 97074 Würzburg, Germany

Received 23 July 2004; received in revised form 18 March 2005; accepted 21 March 2005

KEYWORDS
Diagnostic scores;
Knowledge discovery;
Complexity measures;
Data mining

Summary

Objective: Knowledge acquisition and maintenance in medical domains with a large application domain ontology is a difficult task. To reduce knowledge elicitation costs, semi-automatic learning methods can be used to support the domain specialists. They are usually not only interested in the accuracy of the learned knowledge: the understandability and interpretability of the learned models is of prime importance as well. Then, often simple models are more favorable than complex ones.
Methods and material: We propose *diagnostic scores* as a promising approach for the representation of simple diagnostic knowledge, and present a method for inductive learning of diagnostic scores. It can be incrementally refined by including background knowledge. We present complexity measures for determining the complexity of the learned scores.
Results: We give an evaluation of the presented approach using a case base from the fielded system SonoConsult. We further discuss that the user can easily balance between accuracy and complexity of the learned knowledge applying the presented measures.
Conclusions: We argue that semi-automatic learning methods can support the domain specialist efficiently when building (diagnostic) knowledge systems from scratch. The presented complexity measures allow for an intuitive assessment of the learned patterns.

1. Introduction

Constructing and in particular maintaining a knowledge base in medical domains is a difficult task. Additionally, it is often difficult to identify the relevant attributes (input/output) of the system to be built. However, if these objects are already known, then it is still difficult to determine the relations between these attributes and their appropriate strengths. If the degree of connectivity between attribute values and diagnoses is potentially high, then managing the sheer number of relations is a major problem. Pure automatic learn-

* Corresponding author. Tel.: +49 931 888 6739; fax: +49 931 888 6732.
E-mail addresses: atzmueller@informatik.uni-wuerzburg.de (M. Atzmueller), baumeister@informatik.uni-wuerzburg.de (J. Baumeister), puppe@informatik.uni-wuerzburg.de (F. Puppe).

0933-3657/$ — see front matter © 2005 Elsevier B.V. All rights reserved.
doi:10.1016/j.artmed.2005.03.003

ing methods are usually not good enough to achieve a quality comparable to manually built knowledge bases. However, they can be used to support the domain specialist. In such semi-automatic scenarios, the understandability and interpretability of the learned models is of prime importance. Ideally, the learning method constructs knowledge in the same representation the human expert favors.

A rather wide spread formalism for medical decision making are *diagnostic scores*, e.g., [1,2]. For deriving a concept, a limited number of features is used in a regular and simple to interpret manner, which can be applied even without a computer. In its simplest form, each feature — if observed in a case — individually contributes one point to an account (score), and if the score exceeds a given threshold, the concept is established. Variations concern using several categories instead of one point, acquiring both negative and positive contributions, and utilizing several thresholds to express different degrees, e.g., to differentiate between "possible" and "probable" for deriving a concept.

Diagnostic scores are typically implemented with scoring rules, that add specified points to a diagnosis defined in the rule action. In comparison to general rules, usually scoring rules have no logical combinations in the precondition. Compared to Bayesian networks, scores have much simpler relations and a simpler interpretation of uncertainty. Similar to both, they can be arranged hierarchically, i.e., a concept inferred with a score can be used to infer another concept. Of course, scores can be refined into both formalisms, but are quite expressible in itself. We therefore propose diagnostic scores as a promising approach to support knowledge engineering of medical knowledge systems. Thus, starting with automatically learned knowledge, the expert is able to use the learned knowledge as a starting point in constructing a knowledge base, such that the knowledge can be refined, tuned and extended as needed. One measure for the quality of a score is the accuracy of inferring the respective concept. A second important criterion is the complexity of the score, e.g., depending on the number of features used for the score. We present an inductive method for learning diagnostic scores paying special attention to a compromise between both criteria. The method can be refined incrementally using different types of background knowledge, that improve the learned scores.

Our evaluation is based on cases collected by the knowledge-based documentation and consultation system for sonography SonoConsult[3] (an advanced and isolated part of HepatoConsult[4]), built with the diagnostic shell kit D3 [5]. This system is in routine use in the DRK-hospital in Berlin/Kpenick. The cases are detailed descriptions of examination(s), together with the derived diagnoses. Both observations and diagnoses may be ordered hierarchically according to a specialization hierarchy from general to more detailed elements. For example, the diagnoses are structured in a polyhierarchy with coarse concepts of diagnoses in upper levels of the hierarchy and specializations of these concepts in lower levels of the hierarchy. The observations are semantically grouped in classes w.r.t. their appearance during the examination. Due to a standardized data acquisition strategy only necessary observations are collected for the particular cases. This setting yields a high quality of the case base with detailed and usually correct case descriptions.

The rest of the paper is organized as follows. In Section 2, we introduce the basic notions of our knowledge representation, diagnostic scores and their properties. In Section 3, we first give essential basic definitions for the general learning task. Then, we describe the method of learning diagnostic scores, and discuss additional knowledge that can be applied. In Section 4, we present complexity measures for diagnostic scores and scoring rules. These complexity measures are used to determine the understandability of the learned knowledge. An evaluation with a real-world case base is given in Section 5. We conclude the paper in Section 6 with a discussion of the presented work, and we show promising directions for future work.

2. Diagnostic scores using scoring rules

In the following, we define necessary notions for the learning task. First, we want to consider the objects that are used as an input of a diagnostic knowledge system.

Definition 1 (*Attribute and attribute values*). Let Ω_A be the universe set of all attributes available in the problem domain. A value $v \in \mathrm{dom}(a)$ assigned to an attribute $a \in \Omega_A$ is called an *attribute value* and we call $\Omega_{\mathcal{F}}$ the set of all possible attribute values (findings) in the given problem domain. An attribute value $f \in \Omega_{\mathcal{F}}$ is denoted by $a : v$ for $a \in \Omega_A$ and $v \in \mathrm{dom}(a)$. The set $F_a \subseteq \Omega_{\mathcal{F}}$ of possible attribute values for a given attribute a is defined as $F_a = \{ f \in \Omega_{\mathcal{F}} | \, f = a : v \wedge v \in \mathrm{dom}(a) \}$. Each attribute value $f \in \Omega_{\mathcal{F}}$ is defined as a possible input of a diagnostic knowledge system.

Other common names for attribute values are *findings* and *observations*. A diagnostic system usually comes up with a solution for a given problem. These solutions are defined as diagnoses.

