Wissensmanagement in der Computational Intelligence:

Systematisierung der Beschreibung von Problemen, Methoden und Methodeneinsätzen

Klinkenberg, Ralf
Lehrstuhl für Künstliche Intelligenz (FB 4, LS 8)
Universität Dortmund, 44221 Dortmund
Tel.: 0231/755-5103, Fax: 0231/755-5105
E-Mail: klinkenberg@ls8.cs.uni-dortmund.de

Slawinski, Timo
Lehrstuhl für Elektrische Steuerung und Regelung
Universität Dortmund, 44221 Dortmund
Tel.: 0231/755-4621, Fax: 0231/755-4524
E-Mail: timo@slawinskis.de

Zusammenfassung

In der *Computational Intelligence* (CI) wurden und werden zahlreiche Methoden aus den Bereichen der evolutionären Algorithmen, der neuronalen Netze und der Fuzzy Logik entwickelt. Eine wesentliche Schwierigkeit aus Anwendersicht besteht in der Auswahl von Methoden und deren Anpassung an die konkrete Problemstellung. Wünschenswert wäre also eine Taxonomie, wann und warum welche CI-Methode einzusetzen ist. Hierbei ist auch eine Abgrenzung zu anderen, insbesondere klassischen Verfahren vorzunehmen. Wichtige Voraussetzung dafür sind klare begriffliche Definitionen und ein systematischer Methodenvergleich auf breiter Basis.

Mit dem langfristigen Ziel ein Wissensmanagement–System aufzubauen ist im Sonderforschungsbereich 531 (CI) ein Ansatz für die formalisierte Beschreibung von Problemen, Methoden, Analysemethoden, Methodeneinsätzen und Prozessketten entwickelt und umgesetzt worden. In diesem Beitrag wird zunächst das diesem Ansatz zugrunde liegende Konzept vorgestellt. Zur Demonstration der Tragfähigkeit des Konzepts wird dann exemplarisch für zwei CI–Methoden beschrieben, wie bei ihrer Parametrierung systematisch vorgegangen werden kann. Anhand von sechs Benchmarkproblemen wird für eine der beiden CI-Methoden das Ergebnis dieses systematischen Vorgehens mit dem bisher besten Ergebnis mit dieser Methode verglichen, um die Qualität der Resultate abzuschätzen, die ein nicht so versierter Benutzer mit der systematischen Vorgehensweise erzielen könnte.

1 Einleitung

Der Sonderforschungsbereich 531 „Design und Management komplexer technischer Systeme mit Methoden der Computational Intelligence“ (SFB CI) umfaßt Forschung in den Bereichen Fuzzy-Systeme, neuronale Netze und evolutionäre Algorithmen. Zu seinen Zielen gehört die Weiterentwicklung der Grundlagen der CI, die Entwicklung methodischer Ansätze für Anwendungen der CI, die Erschließung von Synergieeffekten durch Kombination von CI-Methoden, die Abgrenzung der Anwendungsbereiche auch durch Methodenvergleiche und die Validierung der Konzepte anhand von prototypischen Anwendungen.

In der Arbeitsgruppe Systematisierung (AG Sys) des SFB CI soll durch eine formalisierte Beschreibung von Problemen, Methoden, Analysemethoden, Methodeneinsätzen und Prozessketten die Grundlage für einen systematischen Methodenvergleich geschaffen und für eine bessere Vernetzung der Teilprojekte innerhalb des SFBs gesorgt werden. Darauf aufbauend soll untersucht werden, ob eine Taxonomie bezüglich der Methodenwahl und des Experimentdesigns abgeleitet werden kann. Diese interdisziplinäre Arbeitsgruppe setzt sich sowohl aus theoretischen und praktischen Informatikern als auch Wissenschaftlern verschiedener ingenieurswissenschaftlicher Anwendungsfelder von CI-Methoden aus Chemietechnik, Elektrotechnik und Maschinenbau zusammen.

Dieser Beitrag ist wie folgt aufgebaut: Im ersten Teil wird sowohl das begriffliche Konzept für das Wissensmanagement–System (WMS) als auch dessen technische Umsetzung vorgestellt (Abschnitt 2). Zentral dabei ist die Verwendung einer in XML beschriebenen Ontologie, die die kompakte Darstellung der Struktur der Formulare erlaubt, über die Wissen über CI-Methoden und ihren Einsatz eingetragen und dargestellt werden kann. Auf diese Weise wird die Struktur der Formulare, d.h. die begriffliche Struktur (Ontologie), von den Formular-Instanzen, d.h. den Inhalten, und von ihrer Darstellung getrennt. Die Generierung und Auswertung der HTML-Formulare sowie die Verwaltung der Daten erfolgt automatisch auf der Basis der Ontologie. Im zweiten Teil wird auf den prototypischen Einsatz des Wissenmanagement–Systems eingegangen. Gegenstand der Untersuchungen ist eine systematische Vorgehensweise für die problemklassenabhängige Parametrierung der Support Vector Machine (SVM, Stützvektormethode) und des Fuzzy–ROSA–Verfahrens (Abschnitte 3.1 und 3.2). Für das Fuzzy–ROSA–Verfahren werden die damit erzielbaren Resultate anhand von sechs Benchmarkproblemen in Abschnitt 3.3 mit den bisher besten auf diesen Problemen erzielten Ergebnissen verglichen.

2 Wissenserfassung und -darstellung mit Hilfe aus einer Ontologie generierter Formulare

Die *Computational Intelligence* (CI) umfaßt viele verschiedene Methoden aus den Bereichen der evolutionären Algorithmen, der neuronalen Netze und der Fuzzy Logik. Eine wesentliche Schwierigkeit aus Anwendersicht besteht in der Auswahl von Methoden und deren Anpassung an eine konkrete Problemstellung. Wünschenswert wäre also eine Taxonomie, wann und warum welche CI-Methode wie einzusetzen ist. Hierbei ist auch eine Abgrenzung zu anderen, insbesondere klassischen Verfahren vorzunehmen. Wichtige Voraussetzung dafür sind klare begriffliche Definitionen und ein systematischer Methodenvergleich auf breiter Basis.

Mit dem langfristigen Ziel ein Wissensmanagement–System aufzubauen ist im Sonderforschungsbereich 531 (CI) ein Ansatz für die formalisierte Beschreibung von Problemen, Methoden, Analysemethoden, Methodeneinsätzen und Prozessketten entwickelt und umgesetzt worden. Nach einer kurzen Beschreibung des Vorgehens der interdisziplinären Arbeitsgruppe Systematisierung (AG Sys) des SFB 531 (CI) in Abschnitt 2.1, wird in Abschnitt 2.2 der von dieser Arbeitsgruppe gewählte Ansatz zur Trennung der konkreten zu beschreibenden Objekte (inhaltliche Ebene) von der zugrunde liegenden Struktur (konzeptuelle Ebene) und von der Darstellung der Struktur und der Inhalte (Darstellungsebene) beschrieben, der auf der Verwendung einer Ontologie, d. h. einer Begriffshierarchie zur kompakten Darstellung der Formularstrukturen beruht. Die technische Realisierung dieses Ansatzes und der automatischen Formulargenerierung mit Hilfe dieser Ontologie wird in Abschnitt 2.3 dargestellt.

2.1 Ziele und Vorgehen der Arbeitsgruppe Systematisierung im SFB 531

In der Arbeitsgruppe Systematisierung (AG Sys) des SFB CI soll durch eine formalisierte Beschreibung von Problemen, Methoden, Analysemethoden, Methodeneinsätzen und Prozessketten die Grundlage für einen systematischen Methodenvergleich geschaffen und für eine bessere Vernetzung der Teilprojekte innerhalb des SFBs gesorgt werden. Darauf aufbauend soll untersucht werden, ob eine Taxonomie bezüglich der Methodenwahl und des Experimentdesigns abgeleitet werden kann. Diese interdisziplinäre Arbeitsgruppe setzt sich sowohl aus theoretischen und anwendungsorientierten Informatikern als auch Wissenschaftlern verschiedener ingenieurswissenschafltlicher Anwendungsfelder von CI-Methoden aus Chemietechnik, Elektrotechnik und Maschinenbau zusammen.

Bevor entschieden wurde, wie später technisch die formalisierte Beschreibung der zu erfassenden und zu verwaltetenden CI-Konzepte unterstützt werden sollte, fand eine intensive Diskussion über die zu beschreibenden Konzepte sowie ihre zu beschreibenden Eigenschaften und Beziehungen zueinander statt. Relativ früh bestand Konsens darüber, dass *Probleme*, *Methoden* und *Methodeneinsätze* zu den zentralen Konzepten gehören sollten und beispielsweise jeweils mit Hilfe eines Formulars leicht beschreibbar sein sollten. Unter einem *Problem* ist dabei eine zu lösende anwendungsorientierte oder theoretische Aufgabe zu verstehen. Eine *Methode* auf bestimmte Probleme anwendbar und ein *Methodeneinsatz* beschreibt eine konkrete Anwendung einer Methode auf ein Problem (siehe auch Abbildung 1).

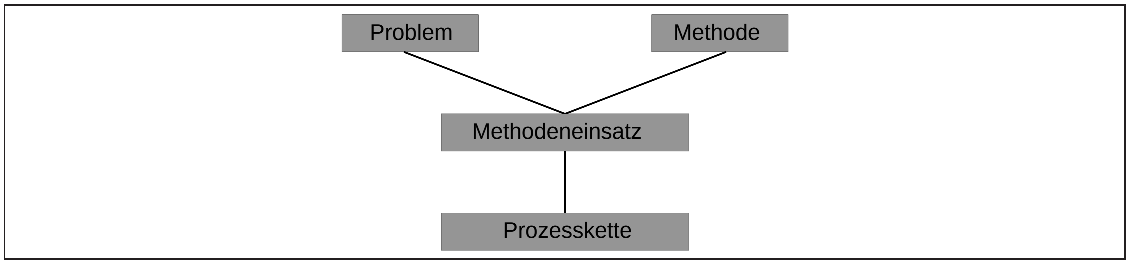

Abbildung 1: Ausgangspunkt der Ontologie: Formulare/Konzepte zur Beschreibung von Methoden, Problemen, Methodeneinsätzen und Prozessketten.

