

Semiüberwachte Paarweise Klassifikation

Bachelor-Thesis von Andriy Nadolskyy aus Jaworiw
Dezember 2013



TECHNISCHE
UNIVERSITÄT
DARMSTADT

Fachbereich Informatik
Knowledge Engineering

Semiüberwachte Paarweise Klassifikation

Vorgelegte Bachelor-Thesis von Andriy Nadolskyy aus Jaworiw

1. Gutachten: Prof. Dr. Johannes Fürnkranz

2. Gutachten: Dr. Eneldo Loza Mencía

Tag der Einreichung:

Erklärung zur Bachelor-Thesis

Hiermit versichere ich, die vorliegende Bachelor-Thesis ohne Hilfe Dritter nur mit den angegebenen Quellen und Hilfsmitteln angefertigt zu haben. Alle Stellen, die aus Quellen entnommen wurden, sind als solche kenntlich gemacht. Diese Arbeit hat in gleicher oder ähnlicher Form noch keiner Prüfungsbehörde vorgelegen.

Darmstadt, den 17. Dezember 2013

(Andriy Nadolskyy)

Inhaltsverzeichnis

Inhaltsverzeichnis	2
1 Einführung	3
1.1 Motivation	3
1.2 Zielsetzung	3
1.3 Gliederung	4
2 Grundlagen	5
2.1 Maschinelles Lernen	5
2.2 Klassifizierung	5
2.3 Trainingsalgorithmen	6
2.3.1 Support Vector Machine	7
2.3.2 Entscheidungsbäume	7
2.4 Multiklassen-Probleme	8
2.5 Paarweise Klassifizierung	8
2.6 Präferenzlernen	10
2.7 Labelranking durch paarweisen Vergleich	10
2.8 Selbsttraining	11
2.9 Kreuzvalidierung (Cross-Validation)	12
3 Grundidee und Verfahren zur Verwendung der aus Trainingsset gewonnenen Information	13
3.1 Grundidee	13
3.2 Übersicht der Verfahren zur Anpassung des Trainingssets	14
3.3 Ursprüngliches Label auf beliebiger Position im Ranking	14
3.3.1 Variable relevante Positionsanzahl	15
3.3.2 Prozentsatz der relevanten Positionsanzahl	15
3.4 Ursprüngliches Label an erster Position im Ranking	16
3.4.1 Variable relevante Positionsanzahl	16
3.4.2 Prozentsatz der relevanten Positionsanzahl	16
3.4.3 Prozentsatz der relevanten Positionsanzahl ab der zweiten Position	18
3.4.4 Ranking ignorieren bei falscher Vorhersage erster Position	20
3.5 Verwandte Arbeiten	20
4 Experimente	21
4.1 Datensätze und Datenaufbereitung	21
4.1.1 Herkunft der Datensätze	21
4.1.2 Format und Datenaufbereitung	22
4.2 Experimenten- und Testumgebung	23
4.3 Evaluation	23
4.4 Ergebnisse	24
4.4.1 Einstellungen	24
4.4.2 Visualisierung der Teilergebnisse	24
4.4.3 Übersicht der Ergebnisse	27
5 Zusammenfassung und Ausblick	29
5.1 Zusammenfassung	29
5.2 Ausblick	29
Literaturverzeichnis	30
Anhang	31

1 Einführung

1.1 Motivation

Im Zeitalter ständig wachsender Datenmengen und Berechnungsmöglichkeiten ist es möglich geworden verschiedene Informationsarten in Datenbanken, speziell auch Data Warehouse zu speichern. Zunehmende Digitalisierung von Texten, Bildern, Audio, Automatisierung der Geschäftsprozesse, Aufbewahrung der alltäglichen Prozessinformation wie Einkäufe, Kreditkartentransaktionen, Telefongespräche usw. sorgen für ein schnelles und hohes Wachstum der Datenmenge. Sammlung und Organisierung dieser Daten spielt weiterhin eine große Rolle. Dank der Fortschritte bei der Aufbau von leistungsfähigen Datenbanksystemen und Fortschritten in Speichertechnologien kann regelmäßig weiter anwachsende riesige Datenmenge langfristig gespeichert werden.

Früher war die Datenmenge kleiner, überschaubar und konnte manuell analysiert werden. Heutzutage ist es für einen Menschen entweder kaum mehr möglich oder führt zu sehr hohen Kosten. Deswegen werden effiziente maschinelle Analyse, Erkennung von Mustern und wichtigen Folgerungen immer wichtiger. Sie sollen ermöglichen vorliegende Daten effektiver zu nutzen. Mit dieser Problematik beschäftigt sich Data-Mining und maschinelles Lernen. Ohne diese Gebiete, besonders Data-Mining, würde viel Wissen unbenutzt bleiben.

Data-Mining ist Prozess der Datenanalyse aus verschiedenen Blickwinkeln und Zusammenfassung davon in nützliche Information. Diese Information kann z.B. für die Erhöhung der Einnahmen, Senkung der Kosten benutzt werden. Data-Mining ist aus diesem Grund bei Firmen mit einem stark ausgeprägten Verbraucherfokus sehr beliebt. Man kann somit sowohl Beziehungen zwischen solchen Faktoren wie Preis, Produktpositionierung, Qualität als auch zwischen verschiedenen Käufern, gekauften Produkten ermitteln, was wiederum zu sinnvollen Produktempfehlungen und Umsatzsteigung führen kann. Somit entsteht auch ein Eindruck der personalisierten Angebote für Bedürfnisse von Kunden, was eine emotionale Bindung entwickelt und gute Auswirkung bei Nutzern hat. Diese Techniken tragen zur Erhöhung der Wettbewerbsfähigkeit von Unternehmen, welche sie einsetzen. Das ermöglicht auch Auswirkungen auf Umsatz, Unternehmensgewinn und Kundenzufriedenheit zu bestimmen und diese zu beeinflussen.

Verschiedenste Bereiche wie Medizin, Biologie, Marketing, Bildverarbeitung, Spracherkennung sind heutzutage ohne Ansätze des maschinellen Lernens kaum vorstellbar. Es kann z.B. aufgrund vorheriger Erfahrungen Kreditwürdigkeit einer Person automatisch bestimmt werden. Das erspart der entsprechenden Institution viel Zeit für manuelle Bearbeitung des Falles und Kosten bei einer möglichen menschlichen Fehlentscheidung. In der Medizin kann die Notwendigkeit der maschinellen Analyse und Entscheidung am Beispiel der Organsegmentierung veranschaulicht werden. Ausbildung der Radiologen ist teuer und dauert lange Zeit. Dazu kommen noch häufige Überstunden und damit verbundene Müdigkeit, aufgrund welcher manuelle Segmentierung von Radiologen schlechtere Ergebnisse als maschinell durchgeführte Segmentierung der Organe aufweist. Zusätzlich ist Genauigkeit der maschinellen Bearbeitung natürlich sehr hoch, was z.B. bei der Planung einer Operation ein mächtiges Werkzeug vom Chirurgen ist.

Ein spezieller Fall der Folgerung aufgrund der analysierten Datenmenge ist die Klassifizierung. Bei der vorherig erwähnten Segmentierung wäre das die Zuweisung zu einem Organ, welche basierend auf dem trainierten Modell gemacht wird. Als Eingabedaten dafür können z.B. von mehreren Radiologen durchgeführte Organsegmentierungen benutzt werden. Oder bei dem Beispiel davor zu der Kreditwürdigkeit, wäre eine Entscheidung: kreditwürdig oder nicht kreditwürdig eine erfolgte Klassifizierung. Als Eingabedaten für das Training des entsprechenden Klassifizierers können einheitlich dokumentierte Beschreibungen der vorherigen Fälle dienen.

1.2 Zielsetzung

Im Rahmen dieser Bachelorarbeit sollen Verfahren entwickelt werden, welche der Verbesserung der Vorhersage bei der Klassifizierung beitragen. Dabei wird folgend vorgegangen: ein Klassifizierer wird zuerst auf der Menge der vorhandenen markierten Eingabedaten trainiert. Dann werden die selben Daten mit dem gelernten Klassifizierer klassifiziert. Abgeleitete Information soll der Anpassung der Eingabedaten dienen. Auf dem auf diese Weise modifizierten Datensatz wird ein neuer Klassifizierer trainiert. Es geht dabei um sogenanntes Selbsttraining, da keine weitere Information von "außen" für die Erweiterung vom Datensatz benötigt wird, es wird nur anhand der analysierten bereits vorhandenen Information neues Wissen generiert und dieses entsprechend für das neue Training verwendet. Gewonnener Klassifizierer soll im Falle eines erfolgreichen Verfahrens bessere Ergebnisse bei der Klassifizierung als mit dem originalen Klassifizierer, welcher nur auf original vorhandener Information trainiert wird, vorweisen. Es wird untersucht, ob eine Verbesserung der Vorhersage möglich ist. Einzelne Verfahren werden miteinander verglichen.

Diese Bachelorthesis spezialisiert sich außerdem speziell auf Multiklassen-Problemen. Sie werden wie gewöhnlich binarisiert: aufgeteilt in zwei-Klassen-Probleme. Handhabung davon ist viel intuitiver und einfacher: anstatt direkt eine Entscheidung zu treffen, welche Klasse aus der ganzen Reihe am besten passend ist, ist es einfacher das Problem aufzuteilen und dann zu entscheiden, welche eine von zwei Klassen besser geeignet für eine konkrete Problemstellung ist

[Nam07]. Dazu kommt aber das Problem der Zusammenführung von gewonnenen Ergebnissen. Es gibt verschiedene Strategien für diese Aggregation. Die intuitivste und beliebteste davon heißt Voting: es wird eine Stimme entweder für die eine oder andere Klasse abgegeben. Nach der Abfrage von allen Modellen wird die Klasse mit der höchsten Anzahl der Stimmen vorhergesagt.

1.3 Gliederung

Im Kapitel 2 werden für diese Bachelorarbeit grundlegende Begriffe beschrieben. Sie sollen zum einfachen Einstieg in das Thema dienen und das Gefühl der folgenden Problematik ermitteln. Am Anfang des Kapitels werden etwas allgemeinere wichtige Definitionen gegeben und mit Fortschreitung innerhalb des Kapitels spezifizieren sie sich mehr auf diese Arbeit. Der Fokus liegt dabei hauptsächlich auf maschinellem Lernen, besonders detailliert wird auf die Klassifizierung eingegangen.

Die Grundidee und im Rahmen dieser Bachelorarbeit entwickelte Verfahren werden im Kapitel 3 erläutert. Er beinhaltet Beschreibung und Pseudocode zu jedem Algorithm. Am Ende dieses Kapitels werden verwandte Arbeiten betrachtet.

Rund um das Thema Experimente und alle damit verbundene Einzelheiten geht es im Kapitel 4. Hier werden die Herkunft, das originale Format und die Formattanpassung der Datensätze, die Testumgebung, die Evaluationsmaße und schlussendlich die Ergebnisse detailliert beschrieben und zum großen Teil visualisiert.

Im Kapitel 5 wird diese Arbeit sowohl zusammengefasst, als auch das Verbesserungspotential und Verbesserungsvorschläge eingegangen werden.

2 Grundlagen

In diesem Kapitel werden relevante Grundbegriffe beschrieben.

2.1 Maschinelles Lernen

Maschinelles Lernen ist ein Teilgebiet der Informatik und beschäftigt sich mit Erforschung der Systeme, die aus Erfahrung Wissen generieren. Das bedeutet u.a., dass auf Basis der Erfahrung das zukünftige Verhalten vorhergesagt werden kann. Technologien aus dem maschinellen Lernen ermöglichen große Datenmengen schnell zu bearbeiten, haben hohe Robustheit gegen neue Daten, können Wissen auf andere Domäne übertragen. Dafür werden unterschiedliche Algorithmen eingesetzt (s. 2.3) [Kod88]. Viele Forschungsgebiete wie natürliche Sprachverarbeitung, DNA-Analyse, automatische Diagnoseerkennung, Organsegmentierung, Schrifterkennung wären kaum vorstellbar ohne diese Technologien.

Als Eingabedaten bei unterschiedlichsten Problemen des maschinellen Lernens dienen strukturierte Datensätze. Sie beschreiben die gegebene Aufgabenstellung, z.B. bei einem Organsegmentierungsproblem werden vorhandene Ergebnisse manueller Segmentierung einen Datensatz bilden.

Ein Datensatz besteht aus Instanzen und jede Instanz bildet einen konkreten Fall aus dem gestellten Problem ab. Eine Instanz setzt sich wiederum aus Attribute zusammen. Ein Attribut hat einen Namen und einen Wert und beschreibt eine relevante Tatsache des Problems.

In der Abbildung 1 wird ein Wetter-Golf Problem dargestellt. Der Datensatz beinhaltet Angaben der Wetterverhältnisse an mehreren Tagen. *Play Golf?* ist das Klassenattribut und hat als mögliche Werte *true* und *no*. Es informiert über Spielmöglichkeit an entsprechenden Tagen: bei *true* konnte Golf gespielt werden und bei *false* nicht.

Aufgrund der Erfahrung (Datensatz mit bekannten Klassenattributen) kann für eine neue unbekannte Instanz das Klassenattribut vorhergesagt werden.

Day	Temperature	Outlook	Humidity	Windy	Play Golf?
07-05	hot	sunny	high	false	no
07-06	hot	sunny	high	true	no
07-07	hot	overcast	high	false	yes
07-09	cool	rain	normal	false	yes
07-10	cool	overcast	normal	true	yes
07-12	mild	sunny	high	false	no
07-14	cool	sunny	normal	false	yes
07-15	mild	rain	normal	false	yes
07-20	mild	sunny	normal	true	yes
07-21	mild	overcast	high	true	yes
07-22	hot	overcast	normal	false	yes
07-23	mild	rain	high	true	no
07-26	cool	rain	normal	true	no
07-30	mild	rain	high	false	yes

today	cool	sunny	normal	false	?
tomorrow	mild	sunny	normal	false	?

Abbildung 1: Wetter-Golf Datensatz¹

2.2 Klassifizierung

Klassifizierung nennt man die Kategorisierung der Objekte anhand ihrer Attribute zu Klassen. Sie kann generell in zwei Phasen unterteilt werden: die Trainingsphase und die Testphase.