Definition 2 (*Diagnosis*). Let d be a *diagnosis* representing a possible output, i.e., a solution, of the diagnostic knowledge system. We define $\Omega_{\mathcal{D}}$ to be the universe of all possible diagnoses for a given problem domain.

A symbolic state $\mathrm{dom}(d) = \{\textit{unlikely}, \textit{probable}\}$ is assigned to a diagnosis $d \in \Omega_{\mathcal{D}}$ with respect to a given problem. The value range of a diagnosis d is denoted by $\mathrm{dom}(d)$.

A collection of cases is given as an input to the learning system.

Definition 3 (*Case*). A *case* c is defined as a tuple

$$c = (\mathcal{F}_c, \mathcal{D}_c, \mathcal{I}_c),$$

where $\mathcal{F}_c \subset \Omega_{\mathcal{F}}$ is a set of attribute values given as input to the case. Often $\mathcal{F}_c$ is also called the set of *observations* for the given case. The set $\mathcal{D}_c \subseteq \Omega_{\mathcal{D}}$ contains the diagnoses describing the solution of the case c, and $\mathcal{I}_c$ contains additional (meta-) information describing the case c in more detail. The set of all possible cases for a given problem domain is denoted by Ω_C.

For the learning task, we consider a case base $\mathrm{CB} \subseteq \Omega_C$ containing all available cases that have been previously solved.

A simple and intuitive way for representing inferential knowledge is the utilization of *diagnostic scores*[6, Ch. 3]. Then, simple scoring rules are applied.

Definition 4 (*Simple scoring rule*). A *simple scoring rule* r is denoted as follows:

$$r = f \xrightarrow{s} d,$$

where $f \in \Omega_{\mathcal{F}}$ is an attribute value, and $d \in \Omega_{\mathcal{D}}$ is the targeted diagnosis. For each rule a symbolic confirmation category $s \in \Omega_{\mathrm{scr}}$ is attached with

$$\Omega_{\mathrm{scr}} \in \{S_3, S_2, S_1, 0, S_{-1}, S_{-2}, S_{-3}\}.$$

Let Ω_R be the universe of all possible rules for the sets $\Omega_{\mathcal{F}}, \Omega_{\mathcal{D}}$ and Ω_{scr}. Then, we call $\mathcal{R} \subseteq \Omega_R$ the *r*ule base containing the inferential knowledge of the problem domain.

Scores are used to represent a qualitative approach for deriving diagnoses with symbolic confirmation categories. These categories state the degree of confirmation or disconfirmation of a particular diagnosis. In this way, a symbolic category s expresses the strength for which the observation of the attribute value f will confirm/disconfirm the diagnosis d. Whereas $s \in \{S_1, S_2, S_3\}$ stand for confirming symbolic categories in ascending order, the categories $s \in \{S_{-1}, S_{-2}, S_{-3}\}$ are ascending categories for disconfirming a diagnosis. A rule with category 0 has no effect on the diagnosis' state, and therefore is usually omitted from the rule base. It is worth noticing that the value range Ω_{scr} of the possible symbolic categories is not fixed. For a more detailed (or coarse) representation of confirmation the value range may be extended (or reduced).

For a given case $c \in \Omega_C$ the final state of each diagnosis $d \in \Omega_{\mathcal{D}}$ is determined by evaluating the available scoring rules $r \in \mathcal{R}$ targeting d. Thus, rules $r = f \xrightarrow{s} d$ contained in $\mathcal{R}$ are activated, if f is observed in case c, i.e., $f \in \mathcal{F}_c$. The symbolic categories of the activated rules are aggregated by adding the categories in a way, so that four equal categories result in the next higher category (e.g., $S_1 + S_1 + S_1 + S_1 = S_2$), and so that two equal categories with opposite sign nullify (e.g., $S_1 + S_{-1} = 0$). For a more detailed or coarse definition of Ω_{scr} the aggregation rules may be adapted. A diagnosis is assumed to be *probable* (i.e., part of the final solution of the case), if the aggregated score is greater or equal than the symbolic category S_3. Analogously, a diagnosis is assumed to be *unlikely*, if the aggregated score is less or equal than the symbolic category S_{-3}.

Discussion: The presented knowledge representation of diagnostic scores differs from probabilistic models and certainty factors in several ways: diagnostic scores do not use a probabilistic model for weighting the relations between attribute values and diagnoses, but attach symbolic confirmation categories to these relations. These categories are aggregated by a simple (linear) sum function. Thus, the weighting of scores is not normalized, but the final state of a score is determined using a fixed threshold value. Therefore, we see that diagnostic scores have deficiencies when compared to probabilistic models due to their independence assumption between attribute values. Furthermore, if the strength of a combination of attribute values is disproportionate when compared to the single observation of the attribute values, then the presented knowledge representation is not appropriate, since the particular attribute values can only contribute to a diagnostic score in a linear way. This problem is commonly tackled by introducing an abstracted attribute, for which its values are derived w.r.t. the values of the "combined" attributes. Subsequently, the abstracted attribute is used instead of the combined attributes.

Furthermore, the application of diagnostic scores for building large knowledge systems is still reasonable due to the following aspects: when compared to confirmation categories the elicitation/adaption and interpretation of probabilities is quite difficult for domain specialists that are not familiar with

statistics. Estimating and adapting a small number of confirmation categories is often much easier than estimating the difference between two probabilistic values (e.g., $p_1 = 0.01$ and $p_2 = 0.001$). Although, diagnostic scores cannot model dependency relations between attribute values, which, e.g., can easily be implemented with Bayesian networks, the effect of this inaccuracy is often negligible when not all dependencies between attributes are known. This is frequently the case in the medical domain.

While diagnostic scores lack a sound probabilistic interpretation they are very suitable for the (simplified) construction and maintenance of diagnostic knowledge systems done by the domain specialists themselves. Due to its wide spread use, the concept of diagnostic scores is often already familiar to the domain experts (e.g., [8]). Additionally, diagnostic scores are simple to model and very easy to comprehend, and have very intuitive explanation capabilities concerning practical applications. In the past, it has been shown that the performance of a knowledge system mainly depends on the quality of the included knowledge and not on the applied knowledge representation [9].