Oft ist Problem aber nicht durch eine einzige Methode lösbar, sondern erst durch den kombinierten Einsatz mehrerer Methoden. Eine *Prozesskette* setzt sich aus mehreren Methodeneinsätzen zusammen, d. h. ein komplexes Problem kann durch Zerlegung in Teilaufgaben und die Anwendung mehrer Methoden gelöst werden. Ein erste Methode könnte beispielsweise gegebene Daten in eine für ein zweites Verfahren günstige Repräsentation transformieren, auf der dann das zweite Verfahren aufsetzen könnte. Die erste Methode könnte beispielsweise der Merkmale generieren, tranformieren und/oder auswählen, auf denen aufbauend das zweite Verfahren z. B. eine Lernaufgabe löst.

Neben dieser sehr anwendungsorientierten Sichtweise von Problemen und Methoden sind aus der Sicht des SFB CI und auch im Hinblick auf das Ziel einer Taxonomie auch formale Eigenschaften von Problemen und Methoden von Interesse, die z. B. die Lösbarkeit von bestimmten Problemen mit bestimmten Methoden oder die Komplexität solcher Lösungen beschreiben. Auch solche Eigenschaften sollten in einem CI-Wissensmanagementsystem erfaßt werden können. Darüberhinaus ist es nicht nur sinnvoll, diese formaleren Eigenschaften zu erfassen, sondern auch die zugehörigen *Analysemethoden*, mit denen sich solche Eigenschaften beweisen lassen. Probleme, Methoden, Analysemethoden, Methodeneinsätze und Prozeßketten sind also die zentralen Konzepte, für die sich die Arbeitsgruppe für ihr Wissensmanagementsystem entschieden hat. Wie im nächsten Abschnitt deutlich wird, ist es jedoch hilfreich, weitere Konzepte hinzuzunehmen und eine relational vernetzte Darstellung zu wählen, die einerseits hilft, Redundanzen bei inhaltlichen Einträgen zu vermeiden und andererseits die spätere Navigation durch das dargestellte Wissen erleichtert.

2.2 Verwendung einer Ontologie zur kompakten Darstellung der Formularstrukturen

Instanzen der beschriebenen Konzepte Problem, Methode, Analysemethode, Methodeneinsatz und Prozeßkette könnte man beispielsweise über Formulare beschreiben, die neben dem einem Namen der Instanz und einer Beschreibung weitere charakteristische Merkmale und hilfreiche Informationen enthalten könnten. Will man beispielsweise zu jeder Instanz festhalten, in welchem Projekt sie (zuerst) aufgetreten oder beschrieben worden ist, und wer geeignete Ansprechpersonen sein könnten, was ähnliche Probleme bzw. Methoden sind etc., stellt man schnell fest, dass viele Dinge wie z.B. Kontaktpersonen mit z.B. ihren Telefonnummern und E-Mail-Adressen häufig einzutragen sind. Wählt man für solche Dinge eigene Konzepte wie z. B. *Person*, deren Instanzen über Relationen wie beispielsweise *ist-Kontaktperson-für* oder *ist-Mitarbeiter-in* mit Instanzen anderer Konzepte in Beziehung gesetzt werden können, reicht ein einfacher Eintrag der Daten beim ersten Auftreten und eine kurze Referenz, d. h. eine einfache Relationsinstanz, bei weiteren Vorkommen.

Ein weiterer Vorteil dieser Art der Redundanzvermeidung ist die Möglichkeit, über ein Wissensmanagementsystem die Navigation entlang solcher Relationen zu erlauben, mit deren Hilfe man beispielsweise Fragen wie die folgenden relativ leicht beantworten kann:

- Welche Probleme sind mit dieser Methode schon bearbeitet worden?
- *Oder umgekehrt:*
  Welche Methoden sind auf dieses Problem schon angewandt worden?

- Welche Methoden sind ähnlich?
  Welche sind auf gleichartige Probleme angewandt worden?
- Welche Eigenschaften hat dieses Problem bzw. diese Methode?
  Mit welchen Analysemethoden kann man das zeigen?

Durch diese Art des Vorgehens entsteht um die zentralen fünf Konzepte schnell eine Menge weiterer Konzepte, die miteinander über Relationen verbunden sind und deren Instanzen durch bestimmte charakteristische Eigenschaften beschrieben werden können. Um die Wartbarkeit der Wissensbasis zu erleichtern, empfiehlt es sich, die Ebene der Konzepte und die Ebene der Instanzen zu trennen. Die konzeptuelle Ebene beschreibt die Strukturen des zu beschreibenden Wissens, während die Instanzen die konkreten Inhalte, d. h. die Beschreibung konkreter Wissensinhalte enthalten. Die strukturelle Information der konzeptuellen Ebene läßt sich kompakt in einer Ontologie darstellen, die Konzepte sowie ihre Eigenschaften und Relationen losgelöst von konkreten Instanzen beschreibt. Die Ontologie wird typischerweise deutlich seltener geändert als die in das Wissensmanagementsystem einzutragenden Instanzen, denn sie gibt die zugrunde liegende Struktur vor, in deren Rahmen sich die Instanzen möglichst gut beschreiben lassen sollen, und diese Beschreibungen möchte man ja nicht unnötig oft einer neuen Struktur anpassen müssen. Allein aus diesem Grund schon ist eine klare Trennung von Struktur und Inhalt wünschenswert.

Ebenso wie es sinnvoll ist, Struktur und Inhalt voneinander zu trennen, ist es sinnvoll, die Darstellung der Inhalte und der Strukturen von eben diesen zu trennen. Auf diese Weise kann man auch die Form der Darstellung unabhängig von den Inhalten und Strukturen verwalten und ändern und ermöglicht eine möglichst kompakte Darstellung von Strukturen und Inhalten. Wie im nächsten Abschnitt gezeigt wird, kann man die Darstellung sogar weitgehend automatisiert aus einer geeignet repräsentierten Ontologie und entsprechenden Inhalten generieren.

2.3 Technische Realisierung und automatische Formulargenerierung

Der hier vorgestellt Ansatz zur technischen Realisierung der Formulare basiert auf zwei XML-Dateien und einer Software namens Information Layer[1], die von Stefan Haustein, einem Mitarbeiter am Lehrstuhl für künstliche Intelligenz der Universität Dortmund, im Rahmen des EU-Projektes COMRIS entstanden ist, die aber auch darüber hinaus weiter entwickelt wird. Die erste der beiden XML-Dateien enthält die oben beschriebene Ontologie (`ontology.xml`) während die zweite die Daten aller eingetragenen Instanzen enthält (`instances.xml`). Aus diesen beiden XML-Dateien generiert der Information Layer automatisch alle HTML-Seiten zur Eingabe neuer Formulare und zur Ansicht oder Überarbeitung bestehender Formulare (siehe Abbildung 2). Deswegen müssen keine Skripten zur Auswertung irgendwelcher HTML-Formulare geschrieben oder gewartet werden. Man muß nur einmal die Ontologie (Struktur) der Formulare (Konzepte) beschreiben. Auch die XML-Datei mit den Instanzen wird automatisch vom Information Layer erzeugt und verwaltet. Der Systembenutzer trägt die Daten einfach in die automatisch generierten HTML-Formulare ein, die dann automatisch vom Server ausgewertet werden.

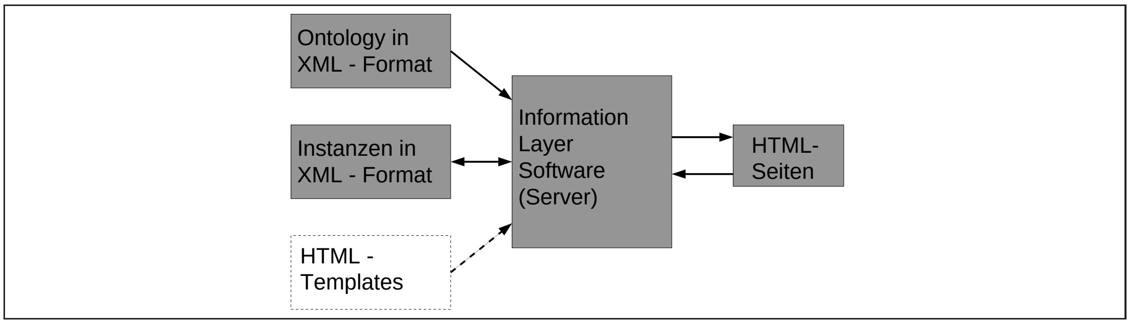

Abbildung 2: Information Server: automatische Generierung von HTML-Seiten aus der Ontologie, den Instanzen und (optional) XHTML-Templates.

Die Struktur der Formulare (Konzepte) sowie die in ihnen verwendeten Begriffe (Konzepte, Attribute) und Querbeziehungen (Relationen) werden kompakt in einer Ontologie (Begriffshierarchie) in XML-Format dargestellt. Das XML-Format ist ein verständliches, leicht von Hand editierbares Format für Textdateien. Es wird zwischen Konzepten (z.B. die einzelnen Formulartypen Problem, Methode, etc.) und ihren Instanzen (ausgefüllte Exemplare der Formulare, d.h. konkrete Beschreibungen von Problem- und Methodeninstanzen, etc.) unterschieden. Die Ontologie beschreibt die einzelnen Konzepte, d.h. ihre Namen, Attribute und Beziehungen.

Nachdem man die Ontologie in einer XML-Datei beschrieben hat, kann man anfangen, Instanzen der in der Ontologie spezifizierten Konzepte über vom Information Layer generierte HTML-Formulare einzutragen, zu ändern oder zu löschen. Die Instanz-Datei in XML-Format wird automatisch verwaltet und es wird auch automatisch für die Konsistenz bi-direktionaler Abbildungen gesorgt, indem beim Eintragen (bzw. Löschen) einer Instanz einer solchen Relation automatisch auch die zugehörige Instanz der inversen Relation angelegt (bzw. gelöscht) wird.