¹ Quelle: <http://www.ke.tu-darmstadt.de/lehre/archiv/ss11/web-mining/wm-intro.pdf> (S.52)

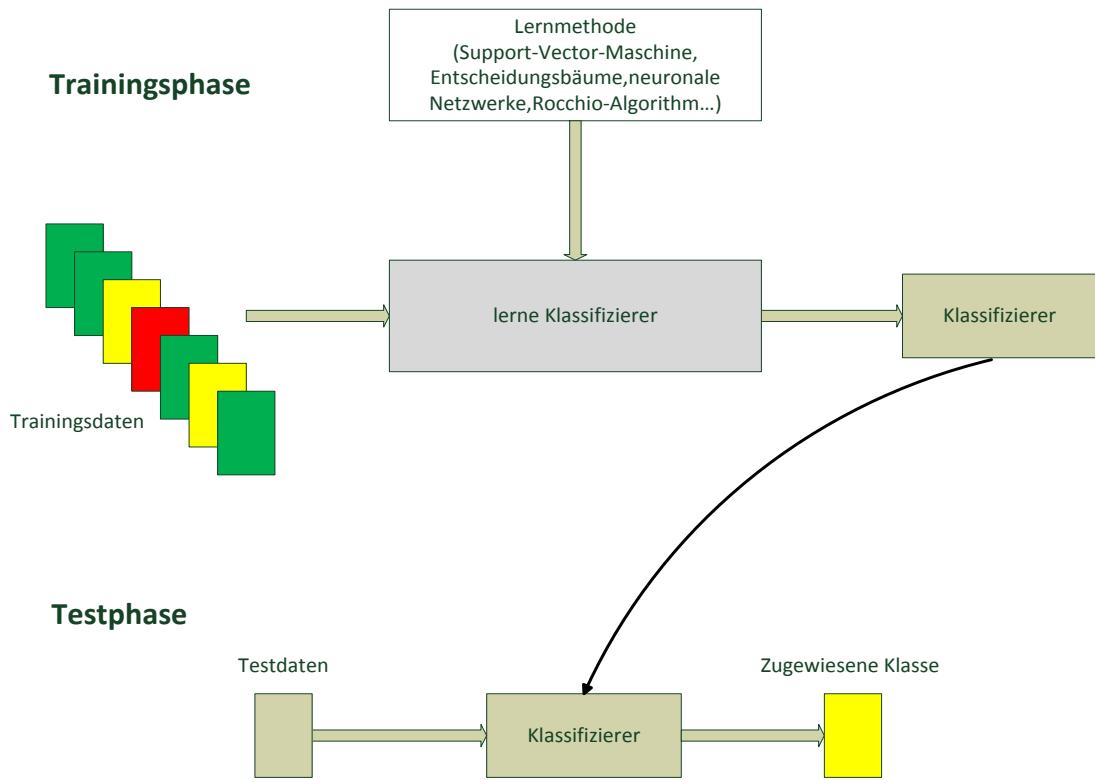


Abbildung 2: Zwei Phasen der Klassifizierung

Erlernen des Zusammenhangs zwischen Attributwerten der Instanzen und deren Klassen ist das Ziel vom Trainingsvorgang. Es gibt unterschiedliche Verfahren, die für die Erstellung des Klassifizierers verwendet werden wie z.B. Naive Bayes, SVM (2.3.1), kNN, Entscheidungsbäume (2.3.2).

In der Trainingsphase bekommt ein ausgewähltes Lernalgorithmus eine Sammlung von Trainingsdaten mit klassifizierten Instanzen und generalisiert sie entsprechend. Der gelernte Klassifizierer ist in der Lage die algorithmababhängig wahrscheinlichste Klasse für eine neue und unbekannte Instanz vorherzusagen.

Ein Testdatensatz ist eine Menge von Instanzen mit bekannten Klassen, die normalerweise zu Trainingsdaten nicht aufgenommen werden. Der Klassifizierer wird auf der Menge der Testinstanzen getestet. Als ein wichtiges Gütekriterium dient die prozentuelle Übereinstimmung der vorhergesagten Klassen mit originalen Klassen der Instanzen (accuracy).

2.3 Trainingsalgorithmen

Alle Algorithmen kann man anhand der vorhandenen Eingabedaten und der gewünschten Ausgabe in folgende Typen aufteilen:

- überwachtes Lernen (supervised learning): Klassenattribute sind bekannt für alle Instanzen aus dem Datensatz. Der Lernalgorithmus versucht eine möglichst gute Abbildung zu finden, welche Eingabewerte zu Ausgabewerten zuordnet. Da das Klassenattribut von Anfang an bekannt ist, kann die Übereinstimmung überwacht werden. Daher kommt der Typname. Nach dem Lernprozess soll der Klassifizierer in der Lage sein für eine unbekannte Instanz ihre Klasse korrekt vorherzusagen.
- Unüberwachtes Lernen (unsupervised learning): Klassenattribute sind nicht bekannt. Algorithmen dieses Problemtyps versuchen Struktur innerhalb der Eingabedaten zu finden. Häufige Anwendungen sind Clustering und Datenkomprimierung.
- Semiüberwachtes Lernen (semi-supervised learning): Klassenattribute sind nur zum Teil bekannt. Das System muss aufgrund der vorhandenen Labels entscheiden welche Merkmale für die Gruppierung der Eingabedaten benutzt werden. Dieser Problemtyp hat wegen seiner hohen Skalierbarkeit und im Vergleich zum überwachten Lernen einfacher Anschaffung von Datensätzen gute Zukunftsaussichten und wird aktuell intensiv erforscht [CSZ⁺06].

- Lernen durch Verstärkung (reinforcement learning): das System besteht aus einem Agenten und seiner Welt. Der Agent lernt aus Belohn- und Bestraferfahrungen als Reaktion des Systems auf seine Aktionen. Das Ziel ist sein Verhalten an die Welt so anzupassen, dass ein maximaler Gewinn erreicht wird [KLM96].
- Lernen des Lernens (learning to learn): lässt sich am besten mit Lernprozessen von einem Menschen zu vergleichen. Ein Mensch bekommt ständig verschiedene Lernaufgaben, welche unterschiedliche Anzahl der Beispiele aufweisen. Oft ist nur ein Beispiel zum Lehren ausreichend, dieses wird vom Menschen auch verallgemeinert. So mit wird für unterschiedliche Aufgaben ein anderer induktiver Bias gelernt: das heißt, es wird gelernt, wie man lernt. Genau so lernt das System seinen eigenen induktiven Bias aufgrund vorheriger Erfahrungen [TP98].
- Kontinuierliches Lernen (developmental learning): das System lernt aus Selbsterforschung und sozialer Interaktion mit menschlichen Lehrern. Es ähnelt der Erforschungs- und Lernvorgehensweisen der Kinder. Dieser Lerntyp spielt eine wichtige Rolle in Robotik [NAH02].

Aufgrund unterschiedlicher Struktur von Datensätzen wurden viele Verfahren entwickelt. Was sich für einen Datensatz als gut geeignet erweist, kann für einen anderen komplett unbrauchbar sein. In folgenden Unterkapiteln werden konkrete Techniken vorgestellt, welche im Rahmen dieser Thesis zur Anwendung kamen.

2.3.1 Support Vector Machine

Eine Support Vector Machine (SVM) ist ein linearer Klassifizierer. Er ist auf Problemen des überwachten Lernens anwendbar. Aufgabe einer SVM ist eine Hyperebene zu finden, die zwei Vektorobjekte aus einem Vektorraum in zwei Klassen teilt. Support Vectors (daher kommt auch der Name des Verfahrens) nennt man die am nächsten liegenden Vektoren zur Hyperbene. Allein diese Vektoren bestimmen die Position der Hyperebene, alle anderen Vektoren haben keinen Einfluss darauf. Zwecks Vermeidung des Overfittings und einer hohen Zuverlässigkeit bei der Klassifizierung von Objekten, die vorher noch nicht gesehen wurden, muss der Abstand zwischen Support Vectors (Margin) maximiert werden [Vap00].

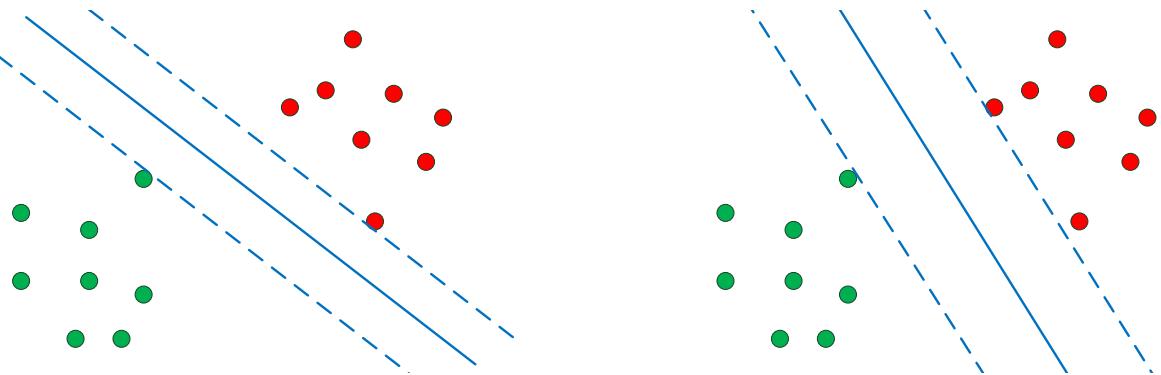


Abbildung 3: SVM: zwei mögliche Trennungen von zwei Klassen

In der Abbildung 3 wird ein Fall mit einem zufällig gewählten Margin (links) und einem maximierten Margin (rechts) dargestellt. Man sieht, dass der Unterschied zwischen den beiden signifikant ist.

2.3.2 Entscheidungsbäume

Ein Entscheidungsbau ist ein gerichteter, geordneter und kreisfreier Baum. Er dient zur Darstellung der Entscheidungsregeln. Der Baum wird ausgehend vom Wurzelknoten traversiert. Eine Entscheidung bringt uns zum Folgerknoten und entspricht einer Testdurchführung für ein Attribut. Kommt man bei einem Blatt an, ist es der gesuchte Klassenwert. Es gibt verschiedene Algorithmen zur Konstruktion der Entscheidungsbäume. Sie werden nach einem Top-down-Prinzip induziert und unterscheiden sich z.B. bei der Auswahl von Attributen, Abbruchskriterien des Induktionsprozesses [PKSR02].

J48 ist eine Implementierung vom Entscheidungsfindungsalgorithmus $c_{4.5}$ und wird in dieser Bachelorarbeit verwendet.

In der Abbildung 4 wird der Entscheidungsbau für das Wetter-Golf Problem repräsentiert. Für die erste noch nicht gesehene Instanz aus der Abbildung 1 bekommen wir den folgenden Pfad: bei der Wurzel namens *Outlook* wird die Verzweigung nach links genommen (für *sunny*), dann befinden wir uns im inneren Knoten namens *Humidity* und nehmen die Verzweigung nach rechts (für *normal*). Somit sind wir bei dem Blatt *yes* angekommen und es ist unser vorhergesagter

Klassenwert für diese Instanz. Wie man auch sieht, wurden nicht alle Attribute bei der Erstellung vom Entscheidungsbaum in Betracht genommen. Das Attribut *Day* kann bei der Suche nach dem Klassenwert komplett ausgelassen werden, unabhängig von der genommenen Verzweigung.

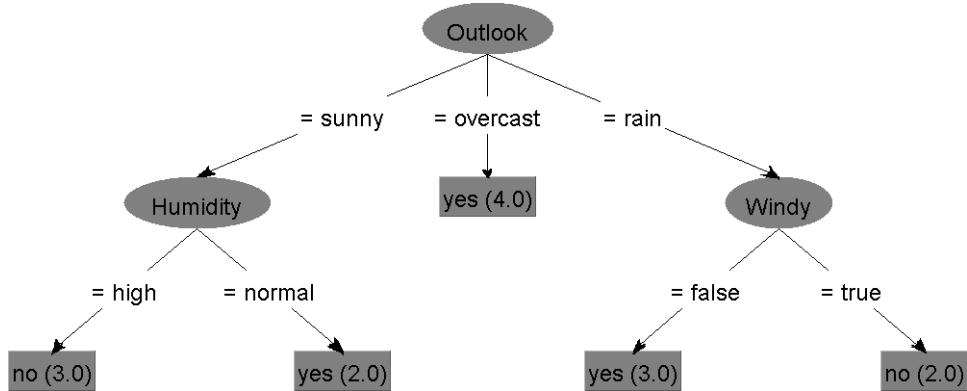


Abbildung 4: Entscheidungsbaum (J48) für das Wetter-Golf Problem²

2.4 Multiklassen-Probleme

Probleme, die über mehr als zwei Klassen verfügen, heißen Multiklassen-Probleme. Für die Lösung dieser Probleme gibt es zwei Vorgehensweisen: man kann entweder direkt einen Klassifizierer nehmen, welcher für Multiklassen-Probleme entwickelt wurde, oder von der Binarisierung (s.2.5) profitieren, um eine größere Auswahl an Klassifizierern zu gewinnen.

2.5 Paarweise Klassifizierung

In der realen Welt hat man meistens mit einem Multiklassen-Problem zu tun. Es kann aber auf mehrere binäre Probleme zurückgeführt werden. Anstatt der direkten Auswahl aus einer Reihe von Labels oder Rankingerstellung anhand ihrer Zweckmäßigkeit ist es viel einfacher zuerst eine Entscheidung bei paarweisen Alternativen zu treffen. Das eigentliche Problem (Auswahl des am besten passenden Labels oder der Rankingerstellung) wird dann im zweiten Schritt basierend auf den Ergebnissen der paarweisen Vergleiche gelöst.

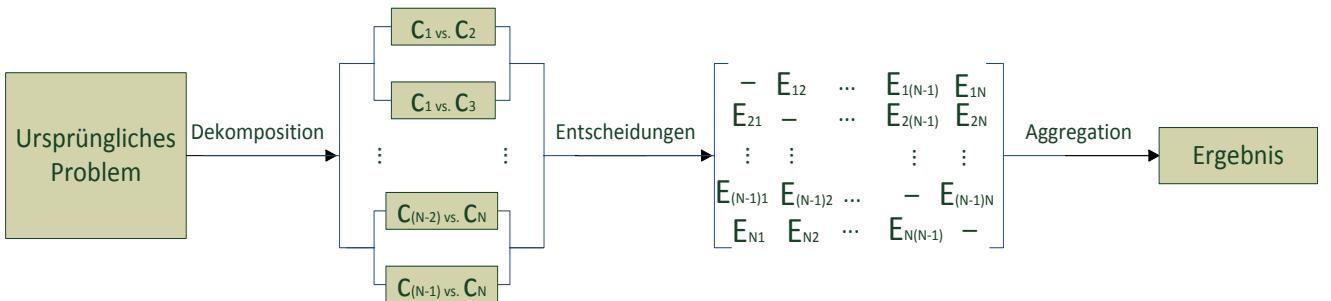


Abbildung 5: Paarweise Klassifizierung

In der Abbildung 5 ist der grobe Verlauf auch noch bildlich dargestellt. Er besteht aus drei Schritten: paarweise Dekomposition, Gewinnung der Entscheidungen und Aggregation. Kommen wir zur detaillierteren Beschreibung.

Zerlegung (Dekomposition) des ursprünglichen Problems in eine Reihe von vermutlich einfacheren Teilproblemen ist nicht nur intuitiv und einfach für Menschen, sondern auch nützlich aus Sicht der automatischen Datenverarbeitung. Ein Multiklassen-Problem wird bei paarweiser Dekomposition, auch Binarisierung genannt, in eine Reihe von zwei-Klassen-Problemen zerlegt. Dabei werden Instanzen aus einem Datensatz für alle möglichen Kombinationen von zwei unterschiedlichen Labelwerten zusammengefasst. Im Vergleich zu alternativen Zerlegungstechniken wie one-against-all, die für jede Klasse ein Modell trainieren, führt paarweise Dekomposition zu maximal einfachen Problemen. Besonders,

² erstellt mit WEKA

was die benötigte Rechenleistung angeht, erweisen paarweise Probleme niedrige Komplexität und haben weniger Trainingsinstanzen im Vergleich zum ursprünglichen Problem, weil nur diejenigen betrachtet werden, die zu einer der beiden Klassen gehören. Diese Probleme haben in der Regel einfachere Entscheidungsgrenzen [Nam07].