Related work: Scoring rules have proved to be useful in large medical knowledge bases, e.g., in the INTERNIST/QMR project [10,11]. Later, probabilistic aspects of the QMR system were investigated [12]. In the context of the PIT system, Fronhöfer and Schramm [13] investigated the probabilistic aspects of scores. The LEXMED [14] project is a successful application developed with the PIT system. The CADIAG [15,16] systems integrate diagnostic scores in a fuzzy framework also using semi-automatic techniques for knowledge acquisition. In our own work with D3, a workbench for developing knowledge systems, scores have been applied successfully in many (large) projects, e.g., in a geo-ecological application [17], in medical domains [4] and technical domains [6] using generalized scores.

3. Learning diagnostic scores

In the following we first discuss diagnostic profiles utilized in the learning method. Then, we briefly discuss necessary data preprocessing steps for the learning task. After that we outline the method for inductive learning of diagnostic scores from cases. Diagnostic profiles describe a compact representation for each diagnosis.

Definition 5 (*Diagnostic profile*). A diagnostic profile P_d for a diagnosis $d \in \Omega_{\mathcal{D}}$ contained in a case base CB is defined as the set of tuples

$$P_d = \{(f, \text{freq}_{f,d}) \mid f \in \Omega_{\mathcal{F}} \wedge \text{freq}_{f,d} \in [0,1]\},$$

where f is an attribute value and $\text{freq}_{f,d} \in [0,1]$ represents the frequency the attribute value f occurs in conjunction with d in the case base CB. Only attribute values f are stored that occur frequently with the diagnosis d.

Learning diagnostic profiles: We construct diagnostic profiles by first learning coarse frequency profiles such that the frequencies of the attribute values are determined according to the given case base. Thus, learning diagnostic profiles entails, that each profile will initially contain all attribute values which occur together with the profile's diagnoses. Therefore, also seldom attribute values occur in the plain profile. However, we want to construct a profile only containing typical attribute values for a given diagnosis. Thus, we apply a statistical pruning method, removing unfrequent attribute values. After that, a diagnostic profile contains at least all relevant attribute values for the specified diagnosis. For a more detailed discussion we refer to [18].

3.1. Basic algorithm

The basic algorithm for learning diagnostic scores can only handle discrete valued attributes. Therefore, for handling continuous data we will transform continuous attributes into attributes with discrete partitions in a preprocessing step. For some continuous attributes the domain specialist already defined reasonable partitions. In the case that there are predefined partitions available, we use these. Otherwise, we used a *k-means* clustering method for discretizing attribute values. The value of k is dependent on the number of related diagnoses, the minimal value is 5.

For learning diagnostic scores, we first have to identify dependencies between attribute values and diagnoses. In general, all possible combinations between diagnoses and attribute values have to be taken into account. However, to reduce the search space, we first learn diagnostic profiles identifying typical attribute values for a diagnosis. Thus, we restrict the set of *important* attribute values for a diagnosis using diagnostic profiles.

In summary, we basically apply three steps for learning a diagnostic scoring rule:

(1) Identify a dependency association between an attribute value $f \in \Omega_{\mathcal{F}}$ and a diagnosis $d \in \Omega_{\mathcal{D}}$
(2) Rate this dependency and map it to a symbolic category $s \in \Omega_{\text{scr}}$
(3) Finally, construct a diagnostic rule: $f \xrightarrow{s} d$

This basic procedure is shown in Algorithm 1, and explained below in more detail.

Algorithm 1 (*Learning simple diagnostic scores*).

Require Case base $CB \subseteq \Omega_C$
1. **for all** diagnoses $d \in \Omega_D$ **do**
2. Learn a diagnostic profile P_d
3. **for all** attributes $a \in \{a \mid a \in \Omega_{(A)}, \exists\ f \in F_a,\ f \in P_d\}$ **do**
4. **for all** attribute values $f \in F_a$ **do**
5. Construct binary variables D, F for d and f, which measure if d and f occur in cases of the case base CB.
6. Compute $\chi^2_{fd} = \chi^2(F, D)$
7. **if** $\chi^2_{fd} \geq \chi^2_\alpha$ **then**
8. Compute the correlation $/\phi_{fd}$ coefficient $\phi_{fd} = \phi(F, D)$
9. **if** $|\phi_{fd}| \geq \text{threshold}_c$ **then**
10. Compute the quasi-probabilistic score qps, $\text{qps} = \text{sgn}(\phi_{fd}) \cdot \text{prec}(r)(1 - \text{FAR}(r))$ using the pseudo-rule: $f \rightarrow d$
11. Map the qps-score to a symbolic category s using a conversion table
12. Apply background knowledge to validate the diagnostic scoring rule, if available
13. Create a diagnostic scoring rule (if valid): $f \xrightarrow{s} d$

For the definitions of prec (precision) and FAR (false alarm rate) see text.

For each diagnosis $d \in \Omega_{\mathcal{D}}$ we create a diagnostic profile. We consider all attributes in the profile selecting the attribute values which are observed in the case base. Then, we create a four-fold contingency-table for each attribute value − diagnosis relation. With the given diagnosis d and attribute value f of attribute a, i.e., $f = a : v$, we construct two binary variables limiting these to cases $\mathcal{C} \subseteq$ CB from the case base in which attribute a is not unknown: a variable D is assigned to the value *true*, if diagnosis d occurs in a case, and false otherwise, and a variable F is assigned to the value *true*, if attribute value f occurs in a case, otherwise F is false likewise. We fill the four-fold table as shown below.

	$D = true$	$D = false$
$F = true$	a	b
$F = false$	c	d

The frequency counts denoted in the table are defined as follows:

$$a = N(D = true \wedge F = true),$$
$$b = N(D = false \wedge F = true),$$
$$c = N(D = true \wedge F = false),$$
$$d = N(D = false \wedge F = false),$$

where $N(\text{cond})$ is the number of times the condition, cond, is true for cases $c \in \mathcal{C}$.

To identify dependencies between attribute values and diagnoses, we apply the χ^2-*test for independence* with a certain threshold χ^2_α corresponding to confidence level α. For binary variables, the formula for the χ^2-test simplifies to

$$\chi^2(F, D) = \frac{(a + b + c + d)(ad - bc)^2}{(a + b)(c + d)(a + c)(b + d)}. \tag{1}$$

We require a certain minimal support threshold for an attribute value f co-occurring with diagnosis d. The default threshold is set to 5, i.e., the attribute value has to occur together with the diagnosis at least five times. Generally, for small sample sizes, we apply the Yates' correction for a more accurate result. For all dependent tuples (F, D) we derive the quality of the dependency using the ϕ-coefficient

$$\phi(F, D) = \frac{ad - bc}{\sqrt{(a + b)(c + d)(a + c)(b + d)}}, \tag{2}$$

which measures the degree of association between two binary variables. We use it to discover positive or negative dependencies. If the absolute value of $\phi(F, D)$ is less than a certain threshold threshold_c, i.e., $|\phi(F, D)| < \text{threshold}_c$, then we do not consider this weak dependency for rule generation. For the remaining dependencies we generate rules described as follows: If $\phi(F, D) < 0$, then we obtain a negative association between the two variables, and we will generate a rule $f \xrightarrow{s} d$ with a negative category s. If $\phi(F, D) > 0$, then we construct a rule $f \xrightarrow{s} d$ with a positive category s.