Ebenso leicht, wie man über die generierten HTML-Formulare Inhalte eingeben und ändern kann, kann man über die Relationsstruktur, die durch Hypertext-Links in den HTML-Seiten automatisch navigierbar gemacht wird, mit einem einfachen WWW-Browser durch die spezifierte Wissensstruktur und die eingegebenen Instanzen browsen und so z. B. Fragen wie die in Abschnitt 2.2 skizzierten beantworten.

Über XHTML-Templates kann die Gestaltung und der Umfang der dargestellten Inhalt der vom Information Layer generierten HTML-Seiten nach eigenen Wünschen gestaltet werden (siehe auch Abbildung 2).

Änderungen der Ontologie wie z.B. das Einführen eines neuen Konzeptes, eines neuen Attributes oder einer neuen Relation zwischen Konzepten sind in der XML-Datei mit der Ontologie einfach durchführbar und werden sofort beim nächsten Neustart des Servers berücksichtigt. Bei der Entfernung von Konzepten, Attributen oder Relationen aus der Ontologie sind gegebenenfalls die entsprechenden Instanzen aus der Instanzendatei zu entfernen. Weitergehende Anpassungen sind nicht notwendig.

Der Information Layer unterstützt eine Reihe von Attributtypen für die Beschreibung von Instanzeigenschaften. Neben einzeiligen und mehrzeiligen Textfeldern sind auch Boolesche Felder möglich. Attribute in der Form einzeiliger Textfelder, die als Wert nur eine URL enthalten, werden automatisch als Link interpretiert und dargestellt. Attribute in der Form von Textfeldern können in der Ontologie als HTML-Felder gekennzeichnet werden. Dann können HTML-Elemente wie Links, Listen und Tabellen in diesen Feldern benutzt werden. Mit Hilfe von Unicode-Zeichen sind auch mathematische Formeln darstellbar.

Bei Relationen zwischen Konzepten erstellt der Information Layer automatisch Auswahllisten mit den akutell vorhandenen Instanzen der jeweils in die gegebene Relation passenden Konzepte. Außerdem erlaubt er bei Relationen auch stets die Erzeugung neuer passender Instanzen. Relationen können als 1:1, 1:n oder m:n definiert werden.

Während dieser Abschnitt des Beitrags sich mit den Konzepten der Wissensdarstellung und ihrer technischen Umsetzung beschäftigt hat, beschreiben die folgenden Abschnitte konkrete Methoden und Methodeneinsätze, die mit Hilfe dieses Systems beschreibbar sind. Die entsprechenden Beschreibungen sind dann z. B. zum Wissenstransfer von Methoden- und Problemexperten an nicht so versierte Personen verwendbar, die sich die für sie in Frage kommenden (ähnlichen) Probleme oder Methoden per Navigation heraussuchen können und anhand der Methoden(einsatz)beschreibungen systematische Vorgehenshinweise für den von ihnen gewünschten Methodeneinsatz erhalten.

3 Prototypische Untersuchungen zum systematischen Einsatz von CI-Methoden

Eine Grundvoraussetzung für den breiten Einsatz von CI–Methoden sind systematische Lösungsstrategien, die den Anwender dabei unterstützen, ein für die jeweilige Aufgabenstellung geeignetes Verfahren auszuwählen und gegebenenfalls anzupassen. Ungeachtet dessen sind in der Literatur diesbezüglich nur sehr wenige Untersuchungen und Vorschläge zu finden. Ohne eine systematisierte Vorgehensweise bestehen aber, insbesondere für den unerfahrenen Anwender, meist erhebliche Schwierigkeiten, den teilweise sehr unterschiedlichen Anforderungen komplexer Aufgabenstellungen gerecht zu werden. In dem hier vorgestellten prototypischen Einsatz des Wissenmanagements–Systems wird für zwei CI–Methoden beschrieben, wie sie systematisch parametrisiert werden können. Für eines der beiden Verfahren wird untersucht, welche Resultate mit dieser systematischen Vorgehensweise im Vergleich zu Einstellungen, die Experten vorgenommen haben, erzielt werden können. Dafür werden sechs bekannte Benchmarkprobleme herangezogen, die auch eine Einordnung der erzielten Resultate in Bezug auf in der Literatur verfügbare Ergebenisse erlauben.

Im Folgenden werden zunächst Support Vector Machines (Abschnitt 3.1) und das Fuzzy–ROSA–Verfahren (Abschnitt 3.2) kurz beschrieben. Dabei wird auch die jeweils verwendete systematische Vorgehensweise vorgestellt. Abschließend wird in Abschnitt 3.3 auf die mit dem Fuzzy–ROSA–Verfahren hiermit erzielten Ergebnisse eingegangen und eine zusammenfassende Bewertung vorgenommen.

3.1 Support Vector Machines

Support Vector Machines (SVMs, Stützvektormethode) [2] sind eine derzeit ein sehr aktiver Forschungsbereich im maschinellen Lernen. Sie sind aus der statistischen Lerntheorie entstanden und wurden bereits in vielen Anwendungsfeldern erfolgreich eingesetzt. SVMs gehören zu einer breiteren Kategorie von Lernverfahren, die das Konzept der Kernfunktionsersetzung nutzen, um Lernaufgaben durch eine implizite Abbildung in einen höherdimensionalen Raum besser lösbar zu machen. SVMs haben einige ansprechende Eigenschaften im Hinblick auf das maschinelle Lernen. Beispielsweise involviert die klassische SVM-Lernaufgabe die Lösung eines quadratischen Optimierungsproblems. Es gibt nur eine Lösung und diese kann durch den Einsatz effizienter Algorithmen aus der Optimierungstheorie gefunden werden. Außerdem muß der Anwender von SVMs beim Einsatz eines Modellselektionsansatzes nur sehr wenige, falls überhaupt irgendwelche Parameter setzen. Nicht zuletzt funktionieren SVMs in vielen praktischen Anwendungen sehr gut.

3.1.1 Grundlagen der Support Vector Machines

Support Vector Machines [2] (SVMs, Stützvektormethode) basieren auf dem Prinzip der strukturellen Risikominimierung [2] aus der statistischen Lerntheorie. In ihrer einfachen Grundform lernen SVMs lineare Entscheidungsregeln

$$h(\vec{x}) = sign\{\vec{w} \cdot \vec{x} + b\} = \begin{cases} +1, & \text{if } \vec{w} \cdot \vec{x} + b > 0 \\ -1, & \text{else} \end{cases} \tag{1}$$

die sich durch einen Gewichtsvektor $\vec{w}$ und einen Treshold b beschreiben lassen. Die Idee der strukturellen Risikominimierung ist es, eine Hypothese h zu finden, für die die kleinste Fehlerwahrscheinlichkeit garantiert werden kann. Für SVMs zeigt [2], dass dieses Ziel durch das Finden der Hyperebene erreicht werden kann, die die Beispiele der beiden zu unterscheidenden Klassen mit maximalen Abstand (*(soft) margin*) trennt.[1] Die Berechnung dieser Hyperebene ist äquivalent zur Lösung des folgenden Optimierungsproblems.

Optimierungsproblem 1 (SVM (primal))

$$\textit{Minimiere:} \quad V(\vec{w}, b, \vec{\xi}) = \frac{1}{2}\vec{w} \cdot \vec{w} + C \sum_{i=1}^{n} \xi_i \tag{2}$$

$$\textit{unter den Nebenbedingungen:} \quad \forall_{i=1}^{n} : y_i[\vec{w} \cdot \vec{x}_i + b] \geq 1 - \xi_i \tag{3}$$

$$\forall_{i=1}^{n} : \xi_i > 0 \tag{4}$$

Für die gegebenen Trainingsbeispiele $\vec{z}_i = (\vec{x}_i, y_i)$, die jeweils aus einem Attributwertevektor $\vec{x}_i$ und einem Klassenlabel y_i bestehen, müssen die Parameter $\vec{w}$ und b der SVM-Entscheidungsregel gefunden werden, die $V(\vec{w}, b, \vec{\xi})$ minimieren. In diesem Optimierungsproblem ist die euklidische Länge $||\vec{w}||$ des Gewichtsvektors umgekehrt proportional zum Abstand der nächsten Beispiele zur Hyperebene der Entscheidungsregel (*soft margin*). Die Randbedingungen (3) erfordern, dass alle

[1] Siehe [3] für eine Einführung in SVMs für Klassifikationsprobleme und [4] für eine Einführung in SVMs für Regressions-/Approximationsprobleme.

Trainingsbeispiele bis auf eine gewisse Ungenauigkeit ξ_i, d. h. bis auf einen gewissen kleinen Trainingsfehler korrekt klassifiziert werden. Wenn ein Trainingsbeispiel auf der „falschen“ Seite der Hyperebene liegt, ist das entsprechende ξ_i größer oder gleich 1. Deswegen ist $\sum_{i=1}^{n} \xi_i$ eine obere Schranke für die Anzahl der Trainingsfehler. Der Faktor C in (2) wird Kapazitätskonstante genannt und ist ein Parameter zur Einstellung des Trade-Offs zwischen Trainingsfehler und Modellkomplexität (siehe auch Abschnitt 3.1.2).

Wegen des Rechenaufwands ist es günstiger das Wolfe-Dual [5] des Optimierungsproblems 1 zu lösen anstatt das Optimierungsproblem 1 direkt [2].