Z.B. in einem Datensatz mit Labels c_0, c_1, c_2 und c_3 werden alle Instanzen mit Labels c_0 und c_1 , c_0 und c_2 , c_0 und c_3 , c_1 und c_2 , c_1 und c_3 , c_2 und c_3 separat gruppiert und für jede Gruppe werden alle anderen Labels ignoriert. Somit entstehen sechs dekomponierte Datensätze.

Lernen durch paarweisen Vergleich (learning by pairwise comparison, kurz LPC), auch bekannt als 1-vs-1 oder Round-Robin, ist eine gut etablierte “teile und herrsche“ Technik im Bereich des maschinellen Lernens. Für jedes Paar von Klassen wird ein Klassifizierer gelernt (das darauf angewendete Verfahren nennt man Basislerner). Es entstehen generell $n * (n - 1)/2$ Klassifizierer, wobei n die Anzahl der Klassen von einem Datensatz ist. Wenn wir zum letzten Beispiel zurückkehren, dann werden sechs Klassifizierer gelernt, je ein für die sechs entstandenen dekomposierten Datensätze [FH03].

Am Ende müssen Ergebnisse zusammengesetzt werden. Es gibt verschiedene Möglichkeiten für die Kombination der Vorhersage von einzelnen Klassifikatoren. Einfache und oft verwendete Strategien bei einem Multiklassen-Problem sind folgende:

- Voting: ist die beliebteste Strategie zur Aggregation der Vorhersagen von paarweisen Klassifikatoren. Bei binärer Abstimmung kann jeder Klassifizierer seine Stimme entweder für die erste oder für die zweite Klasse abgeben. Nachdem alle Modelle abgefragt sind, wird die Klasse mit der höchsten Anzahl der Stimmen vorhergesagt.
- Weighted Voting: diese Strategie unterscheidet sich mit der vorherigen durch die Gewichtung von Votes. Ein Votum wird abhängig von der Wahrscheinlichkeitsschätzung unter den zwei entsprechenden Klassen aufgeteilt. Nachdem alle Modelle abgefragt sind, wird die Klasse mit der höchsten Summe der gewichteten Stimmen vorhergesagt.
- Pairwise Coupling: bei vielen Szenarien werden Wahrscheinlichkeitsschätzungen für Klassenwerte gebraucht. Da viele paarweise Klassifizierer Klassenwahrscheinlichkeiten bereitstellen, haben unterschiedliche Autoren Wahrscheinlichkeitsschätzungen durch ihre Kombination vorgeschlagen. Als Beispiel betrachten wir recht intuitive Kombinationsregel von J.H.Friedmans: der zugeordnete Klassenwert entspricht demjenigen, für welchen die meisten paarweisen Vergleiche gewonnen werden. Und zwar werden bei paarweiser Klassifizierung separate Entscheidungsgrenzen unabhängig zwischen jedem Paar von N Klassen ermittelt. Jede der Grenzen wird dann verwendet, um einer unbekannten Instanz eine der beiden Klassen zuzuweisen. Ein Klassenlabel, welches die meisten Zuweisungen bekommt, wird zur vorhergesagten Klasse der unbekannten Instanz gemacht [WLW04].

Pseudocode für die paarweise Klassifizierung sieht folgend aus:

```

input : dataset - the given dataset; testInstance - instance for which the class should be predicted
output: predicted class
1 numberOf Classes := get the number of classes from the dataset;
2 classNames := get the classes names from the dataset;
3 /* training phase */ 
4 /* do actual decomposing: generate for each pair of classes corresponding binary problem */
5 for i := 0 to numberOf Classes - 1 do
6   for j := i + 1 to numberOf Classes do
7     | decomposedDatasetsi,j = get all instances of classNamesi and classNamesj;
8   end
9 end
10 foreach decomposed dataset dds of decomposedDatasets do
11   | models += trained model for dds;
12 end
13 /* prediction phase */ 
14 foreach model m of models do
15   | classify testInstance with m;
16   | increase counter for the predicted class by 1;
17 end
18 return class with the highest counter

```

Algorithm 1: Paarweise Klassifizierung (Pseudocode)

Basierend auf dem Pseudocode kann paarweise Klassifizierung mit einer Wunschprogrammiersprache implementiert werden.

Paarweise Klassifizierung erweist folgende Vorteile:

- Vereinfachung des Problems: kleine Teilprobleme, die einfacher zu lösen sind. Sie erweisen niedrige Komplexität und einfachere Entscheidungsgrenzen.
- Modularität: ermöglicht einfachen Austausch von einzelnen Modulen. Da Lernen und Vorhersage voneinander entkoppelt sind, ist es z.B. möglich auf der Basis von dem selben Lerner verschiedene Arten der Vorhersage auszuprobieren und dann miteinander zu vergleichen.
- Verfügbarkeit von vielen Verfahren, welche für ein zwei-Klassen-Problem geeignet sind.

Nachteile von diesem Verfahren sind folgende:

- hohe Anzahl der Klassifizierer: es gibt $n * (n - 1)/2$ Klassifizierer, welche bei einem Datensatz mit vielen Labels speicherlastig werden können.
- Aggregation als ein weiterer und unnötiger Schritt beim ursprünglichen Problem.

2.6 Präferenzlernen

Eine Präferenz kann als eine schwache Einschränkung betrachtet werden, die gegebenenfalls bis zu einem gewissen Grad verletzt werden kann. Präferenzlernen ist ein Teilgebiet des maschinellen Lernens, in welchem Lernen des vorausschauenden Präferenzmodells aufgrund der betrachteten Präferenzinformation induktiv untersucht wird. Es wird versucht aus einer Menge von Objekten mit bekannten Präferenzen eine Funktion zu lernen, welche in der Lage ist, Präferenzen für eine neue Menge oder für die selbe Menge aber im anderen Kontext vorherzusagen. Ein besonderer Schwerpunkt von Präferenzlernen ist Lernen zu ordnen (learning to rank), welcher zur Zeit ein der am intensivsten untersuchten Problemen der Präferenzlernenaufgaben ist. Aufgrund der Unterschiede zwischen Präferenzinformationen, werden Aufgabenstellungen in drei Hauptkategorien unterteilt: Labelranking, Instanzranking und Objektranking [CG05], [FH03].

2.7 Labelranking durch paarweisen Vergleich

Labelranking ist eine Lernaufgabe, welche aus der Induktion einer Abbildungsfunktion von gegebenen Instanzen mit einer endlichen Anzahl der Labels auf Ranking besteht. Sei eine gegebene Instanz hat folgende Präferenzen: $a > b$ und $a > c$. Wir wissen aber nicht, in welcher Relation b und c zueinander stehen. Unsere Annahme ist, dass es ein wahres Ranking von Präferenzen gibt und das Ziel ist dieses möglichst genau zu bestimmen.

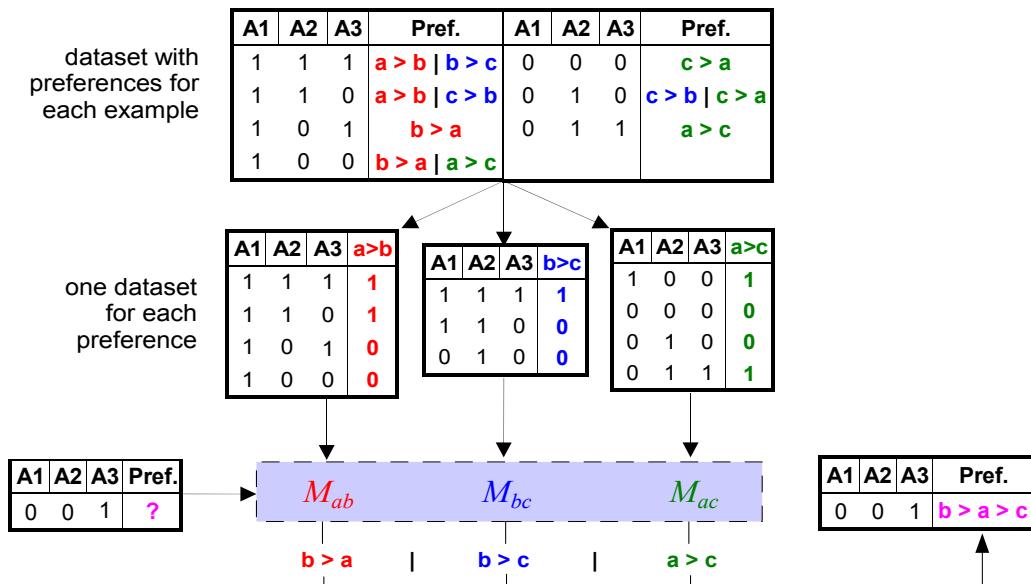


Abbildung 6: Lernen durch paarweisen Vergleich (Quelle [FH03])

In der Abbildung 6 repräsentiert die obige Tabelle ein gegebenes Trainingsset. Es besteht aus Instanzen und entsprechend zugehörigen Präferenzen. Bei einem sogenannten Ranking durch paarweisen Vergleich wird für jede Präferenz ein

entsprechender Datensatz erfasst, wobei Instanzen mit z.B. folgenden Präferenzen: $a > b$ und $b > a$ zum selben Datensatz gehören. Somit wird der ursprüngliche Datensatz in $n * (n - 1) / 2$ Datensätze, in unserem Fall in drei (drei Tabellen im mittleren Bereich der Abbildung) aufgeteilt, wobei n die Anzahl der einzigartigen Präferenzen (mit Ausnahme von gleichen Labels, die in umgekehrter Relation zueinander stehen) darstellt. Anschließend wird es für jeden der gewonnenen Datensätze je ein Modell trainiert: M_{ab} , M_{bc} , M_{ac} . Im untersten Teil der Abbildung werden für eine Testinstanz alle Modelle abgefragt und wir bekommen folgende Vorhersagen: $b > a$, $b > c$, $a > c$. Für die Aggregation davon kann z.B. Voting benutzt werden. Wir bekommen folgende Ergebnisse: b hat zwei Stimmen, a hat eine Stimme und c hat keine Stimmen. Aufgrund der Abstimmung bekommen wir für die Testinstanz das folgende vorhergesagte Ranking: $b > a > c$.

Diese Vorgehensweise ähnelt sich der paarweisen Klassifizierung.

2.8 Selbsttraining

Selbsttraining ist wahrscheinlich die älteste Technik zur Benutzung der Daten ohne Labels bei der Klassifizierung. Die Idee ist im Jahr 1965 in der Literatur bereits erschienen [SPP10]. Selbsttraining ist ein häufig verwendetes Verfahren des semi-überwachten Lernens, bei welchem ein Klassifizierer zuerst auf der Menge der markierten Daten trainiert wird und dann wird er zur Klassifizierung der unmarkierten Daten benutzt. Anschließend werden unmarkierte Instanzen zusammen mit ihren vorhergesagten Labels zum Trainingsset hinzugefügt und dabei sind verschiedene Szenarien möglich:

- es werden alle Instanzen übernommen;
- es werden nur die besten Instanzen in das Trainingsset aufgenommen;
- Instanzen werden gewichtet hinzugefügt.

Meistens wird die zweite Option bevorzugt. Dann wird ein neuer Klassifizierer auf dem bekannten Datensatz trainiert und die komplette Prozedur wiederholt sich wieder. Wichtig ist, dass der Klassifizierer seine eigenen Vorhersagen benutzt, um sich selbst zu lehren. Deswegen wird diese Technik auch als self-training, self-labeling oder bootstrapping genannt. Das Hinzufügen der neuen Instanzen zum Trainingsset kann die Genauigkeit der Vorhersage verschlechtern, deswegen werden bei manchen Algorithmen Instanzen, welche zur Verschlechterung geführt haben, im nächsten Schritt aus dem Trainingsset entfernt und nicht mehr dazu hinzugefügt.

Diese Technik hat wegen ihrer hohen Skalierbarkeit und im Vergleich zu Verfahren des überwachten Lernens einfacherer Anschaffung von Datensätzen gute Zukunftsaussichten und wird aktuell intensiv erforscht. Selbsttraining wird intensiv z.B. bei der maschinellen Übersetzung, Objekterkennung auf Bildern und beim Parsing eingesetzt.

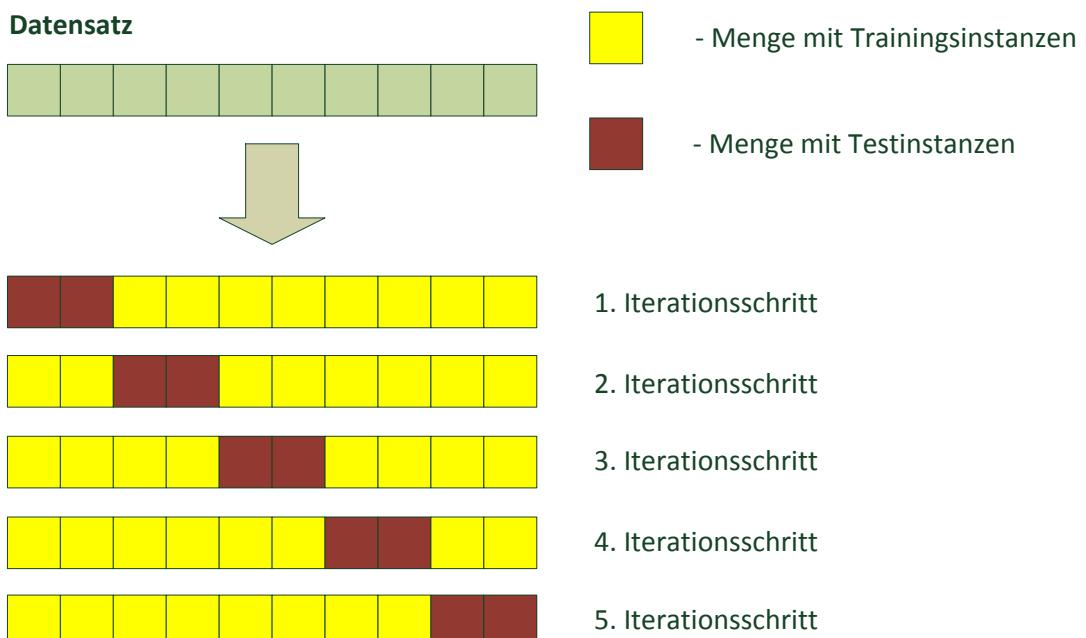


Abbildung 7: Fünffache Kreuzvalidierung

2.9 Kreuzvalidierung (Cross-Validation)

Diese Technik hat folgenden Hintergrund: man möchte die gelernten Modelle evaluieren. Für die Evaluierung braucht man einen geeigneten Datensatz, der homogen zu dem Trainingsset ist. Eine einfache und intuitive Lösung ist die Aufteilung des vorhandenen Datensatzes in mehrere.

Bei Kreuzvalidierung wird die zu untersuchende Datenmenge in N möglichst gleich große Teilmengen zerlegt, wobei sie miteinander disjunkt sind. Dann wird ein Teil von diesen Teilmengen zum Testen und das andere zum Trainieren benutzt, wobei die Testmenge in der Regel kleiner als die Trainingsmenge ist. Diese Vorgehensweise wird N -mal wiederholt. Man kann auch den kompletten Vorgang mit einer anderen Mengenaufteilung mehrmals wiederholen [K⁺95].