For determining the exact symbolic confirmation category of the remaining rules $r = f \rightarrow d$, we utilize two measures used in diagnosis: *precision* and the *false alarm rate (FAR)*, which is also known as the *false positive rate*, or 1 − specificity. The precision of a rule r is defined as

$$\text{prec}(r) = \frac{\text{TP}}{\text{TP} + \text{FP}}, \tag{3}$$

whereas the false alarm rate, FAR, for a rule, r, is defined as

$$\text{FAR}(r) = \frac{\text{FP}}{\text{FP} + \text{TN}}. \tag{4}$$

The symbols TP, TN, FP denote the number of *true positives*, *t* rue negatives, and *false positives*, respectively. These can easily be extracted from the contingency table. For a positive dependency between attribute value f and diagnosis d, $\text{TP} = a, \text{TN} = d$ and $\text{FP} = b$. For a negative dependency the situation is different, since we try to predict the absence of the diagnosis, so $\text{TP} = b, \text{TN} = c$ and $\text{FP} = a$.

To score the dependency, we first compute a *quasi probabilistic score (qps)* which we then map to a symbolic category. The numeric qps score for a rule r is computed as follows

$$\mathrm{qps}(r) = \mathrm{sgn}(\phi(D,F)) \cdot \mathrm{prec}(r)(1 - \mathrm{FAR}(r)). \quad (5)$$

We achieve a trade-off between the accuracy of the diagnostic scoring rule to predict a disease measured against all predictions and the proportion of false predictions. It is worth noting, that often the *t* rue positive rate (TPR) — which is also known as *r* ecall/sensitivity — is used in combination with the FAR as a measure of accuracy.

However, this is mostly applicable to standard rules, that usually contain more complex rule conditions than scoring rules applied in diagnostic scores. Since a diagnostic score is a combination of several diagnostic scoring rules, which support each other in establishing a diagnosis, their accuracy needs to be assessed on localized regions of the diagnosis space. So, precision is more suggestive, since it does not take the entire diagnosis space into account, but it measures only the accuracy of the localized prediction. To ease interpretability of the discovered knowledge, we restrict the mapping process to only six different symbolic confirmation categories, three positive and three negative ones.

The qps-scores are mapped to the symbolic categories according to the following conversion table:

qps(r)		category(r)	qps(r)		category(r)
$[-1.0, -0.9)$	$\rightharpoonup$	S_{-3}	(0.0, 0.5)	$\rightharpoonup$	S_1
$[-0.9, -0.5)$	$\rightharpoonup$	S_{-2}	[0.5, 0.9)	$\rightharpoonup$	S_2
$[-0.5, 0.0)$	$\rightharpoonup$	S_{-1}	[0.9, 1.0]	$\rightharpoonup$	S_3

We accept the loss of information to increase the understandability and to facilitate a user-friendly adaptation of the learned diagnostic scores.

3.2. Including background knowledge

The presented algorithm can be augmented with background knowledge in order to achieve better learning results. We introduce abnormality information and knowledge about partition classes as appropriate background knowledge.

Abnormality information: Sometimes *abnormality* information about attribute values may be available. Then, each value v of an attribute a is attached with a label that explains, if v is describing a normal or an abnormal state of the attribute. For example, consider the attribute *temperature* with the value range

$$\mathrm{dom}(\mathrm{temperature}) = \{\mathrm{normal}, \mathrm{marginal}, \mathrm{high}, \mathrm{very\ high}\}.$$

The values *normal* and *marginal* denote normal values of the attribute, and the values *high* and *very high* describe abnormal states of the attribute. We will use these abnormalities, for further reducing the size of the generated rule base.

Let $r = a : v \xrightarrow{s} d$ be a given scoring rule. If $s \in \Omega_{\mathrm{scr}}$ denotes a positive category and v is a normal value of attribute a, then we do not create rule r, since attribute values describing normal behavior usually should not increase the confirmation of a diagnosis. Furthermore, if s denotes a negative category and v is an abnormal value of attribute a, then we likewise do not create rule r, because an abnormal attribute value usually should not decrease the confirmation of a diagnosis, but increase the confirmation of other diagnoses.

Partition class knowledge: As a second type of background knowledge the expert can provide *partition class* knowledge describing how to divide the set of diagnoses and attributes into partially disjunctive subsets, i.e., partitions. These subsets correspond to certain problem areas of the application domain.

For example, in the medical domain of sonography, we have subsets corresponding to problem areas like *liver*, *pancreas*, *k* idney, *stomach* and *intestine*. This knowledge is especially useful when diagnosing multiple faults. Since a case may contain multiple solutions, attributes occurring with several diagnoses will be contained in several diagnostic profiles. We reduce noise and irrelevant dependencies by pruning such discovered dependencies $f \rightarrow d$, for which f and d are not in the same partition class.

3.3. Post-processing

To reduce the number of generated rules and to make them more comprehensible, we merge rules for discretized continuous attributes. We combine rules covering neighboring discretized partitions of attributes into one rule, if they have equal symbolic categories. This does not result in a loss of accuracy, because the rule base is not changed semantically, but only tightened syntactically.

Pruning the learned scores: For making the learned scores even more compact, an optional pruning method can be applied. This reduces the number of learned scoring rules even further, and thus increases the understandability of the learned scores. The pruning approach is a heuristic, which can be applied by the user to simplify the learned scores even more.

We apply the following three steps for pruning a score: The score of a particular diagnosis is first divided into different sets of scoring rules according

to the symbolic confirmation category $s(r)$ of a rule r. We define a set of *positive scoring rules* $\mathrm{PSR} = \{r|s(r) \in \{S_3, S_2, S_1\}\}$, and a set of negative scoring rules $\mathrm{NSR} = \{r|s(r) \in \{S_{-3}, S_{-2}, S_{-1}\}\}$.