Optimierungsproblem 2 (SVM (dual))

$$\textit{Minimiere: } W(\vec{\alpha}) = -\sum_{i=1}^{n} \alpha_i + \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} y_i y_j \alpha_i \alpha_j (\vec{x}_i \cdot \vec{x}_j) \tag{5}$$

$$\textit{unter den Nebenbedingungen: } \sum_{i=1}^{n} y_i \alpha_i = 0 \tag{6}$$

$$\forall_{i=1}^{n} : 0 \leq \alpha_i \leq C \tag{7}$$

Für die gegebenen Trainingsbeispiele $\vec{z}_i = (\vec{x}_i, y_i)$ sind die Parameter α_i zu finden, die $W(\vec{\alpha})$ minimieren. In dieser Arbeitet wird die SVM-Implementation *mySVM* [6] für die Berechnung der Lösung dieses Optimierungsproblems benutzt.[2] Stützvektoren (*Support Vectors*) sind die Trainingsbeispiele $\vec{x}_i$ mit $\alpha_i > 0$ bei der Lösung. Aus der Lösung des Optimierungsproblems 2 kann die SVM-Entscheidungsregel wie folgt berechnet werden

$$\vec{w} \cdot \vec{x} = \sum_{i=1}^{n} \alpha_i y_i (\vec{x}_i \cdot \vec{x}) \quad \text{and} \quad b = y_{usv} - \vec{w} \cdot \vec{x}_{usv} \tag{8}$$

Das Trainingsbeispiel $(\vec{x}_{usv}, y_{usv})$ für die Berechnung von b muß ein Stützvektor mit $\alpha_{usv} < C$ sein. Schließlich erhält man die Trainingsfehler ξ_i als $\xi_i = max(1 - y_i [\vec{w} \cdot \vec{x}_i + b], 0)$.

Sowohl für das Lösen des Optimierungsproblems 2 als auch für das Anwenden der gelernten Entscheidungsregel reicht es aus, innere Produkte (Skalarprodukte) zwischen Attributwertevektoren zu berechnen. Durch das Ausnutzen dieser Eigenschaft haben Boser et al. den Einsatz von Kernfunktionen $\mathcal{K}(\vec{x}_i, \vec{x}_j)$ für das Lernen nichtlinearer Entscheidungsregeln eingeführt. Abhängig von der Art der Kernfunktion lernen SVMs anstelle von linearen Klassifikatoren[3] Polynomklassifikatoren, auf Radialen Basisfunktionen (RBF) basierende Klassifikatoren oder zwei-lagige sigmoide neuronale Nezte. Solche Kernfunktionen berechnen ein inneres Produkt in einem Merkmalsraum und ersetzen das innere Produkt (Skalarprodukt) in den obigen Formeln.

[2] *mySVM* ist verfügbar unter `http://www-ai.cs.uni-dortmund.de/SOFTWARE/MYSVM/` und kann sowohl Klassifikations- als auch Approximations-/Regressionsprobleme lösen.

[3] Klassifikatoren bei SVMs für Klassifikationsaufgaben bzw. Approximationen bei SVMs für Regressionsaufgaben.

3.1.2 Systematisches Vorgehen zur Parametrisierung von SVMs

Offensichtlich ist die Wahl der *Kernfunktion* und ihrer Parameter von entscheidender Bedeutung für die Parametrisierung einer SVM, da die Kernfunktion die Art des Hypothesenraums und damit die Ausdruckskraft der durch die SVM lernbaren Modelle vorgibt. In manchen Anwendungen liefern bereits lineare Kernel sehr gute Ergebnisse, so z. B. in der Klassifikation von Texten [7], wo kompliziertere Kernfunktionen nicht signifikant besser oder schlechter abschneiden, so dass man dort in der Regel einfach lineare Kernfunktionen einsetzt. Sehr oft werden die besten oder zumindest sehr gute Ergebnisse mit radialen Basisfunktionen (RBF) als Kernfunktionen erreicht. [8] beschreibt beispielsweise Experimente mit verschiedenen Kernfunktionen für Zeitreihendaten, bei denen RBF-Kernfunktionen auch im Vergleich mit spezieller angepaßten Kernfunktionen meist sehr gut abschneiden. Deswegen reicht es in den meisten Anwendungen, sich bei der Variation der Parameter für das Finden einer guten Parametrisierung auf lineare und RBF-Kernfunktionen zu beschränken, weil vom Einsatz von beispielsweise polynomiellen oder sigmoiden Kernfunktionen (in der Regel) keine (signifikant) besseren Ergebnisse zu erwarten sind.

Manche *Kernfunktionen* haben weitere *Parameter*, über die sich die Form der Kernfunktion genauer spezifizieren läßt. Bei RBF-Kernfunktionen ist dies z. B. der Parameter γ, der die Breite der radialen Basisfunktionen angibt ($\mathcal{K}(x_i, x_j) = e^{-\gamma||x_i - x_j||^2}$). Bei polynomiellen Kernfunktionen sind dies beispielsweise der Grad des Polynoms sowie ein Skalierungsfaktor.

Ein weiterer wichtiger Parameter beim Training von SVMs ist die *Kapazitätskonstante* C, über die man den Trade-Off zwischen Trainingsfehler und Modellkomplexität einstellen kann. Ein großer Wert von C erlaubt einzelnen Trainingsbeispielen einen größeren Einfluss auf die Lage der Hyperebene und führt damit zu einer stärkeren Anpassung an die Trainingsdaten, während ein kleinerer Wert von C zu einer glatteren Hyperebene bzw. Trennfunktion führt, also zu einem stärker generalisierenden Modell. Durch geeignete Wahl von C läßt sich also eine Überanpassung an die Trainingsdaten (*Overfitting*) vermeiden, die durch das Lernen eines komplexen Modells zwar einen geringen Trainingsfehler erreichen würde, aber einen hohen Fehler auf ungesehenen (Test)beispielen zur Folge hätte.

Bei manchen Klassifikationsanwendungen, insbesondere bei solchen mit sehr ungleicher Klassenverteilung, kann es sinnvoll sein, durch unterschiedliche *Gewichte (Fehlklassifikationskosten)* für positive und negative Beispiele die ungleiche Klassenverteilung auszugleichen.

Um für ein gegebenes Klassifikations- oder Regressionsproblem einen guten Parametersatz zu finden, bietet sich der Einsatz einer automatischen Modellselektion an. So kann mit Hilfe einer n-fachen Kreuzvalidierung auf den Trainingsdaten, die hierzu zufällig in n möglichst gleich große Teile aufgeteilt werden, in n Trainingsläufen ein Modell auf jeweils $n-1$ dieser Teile gelernt und auf dem jeweils verbleibendem Teil evaluiert werden. Führt man z.B. für jede zu betrachtende Kombination von Kernelfunktion (z.B. linear und RBF), ihrer Parameter (bei RBF z.B. $\gamma \in \{10^{-2}, 10^{-1}, 1, 10^1, 10^2\}$) und der Kapazitätskonstante C (z.B. $C \in \{10^{-2}, 10^{-1}, 1, 10^1, 10^2\}$) aus, so kann man automatisch die Parameterkombination mit dem kleinsten durchschnittlichen Fehler bei dieser Kreuzvalidierung auswählen lassen. Mit dieser Parameterkombination trainiert man dann eine SVM auf den gesamten Trainingsdaten, die man dann z. B. auf einer zuvor noch nicht benutzten Menge von Testbeispielen evaluieren kann, um die Performanz des SVM-Ansatzes zu schätzen.

Bei binären Klassifikationsaufgaben kann man anstelle der relativ aufwendigen Kreuzvalidierung auch die in [9] beschriebenen ξ-α-Schätzer für die Performanzschätzung einer SVM mit einer bestimmten Parameterkombination einsetzen. Die auf der Idee der Leave-One-Out-Schätzung basierenden ξ-α-Schätzer sind sehr effizient, weil sie mit einem einzigen SVM-Trainingslauf bestimmt werden können, und gleichzeitig auch sehr effektiv, d. h. sie liefern eine relative genaue Schätzung. Bei den in dieser Arbeit betrachteten Benchmark-Datensätzen handelt es sich allerdings nicht um binäre Klassifikationsprobleme, sondern um zwei Regressionsprobleme und vier Klassifikationsprobleme mit mehr als zwei Klassen, so dass hier die Kreuzvalidierung der Ansatz der Wahl ist.

Bei Klassifikationsaufgaben mit mehr als zwei Klassen kann folgender Ansatz gewählt werden, um mit mehreren, nur binäre Klassifikationsprobleme lösenden SVMs das Gesamtproblem zu lösen. Für jede Klasse wird eine eigene SVM trainiert, die für ein neues Beispiel entscheiden können soll, ob das Beispiel zu dieser Klasse gehört oder nicht. Für die Klassifikation eines neuen Beispiels wird dann jede dieser SVMs auf dieses Beispiel angewendet, und die Klasse der SVM mit der höchsten Konfidenz in ihre Vorhersage, d. h. mit dem größten Abstand seiner trennenden Hyperebene zu diesem Beispiel, gewählt. Beim Training sind die Parameter dieser SVMs also gemeinsam zu optimieren, damit ihre Vorhersagen vergleichbar sind, und als Optimierungskriterium dient der gemeinsam erreichte Klassifikationsfehler.

3.2 Fuzzy–ROSA–Verfahren

Die grundlegende Idee des Fuzzy–ROSA–Verfahrens besteht darin, einzelne Regeln daraufhin zu testen, ob sie einen relevanten Teilaspekt des zu modellierenden Systems beschreiben [10, 11, 12]. Damit wird das komplexe Problem einen guten Regelsatz zu finden, auf das einfachere Problem, einzelne relevante Regeln zu finden, zurückgeführt. Dieser Ansatz ermöglicht einerseits, befriedigende Lösungen auch für komplexere Aufgabenstellungen in akzeptabler Zeit zu erzielen, andererseits wird das globale Optimum in der Regel nicht erreicht. Der Regelgenerierungsprozess des Fuzzy–ROSA–Verfahrens lässt sich grob in die folgenden vier Schritte einteilen:

Projektdefinition: Vor der eigentlichen Regelgenerierung müssen die Zugehörigkeitsfunktionen für die Ein-/Ausgangsvariablen des betrachteten Systems festgelegt werden. Dies kann wissensbasiert, datenbasiert oder heuristisch erfolgen [13]. Außerdem kann die maximale Kombinationstiefe c_{max} für die Prämisse und gegebenenfalls eine maximale Zeittiefe t_{max} festgelegt werden, um den Rechenaufwand zu beschränken

Regelgenerierung: Die Regelsuche erfolgt im Fuzzy–ROSA–Verfahren so, dass Regelkandidaten (Hypothesen) aufgestellt werden, die dann auf ihre Relevanz getestet werden. Abhängig von der Suchraumgröße kann eine komplette

Suche, eine evolutionäre Suche oder eine Kombination von beiden gewählt werden [14]. Der Regelsatz wird sukzessive aus allen relevanten, nicht redundanten Regeln aufgebaut. Das Ergebnis des Regelgenerierungsprozess hängt im starken Maße von der gewählten Regeltest- und Bewertungsstrategie ab [15].