In der Abbildung 7 wird eine mögliche Aufteilung in Trainings- und Testmenge für eine fünffache Kreuzvalidierung dargestellt. Sie kann auch anders erfolgen.

3 Grundidee und Verfahren zur Verwendung der aus Trainingsset gewonnenen Information

3.1 Grundidee

Die Idee ist zu überprüfen, inwieweit sich ein paarweiser Klassifizierer oder Ranker verbessern lässt, indem man die Information aus den Rankings berücksichtigt.

Zur Veranschaulichung nehmen wir uns ein einfaches Beispiel: ein paarweiser Klassifizierer bzw. Ranker lernt aus Trainingsdaten, eine Menge von Labels $\{A, B, C, D, E\}$ in eine Reihung zu bringen. Dabei wird für jedes Paar der Labels ein Klassifizierer trainiert. In unserem Fall gibt es Klassifizierer für $A > B, A > C, A > D, A > E, B > C, B > D, B > E, C > D, C > E$ und $D > E$. Um für ein Beispiel eine Reihung vorherzusagen, wird jeder dieser 10 Klassifizierer abgefragt, und dann wird die endgültige Ordnung der Klassen durch Abstimmung bestimmt.

Dabei kann es vorkommen, dass die endgültige Ordnung nicht mit den Vorhersagen übereinstimmt. Wenn sich z.B. $A > B, A > C, A > D, E > A, B > C, D > B, B > E, C > D, C > E$ und $D > E$ ergibt, dann hat A 3 Stimmen, B 2 Stimmen, C 2 Stimmen, D 2 Stimmen und E 1 Stimme. Es wird eine Ordnung $A > B, C, D > E$ vorhergesagt. Jetzt hat aber der binäre Klassifizierer für das Klassenpaar A und E vorhergesagt, dass $E > A$ gilt, während in der durch Voting entstandenen Reihung $A > E$ herausgekommen ist. Es kann sogar vorkommen, dass in den Trainingsbeispielen gegebene Präferenzen nach dem Lernen entgegengesetzt werden.

Die Idee für die Thesis ist zu untersuchen, ob man die Trainingsdaten korrigieren kann, indem man:

- in den Trainingsdaten gegebene Präferenzen, die einer gelernten Reihung widersprechen, „korrigiert“ und dann nochmals trainiert;
- Präferenzen, die in den Trainingsdaten nicht auftreten, aber in der ersten Iteration gelernt werden, werden ganz (oder zum Teil) dem ursprünglichen Trainingsset hinzugefügt.

Beides kann man auch mehrmals iterieren. Experimentell muss man Ergebnisse dann mit dem klassischen Ansatz ohne Korrektur vergleichen, bzw. mit einem Ansatz, der die Beispiele nur aufgrund des Lernergebnisses der binären Klassifizierer korrigiert (ohne die Reihungen zu berücksichtigen).

Diese Ideen sind verwandt zu „Noise Elimination“ in der Klassifikation bzw. „Selbsttraining“ beim Aktiven Lernen.

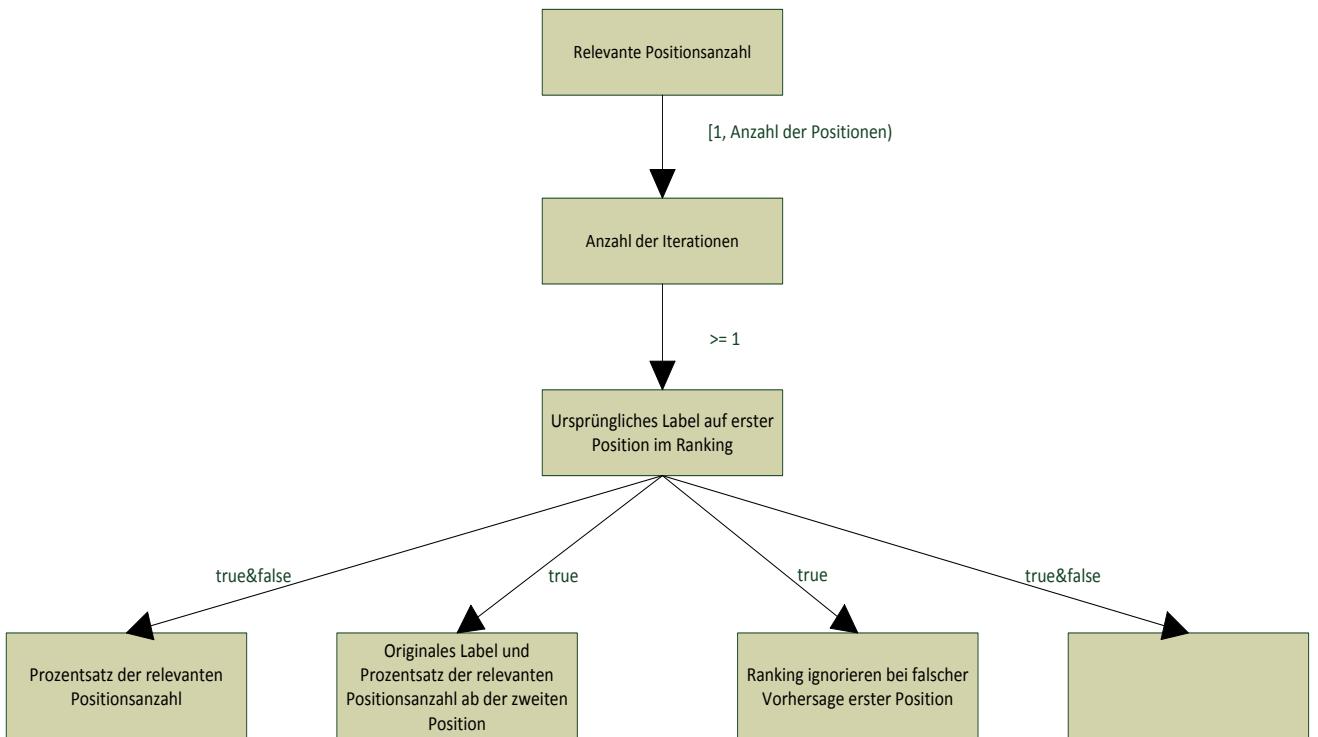


Abbildung 8: Verfahrensübersicht

3.2 Übersicht der Verfahren zur Anpassung des Trainingssets

Es gibt mehrere Verfahren, die in dieser Bachelorthesis untersucht werden, manche davon sind miteinander kombinierbar. Es gibt gemeinsame Einstellungsparameter, die bei allen Vorgehensweisen gleich sind. Sie werden hier zuerst genauer erläutert.

“Relevante Positionsanzahl“ ist das erste gemeinsame Parameter und symbolisiert die Anzahl der höchstrangigen Labels vom gewonnenen Ranking, die zu paarweisen Vergleichen umgewandelt werden.

“Anzahl der Iterationen“ zeigt wie oft das aktuelle Verfahren auf dem Trainingsset wiederholt wird.

Abhängig davon, ob das ursprüngliche Label sich immer an der ersten Position des Rankings befinden muss oder nicht, werden untersuchte Verfahren logisch aufgeteilt. In der Abbildung 8 ist schematische Übersicht der Verfahren dargestellt.

Pseudocode (s. Algorithm 2) beschreibt einen abstrakten Ablauf zur Realisierung der im Abschnitt 3.1 erklärten Idee. In jedem Iterationsschritt einer Kreuzvalidierung wird der geladene Datensatz in Trainings- und Testset aufgeteilt. Das gebildete Trainingsset wird binarisiert und auf den zwei-Klassen-Problemen werden Modelle trainiert. Dann kommen wir zum Teil, welches sich abhängig von später in diesem Kapitel beschriebenen Verfahren entsprechend unterscheidet. Deswegen wurde dieser Schritt im Pseudocode möglichst abstrakt gehalten und entsprechend markiert. Nach der entsprechenden Anpassung vom Trainingsset wird es noch mal binarisiert und auf den neu gewonnenen zwei-Klassen-Problemen werden Modelle trainiert. Mit diesen Modellen wird das Testset klassifiziert und anschließend wird evaluiert, wie gut diese Klassifizierung war. Nachdem alle Iterationsschritte zu Ende sind, wird das Ergebnis der Evaluierung zurückgegeben.

```
input : dataset - the given dataset; numberOfFolds - number of folds for cross-validation
output: evaluation results of classification
1 for fold := 0 to numberOfFolds do
2   | trainingSet := get the training set from the dataset;
3   | testSet := get the test set from the dataset;
4   | /* do decomposing of the trainingSet */ *
5   | decomposedDatasets = for each pair of classes generate a corresponding problem;
6   | /* do the classification of each decomposed dataset with a binary classifier */ /
7   | foreach decomposed dataset dds of decomposedDatasets do
8     |   | models += trained model for dds
9   | end
10  | /* this part is depending on the chosen algorithm */ /
11  | /*
12  | improvedTrainingSet := get improved training set(trainingSet, models);
13  | /*
14  | /* do decomposing of the improvedTrainingSet */ /
15  | improvedDecomposedDatasets = for each pair of classes generate a corresponding problem;
16  | /* do the classification of each improved decomposed dataset with a binary classifier */ /
17  | foreach improved decomposed dataset idds of improvedDecomposedDatasets do
18    |   | improvedModels += trained model for idds
19  | end
20  | foreach test instance ti of testSet do
21    |   | classify ti with improvedModels;
22  | end
23  | evaluationResults += evaluation of the classification of testSet;
24 end
25 return evaluationResults
```

Algorithm 2: Allgemeiner Ablauf (Pseudocode)

Kommen wir zu detaillierter Beschreibung der einzelnen Algorithmen.

3.3 Ursprüngliches Label auf beliebiger Position im Ranking

Bei dieser Vorgehensweise nimmt man keine Rücksicht auf das ursprüngliche Label und es wird ausschließlich das gewonnene Ranking in Betracht genommen. Dabei ist die Anzahl der Iterationen gleich eins, weil dieses Verfahren sich am Anfang, mit wenigen getesteten Datensätzen, nicht bewährt hat.

Diese Vorgehensweise hat folgende Variationen.

3.3.1 Variable relevante Positionsanzahl

Bei diesem Verfahren werden Ergebnisse abhängig von unterschiedlicher relevanten Positionsanzahl untersucht. Man geht dabei folgend vor: man nimmt entsprechend zur Anzahl der relevanten Positionen die höchstrangigen Labels und wandelt sie zu paarweisen Vergleichen um.

Schauen wir uns ein Beispiel an: sei $[c_0, c_1, c_2, c_3]$ die gewonnene totale Ordnung von einer beliebigen Instanz. Dann bekommen wir $c_1 < c_0$, $c_2 < c_0$, $c_3 < c_0$ als paarweise Vergleiche bei relevanter Positionsanzahl ist gleich eins, bei zwei kommen noch $c_2 < c_1$, $c_3 < c_1$ dazu, bei drei kommt $c_3 < c_2$ dazu. Weitere Erhöhung von diesem Parameter wird nichts ändern, da es keine weiteren Labels gibt, die im Ranking nach c_3 stehen. Damit variiert sich dieser Wert zwischen eins und der Anzahl der Positionen im Ranking.

```
input : trainingSet - the training set to improve; models - models of decomposed datasets;
        numberOfRelevantPositions - number of relevant for ranking positions
output: improved training set
1 classifiedInstances := classify instances from the trainingSet with models;
2 foreach classified instance ci of classifiedInstances do
3   orderedVotes := order votes of the ci;
4   for position := 0 to numberOfRelevantPositions do
5     pairwisePreferences := converted total order of orderedVotes to pairwise preferences corresponding
      to value of position;
6     foreach pairwise preference pp of pairwisePreferences do
7       newInstance := copy of ci;
8       replace class attribute of newInstance by pp;
9       trainingSet += newInstance;
10    end
11  end
12  remove ci from the trainingSet;
13 end
14 return trainingSet
```

Algorithm 3: Variable relevante Positionsanzahl (Pseudocode)

Der obige Pseudocode veranschaulicht den programmatischen Ablauf von diesem Verfahren: es werden zuerst alle Instanzen aus dem Trainingsset anhand der bereits trainierten Modelle klassifiziert. Dadurch entstehen verschiedene Voteswerte für einzelne Klassen. Mit der Anordnung von diesen gewinnen wir gleichzeitig die totale Ordnung der Labels für die entsprechend zugehörige Instanz.

Abhängig von der relevanten Positionsanzahl, bekommt man unterschiedlich viele paarweise Präferenzen für jede Instanz, welche dann neue Instanzen mit alten Attributwerten (sind Attributwerte der aktuell untersuchten Instanz gemeint) und neu gesetzter Klasse (entsprechende paarweise Präferenz) bilden. Sie werden zum Trainingsset hinzugefügt und die Instanz mit ursprünglichem Klassenlabel wird aus dem Trainingsset entfernt. Auf diese Weise modifiziertes Trainingsset wird zurückgegeben.

3.3.2 Prozentsatz der relevanten Positionsanzahl

Dieses Verfahren unterscheidet sich mit dem Vorherigen dadurch, dass man einen Prozentsatz von den entstandenen paarweisen Vergleichen zufällig auswählt. Verbleibend beim letzten Beispiel mit der relevanten Positionsanzahl ist gleich zwei, haben wir insgesamt folgende Vergleiche: $c_1 < c_0$, $c_2 < c_0$, $c_3 < c_0$, $c_2 < c_1$, $c_3 < c_1$. Bei einem Prozentsatz von 60 werden drei von fünf existierenden übernommen, z.B. $c_1 < c_0$, $c_3 < c_0$, $c_2 < c_1$. Bei diesem Verfahren werden mehrere Prozentsätze pro jede relevante Positionsanzahl untersucht.

Beim Algorithm 4 wird Pseudocode für dieses Verfahren abgebildet. Auch hier werden zuerst alle Instanzen aus dem Trainingsset anhand der bereits trainierten Modelle klassifiziert. Dadurch entstehen verschiedene Voteswerte für einzelne Klassen. Mit der Anordnung von diesen gewinnen wir gleichzeitig die totale Ordnung der Labels für die entsprechend zugehörige Instanz.

Aufgrund des Prozentsatzes und der Anzahl der paarweisen Präferenzen wird berechnet, wie viele Präferenzen beibehalten werden. Anschließend werden neue Instanzen mit alten Attributwerten (sind Attributwerte der aktuell untersuchten Instanz gemeint) und neu gesetzter Klasse (zufällig ausgewählter paarweiser Präferenz) zum Trainingsset hinzugefügt. Die Instanz mit ursprünglichem Klassenlabel wird aus dem Trainingsset entfernt. Am Ende wird das modifizierte Trainingsset zurückgegeben.

```

input : trainingSet - the training set to improve; models - models of decomposed datasets;
        number Of Relevant Positions - number of relevant for ranking positions; percents To Take - percents
        of preferences to be random taken
output: improved training set
1 classifiedInstances := classify instances from the trainingSet with models;
2 foreach classified instance ci of classifiedInstances do
3   orderedVotes := order votes of the ci;
4   for position := 0 to number Of Relevant Positions do
5     pairwisePreferences := converted total order of orderedVotes to pairwise preferences corresponding
      to value of position;
6   end
7   numberOfPreferencesToTake := number of pairwisePreferences * percents To Take / 100;
8   for takenPreferenceNumber := 0 to numberOfPreferencesToTake do
9     randomNumber := get the number with the upper limit of number of pairwisePreferences;
10    newInstance := copy of ci;
11    replace class attribute of newInstance by pairwisePreferences[randomNumber];
12    trainingSet += newInstance;
13  end
14  remove ci from the trainingSet;
15 end
16 return trainingSet

```

Algorithm 4: Prozentsatz der relevanten Positionsanzahl (Pseudocode)

3.4 Ursprüngliches Label an erster Position im Ranking

Bei dieser Vorgehensweise wird überprüft, ob das höchstrangige Label vom gewonnenen Ranking mit dem ursprünglichen Label übereinstimmt. Wenn das der Fall ist, wird nichts verändert, ansonsten wird die totale Ordnung bezüglich dem originalen Label angepasst und zwar so, dass es auf die erste Position gesetzt wird und die restlichen Labels um eins nach rechts in der Positionsordnung verschoben werden.