- If there are no positive rules, i.e., $\mathrm{PSR} = \varnothing$ but only negative rules, i.e., $\mathrm{NSR} \neq \varnothing$, then we remove all rules in NSR. This is motivated by the fact that knowledge concerning a score is obviously more helpful, if the diagnosis of the score can be established. If only rules with negative categories are available, then this still can be useful information for the user. However, if the goal is to find compact and simple scores, then the rules can be pruned.
- Suppose we can extract a set of rules contained in PSR that cannot categorically establish a diagnosis together, i.e., all rules with a confirmation category $s(r) \in \{S_1, S_2\}$. Then this subset can be pruned as well. In this respect the user has to balance the loss of information versus the increase of understandability.
- Considering the minimum confirmation category $\min_s$ of a rule in PSR, i.e., $\min_s = n\{s = s(r)|r \in \mathrm{PSR}\}$ we can remove the negative rules of NSR if they cannot deestablish the diagnosis of the score together given only $\min_s$. This heuristic is motivated by the fact that the negative score will not change anything concerning classification, if removed. However, the user has to consider the loss of information as well, since adjusting the confirmation categories of the removed negative scores may improve the score.

In Section 5, the significance and impact of the presented pruning methods are shown in comparison to the plain learning method.

3.4. Discussion

In this section, we presented a semi-automatic method for learning diagnostic scores. In such a semi-automatic process, the user can guide the learning method by adapting the following parameters:

- With the *number of different confirmation* categories the user controls the granularity of the learned knowledge. A smaller number of different categories may simplify the interpretation of the learned scores, but may decrease its accuracy.
- Using $\mathrm{threshold}_c$ for the ϕ *-coefficient* we determine the strengths of the dependencies which are taken into account for the generation of scoring rules, e.g., if $\mathrm{threshold}_c$ is set to a higher value, then only stronger dependencies are considered.
- *Background knowledge*: The application of abnormality knowledge or partition class knowledge can be used to potentially reduce the score size without a significant decrease of accuracy.

Additionally, an optional pruning step can be performed in order to reduce the number of generated rules. No further parameters are necessary for this step.

4. Complexity measures for diagnostic scores

The presented method describes an approach for semi-automatically learning diagnostic scores consisting of scoring rules. In contrast to classical machine learning approaches, the method is not only interested in the accuracy of the resulting scores, but also emphasizes the understandability of the learned rules.

Understandability, unexpectedness, actionability, surprisingness, validity and simplicity measured on rules, or patterns in general, are several interestingness measures used in data mining research [19,20]. Validity is most often measured, and together with the simplicity it can be regarded as an objective measure. We will focus on these in our evaluation, for which we will assess the accuracy, corresponding to validity, and the complexity corresponding to the simplicity measure.

The understandability of the learned scores is typically defined by its complexity which can be measured with respect to the learned scoring rules in the rule base $\mathcal{R} \subseteq \Omega_R$. If the learned rules have a low complexity, then it is easier for the expert/user to understand the rules.

In general, a score is considered to be the more complex, the more attribute values it contains. This directly corresponds to the number of learned rules per diagnosis. An overall impression of the complexity of the learned scores is given by the total number of learned rules. Furthermore, as a global complexity measure we count the total number of attribute values used in scoring rules of the rule base. Usually a moderate number of attribute values is considered more comprehensible than a huge number of attribute values.

It is obvious, that it is difficult to determine the complexity of a rule base by only one complexity measure, since there are several issues influencing the complexity. We consider the following issues and define corresponding complexity functions applied on scoring rule bases:

Applied attribute values: Number of attribute values occurring in the rules of the rule base; the rule base

is much simpler to survey, if fewer attribute values are used for describing the scores. The value of this measure is influenced by the pruning parameters specified for the learning method.

RULE BASE SIZE: Overall number of learned scoring rules; obviously the number of scoring rules is a direct measure for the complexity of the learned knowledge. However, for a more detailed analysis of the rule base complexity the applied classes of confirmation categories should be considered. Thus, the interpretation of scoring rules categorically establishing or excluding a diagnosis, i.e., S_3, S_{-3}, is very simple, when compared to scoring rules with less certain confirmation categories, e.g., S_1, S_{-1}.

Therefore, it is suggestive to define a weighting function w for confirmation categories. In the context of our work, we defined $w(s) = 1$ for $s \in \{S_3, S_{-3}\}$, and $w(s) = 2$ otherwise. Thus, we define a category $s \in \Omega_{scr} \backslash \{S_3, S_{-3}\}$ to be as double complex as the certain categories S_3, S_{-3}.

In summary, the measure RULE BASE SIZE is simply defined by the count of the rules contained in the rule base. A more refined measure RULE BASE SIZE for a rule base $\mathcal{R} \subseteq \Omega_R$ is defined as follows

$$\text{Rule base size}\,(\mathcal{R}) = \sum_{r \in \mathcal{R}} w(\text{category}(r)).$$

MEAN RULES: Mean number of rules for scoring a single diagnosis; obviously, less rules for scoring a diagnosis are much simpler to understand than more rules. This measure directly depends on the RULE BASE SIZE. Likewise, it can be measured using the weighted categories or directly measuring the count of rules.

CATEGORY RANGE: Number of different confirmation categories applied in the rule base, i.e., $|\Omega_{scr}|$; a small universe of possible confirmation categories allows for a simpler distinction between the single categories. This measure is predefined by the settings of the learning method, i.e., we specify the universe of categories beforehand.

SCORE CATEGORIES: Mean number of different confirmation categories applied for scoring a single diagnosis. A smaller number of distinct categories allows for a much simpler interpretation of the diagnosis score, since confirmation strengths of the attribute values contributing to a score are less distributed. This measure is indirectly dependent on the CATEGORY RANGE measure described above.

We apply the presented complexity measures in the next section when evaluating the learned diagnostic scores.

Related work: Favoring simple rules is in line with a classic principle of inductive learning methods called Ockham's Razor [21]. Existing interestingness measures applying this principle generate compact rules [22], for example, which takes the number of rules, the number of conditions in a rule, and the classification accuracy of a rule into account. A general measure discussed by [20] takes the size of the disjuncts of a rule into account. Due to the fact that we only consider simple scoring rules not containing disjuncts, this measure is not applicable to diagnostic scores. We purely concentrate on the syntactic elements contained in the rule base $\mathcal{R}$. Instead of combining several aspects in an interestingness measure, we determine the complexity measure and accuracy measure separately.