Regelreduktion: Die Anzahl der Regeln kann anschließend durch Offline–Regelreduktionsverfahren verringert werden [16, 17]. Dabei können verschiedene Anforderungen, wie z. B. komplette Überdeckung aller Eingangssituationen (*situationsbasierte Konfliktreduktion*), gleichmäßige Ausnutzung der Daten (*datenbasierte Konfliktreduktion*), Verringerung des Modellierungsfehlers oder der Regelanzahl (*optimierende Konfliktreduktion*), berücksichtigt werden.

Regelsatzanalyse und -optimierung: Durch die abschließende Analyse des Regelsatzes kann der Modellierungsprozess und das Modellierungsergebnis bewertet und gegebenenfalls Feedback für die Problemformulierung erhalten werden. Zusätzlich kann das Ein-/Ausgangsverhalten des erhaltenen Fuzzy–Systems durch Anpassung der verbleibenden freien Parameter optimiert werden.

In [18] wird eine systematische Vorgehensweise zur Einstellung der Strategieparameter des Fuzzy–ROSA–Verfahrens vorgeschlagen, die es auch weniger versierten Benutzern ermöglichen soll, befriedigende Ergebnisse in akzeptabler Zeit zu erzielen. Darauf wird im Folgenden genauer eingegangen.

3.2.1 Systematische Festlegung der Zugehörigkeitsfunktionen

Es wird davon ausgegangen, dass kein Expertenwissen verfügbar ist und somit wird bei der Festlegung der Zugehörigkeitsfunktionen heuristisch vorgegangen. Die dabei eingesetzte, Vorgehensweise ist problemabhängig:

Im Falle *kontinuierlicher* Einflussgrößen wird eine äquidistante Partitionierung vorgenommen. Bei der systematischen Vorgehensweise wird dafür zunächst der Wertebereich der Einflussgröße bestimmt und anschließend werden automatisch 50 % überlappende Trapeze generiert[4]. Die Anzahl der linguistischen Werte pro Einflussgröße ist je nach Datenumfang D mit vier, fünf, sieben oder neun vorgegeben worden. Eine ungerade Anzahl von linguistischen Werten hat sich insbesondere bei einem zur Null symmetrischen Wertebereich als vorteilhaft erwiesen. Dabei ist in den meisten Fällen die Einteilung in fünf linguistische Werte ausreichend.

Einfacher ist die Vorgehensweise bei der Definition der Zugehörigkeitsfunktionen für *diskrete* Einflussgrößen, die zum Beispiel für Klassen stehen können[5]. Im Prinzip ist für jeden diskreten Wert einfach ein Singleton zu definieren.

[4] Einzige Ausnahme ist das Benchmarkproblem MACKEY, bei dem Dreiecke statt Trapeze verwendet worden sind, um die Vergleichbarkeit zu anderen Ansätzen mit dem Fuzzy–ROSA–Verfahren zu gewährleisten [19, 11].

[5] Gegebenenfalls ist den Klassen eine Nummer zuzuordnen, da im Fuzzy–ROSA–Vefahren nur numerische Größen verarbeitet werden können.

3.2.2 Systematische Konfiguration des Fuzzy–Systems

Bevor auf die problemklassenspezifischen Einstellungen des Fuzzy–Systems eingegangen wird, erfolgt eine Übersicht über die Parameter, die im Rahmen der systematischen Vorgehensweise nicht variiert werden: Das Algebraische Produkt ist als UND–Operator für die Prämissenauswertung, die Aktivierung und die UND–Verknüpfung des Glaubensmaßes (Bewertungsindex) voreingestellt. Entsprechend ist die Algebraische Summe als ODER–Operator für die Akkumulation gewählt worden. Die Wahl dieser Standardeinstellungen wird in [11] vorgeschlagen und ist zum Teil aus der Statistik motiviert. Weiterhin werden bei Anwendung des Fuzzy–Systems ausgangsseitig immer Singletons verwendet [11]. Diese Auswertung ist zum einen effizienter und zum anderen konnten keine signifikanten Unterschiede bezüglich der Modellierungsgüte im Vergleich zur Verwendung der Originalzugehörigkeitsfunktionen festgestellt werden. Im Gegensatz dazu ist sowohl die Verfahrensweise bei nicht abgedeckten Eingangssituationen sowie die Wahl der Defuzzifizierungsmethode problemabhängig wie folgt vorzunehmen:

Defuzzifizierung: Bei diskreten Ausgangsgrößen ist es häufig sinnvoll, den am stärksten empfohlenen Ausgangsgrößenwert zu wählen, was durch die MOM–Defuzzifizierung (*Mean of Maximum*) erreicht wird. Insbesondere bei *Klassifikationsproblemen* wird damit vermieden, dass eine unerwünschte Mittelwertbildung zweier Klassen, wie z. B. *Mann* und *Frau* vorgenommen wird. Hingegen ist bei *kontinuierlichen* Ausgangsgrößen meist ein Kompromiss zwischen den verschiedenen Regeln entsprechend ihres Aktivierungsgrads und Glaubensmaßes wünschenswert, da sich dies insbesondere bei Approximationsaufgaben günstig auf die Modellierungsgüte auswirken kann. Deswegen wird unter diesen Voraussetzungen üblicherweise die COG–Defuzzifizierung (*Center of Gravity*) eingesetzt.

Nicht abgedeckte Eingangssituationen: Die Vorgehensweise bei nicht abgedeckten Eingangssituationen kann sich entscheidend auf das Modellierungsergebnis auswirken. Bei *Klassifikationsaufgaben* ist es naheliegend, einen Defaultwert für *nicht klassifiziert* auszugeben. In Abhängigkeit von der konkreten Aufgabenstellung muss dann entschieden werden, wie in einem solchen Fall weiter verfahren werden soll. Im Gegensatz dazu ist es bei *Approximationsaufgaben* vielfach günstiger, den Mittelwert der Ausgangsgröße als Defaultwert festzulegen, da dadurch im Mittel die Modellierungsgüte erhöht wird. Eine Ausnahme bilden dynamische Probleme, bei denen häufig ein „Halten“ des letzten Wertes vorzuziehen ist, insbesondere dann, wenn eine sprunghafte Änderung der Ausgangsgröße vermieden werden soll.

Obwohl im Fuzzy–ROSA–Verfahren auch erweiterte Strategieelemente, wie Hyperinferenz und Inferenzfilter verfügbar sind [10], ist hier auf eine Ausdehnung der systematischen Untersuchung verzichtet worden, um den Aufwand in akzeptablen Grenzen zu halten.

3.2.3 Systematische Wahl der Strategieparameter der Regelsuche

Wie bereits ausgeführt, ist die Wahl der Regeltest- und Bewertungsstrategie zentral für das Modellierungsergebnis. Obwohl heuristisches Wissen über günstige Einstellungen für bestimmte Problemklassen vorhanden ist, konnten bisher noch keine allgemein gültigen Handlungsvorschläge abgeleitet werden. In Anbetracht dessen wird in [18] ein Schema zur Einstellung der Strategieparameter der Regelsuche vorgeschlagen und eingesetzt. Die Einstellung der Strategieparameter der Regelsuche basiert dabei auf einer kompletten Suche mit maximaler Kombinationstiefe $c_{max} = 2$. Damit ist für alle im Rahmen dieses Beitrags untersuchten Benchmarkprobleme gewährleistet, dass die Suche in akzeptabler Zeit, d. h. in wenigen Minuten, durchgeführt werden kann. Außerdem kann oft schon mit einer maximalen Kombinationstiefe $c_{max} = 2$ eine hohe Modellierungsgüte bei vergleichsweise kleiner Regelanzahl R erzielt werden. Andernfalls ist meistens zumindest eine Aussage darüber möglich, ob eine Modellierung mit dem Fuzzy–ROSA–Verfahren überhaupt erfolgsversprechend ist. Bei der Wahl der Strategieparameter der Regelsuche werden im Wesentlichen die folgenden drei Zielsetzungen verfolgt:

- eine hohe Modellierungsgüte,
- kompakte Regelsätze und
- eine hohe Abdeckung der Lerndaten.

Entscheidenden Einfluss auf die Modellierungsgüte hat die Wahl der Regeltest- und Bewertungsstrategie. Weiterhin können kompakte und damit besser interpretierbare Regelsätze dadurch erhalten werden, dass nur Regeln mit einem Bewertungsindex $J \geq J_{min}$ in den Regelsatz aufgenommen werden. In einem Teil der Benchmarkanwendungen führt die Erhöhung des Schwellwerts J_{min} sogar zunächst auch zu einer Erhöhung der Modellierungsgüte. Allerdings kann ein zu hoher Schwellwert J_{min} dazu führen, dass nicht mehr alle Datenpunkte abgedeckt werden und infolgedessen die Modellierungsgüte wieder sinkt. Zur Erreichung eines möglichst günstigen Kompromisses zwischen den eingangs aufgeführten Zielsetzungen wird deswegen nach folgendem Schema vorgegangen [18]:

1. Eine komplette Durchmusterung der Kombinationstiefe $c_{max} = 2$ für alle Regeltest- und Bewertungsstrategien mit einem Schwellwert $J_{min} = 0.0$.

2. Schrittweise Reduzierung der erhaltenen Regelsätze durch Erhöhung des Schwellwerts J_{min} mit einer Schrittweite von $\Delta J = 0.1$.