Zur Veranschaulichung sehen wir uns ein Beispiel dazu an: sei das ursprüngliche Label ist gleich c_0 und die gewonnene totale Ordnung ist gleich $[c_1, c_2, c_0, c_3]$. Da c_0 ungleich c_1 ist, muss das Ranking bezüglich c_0 verändert werden und wir bekommen $[c_0, c_1, c_2, c_3]$ als neue totale Ordnung.

Diese Vorgehensweise hat folgende Variationen.

3.4.1 Variable relevante Positionsanzahl

Bei diesem Verfahren werden Ergebnisse abhängig von unterschiedlicher relevanten Positionsanzahl untersucht. Es gibt keine Unterschiede im Vergleich zum obig beschriebenen gleichnamigen Verfahren (s. 3.3.1 auf der vorherigen Seite), außer der Position vom ursprünglichen Label natürlich.

Bei diesem Verfahren ist die Anzahl der Iterationen gleich eins, weil es sich am Anfang, mit wenigen getesteten Datensätzen, nicht bewährt hat.

Der einzige Unterschied im Pseudocode (s. Algorithm 5) im Vergleich zum Pseudocode vom Algorithm 3 besteht darin, dass Ranking bei der falschen Vorhersage vom höchstrangigen Label angepasst wird. Und zwar wird das originelle Label auf die erste Position gesetzt und alle anderen werden entsprechend um eine Position “tiefer” versetzt.

3.4.2 Prozentsatz der relevanten Positionsanzahl

Dieses Verfahren unterscheidet sich mit dem Vorherigen dadurch, dass man einen Prozentsatz von den entstandenen paarweisen Vergleichen zufällig auswählt. Es gibt keine Unterschiede im Vergleich zum obig beschriebenen gleichnamigen Verfahren (s. 3.3.2 auf der vorherigen Seite), außer der Position vom ursprünglichen Label natürlich.

Bei diesem Verfahren ist die Anzahl der Iterationen gleich eins, weil es sich am Anfang, mit wenigen getesteten Datensätzen, nicht bewährt hat.

Der einzige Unterschied im Pseudocode (s. Algorithm 6) im Vergleich zum Pseudocode vom Algorithm 4 besteht darin, dass Ranking bei der falschen Vorhersage vom höchstrangigen Label angepasst wird. Und zwar wird das originelle Label auf die erste Position gesetzt und alle anderen werden entsprechend um eine Position “tiefer” versetzt.

```

input : trainingSet - the training set to improve; models - models of decomposed datasets;
        numberOfRelevantPositions - number of relevant for ranking positions
output: improved training set
1 classifiedInstances := classify instances from the trainingSet with models;
2 foreach classified instance ci of classifiedInstances do
3   | orderedVotes := order votes of the ci;
4   | highestVotedLabel := get label with the highest vote;
5   | originalLabel := get the original label of ci;
6   | if highestVotedLabel != originalLabel then
7     |   | set originalLabel to the first position at the orderedVotes and shift other labels one position down
8       |   | accordingly;
9   | end
10  | for position := 0 to numberOfRelevantPositions do
11    |   | pairwisePreferences := converted total order of orderedVotes to pairwise preferences corresponding
12      |   | to value of position;
13    |   | foreach pairwise preference pp of pairwisePreferences do
14      |   |   | newInstance := copy of ci;
15      |   |   | replace class attribute of newInstance by pp;
16      |   |   | trainingSet += newInstance;
17    |   | end
18  | end
19  | remove ci from the trainingSet;
20 end
21 return trainingSet

```

Algorithm 5: Variable relevante Positionsanzahl (Pseudocode)

```

input : trainingSet - the training set to improve; models - models of decomposed datasets;
        numberOfRelevantPositions - number of relevant for ranking positions; percentsToTake - percents
        of preferences to be random taken
output: improved training set
1 classifiedInstances := classify instances from the trainingSet with models;
2 foreach classified instance ci of classifiedInstances do
3   | orderedVotes := order votes of the ci;
4   | highestVotedLabel := get label with the highest vote;
5   | originalLabel := get the original label of ci;
6   | if highestVotedLabel != originalLabel then
7     |   | set originalLabel to the first position at the orderedVotes and shift other labels one position down
8       |   | accordingly;
9   | end
10  | for position := 0 to numberOfRelevantPositions do
11    |   | pairwisePreferences := converted total order of orderedVotes to pairwise preferences corresponding
12      |   | to value of position;
13  | end
14  | numberOfPreferencesToTake := number of pairwisePreferences * percentsToTake / 100;
15  | for takenPreferenceNumber := 0 to numberOfPreferencesToTake do
16    |   | randomNumber := get the number with the upper limit of number of pairwisePreferences;
17    |   | newInstance := copy of ci;
18    |   | replace class attribute of newInstance by pairwisePreferences[randomNumber];
19    |   | trainingSet += newInstance;
20  | end
21  | remove ci from the trainingSet;
22 end
23 return trainingSet

```

Algorithm 6: Prozentsatz der relevanten Positionsanzahl (Pseudocode)

3.4.3 Prozentsatz der relevanten Positionsanzahl ab der zweiten Position

Hier werden zuerst alle paarweisen Vergleiche für die relevante Positionsanzahl gebildet. Alle davon, die als präferiertes Label das originale Label haben, werden komplett übernommen. Paarweise Vergleiche bezüglich der Labels ab der zweiten Position im Ranking werden entsprechend zum Prozentsatz für jede Position zufällig beibehalten.

Zur Veranschaulichung sehen wir uns ein Beispiel dazu an: sei das ursprüngliche Label ist gleich c_0 , die relevante Positionsanzahl ist gleich drei, der Prozentsatz für die zweite Position beträgt 70, für die dritte 60 und das gewonnene Ranking ist gleich $[c_1, c_2, c_0, c_3, c_4]$. Da $c_0 \neq c_1$ ist, muss das Ranking zuerst bezüglich c_0 verändert werden und wir bekommen $[c_0, c_1, c_2, c_3, c_4]$ als neue totale Ordnung.

Wir haben insgesamt folgende paarweise Vergleiche: $c_1 < c_0, c_2 < c_0, c_3 < c_0, c_4 < c_0, c_2 < c_1, c_3 < c_1, c_4 < c_1, c_3 < c_2, c_4 < c_2$.

$c_1 < c_0, c_2 < c_0, c_3 < c_0, c_4 < c_0$ werden komplett übernommen. Für die zweite Position haben wir 70 als entsprechender Prozentsatz, deswegen können nur zwei von drei vorhandenen paarweisen Vergleichen beibehalten werden, z.B. $c_2 < c_1$ und $c_4 < c_1$. Für die dritte Position haben wir 60 als entsprechender Prozentsatz, deswegen kann eins von zwei vorhandenen paarweisen Vergleichen beibehalten werden, z.B. $c_3 < c_2$. Somit wird insgesamt folgendes übernommen: $c_1 < c_0, c_2 < c_0, c_3 < c_0, c_4 < c_0, c_2 < c_1, c_4 < c_1, c_3 < c_2$.

Dieses Verfahren wurde mit unterschiedlicher Anzahl an Iterationen untersucht.

```

input : trainingSet - the training set to improve; models - models of decomposed datasets;
        percentsAfterFirstPosition - values of percents to take after first position (ordered);
        numberOfIterations - number of iterations to be made
output: improved training set
1 for iteration := 0 to numberOfIterations do
2   classifiedInstances := classify instances from the trainingSet with models;
3   foreach classified instance ci of classifiedInstances do
4     orderedVotes := order votes of the ci;
5     highestVotedLabel := get label with the highest vote;
6     originalLabel := get the original label of ci;
7     if highestVotedLabel != originalLabel then
8       | set originalLabel to the first position at the orderedVotes and shift other labels one position down
       | accordingly;
9     end
10    localPairwisePreferences := get the pairwise preferences for the originalLabel;
11    /* add instances with pairwise preferences, relating to the original label, as classes
       to the training set */*
12    foreach pairwise preference pp of localPairwisePreferences do
13      | newInstance := copy of ci;
14      | replace class attribute of newInstance by pp;
15      | trainingSet += newInstance;
16    end
17    /* handle all other preferences */*
18    for position := 0 to percentsAfterFirstPosition do
19      | localPairwisePreferences := get pairwise preferences for the position's + 1 label in the ranking;
20      | numberOfPreferencesToTake := number of localPairwisePreferences *
21      | percentsAfterFirstPosition[position] / 100;
22      | for takenPreferenceNumber := 0 to numberOfPreferencesToTake do
23        |   | randomNumber := get the number with the upper limit of number of pairwisePreferences;
24        |   | newInstance := copy of ci;
25        |   | replace class attribute of newInstance by localPairwisePreferences[randomNumber];
26        |   | trainingSet += newInstance;
27      |   end
28    | end
29    | remove ci from the trainingSet;
30  end
31 end
32 return trainingSet
```

Algorithm 7: Prozentsatz der relevanten Positionsanzahl ab der zweiten Position (Pseudocode)

Algorithm 7 stellt Pseudocode zu diesem Verfahren dar. Auch hier werden zuerst alle Instanzen aus dem Trainingsset anhand der bereits trainierten Modelle klassifiziert. Dadurch entstehen verschiedene Voteswerte für einzelne Klassen. Mit der Anordnung von diesen gewinnen wir gleichzeitig die totale Ordnung der Labels für die entsprechend zugehörige Instanz.

Es wird zuerst sichergestellt, dass das originale Label sich an der ersten Position des Rankings befindet und anschließend werden alle Instanzen mit paarweisen Präferenzen als Klassenattribute dafür übernommen. Ab der zweiten Rankingsposition wird die Anzahl zu behaltenden Präferenzen anhand des Prozentsatzes und der Anzahl von möglichen paarweisen Präferenzen für dieses Label berechnet. Sie werden zufällig ausgewählt und als neue Klassenlabels für die Kopie der ursprünglichen Instanz gesetzt. Auf diese Weise gebildete Instanzen werden zum Trainingsset hinzugefügt und die ursprüngliche Instanz wird aus dem Trainingsset entfernt. Das neue Trainingsset wird binarisiert und für neue zwei-Klassen-Probleme werden Modelle trainiert.

Im Gegensatz zu allen vorher beschriebenen Verfahren wird diese komplette Prozedur abhängig von *numberOfIterations* mehrmals wiederholt (mit neu trainierten Modellen und modifiziertem Trainingsset). Am Ende wird das gewonnene Trainingsset zurückgegeben.

```

input : trainingSet - the training set to improve; models - models of decomposed datasets;
        numberOfRelevantPositions - number of relevant for ranking positions; numberOfIterations -
        number of iterations to be made
output: improved training set
1 for iteration := 0 to numberOfIterations do
2   classifiedInstances := classify instances from the trainingSet with models;
3   foreach classified instance ci of classifiedInstances do
4     orderedVotes := order votes of the ci;
5     highestVotedLabel := get label with the highest vote;
6     originalLabel := get the original label of ci;
7     if highestVotedLabel != originalLabel then
8       localPairwisePreferences := get the pairwise preferences for the originalLabel;
9       /* add instances with pairwise preferences, relating to the original label, as
          classes to the training set */*
10      foreach pairwise preference pp of localPairwisePreferences do
11        newInstance := copy of ci;
12        replace class attribute of newInstance by pp;
13        trainingSet += newInstance;
14      end
15      set originalLabel to the first position at the orderedVotes and shift other labels one position down
          accordingly;
16    end
17  else
18    /* handle all preferences for labels from ranking until numberOfRelevantPositions
       */
19    for position := 0 to numberOfRelevantPositions do
20      pairwisePreferences := converted total order of orderedVotes to pairwise preferences
          corresponding to value of position;
21      foreach pairwise preference pp of pairwisePreferences do
22        newInstance := copy of ci;
23        replace class attribute of newInstance by pp;
24        trainingSet += newInstance;
25      end
26    end
27  end
28  remove ci from the trainingSet;
29 end
30 do decomposing on the trainingSet; train new models on the decomposed trainingSet;
31 end
32 return trainingSet

```

Algorithm 8: Ranking ignorieren bei falscher Vorhersage erster Position (Pseudocode)

3.4.4 Ranking ignorieren bei falscher Vorhersage erster Position

Bei diesem Verfahren hat die Korrektur von der höchsträngigen Position einen radikalen Charakter. Wenn die erste Position im Ranking mit dem originalen Label übereinstimmt, wird es, wie bei allen anderen Verfahren, auf die relevante Positionsanzahl Rücksicht genommen und entsprechend dazu werden paarweise Vergleiche gebildet. Wenn sie nicht übereinstimmt, wird das gewonnene Ranking verweigert und anstatt davon nur das ursprüngliche Label genommen und daraus werden entsprechende paarweise Vergleiche gebildet.

Zum Beispiel: sei das ursprüngliche Label ist gleich c_0 und die gewonnene totale Ordnung ist gleich $[c_1, c_2, c_0, c_3]$. Da c_0 ungleich c_1 ist, wird das Ranking verweigert. Anstatt davon wird das originale Label genommen und daraus werden paarweise Vergleiche gebildet. Man bekommt folgendes Ergebnis: $c_1 < c_0, c_2 < c_0, c_3 < c_0$.

Dieses Verfahren wurde mit unterschiedlicher Anzahl an Iterationen untersucht.

Der obige Pseudocode (s. Algorithm 8) dient zur algorithmischen Veranschaulichung dieses Verfahrens. Hier werden zuerst alle Instanzen aus dem Trainingsset anhand der bereits trainierten Modelle klassifiziert. Dadurch entstehen verschiedene Voteswerte für einzelne Klassen. Mit der Anordnung von diesen gewinnen wir gleichzeitig die totale Ordnung der Labels für die entsprechend zugehörige Instanz.

Es wird anfangs überprüft, ob das originale Label sich an der ersten Position des Rankings befindet. Falls nicht, dann wird das Ranking verweigert und es werden nur paarweise Präferenzen vom originalen Label zum Trainingsset hinzugefügt (s. Beispiel am Anfang dieses Abschnitts). Falls ja, dann werden entsprechende Präferenzen für Labels bis zur relevanten Position erzeugt und das Trainingsset wird um damit gebildete Instanzen ergänzt. Die ursprüngliche Instanz wird gelöscht. Das neue Trainingsset wird binarisiert und für neue zwei-Klassen-Probleme werden Modelle trainiert.