5. Evaluation

We evaluated the introduced learning method with a case base collected by the fielded knowledge system SONOCONSULT. We first describe the properties of the case base, and then discuss the F-measure used in the context of the evaluation. Finally, we present and discuss the results of the evaluation.

5.1. Properties of the case base

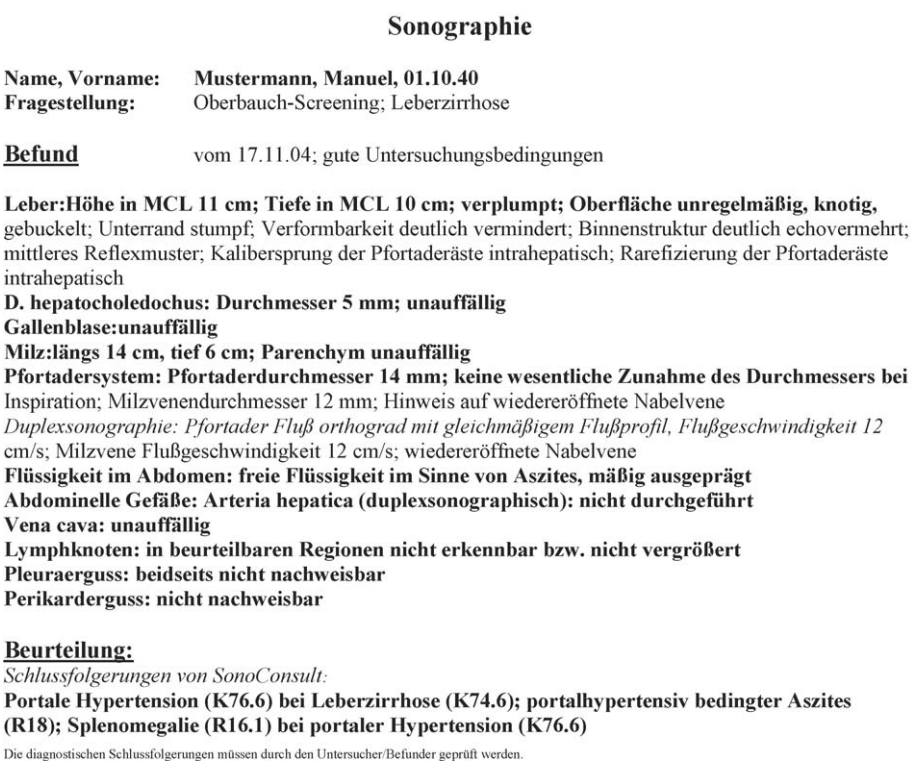
We evaluated the methods with cases taken from a medical application, which is currently in routine use. The applied SONOCONSULT case base contains 1340 cases, with a mean of diagnoses $M_d = 4.32 \pm 2.79$ and a mean of relevant attribute values $M_f = 76.89 \pm 20.59$ per case.

SONOCONSULT is a knowledge-based documentation and consultation system for sonography, an advanced and isolated part of HEPATOCONSULT[4], developed and maintained by the domain specialists using the shell-kit D3 [5]. The quality of the derived diagnoses usually is very good as checked by experts in a medical evaluation, cf. [3]. In Fig. 1, the data of a case acquired with SONOCONSULT is shown (in german).

5.2. Evaluating accuracy in multiple fault problems

For the evaluation of the experiments we adopted the commonly used F-measure known from information extraction theory which is appropriate for comparing solutions with multiple faults. For the correct solution $\mathcal{D}_1$ and an inferred solution $\mathcal{D}_2$ the F-measure is defined as follows ($D_1, D_2 \subseteq \Omega_{\mathcal{D}}$):

$$f(\mathcal{D}_1, \mathcal{D}_2) = \frac{(\beta^2 + 1) \cdot \text{prec}(\mathcal{D}_1, \mathcal{D}_2) \cdot \text{recall}(\mathcal{D}_1, \mathcal{D}_2)}{\beta^2 \cdot \text{prec}(\mathcal{D}_1, \mathcal{D}_2) + \text{recall}(\mathcal{D}_1, \mathcal{D}_2)}, \tag{6}$$



Sonographie

Name, Vorname: **Mustermann, Manuel, 01.10.40**
Fragestellung: Oberbauch-Screening; Leberzirrhose

Befund vom 17.11.04; gute Untersuchungsbedingungen

Leber:Höhe in MCL 11 cm; Tiefe in MCL 10 cm; verplumpt; Oberfläche unregelmäßig, knotig, gebuckelt; Unterrand stumpf; Verformbarkeit deutlich vermindert; Binnenstruktur deutlich echovermehrt; mittleres Reflexmuster; Kalibersprung der Pfortaderäste intrahepatisch; Rarefizierung der Pfortaderäste intrahepatisch
D. hepatocholedochus: Durchmesser 5 mm; unauffällig
Gallenblase:unauffällig
Milz:längs 14 cm, tief 6 cm; Parenchym unauffällig
Pfortadersystem: Pfortaderdurchmesser 14 mm; keine wesentliche Zunahme des Durchmessers bei Inspiration; Milzvenendurchmesser 12 mm; Hinweis auf wiedereröffnete Nabelvene
Duplexsonographie: Pfortader Fluß orthograd mit gleichmäßigem Flußprofil, Flußgeschwindigkeit 12 cm/s; Milzvene Flußgeschwindigkeit 12 cm/s; wiedereröffnete Nabelvene
Flüssigkeit im Abdomen: freie Flüssigkeit im Sinne von Aszites, mäßig ausgeprägt
Abdominelle Gefäße: Arteria hepatica (duplexsonographisch): nicht durchgeführt
Vena cava: unauffällig
Lymphknoten: in beurteilbaren Regionen nicht erkennbar bzw. nicht vergrößert
Pleuraerguss: beidseits nicht nachweisbar
Perikarderguss: nicht nachweisbar

Beurteilung:
Schlussfolgerungen von SonoConsult:
Portale Hypertension (K76.6) bei Leberzirrhose (K74.6); portalhypertensiv bedingter Aszites (R18); Splenomegalie (R16.1) bei portaler Hypertension (K76.6)

Die diagnostischen Schlussfolgerungen müssen durch den Untersucher/Befunder geprüft werden.