3. Ermittlung der Modellierungsgüte und der nicht abgedeckten Datensätze für die in den vorherigen Schritten erzeugten Regelsätze auf den Lerndaten.

4. Wahl der Regeltest- und Bewertungsstrategie sowie des Schwellwerts J_{min} unter Berücksichtigung der erzielten Modellierungsgüte ϵ, Regelanzahl R und nicht abgedeckten Datensätze D_{nc}.

5. Optionale Anwendung der optimierenden Konfliktreduktion (OCR) zur Erhöhung der Modellierungsgüte und Erniedrigung der Regelanzahl R.

Es hat sich für die Wahl der Regeltest- und Bewertungsstrategie sowie des Schwellwerts J_{min} als sinnvoll erwiesen, die Kenngrößen der erzielten Modellierungsergebnisse, wie Modellierungsfehler ϵ, Regelanzahl R und die nicht abgedeckten Datensätze D_{nc}, in eine Tabelle einzutragen. Dabei hat es sich in fast allen bisher untersuchten Anwendungen als ausreichend erwiesen, zunächst nur die Modellierungsgüte zu betrachten, da eine erhöhte Anzahl von nicht abgedeckten Datensätzen im Allgemeinen zu einem höheren Modellierungsfehler ϵ führt. Im Falle gleichwertiger Modellierungsgüten ist der kompaktere Regelsatz mit der kleineren Regelanzahl R vorzuziehen.

3.3 Experimentelle Ergebnisse und Bewertung

Anhand von sechs Benchmarkdatensätzen soll untersucht werden, ob die in Abschnitt 3.2 beschriebene systematische Vorgehensweise es auch weniger versierten Anwendern ermöglicht, ohne spezielle Vorkenntnisse in akzeptabler Zeit befriedigende Modellierungsergebnisse mit dem Fuzzy-ROSA-Verfahren zu erzielen. Im Folgenden wird zunächst ein Überblick über die Resultate gegeben, die bei den Benchmarkproblemen erreicht wurden (siehe Tabelle 1). Anschließend werden die Vorgehensweise und die Ergebnisse der Experimente auf den einzelnen Benchmarkdatensätzen erläutert.

Tabelle 1: Mit der Fuzzy–ROSA–Methode (FRM) erzielter durchschnittlicher relativer Fehler ϵ_{vali} auf den Validierungsdaten in Prozent und durchschnittliche Regelanzahl R bei 2-facher Kreuzvalidierung.

Datensatz	FRM_{sys}		$\text{FRM}_{sys+OCR}$		FRM_{best}	
	ϵ_{vali}	R	ϵ_{vali}	R	ϵ_{vali}	R
IRIS	6.0	7	6.0	5	4.1	9
WINE	11.2	105	10.7	15	6.2	141
MACKEY	0.07	59	0.05	20	0.00014	92
KIN	0.22	1530	0.12	457	0.16	309
GENE	5.5	1567	5.4	500	5.8	221
SAT	18.8	2683	18.2	1044	12.7	204

Ziel der Anwendung der Fuzzy–ROSA–Methode (FRM) auf die Benchmarkprobleme ist, einen Vergleich zu anderen Ansätzen aus der Literatur zu ermöglichen. Hinsichtlich der Modellierungsgüte wird dafür eine Einordnung der erzielten Ergebnisse vorgenommen. Dabei wird auch darauf eingegangen, welche Modellierungsgüte mit der systematischen Vorgehensweise (FRM_{sys} bzw. $\text{FRM}_{sys+OCR}$), verglichen mit dem bisher besten von Experten vorgenommenen Modellierungsansatz mit dem Fuzzy–ROSA–Verfahren (FRM_{best}), erreicht werden kann. Da viele in der Literatur verwendete Ansätze nicht zu interpretierbaren Modellen führen, wird diesbezüglich auf einen umfassenden Vergleich verzichtet. Stattdessen wird die Interpretierbarkeit der mit dem Fuzzy–ROSA–Verfahren erzeugten Modelle bewertet und gegebenenfalls auf weiterführende Literatur verwiesen.

3.3.1 IRIS

Dieses Klassifikationsproblem basiert auf dem in der Literatur weit verbreiteten IRIS–Datensatz[6] [21, 22]. Die Klassifikationsaufgabe besteht darin, drei Schwertlilienarten anhand von vier Charakteristika zu bestimmen.

Anders als bei den in Tabelle 1 zusammengefaßten Ergebnissen, wird hier eine 10-fache Kreuzvalidierung analog zu [23] vorgenommen[7], um eine bessere Vergleichbarkeit der Ergebnisse zu ermöglichen. Dabei ergaben sich folgende Resultate für FRM_{sys} bzw. FRM_{best}: $R = 9$, $\epsilon_{learn} = 4.1$, $\epsilon_{vali} = 4.1$ und $\mathrm{FRM}_{sys+OCR}$: $R = 5$, $\epsilon_{learn} = 2.9$, $\epsilon_{vali} = 5.4$. Der relative Klassifikationsfehler auf Validierungsdaten ϵ_{vali} wird in Abbildung 3 mit den Ansätzen aus [23] verglichen.

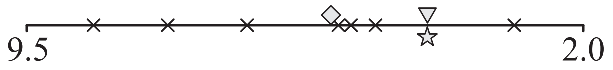

Abbildung 3: Relativer Klassifikationsfehler auf Validierungsdaten in Prozent ϵ_{vali} für IRIS bei folgenden Ansätzen: ∇ FRM_{sys}, $\diamond$ $\mathrm{FRM}_{sys+OCR}$, $\star$ FRM_{best} und $\times$ Literatur.

Insgesamt zeigt sich, dass das beste Modellierungsergebnis mit dem Fuzzy–ROSA–Verfahren FRM_{best} mit der systematischen Vorgehensweise FRM_{sys} erzielt werden konnte. Die Anwendung der optimierenden Konfliktreduktion führt in diesem Beispiel zu einem Overfitting und somit zu einem höheren Klassifikationsfehler auf Validierungsdaten. Im Vergleich zu den Ansätzen aus der Literatur belegt das Fuzzy–ROSA–Verfahren mit FRM_{sys} bzw. FRM_{best} einen der vorderen Plätze. Mit einer Regelanzahl von $R = 9$ ist auch die Interpretierbarkeit des erhaltenen Fuzzy–Modells in einem sehr hohen Maße gewährleistet.

3.3.2 WINE

Dieses Klassifikationsproblem aus [24] hat zum Ziel, die Herkunft von Weinen anhand ihrer chemischen Zusammensetzung zu bestimmen. Wie beim Benchmarkproblem IRIS wird hier eine 10-fache Kreuzvalidierung analog zu [23] vorgenommen. Dabei ergaben sich folgende Resultate für FRM_{sys} bzw. FRM_{best}: $R = 141$, $\epsilon_{learn} = 3.2$, $\epsilon_{vali} = 6.2$ und $\mathrm{FRM}_{sys+OCR}$: $R = 35$, $\epsilon_{learn} = 1.8$, $\epsilon_{vali} = 6.7$. Der relative Klassifikationsfehler auf Validierungsdaten ϵ_{vali} wird in Abbildung 4 mit den Ansätzen aus [23] verglichen.

Abbildung 4: Relativer Klassifikationsfehler auf Validierungsdaten in Prozent ϵ_{vali} für WINE bei folgenden Ansätzen: ∇ FRM_{sys}, $\diamond$ $\mathrm{FRM}_{sys+OCR}$, $\star$ FRM_{best} und $\times$ Literatur.

[6] Wie die meisten anderen der hier verwendeten Datensätze kann der IRIS-Datensatz aus dem Repository der University of California at Irvine (UCI)[20] heruntergeladen werden.

[7] In [23] werden die Daten zufällig in zehn Teile geteilt. In zehn Modellierungen wird auf jeweils neun Teilen gelernt und auf dem verbleibenden Teil validiert. Das Modellierungsergebnis ergibt sich aus Mittelung über die zehn Durchläufe.

Zwar weist in diesem Beispiel die systematische Vorgehensweise FRM_{sys} bzw. FRM_{best} den niedrigsten Klassifikationsfehler auf. Allerdings ist die Regelanzahl $R = 141$ in Anbetracht dessen, dass nur $D = 178$ Datenpunkte verfügbar sind, als sehr hoch zu bezeichnen. Daher ist der Ansatz mit optimierender Konfliktreduktion, der zu einer Regelanzahl $R = 35$ führt, vorzuziehen. Im Vergleich zu Ansätzen aus der Literatur werden in diesem Beispiel mit dem Fuzzy–ROSA–Verfahren nur hintere Plätze belegt.

3.3.3 MACKEY

Die chaotische Mackey–Glass–Zeitfolge [25] wird in der Literatur häufig zum Testen von Lern- und Modellierungsverfahren genutzt. Ein Überblick und eine Einordnung der Ergebnisse, die für dieses Approximationsproblem mit dem Fuzzy–ROSA–Verfahren erzielt werden können, ist in [19] zu finden. Es ergeben sich folgende Resultate[8] für FRM_{sys}: $R = 59$, $\epsilon_{learn} = 0.007$, $\epsilon_{vali} = 0.008$; $\mathrm{FRM}_{sys+OCR}$: $R = 20$, $\epsilon_{learn} = 0.004$, $\epsilon_{vali} = 0.005$ und FRM_{best}: $R = 92$, $\epsilon_{learn} = 0.00011$, $\epsilon_{vali} = 0.00014$. Der mittlere quadratische Modellierungsfehler auf Validierungsdaten ϵ_{vali} wird in Abbildung 5 mit den Ansätzen aus [19] verglichen.

Abbildung 5: Mittlerer quadratischer Fehler auf Validierungsdaten ϵ_{vali} für MACKEY bei folgenden Ansätzen: $\triangledown$ FRM_{sys}, $\diamond$ $\mathrm{FRM}_{sys+OCR}$, $\star$ FRM_{best} und $\times$ Literatur.