Im Gegensatz zu allen davor beschriebenen Verfahren, außer dem vorherigen Verfahren (s. 3.4.3), wird diese komplette Prozedur abhängig von *numberOfIterations* mehrmals wiederholt (mit neu trainierten Modellen und modifiziertem Trainingsset). Am Ende wird das gewonnene Trainingsset zurückgegeben.

3.5 Verwandte Arbeiten

Es gab bis jetzt keine mir bekannte veröffentlichte wissenschaftliche Arbeit, die sich mit dieser Aufgabenstellung beschäftigte und somit liegen keine Ergebnisse zum Vergleich vor. Diese Arbeit kann deswegen als Basis für weitere Untersuchung der Problematik benutzt werden.

4 Experimente

In diesem Kapitel werden alle Details zum Experimentenverlauf beschrieben: Herkunft und Format der Datensätze, Testumgebung, Evaluation und anschließend Ergebnisse.

4.1 Datensätze und Datenaufbereitung

Datensätze und damit verbundene Formattanpassung werden in folgenden Abschnitten beschrieben.

4.1.1 Herkunft der Datensätze

Alle im Rahmen dieser Bachelorarbeit benutzten Datensätze stammen aus der UCI-Repository [BL13] und aus 19 von George Forman (Hewlett-Packard Laboratorien) gestifteten Multiklassen-Textdatensätzen. Sie wurden der WEKA-Website entnommen³.

UCI Machine Learning Repository ist eine Sammlung von Datenbanken, Domaintheorien und Datengeneratoren, die für empirische Analyse von Algorithmen des maschinellen Lernens eingesetzt werden. Sie wurde als ein FTP-Archiv im Jahr 1987 an der University of California in Irvine von David Aha und seinen Kommilitonen erstellt. Seitdem wurde UCI-Repository weitgehend von Studenten, Pädagogen und Forschern als die Hauptquelle der Datensätze im Bereich des maschinellen Lernens benutzt [BL13].

19 Multiklassen-Datensätze wurden aus realen Texten erstellt und besitzen folgende Eigenschaft: Dünnbesetztheit. Um sie auszunutzen, werden Datensätze im Sparse-Format gespeichert: jeder Eintrag einer Instanz besteht aus zwei Indizes, wobei die erste die Nummer des Attributes und die zweite den dazugehörigen Wert repräsentiert.

Datensätze aus beiden Quellen bilden reale Probleme aus der Praxis ab.

In der Abbildung 9 werden alle benutzten Datensätze und ihre wichtigsten Parameter tabellarisch dargestellt.

Datensatz	Instanzen	Attribute	Klassen	Sparse-Format
letter	20000	17	26	nein
mfeat-fourier	2000	77	10	nein
optdigits	5620	65	10	nein
segment	2310	20	7	nein
fbis.wc	2463	2001	17	ja
la1s.wc	3204	13196	6	ja
la2s.wc	3075	12433	6	ja
new3s.wc	9558	26833	44	ja
oh0.wc	1003	3183	10	ja
oh5.wc	918	3013	10	ja
oh10.wc	1050	3239	10	ja
oh15.wc	913	3101	10	ja
ohscal.wc	11162	11466	10	ja
re0.wc	1504	2887	13	ja
re1.wc	1657	3759	25	ja
tr11.wc	414	6430	9	ja
tr12.wc	313	5805	8	ja
tr21.wc	336	7903	6	ja
tr23.wc	204	5833	6	ja
tr31.wc	927	10129	7	ja
tr41.wc	878	7455	10	ja
tr45.wc	690	8262	10	ja
wap.wc	1560	8461	20	ja

Abbildung 9: Auflistung der Datensätze

³ <http://www.cs.waikato.ac.nz/ml/weka/datasets.html>

4.1.2 Format und Datenaufbereitung

Alle in dieser Arbeit benutzten Datensätze sind ursprünglich im ARFF-Format zugänglich. ARFF(Attribut-Relation File-format) wurde an der University of Waikato für die Benutzung mit der WEKA-Software entwickelt. Es besteht aus zwei Teilen: einem Kopf (header) und Daten.

Der Kopf besteht wiederum aus dem Relationsnamen, einer Liste mit Attributen und den zugehörigen Typen. Es sind folgende Attributtypen verfügbar: *numeric*, *string*, *date* (kann auch selbst spezifiziert werden) und selbstspezifizierte Auflistung aller möglichen Werte (z.B. {yes, no, maybe}). Ein Beispielkopf wird aus dem Wetter-Golf Problem bei Listing 1 dargestellt.

```
@relation Wetter_Golf

@attribute Day          date "MM-dd"
@attribute Temperature {hot , cool , mild}
@attribute Outlook      {sunny , overcast , rain}
@attribute Humidity    {high , normal}
@attribute Windy        {false , true}
@attribute class         {no, yes}
```

Listing 1: ARFF-Kopf

Der Datenabschnitt fängt mit *@data* an und besteht aus Instanzen (eine pro Zeile), die mit Attributwerten in der selben Reihenfolge wie bei der Deklaration im Kopfabschnitt gefüllt werden. Attributwerte werden voneinander durch Komma getrennt. Fehlende Werte werden durch ein Fragezeichen repräsentiert. Folgend wird ein Beispiel-Datenabschnitt dargestellt.

```
@data
07-05,hot , sunny , high , false , no
07-06,hot , sunny , high , true , no
07-07,hot , overcast , high , false , yes
07-09,cool , rain , normal , false , yes
```

Listing 2: ARFF-Datenteil

Im Sparse-Format sieht der Datenabschnitt etwas anders aus: es werden nur Attribute, die ungleich null sind, explizit mit der selben Attributnummer wie bei der Deklaration (beginnend mit Null) und dem zugehörigen Wert gespeichert. Jede Instanz befindet sich dabei innerhalb von geschweiften Klammern. Z.B.:

```
@data
{0 07-05,1 hot,5 no}
{1 hot,2 sunny,3 high,5 no}
{0 07-07,4 false ,5 yes}
{1 cool,3 normal,5 yes}
```

Listing 3: ARFF-Datenteil: Sparse-Format

Für detailliertere Information bezüglich dem ARFF-Format s. Quelle⁴.

Datensätze werden aus dem ARFF- zum EARFF-Format überführt. EARFF ist ein erweitertes ARFF-Format. Es wurde speziell für das im nächsten Abschnitt beschriebene Framework entwickelt. Diese Erweiterung beinhaltet vor allem mehr Optionen für das “Class”-Attribut aus dem Kopfabschnitt, das in drei logisch getrennte Einträge durch “|“ unterteilt wird. Der erste Eintrag gibt den Namen des Problems an, der zweite die Klasseninformation im entsprechenden Format und der dritte die Anzahl miteinander zu vergleichenden Klassen.

Zusätzlich muss im Datenbereich der Attributwert für die Klasse jeder Instanz in geschweifte Klammer gesetzt werden. Beim Sparse-Format bleibt er aber unverändert.

Listing 4 stellt einen kleinen Beispiel-Datensatz im EARFF-Format dar.

```
@relation example
```

```
@attribute a0 {false , true}
@attribute a1 {false , true}
```

⁴ <http://www.cs.waikato.ac.nz/ml/weka/arff.html>

```

@attribute a2 {false ,true}
@attribute a3 {false ,true}
@attribute Class [Multiclass | c0 ,c1 ,c2 | 2]

@data
true ,false ,false ,false ,{ c0}
true ,true ,false ,false ,{ c0}
false ,true ,false ,true ,{ c1}
false ,true ,true ,false ,{ c2}

```

Listing 4: EARFF-Format: Beispiel-Datensatz

4.2 Experimenten- und Testumgebung

LPCforSOS (Learning by Pairwise Comparisons for Structured Output Spaces) ist ein frei verfügbares Framework für maschinelles Lernen mit einem speziellen Fokus auf strukturierte Ausgaberäumen und paarweises Lernen. Es unterstützt zur Zeit Multilabel-, Multiklassen-, ordinale, hierarchische Klassifizierung und Label-Ranking Einstellungen (s. Quelle⁵). Es enthält solche wichtigen Komponenten wie:

- Decomposer: aktuell stehen *PairwiseComposer*, *OneAgainstAllComposer*, *TopicRankingComposer* und *ReinforcementLearningComposer* zur Verfügung.
- Aggregator: *Voting*, *WeightedVoting*, *PairwiseCoupling* und *WuLingWeng2* sind mögliche Optionen.
- Evaluator: es werden abhängig von dem Problemtyp unterschiedliche Maße für die Evaluierung benutzt.

Alle diese Komponenten werden bei Experimenten verwendet.

LPCforSOS baut auf der Funktionalität von WEKA auf. WEKA ist auch ein frei verfügbares, in Java geschriebenes Framework, das verschiedene Techniken aus folgenden Bereichen bereitstellt: maschinelles Lernen und Data-Mining. Es wurde an der University of Waikato entwickelt. Weka enthält Tools für Datenvorverarbeitung, Klassifikation, Regression, Clustering, Assoziationsregeln und Visualisierung.

Im Rahmen dieser Bachelorthesis wurden Weka 3.6.5 und Revision 19 von LPCforSOS verwendet. Im Kapitel 3 beschriebene Verfahren wurden innerhalb von LPCforSOS implementiert und getestet.

4.3 Evaluation

Es werden bei der Evaluierung folgende Maße in Betracht genommen:

- **Accuracy** kann als Prozentsatz der durch das trainierte Modell korrekt klassifizierten positiven und negativen Beispiele definiert werden und wird nach folgender Formel berechnet:

$$\frac{\sum(N_{\text{korrekte_positive}} + N_{\text{korrekte_negative}})}{N}, \quad (1)$$

wobei $N_{\text{korrekte_positive}}$, $N_{\text{korrekte_negative}}$ und N die Anzahl der korrekt klassifizierten positiven bzw. negativen Beispiele und die Anzahl aller Beispiele darstellen.

Abbildung 10 stellt Wahrheitstmatrizen für ein zwei-Klassen- und ein Multiklassen-Problem (das linke bzw. das rechte Bild) dar. Wahrheitstmatrix ist in diesem Fall eine Matrix mit tatsächlichen und vorhergesagten Klassifikationen. Accuracy-Wert wird abhängig vom Problemtyp mit einer der folgenden Formeln berechnet: $\frac{N_{A,A} + N_{B,B}}{N}$ bzw. $\frac{N_{A,A} + N_{B,B} + N_{C,C} + N_{D,D}}{N}$.

- **Positionsfehler (Position Error)** ist Distanz zu der Position des tatsächlichen Labels im Ranking. Wenn sich das wahre Label an der ersten Position des vorhergesagten Rankings befindet, dann liegt der Positionsfehler bei null, wenn an der zweiten Position, dann bei eins usw. Dieses Maß wird folgend normalisiert: es werden Position Errors für jedes Label berechnet, aufaddiert und durch die Anzahl der Labels geteilt. Die entsprechende Formel sieht folgend aus:

$$\frac{\sum_{i=1}^n \text{positionError}_i}{n} \quad (2)$$

⁵ <http://sourceforge.net/projects/lpcforsos>

		Classified as				
		A	B	C	D	
True class	A	$N_{A,A}$	$N_{B,A}$	$N_{C,A}$	$N_{D,A}$	N_A
	B	$N_{A,B}$	$N_{B,B}$	$N_{C,B}$	$N_{D,B}$	N_B
	C	$N_{A,C}$	$N_{B,C}$	$N_{C,C}$	$N_{D,C}$	N_C
	D	$N_{A,D}$	$N_{B,D}$	$N_{C,D}$	$N_{D,D}$	N_D
		$N_{A,A} + N_{A,B}$	$N_{B,A} + N_{B,B}$	N	N	

Abbildung 10: Wahrheitsmatrizen (confusion matrices)

- Durchschnittliche Häufigkeit der Rankingskorrektur vom höchstrangigen Label beschreibt wie oft das höchstrangige Label im Durchschnitt korrigiert wird. Dieses Maß ist nur für Verfahren aus dem Abschnitt 3.4 geeignet. Es wird zuerst überprüft, ob das höchstrangige Label vom gewonnenen Ranking mit dem ursprünglichen Label übereinstimmt. Und wenn das nicht zutrifft, wird der Zähler um Eins erhöht. Abschließend wird der Zählerwert durch die Anzahl der Instanzen geteilt:

$$\frac{\sum_{i=1}^n \text{korrigiertesLabel}_i}{n} \quad (3)$$

4.4 Ergebnisse

In diesem Kapitel werden Experimentenergebnisse dargestellt und analysiert.

4.4.1 Einstellungen

Im Kapitel 4.4 beschriebene Experimente wurden mit folgenden Einstellungen durchgeführt:

- zehnfache Kreuzvalidierung.
- Abhängig von der Größe des Datensatzes wurden J48 oder LibLINEAR zwecks akzeptabler Laufzeit zur Klassifizierung verwendet. Kleine Datensätze wurden mit J48 und große mit LibLINEAR klassifiziert. Es gab keine auffallenden Unterschiede der Ergebnisse zwischen diesen Klassifizierern.
- Ergebnisse wurden mit im Kapitel 4.3 beschriebenen Maßen ausgewertet.
- Alle weiteren Einstellungsparameter können entsprechenden Experimententabellen 17 - 27 entnommen werden.

4.4.2 Visualisierung der Teilergebnisse

Im Zusammenhang mit großer Experimentenanzahl werden hier nur Ergebnisse des repräsentativen "segment"-Datensatzes grafisch dargestellt, der mit J48 trainiert wurde. Wegen seiner niedrigen Klassenanzahl, die sich gut anzeigen lässt, und zu erwartender Ergebnisrepräsentation wurde ausgerechnet er ausgewählt.

In der Abbildung 11 werden zwei gleiche Verfahren (mit der Ausnahme der Labelposition im Ranking) miteinander verglichen. Blaue Säulen stellen das Verfahren dar, bei dem sich das originale Label auf beliebiger Position im Ranking befindet, und bei roten Säulen geht es um das Verfahren mit originalem Label an erster Position. Die gelbe Linie ist das Ergebnis vom klassischen Ansatz ohne Korrektur.

Man kann hier zwei Trends erkennen:

- je mehr Positionen aus dem Ranking genommen werden (je größer relevante Positionsanzahl ist), desto schlechter wird das Ergebnis.
- Das Verfahren mit dem originalen Label an erster Position liefert bessere Ergebnisse. Das ist nachvollziehbar, weil in diesem Fall sichergestellt wird, dass das originale Label den größten Einfluss auf die Vorhersage im Vergleich zu anderen einzelnen Labels hat.