Figure 1 Exemplary case data acquired by the SonoConsult system.

where β denotes a constant weight for the precision, and

$$\text{prec}(\mathcal{D}_1, \mathcal{D}_2) = \frac{|\mathcal{D}_1 \cap \mathcal{D}_2|}{|\mathcal{D}_2|},$$

$$\text{recall}(\mathcal{D}_1, \mathcal{D}_2) = \frac{|\mathcal{D}_1 \cap \mathcal{D}_2|}{|\mathcal{D}_1|}.$$

We used $\beta = 1$ in the context of our experiments.

5.3. Experimental results and discussion

For the evaluation we applied a stratified 10-fold cross-validation method. We performed several experiments to determine the impact of including background knowledge into the learning process, and also to measure the significance of pruning on the learned scores. The significance and impact was measured using the accuracy and the understandability of the learned knowledge, evaluated by the complexity measures.

We present the results of three experiments. For experiment *E*0 we applied no background knowledge at all. To show how the application of knowledge improves the results, we used both partition class knowledge and abnormality knowledge for experiment *E*1. Then, for experiment *E*2 we applied the pruning methods, in addition to using the same knowledge as for experiment *E*1.

We created several sets of scores depending on the parameter threshold$_c$, which describes the correlation threshold used in the learning algorithm two criteria — accuracy and complexity — as outlined in Section 4 were used to define the quality of the scores.

The results are presented in the tables. Column threshold$_c$ specifies the correlation threshold, *MR* corresponds to the complexity measure Mean rules, attached with standard deviation. RBS describes the

No.	threshold$_c$	MR[a]	RBS[b] (*uncrt. cat*)[c]	AF[d]	SC[e]	ACC[f]
Experiment *E*0: no knowledge used, no pruning						
1	0.2	30.58 ± 15.93	2201.50 (1586.40)	391.60	3.52	0.94
2	0.3	20.75 ± 10.14	1493.90 (929.70)	348.90	3.26	0.93
3	0.4	14.82 ± 6.93	1067.20 (566.80)	293.70	2.99	0.91
4	0.5	10.93 ± 5.18	786.80 (334.00)	245.80	2.69	0.90
5	0.6	8.46 ± 3.71	609.20 (205.40)	208.00	2.38	0.84

Experiment *E*1: using partition class and abnormality knowledge, no pruning						
1	0.2	8.35 ± 5.09	600.90 (395.10)	177.10	2.59	0.89
2	0.3	5.97 ± 3.25	430.10 (241.90)	149.20	2.36	0.87
3	0.4	4.50 ± 2.09	323.90 (155.60)	130.20	2.10	0.85
4	0.5	3.37 ± 1.45	242.90 (90.70)	112.60	1.77	0.85
5	0.6	2.65 ± 1.03	190.50 (55.80)	99.90	1.53	0.79
Experiment *E*2: using partition class and abnormality knowledge and pruning						
1	0.2	3.60 ± 4.05	258.90 (53.10)	122.70	1.09	0.89
2	0.3	2.86 ± 2.19	205.60 (17.40)	99.50	0.99	0.87
3	0.4	2.42 ± 1.31	174.50 (6.20)	90.00	0.95	0.85
4	0.5	2.12 ± 0.96	152.70 (0.50)	82.50	0.92	0.85
5	0.6	1.87 ± 0.73	134.70 (0.00)	77.90	0.91	0.79

[a] Mean rules: average number of rules per diagnostic score.
[b] Rule base size: total number of rules (confirmation categories equally weighted, cf. Section 4).
[c] Total number of rules with an uncertain confirmation category.
[d] Applied attribute values: number of different attribute values used.
[e] Score categories: number of different confirmation categories used.
[f] Accuracy of the learned diagnostic scores.

complexity measure Rule base size with total number of rules in addition to the number of uncertain rules, i.e., rules with uncertain confirmation categories are given in parentheses. The column *SC* corresponds to the measure Score categories. Column AF shows the number of applied attribute values, i.e., the values of the measure Applied attribute values; in total the knowledge base contains about 2000 attribute values. Finally, we depict the accuracy of the rule base using the F-measure in column ACC.

The high values of the accuracy for low values of threshold_c and the large number of rules per diagnosis indicate overfitting of the learned knowledge. This is of course domain dependent, and therefore the expert needs to tune the threshold carefully. With greater values for threshold_c less rules are generated, since only strong dependencies are taken into account. If threshold_c is too high, i.e., if too many rules are pruned, this obviously degrades the accuracy of the learned scores. In our experiments *E*0− *E*2, this occurs for $\text{threshold}_c = 0.6$, for which the accuracy decreases significantly in comparison to threshold $= 0.5$. Furthermore, the number of rules per diagnosis (MR) is reduced by about a third without decreasing the accuracy (ACC) significantly for the experiments with values $\text{threshold}_c = 0.2$ and $\text{threshold}_c = 0.3$. Analogously, the number of applied attribute values (AF) is reduced with an increasing value of threshold_c but a decreasing accuracy.

Column SC indicates that the number of applied confirmation categories is reduced by an increased threshold_c, i.e., simpler scoring rules are learned. It is worth noticing that for experiment *E*2 the values of measure SC falls below 1.0; then, for some diagnoses no learned scoring rules are available. This is shown in experiments *E*2.2− *E*2.4, where this occurred for unfrequent diagnoses, where the accuracy did not decrease significantly.

Additionally, it is easy to see that an increasing threshold_c suppresses the creation of rules with uncertain confirmation categories, i.e., categories that do not establish a diagnosis categorically. Therefore, the complexity of the learned knowledge is decreased while its understandability is increased using higher thresholds with a lower number of uncertain rules.

In summary, applying background knowledge significantly reduces the number of learned scoring rules by removing irrelevant and noisy attribute values from the generated rule set. Obviously, pruning the profiles decreases the number of learned scoring rules even further without reducing the accuracy.

For experiment *E*0.5 (with $\text{threshold}_c = 0.6$) we obtain 8.46 ± 3.71 rules per diagnosis (MR). A comparable value is given for MR in experiment *E*1.1 (with $\text{threshold}_c = 0.2$), but an increased accuracy. Likewise, for experiment *E*2.1 (with $\text{threshold}_c = 0.2$) we have 3.6 ± 4.05 rules per diagnosis. Comparable values for experiments *E*1.4 and *E*1.3 (with $\text{threshold}_c = 0.5$ or $\text{threshold}_c = 0.4$) obtain worse accuracies. Thus, the experiments show that the background knowledge did not help to increase the accuracy of the learned rule base but helped to improve its understandability.