Das Modellierungsergebnis mit der systematischen Vorgehensweise FRM_{sys} ist deutlich schlechter als die anderen Ansätze. Obwohl es durch die Durchführung der optimierenden Konfliktreduktion (OCR) stark verbessert werden kann, belegt der Ansatz $\mathrm{FRM}_{sys+OCR}$ immer noch einen der hinteren Plätze. Ein Grund dafür liegt in der Granularisierung der Ausgangsgröße. Dem kann, wie in [19] gezeigt wird, durch Regeln vom TSK–Typ entgegengewirkt werden. Dementsprechend ergibt sich das beste Ergebnis mit dem Fuzzy–ROSA–Verfahren FRM_{best} genau für einen solchen Ansatz. Dabei ergibt sich eine vergleichbar hohe Modellierungsgüte wie bei den besten Ansätzen aus der Literatur. Allerdings muss dafür auch eine erhöhte Regelanzahl von $R = 92$ in Kauf genommen werden. Liegt die Priorität auf kompakten interpretierbaren Regelsätzen, ist somit der Ansatz $\mathrm{FRM}_{sys+OCR}$ vorzuziehen.

3.3.4 KIN

Die Aufgabenstellung bei diesem Approximationsproblem ist, den Abstand des Endeffektors eines 8–Gelenk–Robotorarms von einem Zielpunkt vorherzusagen. Es ergeben sich folgende Resultate für FRM_{sys}: $R = 1530$, $\epsilon_{learn} = 0.21$, $\epsilon_{vali} = 0.22$; $\mathrm{FRM}_{sys+OCR}$: $R = 457$, $\epsilon_{learn} = 0.11$, $\epsilon_{vali} = 0.12$ und FRM_{best}: $R = 309$, $\epsilon_{learn} = 0.16$, $\epsilon_{vali} = 0.16$. Der mittlere absolute Modellierungsfehler auf Validierungsdaten

[8] Abweichend von [19] wird hier als bestes Ergebnis FRM_{best} der multilineare TSK–Ansatz für den vollständigen Regelsatz mit Kombinationstiefe $c_{max} = 4$ angegeben. Die starke Verbesserung der Modellierungsgüte ist auf eine Erhöhung der numerischen Rechengenauigkeit zurückzuführen.

ϵ_{vali} wird in Abbildung 6 mit den auf den Web-Seiten von Delve[9] verfügbaren Ansätzen verglichen.

Abbildung 6: Mittlerer absoluter Fehler auf Validierungsdaten ϵ_{vali} für KIN bei folgenden Ansätzen: ∇ FRM$_{sys}$, $\diamond$ FRM$_{sys+OCR}$, $\star$ FRM$_{best}$ und $\times$ Literatur.

Auffällig ist die vergleichsweise schlechte Modellierungsgüte bei der rein systematischen Vorgehensweise FRM$_{sys}$. Ein ähnliches Verhalten konnte auch schon bei der anderen Approximationsaufgabe MACKEY festgestellt werden. Allerdings fällt hier die Erhöhung der Modellierungsgüte durch die optimierende Konfliktreduktion (OCR) so hoch aus, dass der Modellierungsfehler kleiner als beim bisher besten TSK–Ansatz mit dem Fuzzy–ROSA–Verfahren FRM$_{best}$ ist (siehe unten). Insgesamt wird damit zwar ein Platz im oberen Mittelfeld bezüglich der Modellierungsgüte belegt. Jedoch kann bei einer Regelanzahl von $R = 457$ kaum noch von Interpretierbarkeit gesprochen werden. Kompaktere Regelsätze ergeben sich bei der Durchführung der schrittweise lokaler werdenden Suche SLS (siehe [18]). Diese weisen allerdings auch eine deutlich schlechtere Modellierungsgüte auf, welche jedoch durch eine Transformation in TSK–Regeln drastisch verbessert werden kann. Das bisher beste Ergebnis FRM$_{best}$ ist durch die Transformation eines Regelsatzes, generiert in einem SLS–Ansatz vergleichbar mit dem in [18] erzielt worden.

3.3.5 GENE

Bei diesem Benchmarkproblem aus [26] geht es um die Klassifikation von Intron–Exon–Verbindungen in Nukleotidsequenzen. Ein Überblick über die Ergebnisse, die mit dem Fuzzy–ROSA–Verfahren erzielt werden können, ist in [11] zu finden. Dort wird auch eine Einordnung bezüglich der Ansätze, die in [27] aufgeführt sind, vorgenommen. Es ergeben sich folgende Resultate für FRM$_{sys}$: $R = 1567$, $\epsilon_{learn} = 4.1$, $\epsilon_{vali} = 5.5$; FRM$_{sys+OCR}$: $R = 500$, $\epsilon_{learn} = 1.4$, $\epsilon_{vali} = 5.4$ und FRM$_{best}$: $R = 221$, $\epsilon_{learn} = 5.1$, $\epsilon_{vali} = 5.8$. Der relative Klassifikationsfehler auf Validierungsdaten ϵ_{vali} wird in Abbildung 7 mit den Ansätzen aus [27] verglichen.

Abbildung 7: Relativer Klassifikationsfehler auf Validierungsdaten in Prozent ϵ_{vali} für GENE bei folgenden Ansätzen: ∇ FRM$_{sys}$, $\diamond$ FRM$_{sys+OCR}$, $\star$ FRM$_{best}$ und $\times$ Literatur.

Insgesamt werden mit allen Modellierungsansätzen mit dem Fuzzy–ROSA–Verfahren vordere Plätze belegt. Aufgrund des vergleichsweise kompakten Regelsatzes ist FRM$_{best}$ zu favorisieren. Allerdings resultiert dieser Regelsatz aus einem aufwändigen und nicht deterministischen Regelgenerierungsprozess [11], der kaum ohne Expertenwissen vorgenommen werden kann.

[9] **D**ata for **E**valuation **L**earning in **V**alid **E**xperiments (Delve), University of Toronto, Kanda: `http://www.cs.utoronto.ca/~delve/data/datasets.html`

3.3.6 SAT

Die Aufgabe bei diesem Benchmarkproblem besteht darin, den Bodentyp eines Pixels in Satellitenbildern zu klassifizieren Es ergeben sich folgende Resultate für FRM_{sys}: $R = 2683$, $\epsilon_{learn} = 18.2$, $\epsilon_{vali} = 18.8$; $\mathrm{FRM}_{sys+OCR}$: $R = 1044$, $\epsilon_{learn} = 15.4$, $\epsilon_{vali} = 18.2$ und FRM_{best}: $R = 204$, $\epsilon_{learn} = 11.0$, $\epsilon_{vali} = 12.7$. Der relative Klassifikationsfehler auf Validierungsdaten ϵ_{vali} wird in Abbildung 8 mit den Ansätzen aus [27] verglichen.

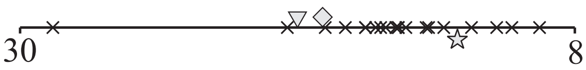

Abbildung 8: Relativer Klassifikationsfehler auf Validierungsdaten in Prozent ϵ_{vali} für SAT bei folgenden Ansätzen: $\triangledown$ FRM_{sys}, $\diamond$ $\mathrm{FRM}_{sys+OCR}$, $\star$ FRM_{best} und $\times$ Literatur.

Mit den systematischen Ansätzen FRM_{sys} und $\mathrm{FRM}_{sys+OCR}$ werden in diesem Beispiel nur hintere Plätze bezüglich der Modellierungsgüte belegt. Hinzu kommt, dass aufgrund der hohen Regelanzahl in beiden systematischen Ansätzen nicht mehr von Interpretierbarkeit gesprochen werden kann. Eine verfeinerte Vorgehensweise FRM_{best} zur Generierung eines kompakten Regelsatzes mit hoher Modellierungsgüte wird in [17] beschrieben. Entscheidend dabei ist die Durchführung einer datenbasierten Konfliktreduktion sowie der Einsatz eines Parallelrechners.

3.3.7 Zusammenfassende Bewertung

Insgesamt kann festgestellt werden, dass die besten Modellierungsansätze mit dem Fuzzy–ROSA–Verfahren FRM_{best} in fast allen Benchmarkproblemen bezüglich der Modellierungsgüte die vorderen Plätze belegen. Einzige Ausnahme ist das Beispiel WINE. Weiterhin wird für IRIS, WINE, GENE und KIN schon mit den systematischen Ansätzen FRM_{sys} bzw. $\mathrm{FRM}_{sys+OCR}$ eine Modellierungsgüte vergleichbar mit dem besten Ansatz FRM_{best} erzielt. Hingegen führen bei SAT und MACKEY erst die verfeinerten Ansätze FRM_{best} zu den hohen Modellierungsgüten. Dennoch ist auch bei diesen beiden Beispielen die Durchführung der systematischen Vorgehensweise als sehr hilfreich einzustufen, da sie gewissermaßen den Ausgangspunkt und die Referenz für die verfeinerten Ansätze bildet.

Die Anwendung der optimierenden Konfliktreduktion (OCR) hat sich teilweise als sehr erfolgreich erwiesen. Nur bei wenigen der untersuchten Datensätze, wie bei WINE und IRIS, führt die optimierende Konfliktreduktion (OCR) zu einem Overfitting. Ansonsten konnten bei den komplexeren Klassifikationsaufgaben GENE und SAT die Regelsätze deutlich verkleinert und gleichzeitig die Modellierungsgüte leicht verbessert werden. Noch viel deutlicher fällt die Erhöhung der Modellierungsgüte durch die optimierende Konfliktreduktion (OCR) bei den Approximationsaufgaben MACKEY und KIN aus.