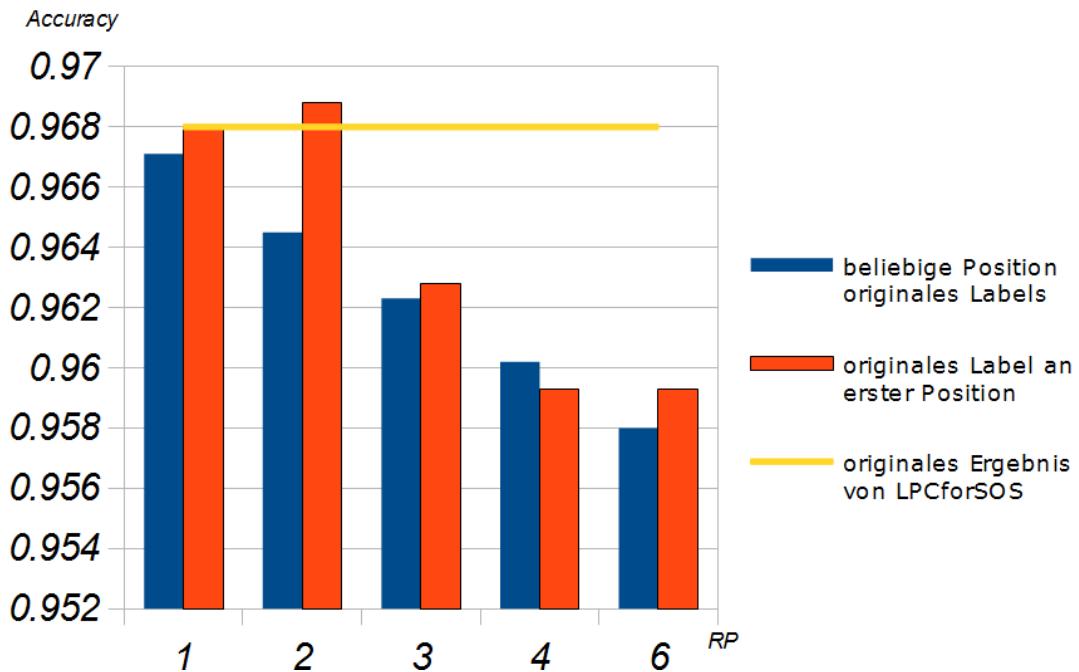


Abbildung 11: Variable relevante Positionsanzahl: beliebige gegen erste Position originales Labels

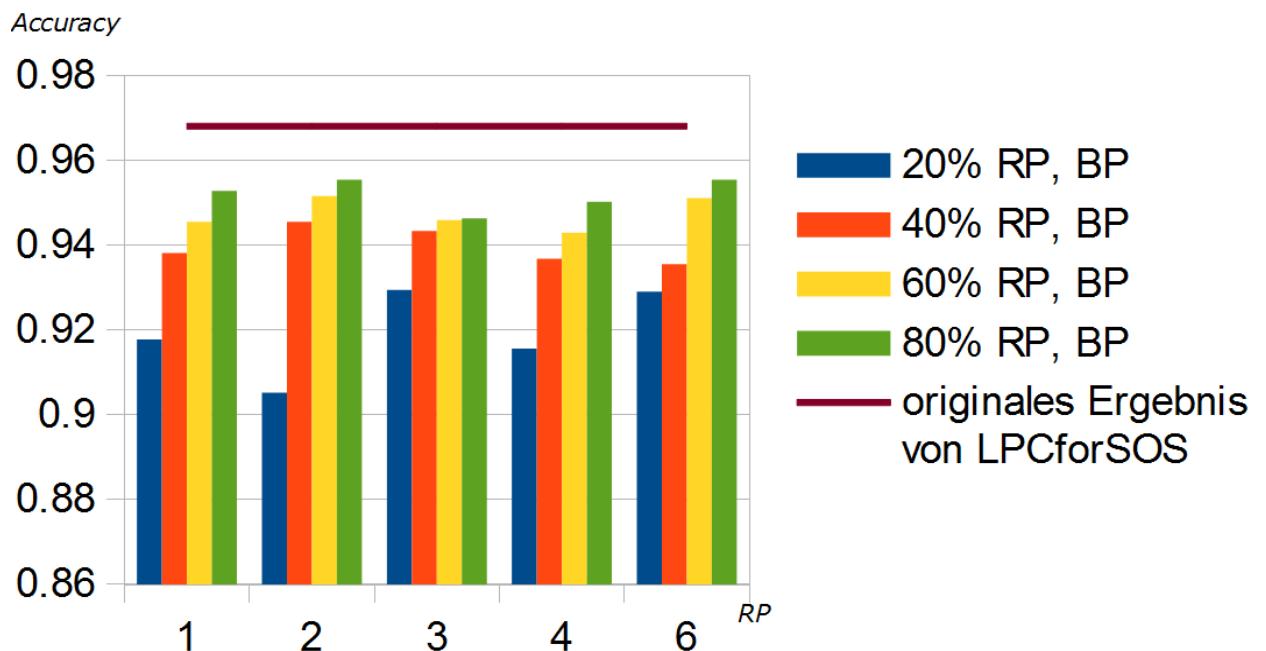


Abbildung 12: Prozentsatz relevanter Positionsanzahl: originales Label an beliebiger Position

In der Abbildung 12 wird ein weiteres Verfahren dargestellt. Dabei befindet sich originales Label an beliebiger Position im Ranking und es wird Prozentsatz der relevanten Positionsanzahl betrachtet. Auf dieser Abbildung lässt sich folgender Trend erkennen:

- je größer der Prozentsatz relevanter Positionsanzahl ist, desto höher ist der Accuracy-Wert.

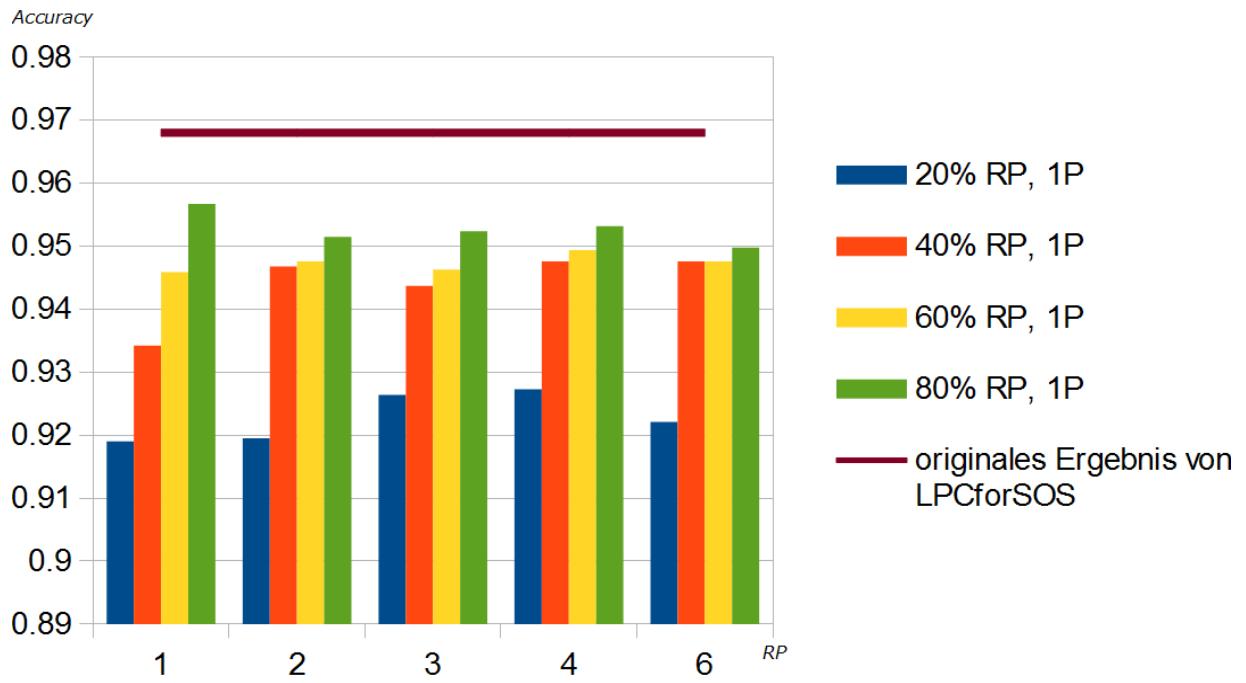


Abbildung 13: Prozentsatz relevanter Positionsanzahl: originales Label an erster Position

In der Abbildung 13 wird weiterhin Prozentsatz der relevanten Positionsanzahl betrachtet, aber originales Label befindet sich an erster Position im Ranking. Auf dieser Grafik lässt sich der gleiche Trend wie davor erkennen:

- je größer der Prozentsatz relevanter Positionsanzahl ist, desto höher ist der Accuracy-Wert.

Im Vergleich zur vorherigen Abbildung sind Accuracy-Werte bei diesem Verfahren im Durchschnitt etwas höher. Das entspricht auch der Erwartung aufgrund der Position des originalen Labels.

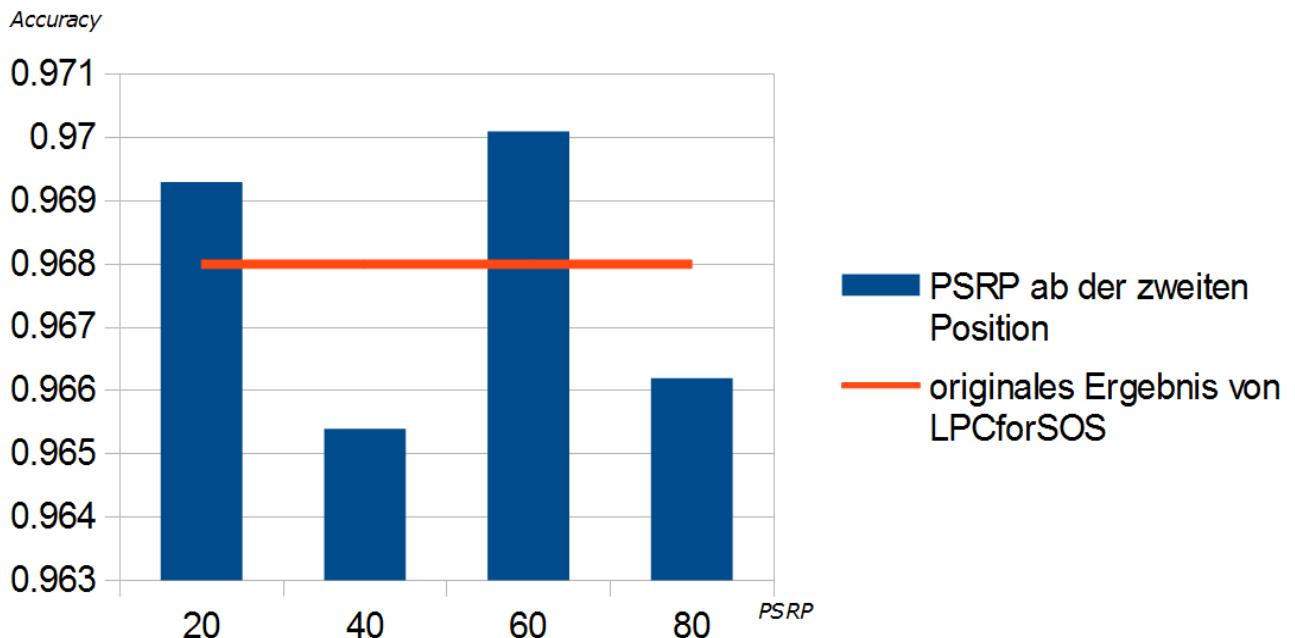


Abbildung 14: Prozentsatz relevanter Positionsanzahl ab der zweiten Position: originales Label an erster Position

In der Abbildung 14 wird variabler Prozentsatz der relevanten Positionsanzahl ab der zweiten Position betrachtet, wobei originales Label sich an erster Position im Ranking befindet.

Es gibt keine erkennbaren Trends.

Wie auf der Grafik zu sehen ist, gibt es zwei Treffer: Accuracy-Werte, die das Ergebnis vom klassischen Ansatz ohne Korrektur übersteigen. Sie sind im Vergleich zum einzigen davor gesehenem Treffer auch signifikanter. Dieses Verfahren

verspricht den größten Potenzial zu besitzen. Es ist auch logisch nachvollziehbar: das originale Label wird beibehalten und es kommen nur wenige der am besten geeigneten Instanzen dazu. Diese werden zufällig ausgewählt.

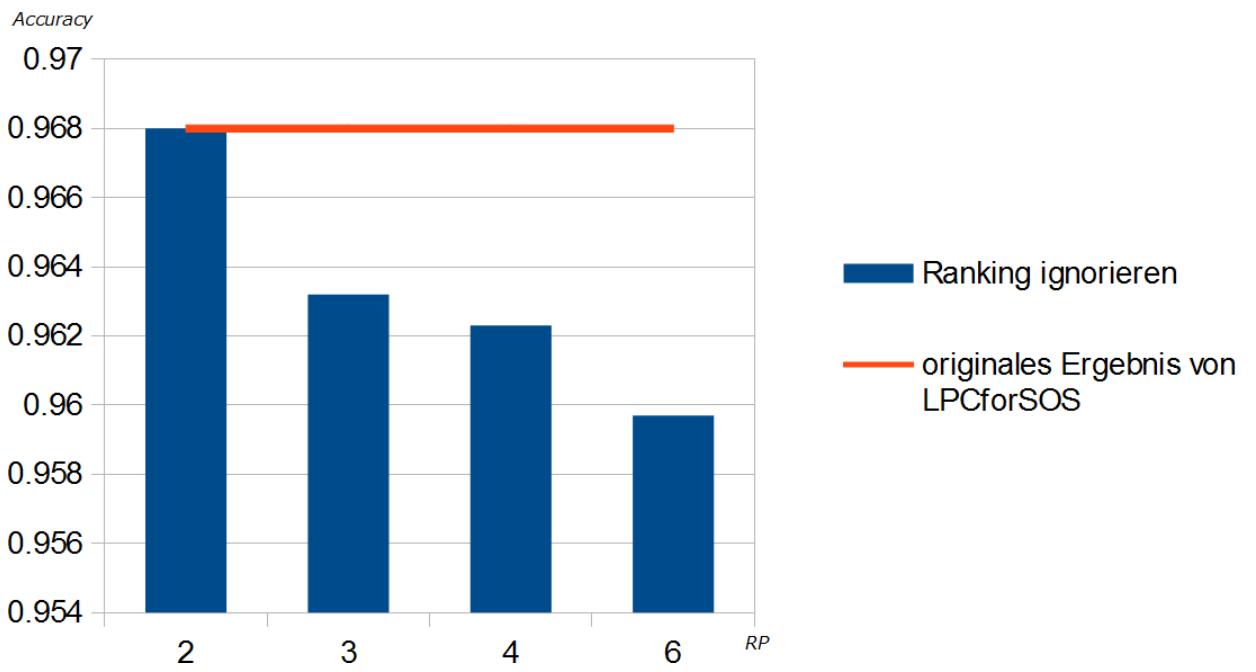


Abbildung 15: Ranking bei falscher Vorhersage erster Position ignorieren

In der Abbildung 15 wird Ranking bei falscher Vorhersage erster Position komplett ignoriert und anstatt davon werden nur paarweise Vergleiche bezüglich des originalen Labels gebildet.

Es lässt sich folgender Trend erkennen:

- je größer relevante Positionsanzahl wird, desto schlechter wird das Ergebnis.

Dieses Verfahren verspricht in aktueller Form keine guten Ergebnisse zu liefern. Die Idee dahinter war: bei richtiger Vorhersage erster Position ist sehr wahrscheinlich, dass die Nachfolger im Ranking auch korrekt vorhergesagt werden. Diese Vermutung hat sich aber empirisch im Rahmen dieser Bachelorarbeit nicht bestätigt.