In practice, the user does not need to choose between the extreme positions of learning complex scores with a high accuracy and a set of simple scores with a lower accuracy. Instead, he is able to balance these two criteria and to find an appropriate trade-off dependent on the specific applica-

tion domain. In this particular experiment, the domain specialist may choose the bases of $E0.4$, $E1.1$ and $E2.1$ in order to obtain a comprehensible sized score for each diagnosis (about 5—10 rules for each score) and a reasonable accuracy of the entire rule base. Of course, the expert needs to choose a suitable value of the maximum number of rules contained in a diagnostic score depending on the application.

In our experience, a semi-automatic learning method for diagnostic scores is a promising direction for the initial construction of a knowledge base. However, so far the quality of the learned knowledge is still not comparable to a knowledge base that was manually built from scratch.

6. Conclusions

We presented a method for learning simple scoring rules applied for diagnostic tasks. Scoring rules are appealing because of their simplicity and practical relevance in medical decision making. The presented work investigates methods for learning small and simple sets of rules with an acceptable quality concerning the diagnostic accuracy. Background knowledge, like abnormalities for attribute values further helped to reduce the size of the rule base, without significantly decreasing the accuracy. The evaluation of the methods was implemented using a stratified 10-cross-validation applying 1340 cases from a real-life medical application.

The proposed learning method is primarily intended to support the domain specialists when building a diagnostic knowledge system from scratch. Then, a set of simple diagnostic scores defining the knowledge base can be learned, if a sufficient number of cases is available for the application domain. The learning method focuses on the understandability of the learned patterns in order to allow for a simple manual adaptation done by the domain specialist.

In the future, we are planning to improve the presented work by considering subgroups for score extraction, which can focus the scores on significant subspaces of the diagnoses' space. Furthermore, scores can be refined and simplified by aggregating attribute values into sub-concepts by learning sub-scores first, and using these sub-scores in combination with other attribute values for the final scores. Also, we want to consider combinations of related attribute values in the scoring rule construction step. Another promising approach is the automatic adaptation of thresholds for scores. In addition, we plan to integrate the complexity measures into general interestingness measures. We expect such ideas to be a good start for significantly improving the combination of accuracy and complexity of the scores.

References

[1] Ohmann C, Franke C, Yang Q. Clinical benefit of a diagnostic score for appendicitis: results of a prospective interventional study. Arch Surg 1999;134:993—6.

[2] Eich H-P, Ohmann C. Internet-based decision-support server for acute abdominal pain. Artif Intell Med 2000;20(1):23—36.

[3] Huettig M, Buscher G, Menzel T, Scheppach W, Puppe F, Buscher H-P. A diagnostic expert system for structured reports, quality assessment, and training of residents in sonography. Med Klin 2004;99(3):117—22.

[4] Buscher HP, Engler C, Fuhrer A, Kirschke S, Puppe F. HepatoConsult: a knowledge-based second opinion and documentation system. Artif Intell Med 2002;24:205—16.

[5] Puppe F. Knowledge reuse among diagnostic problem-solving methods in the shell-kit D3. Int J Hum Comput Stud 1998;49:627—49.

[6] Puppe F, Ziegler S, Martin U, Hupp J. Wissensbasierte Diagnosesysteme im Service-Support (diagnostic knowledge systems for service-support). Berlin: Springer Verlag, 2001.

[8] Paetz J. Finding optimal descision scores by evolutionary strategies. Artif Intell Med 2004;32:85—95.

[9] Puppe B, Ohmann C, Goos K, Puppe F, Mootz O. Evaluating four diagnostic methods with acute abdominal pain cases. Methods Inf Med 1995;34(4):369—70.

[10] Miller R, Pople HE, Myers J. Internist-1, an experimental computer-based diagnostic consultant for general internal medicine. N Engl J Med 1982;307:468—76.

[11] Pople HE. Heuristic methods for imposing structure on ill-structured problems: the structuring of medical diagnostics. In: Szolovits P, editor. Artificial Intelligence in Medicine. Boulder, Colorado: AAAS, Westview Press, 1982.

[12] Middleton B, Shwe M, Heckerman D, Henrion M, Horvitz E, Lehmann H. Probabilistic diagnosis using a reformulation of the INTERNIST-1/QMR knowledge base. II. Evaluation of diagnostic performance.. Methods Inf Med 1991;4(30): 256—67.

[13] Fronhöfer B, Schramm M. Probabilistic aspects of score systems. Int J Uncertainty Fuzziness Knowledge-Based Syst 2003;11(Suppl.):51—73.

[14] Schramm M, Ertel W. Reasoning with probabilities and maximum entropy: the system PIT and its application in LEXMED. In: Inderfurth K, Schwdiauer G, Domschke W, Juhnke F, Kleinschmidt P, Wscher G, editors. Operations research proceeedings. Berlin: Springer Verlag; 1999. p. 274—80.

[15] Adlassnig K-P, Kolarz G. Representation and semiautomatic acquisition of medical knowledge in CADIAG-1 and CADIAG-2. Comput Biomed Res 1986;19:63—79.

[16] Adlassnig K-P, Kolarz G, Scheithauer W, Grabner H. Approach to a hospital-based application of a medical expert system. Med Inf 1986;11:205—23.

[17] Neumann M, Baumeister J, Liess M, Schulz R. An expert system to estimate the pesticide contamination of small streams using Benthic macroinvertebrates as bioindicators, Part 2. Ecol Indicators 2003;2(4):391—401.

[18] Baumeister J, Atzmueller M, Puppe F. Inductive learning for case-based diagnosis with multiple faults. In: Craw S, Preece A, editors. Advances in case-based reasoning, vol. 2416 of LNAI. Berlin: Springer Verlag; 2002. p. 28—42. Proceedings of

the sixth European conference on case-based reasoning (ECCBR-2002).

[19] Tuzhilin A, Klösgen. Zytkow: handbook of data mining and knowledge discovery. New York: Oxford University Press; 2002. Ch. 19.2.2: Usefulness, novelty, and integration of interestingness measures.

[20] Freitas AA. On rule interestingness measures. Knowledge-Based Syst 1999;12:309—25.

[21] Mitchell T. Machine learning. Boston, Massachusetts: McGraw-Hill Comp, 1997.

[22] Yen S-J, Chen AL. An efficient algorithm for deriving compact rules from databases. In: Ling TW, Masunaga Y, editors. Proceedings of the fourth international conference on database systems for advanced applications. Singapore: World Scientific; 1995. p. 364—71.