Hinsichtlich der Interpretierbarkeit sind die Ergebnisse wie folgt zu bewerten: Für die Beispiele IRIS, WINE und MACKEY können kompakte Regelsätze ($R < 50$) mit den systematischen Ansätzen FRM_{sys} bzw. $\mathrm{FRM}_{sys+OCR}$ generiert werden, ohne zu große Abstriche bei der Modellierungsgüte machen zu müssen. Bei GENE, SAT und KIN hingegen ist die Regelanzahl auch nach Durchführung der optimierenden Konfliktreduktion (OCR) immer noch sehr hoch ($R > 400$). Mit den verfeinerten Ansätzen kann die Regelanzahl bei GENE und SAT noch deutlich reduziert werden ($R \approx 200$). Dies ist eine Größenordnung, die durchaus auch von wissensbasiert aufgestellten Regelsätzen in der Praxis erreicht wird. Einzige Ausnahme bildet somit das Beispiel KIN, für das bisher kein kompakter und damit interpretierbarer Regelsatz mit akzeptabler Güte gefunden werden konnte.

Aufgrund der vielversprechenden Resultate kann insgesamt davon gesprochen werden, dass mit der hier vorgestellten systematischen Vorgehensweise, es weniger versierten Anwendern erstmals ermöglicht wird, ohne spezielle Vorkenntnisse in akzeptabler Zeit befriedigende Modellierungsergebnisse mit dem Fuzzy–ROSA–Verfahren zu erzielen. Somit ist diese zu Beginn der Experimente formulierte Zielsetzung weitestgehend eingelöst worden.

3.4 Fazit

Dieser Beitrag hat das Konzept und die technische Realisierung eines Ansatzes für die Erfassung und den Transfer von Wissen im Bereich der Computational Intelligence beschrieben. Der vorgestellte Ansatz trennt die Struktur des dargestellten Wissens klar von den Wissensinhalten und beides wiederum von der Darstellung. Durch diese klare Trennung sowie durch eine objekt- und relational orientierte kompakte Darstellung der Ontologie und der Instanzen in zwei XML-Dateien werden Redundanzen vermieden und die Wartung des Systems und seiner Inhalte erleichtert. Diese Art der Darstellung erlaubt eine automatische Generierung von HTML-Seiten zur Eingabe, Änderung und Ansicht von Wissensinhalten sowie das einfache Navigieren entlang der Relationen zwischen den Wissensinhalten.

Neben diesem Wissensmanagementansatz wurden außerdem für zwei CI–Methoden beschrieben, wie diese systematisch parametrisiert werden können. Solche systematischen Vorgehensweisen stellen beispielhafte Einträge in Methoden oder Methodeneinsatzbeschreibungen dar und erlauben es auch Anwendern, die mit einer CI–Methode nicht so vertraut sind, diese in annehmbarer Zeit mit zufriedenstellendem Ergebnis auf ein ihnen vorliegendes Problem anzuwenden. Anhand von Experimenten mit sechs Benchmarkdatensätzen wurde für eine dieser beiden CI–Methoden gezeigt, dass dieses Vorgehen erfolgsversprechend ist.

3.5 Literaturverzeichnis

[1] Stefan Haustein. Serving both Worlds – Infolayer Status Report. Interner Bericht Nr. 26, Lehrstuhl Informatik VIII *(Künstliche Intelligenz)*, Universität Dortmund, Februar 2001. `http://www-ai.cs.uni-dortmund.de/SOFTWARE/IL/`.

[2] Vladimir N. Vapnik. *Statistical Learning Theory.* Wiley, Chichester, GB, 1998.

[3] C. Burges. A Tutorial on Support Vector Machines for Pattern Recognition. *Data Mining and Knowledge Discovery*, 2(2):121–167, 1998.

[4] Alex J. Smola und Bernhard Schölkopf. A Tutorial on Support Vector Regression. Interner bericht, NeuroCOLT2 Technical Report Series, 1998.

[5] R. Fletcher. *Practical Methods of Optimization*. Wiley, Chichester, GB, 2. Auflage, 1987.

[6] Stefan Rüping. *mySVM-Manual*. Universität Dortmund, Lehrstuhl Informatik VIII, 2000. `http://www-ai.cs.uni-dortmund.de/SOFTWARE/MYSVM/`.

[7] Thorsten Joachims. Text Categorization with Support Vector Machines: Learning with Many Relevant Features. In *Proceedings of the European Conference on Machine Learning (ECML-98)*, Seiten 137 – 142, Berlin, 1998. Springer.

[8] Stefan Rüping. SVM Kernels for Time Series Analysis. In Ralf Klinkenberg, Stefan Rüping, Andreas Fick, Nicola Henze, Christian Herzog, Ralf Molitor, und Olaf Schröder, Hrsg., *LLWA 01 – Tagungsband der GI-Workshop-Woche Lernen – Lehren – Wissen – Adaptivität*, Nummer 763 in Forschungsberichte des Fachbereichs Informatik der Universität Dortmund, Oktober 2001.

[9] Thorsten Joachims. Estimating the Generalization Performance of a SVM Efficiently. In *Proceedings of the International Conference on Machine Learning (ICML-2001)*, San Francisco, CA, USA, 2000. Morgan Kaufman.

[10] H. Kiendl. *Fuzzy Control methodenorientiert*. Oldenbourg, München, 1. Auflage, 1997.

[11] A. Krone. *Datenbasierte Generierung von relevanten Fuzzy–Regeln zur Modellierung von Prozesszusammenhängen und Bedienstrategien*. Fortschritt–Berichte VDI, Reihe 10, Nr. 615. VDI Verlag, Düsseldorf, 1. Auflage, 1999.

[12] T. Slawinski, A. Krone, P. Krause, und H. Kiendl. The Fuzzy–ROSA Method: A Statistically Motivated Fuzzy Approach for Data–Based Generation of Small Interpretable Rule Bases in High–Dimensional Search Spaces. In M. Last, A. Kandel, und H. Bunke, Hrsg., *Data Mining and Computational Intelligence*, Seiten 141–166. Physica-Verlag, Heidelberg, 2001.

[13] A. Krone und T. Slawinski. Data–Based Extraction of Unidimensional Fuzzy Sets for Fuzzy Rule Generation. In *Proceedings of the Seventh IEEE International Conference on Fuzzy Systems (FUZZ–IEEE '98), Anchorage, USA, 1998*, Band 2, Seiten 1032–1037, Piscataway, NJ, USA, 1998. IEEE Press.

[14] T. Slawinski, A. Krone, U. Hammel, D. Wiesmann, und P. Krause. A Hybrid Evolutionary Search Concept for Data–based Generation of Relevant Fuzzy Rules in High Dimensional Spaces. In *Proceedings of IEEE International Conference on Fuzzy Systems (FUZZ–IEEE '99) Seoul, Korea, 1999*, Band 3, Seiten 1432–1437, Piscataway, NJ, USA, 1999. IEEE Press.

[15] H. Jessen und T. Slawinski. Test- and Rating Strategies for Data–based Rule Generation. In *Reihe Computational Intelligence*, Seiten 1–9. CI–39/98, Sonderforschungsbereich 531, Universität Dortmund, 1998.

[16] A. Krone. Advanced Rule Reduction Concepts for Optimizing Efficiency of Knowledge Extraction. In *Proceedings of the Fourth European Congress on Intelligent Techniques and Soft Computing (EUFIT '96)*, Band 2, Seiten 919–923, Aachen, 1996. Verlag Mainz.

[17] A. Krone, P. Krause, und T. Slawinski. A New Rule Reduction Method for Finding Interpretable and Small Rule Bases in High Dimensional Search Spaces. In *Proceedings of the Ninth IEEE International Conference on Fuzzy Systems, (FUZZ–IEEE '00), San Antonio, USA*, Band 2, Seiten 696–699, Piscataway, NJ, USA, 2000. IEEE Press.

[18] T. Slawinski. *Analyse und effiziente Generierung von relevanten Fuzzy–Regeln in hochdimensionalen Suchräumen.* Dissertation (angenommen). Fakultät für Elektrotechnik und Informationstechnik, Universität Dortmund, 2001.

[19] P. Krause. Generierung von Takagi–Sugeno–Fuzzy–Systemen aus relevanten Fuzzy–Regeln. In *Tagungsband des 10. Workshops Fuzzy Control des GMA–FA 5.22, Dortmund*, Seiten 84–97, Karlsruhe, 2000. VDI/VDE GMA–FA 5.22, Wissenschaftliche Berichte Forschungszentrum Karlsruhe (FZKA 6509).

[20] C.L. Blake und C.J. Merz. UCI Repository of machine learning databases, 1998. `http://www.ics.uci.edu/~mlearn/MLRepository.html`.

[21] E. Anderson. The IRISes of the Gaspe Peninsula. *Bull. Amer. IRIS Soc.*, 59:2–5, 1935.

[22] N. R. Pal, K. Pal, und J. C. Bezdek. A Mixed c–Means Clustering Model. In *Proceedings of the Sixth IEEE International Conference on Fuzzy Systems (FUZZ–IEEE '97), Barcelona, Spanien, 1997*, Band 1, Seiten 11–21, Piscataway, NJ, USA, 1997. IEEE Press.

[23] R. Holve. Investigation of Automatic Rule Generation for Hierarchical Fuzzy Systems. In *Proceedings of the Seventh IEEE International Conference on Fuzzy Systems (FUZZ–IEEE '98), Anchorage, USA, 1998*, Band 2, Seiten 973–978, Piscataway, NJ, USA, 1998. IEEE Press.

[24] A. L. Corcoran und S. Sandip. Using Real–Valued Genetic Algorithms to Evolve Rule Sets for Classifications. In *Proceedings of the First IEEE Conference on Evolutionary Computation (ICEC '94), Orlando, USA, 1994*, Band 1, Seiten 120–124, Piscataway, NJ, USA, 1994. IEEE Press.

[25] M. Mackey und L. Glass. Oscillation and Chaos in Physiological Control Systems. *Science*, 197:287–289, 1977.

[26] L. Prechelt. PROBEN 1 – A Set of Neural Network Benchmark Problems and Benchmarking Rules. Interner Bericht 21, Fakultät für Informatik, Universität Karlsruhe, 1994.

[27] D. Michie, D.J. Spiegelhalter, und C.C. Taylor. *Machine learning, Neural and Statistical Classification.* Ellis Horwood, Hemel Hempstead, GB, 1994.