4.4.3 Übersicht der Ergebnisse

Experimente wurden in zwei Phasen aufgeteilt: in erster Phase befanden sich im Kapitel 3 beschriebene Verfahren noch im Entwicklungsstadium und deswegen wurden sie mit wenigen Datensätzen getestet und in zweiter Phase wurden Verfahren auf allen zur Verfügung stehenden Datensätzen getestet.

Experimententabellen 17, 18 und 19 gehören zur ersten Testphase. Datensätze wurden mit J48 klassifiziert.

Experimententabellen zwischen 20 und 27 gehören zur zweiten Testphase. Experimente aus Tabellen 20 - 25 wurden mit J48 und 26 - 27 mit LibLINEAR durchgeführt.

In der Abbildung 16 wird die gesamte Übersicht zu erzielten Treffern je nach Verfahren dargestellt. Sie bestätigt nochmal, dass Verfahren aus Abschnitt 3.4.3 ("Prozentsatz der relevanten Positionsanzahl ab der zweiten Position") den größten Potenzial besitzt.

Detailliertere Information zu einzelnen Experimenten findet man im Anhang, wobei folgende Legende zu beachten ist:

- Treffer sind rot markiert.
- einzelne Evaluierungsmaße werden mit "|" als einem Trennsymbol getrennt. Sie kommen nach folgendem Schema vor: Accuracy|Positionsfehler|Durchschnittliche Häufigkeit der Rankingskorrektur vom höchstrangigen Label, wobei letztes Maß eher optional ist und wird bei der Auswertung von Ergebnissen kaum berücksichtigt. Positionsfehler kann zum besseren Verständnis des zu untersuchenden Verfahrens beitragen, zur Auswertung leistet er aber auch kaum einen Beitrag.

Experimente zum "new3s.wc"-Datensatz wären in absehbarer Zeit nicht fertig, deswegen wurden sie nach einiger Zeit abgebrochen. Bei Datensätzen dieser Größenordnung müsste eine geschickte Auswahl des Klassifizierers bezüglich der Laufzeit gemacht werden.

RP - relevante Positionsanzahl
 PSRP - variabler Prozentsatz relevanter Positionsanzahl
 ab 2P - ab relevanter Positionsanzahl = 2
 BP - originales Label an beliebiger Position im Ranking
 1P - originales Label an erster Position im Ranking

Datensatz	RP, BP	PSRP, BP	RP, 1P	PSRP, 1P	PSRP ab 2P, 1P	ignore Ranking
letter	0	0	0	0	0	0
mfeat-fourier	0	0	0	0	3	0
optdigits	0	0	0	0	0	0
segment	0	0	1	0	2	1
fbis.wc	0	0	0	0	0	0
la1s.wc	0	0	0	0	2	0
la2s.wc	0	0	0	0	2	0
new3s.wc	-	-	-	-	-	-
oh0.wc	0	0	0	0	0	0
oh5.wc	0	0	0	0	0	0
oh10.wc	0	0	0	0	0	0
oh15.wc	0	0	0	0	0	0
ohscal.wc	0	0	0	0	0	0
re0.wc	0	0	0	0	0	0
re1.wc	0	0	0	0	0	0
tr11.wc	0	0	0	0	0	0
tr12.wc	3	0	5	3	4	5
tr21.wc	5	0	4	0	3	4
tr23.wc	0	0	0	0	2	0
tr31.wc	0	0	1	0	4	0
tr41.wc	0	0	0	0	1	0
tr45.wc	0	0	0	0	0	0
wap.wc	0	0	0	0	0	0
Treffersumme	8	0	11	3	23	10

Abbildung 16: Trefferliste

5 Zusammenfassung und Ausblick

5.1 Zusammenfassung

In dieser Arbeit wurden mehrere Verfahren zur Korrektur von Trainingsdaten untersucht. Sie wurden abhängig von der Position des originalen Labels im Ranking in zwei Untergruppen logisch aufgeteilt, die sich in diesem Bezug voneinander grundlegend unterscheiden. Bei Verfahren aus der ersten Gruppe befindet sich das originale Label an beliebiger Position im Ranking und bei Verfahren aus der zweiten Gruppe befindet sich das originale Label an erster Position im Ranking (Position wird entsprechend angepasst).

Die zweite Gruppe liefert durchschnittlich im Vergleich zu den gleichgenannten Verfahren aus der ersten Gruppe bessere Ergebnisse. Das kann dadurch erklärt werden, dass das originale Label als das "wahre" Label gilt und weiterhin den größten Einfluss auf die Vorhersage im Vergleich zu weiteren Labels im Ranking hat. Somit erweist die zweite Gruppe eine bessere Perspektive als die erste.

In dieser Arbeit ist es gelungen Verfahren zu entwickeln, mit welchen eine Verbesserung des Accuracy-Wertes erreicht werden kann. Die Anpassung des Trainingssets bei Verfahren, die sich als besonders gut erwiesen, wurde auch mehrmals durchgeführt.

Es ist in dieser Arbeit nicht gelungen eine systematische Vorgehensweise für steuerbare Verbesserung des Accuracy-Wertes vom beliebigen Datensatz zu gewinnen. Jedoch sind im Ausblick Vorschläge zu finden, die das bewerkstelligen könnten oder mindestens zur Steuerung dieses Prozesses wesentlich beitragen sollten.

5.2 Ausblick

Bei der Vorgehensweise mit dem ursprünglichen Label an erster Position im Ranking (s. 3.4 auf Seite 16) gab es die meisten positiven Ergebnisse, deswegen lege ich einen großen Wert ausgerechnet auf Verfahren, die darauf basieren. Wie man auch anhand der Experimente sieht, werden gute Ergebnisse bei der zufälligen Auswahl von paarweisen Vergleichen erzielt. Es ist ein Zeichen für das Potential dieser Vorgehensweise. Es fehlen aber, meiner Meinung nach, weitere Maße mit welchen man dieses Potential beschreiben und maximal gezielt ausnutzen kann. Hier werden solche möglichen Features aufgelistet:

- Distanz zwischen Labels im Ranking:
 - Abstand der Votes für Labels nach dem Voting könnte dafür geeignet sein. Man könnte auch verschiedene Varianten von diesem Ansatz ausprobieren: z.B. nur die Labels beibehalten, Distanz zu welchen einen bestimmten Wert nicht überschreitet, oder nur 10% der "Distanz"-besten Kandidaten usw.
 - Abstand der Votes kombiniert mit einem Koeffizient von jedem Label, das sich aus Eigenschaften vom entsprechenden Label ergibt.
- Einzelne paarweise Vergleiche: untersuchen wie gut sie im Vergleich zueinander stehen und nur die besten nehmen.
- Gewichtete Labels: paarweise Vergleiche, die von einem nicht höchsträngigen Label stammen, unterstufen.

Es können auch andere Algorithmen zur Klassifizierung ausprobiert und anschließend miteinander verglichen werden.

Literaturverzeichnis

- [BL13] BACHE, K. und M. LICHMAN: *UCI Machine Learning Repository*. <http://archive.ics.uci.edu/ml>, 2013.
- [CG05] CHU, WEI und ZOUBIN GHAHRAMANI: *Preference Learning with Gaussian Processes*. In: *Proceedings of the 22Nd International Conference on Machine Learning*, ICML '05, Seiten 137–144, New York, NY, USA, 2005. ACM.
- [CSZ⁺06] CHAPELLE, OLIVIER, BERNHARD SCHÖLKOPF, ALEXANDER ZIEN et al.: *Semi-supervised learning*, Band 2. MIT press Cambridge, 2006.
- [FH03] FÜRNKRANZ, JOHANNES und EYKE HÜLLERMEIER: *Pairwise preference learning and ranking*. In: *Machine Learning: ECML 2003*, Seiten 145–156. Springer, 2003.
- [K⁺95] KOHAVI, RON et al.: *A study of cross-validation and bootstrap for accuracy estimation and model selection*. In: *International Joint Conference on Artificial Intelligence (IJCAI)*, Band 14, Seiten 1137–1145, 1995.
- [KLM96] KAEFLING, LESLIE PACK, MICHAEL L LITTMAN und ANDREW W MOORE: *Reinforcement learning: A survey*. arXiv preprint cs/9605103, 1996.
- [Kod88] KODRATOFF, YVES: *Introduction to machine learning*. Morgan Kaufmann Pub, 1988.
- [NAH02] NAGAI, YUKIE, MINORU ASADA und Koh HOSODA: *Developmental learning model for joint attention*. In: *Intelligent Robots and Systems, 2002. IEEE/RSJ International Conference on*, Band 1, Seiten 932–937. IEEE, 2002.
- [Nam07] NAM, GE HYUN: *Paarweise Klassifikation für Probleme mit geordneten Klassenwerten*. Diplomarbeit, 2007.
- [PKSR02] PODGORELEC, VILI, PETER KOKOL, BRUNO STIGLIC und IVAN ROZMAN: *Decision trees: an overview and their use in medicine*. Journal of medical systems, 26(5):445–463, 2002.
- [SPP10] SUBRAMANYA, AMARNAG, SLAV PETROV und FERNANDO PEREIRA: *Efficient Graph-based Semi-supervised Learning of Structured Tagging Models*. In: *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, EMNLP '10, Seiten 167–176, Stroudsburg, PA, USA, 2010. Association for Computational Linguistics.
- [TP98] THRUN, SEBASTIAN und LORIEN PRATT: *Learning to learn: Introduction and overview*. In: *Learning to learn*, Seiten 3–17. Springer, 1998.
- [Vap00] VAPNIK, VLADIMIR: *The nature of statistical learning theory*. Springer, 2000.
- [WLW04] WU, TING-FAN, CHIH-JEN LIN und RUBY C. WENG: *Probability Estimates for Multi-class Classification by Pairwise Coupling*. J. Mach. Learn. Res., 5:975–1005, Dezember 2004.

Anhang

	letter_reduziert.earff 26 Klassen, 17 Attribute 8978 Instanzen	segment.earff
originales LPCforSOS	0.8796 ± 0.0106	0.9680 ± 0.0178
Ursprüngliches Label auf beliebiger Position im Ranking		
relevante Positionsanzahl = 1	0.8618 ± 0.0104	0.9671 ± 0.0159
relevante Positionsanzahl = 2	0.8420 ± 0.0128	0.9645 ± 0.0139
relevante Positionsanzahl = 3	0.8377 ± 0.0133	0.9623 ± 0.0188
relevante Positionsanzahl = 4	0.8349 ± 0.0112	0.9602 ± 0.0219
relevante Positionsanzahl = 6	0.8290 ± 0.0120	0.9580 ± 0.0196
relevante Positionsanzahl = 8	0.8262 ± 0.0116	-
relevante Positionsanzahl = 12	0.8289 ± 0.0114	-
relevante Positionsanzahl = 16	0.8238 ± 0.0079	-
relevante Positionsanzahl = 20	0.8227 ± 0.0100	-
Zufällig genommene paarweise Präferenzen		
20,00%	0.7181 ± 0.0114	$0.9260 \pm 0.0192 0.9338 \pm 0.0183$
40,00%	0.7701 ± 0.0139	$0.9390 \pm 0.0212 0.9424 \pm 0.0110$
60,00%	0.7892 ± 0.0141	$0.9455 \pm 0.0137 0.9476 \pm 0.0147$
80,00%	0.7965 ± 0.0149	$0.9550 \pm 0.0144 0.9528 \pm 0.0139$
20% relevanter Positionsanzahl = 1	0.7501 ± 0.0086	$0.9139 \pm 0.0200 0.9130 \pm 0.0248$
20% relevanter Positionsanzahl = 2	0.7303 ± 0.0126	$0.9208 \pm 0.0275 0.9203 \pm 0.0206$
20% relevanter Positionsanzahl = 3	0.7224 ± 0.0163	$0.9368 \pm 0.0203 0.9342 \pm 0.0240$
20% relevanter Positionsanzahl = 4	0.7234 ± 0.0155	$0.9221 \pm 0.0163 0.9165 \pm 0.0170$
20% relevanter Positionsanzahl = 6	0.7222 ± 0.0135	$0.9242 \pm 0.0244 0.9173 \pm 0.0167$
20% relevanter Positionsanzahl = 8	0.7251 ± 0.0172	-
20% relevanter Positionsanzahl = 12	0.7225 ± 0.0169	-
20% relevanter Positionsanzahl = 16	0.7284 ± 0.0167	-
20% relevanter Positionsanzahl = 20	0.7181 ± 0.0199	-
40% relevanter Positionsanzahl = 1	0.8022 ± 0.0118	$0.9355 \pm 0.0171 0.9346 \pm 0.0162$
40% relevanter Positionsanzahl = 2	0.7746 ± 0.0103	$0.9411 \pm 0.0183 0.9377 \pm 0.0168$
40% relevanter Positionsanzahl = 3	0.7696 ± 0.0082	$0.9442 \pm 0.0195 0.9359 \pm 0.0195$
40% relevanter Positionsanzahl = 4	0.7676 ± 0.0092	$0.9390 \pm 0.0277 0.9377 \pm 0.0166$
40% relevanter Positionsanzahl = 6	0.7665 ± 0.0085	$0.9338 \pm 0.0188 0.9442 \pm 0.0148$
40% relevanter Positionsanzahl = 8	0.7670 ± 0.0120	-
40% relevanter Positionsanzahl = 12	0.7659 ± 0.0129	-
40% relevanter Positionsanzahl = 16	0.7641 ± 0.0152	-
40% relevanter Positionsanzahl = 20	0.7616 ± 0.0136	-
60% relevanter Positionsanzahl = 1	0.8250 ± 0.0108	$0.9489 \pm 0.0128 0.9416 \pm 0.0192$
60% relevanter Positionsanzahl = 2	0.7971 ± 0.0094	$0.9506 \pm 0.0167 0.9494 \pm 0.0183$
60% relevanter Positionsanzahl = 3	0.7911 ± 0.0102	$0.9489 \pm 0.0119 0.9515 \pm 0.0139$
60% relevanter Positionsanzahl = 4	0.7910 ± 0.0092	$0.9468 \pm 0.0146 0.9489 \pm 0.0156$
60% relevanter Positionsanzahl = 6	0.7898 ± 0.0132	$0.9442 \pm 0.0231 0.9459 \pm 0.0187$
60% relevanter Positionsanzahl = 8	0.7893 ± 0.0143	-
60% relevanter Positionsanzahl = 12	0.7866 ± 0.0120	-
60% relevanter Positionsanzahl = 16	0.7830 ± 0.0112	-
60% relevanter Positionsanzahl = 20	0.7862 ± 0.0118	-
80% relevanter Positionsanzahl = 1	0.8360 ± 0.0088	$0.9450 \pm 0.0164 0.9489 \pm 0.0127$
80% relevanter Positionsanzahl = 2	0.8104 ± 0.0102	$0.9554 \pm 0.0136 0.9498 \pm 0.0208$
80% relevanter Positionsanzahl = 3	0.8084 ± 0.0148	$0.9550 \pm 0.0176 0.9537 \pm 0.0167$
80% relevanter Positionsanzahl = 4	0.8057 ± 0.0089	$0.9498 \pm 0.0217 0.9541 \pm 0.0149$
80% relevanter Positionsanzahl = 6	0.7925 ± 0.0136	$0.9519 \pm 0.0194 0.9485 \pm 0.0153$
80% relevanter Positionsanzahl = 8	0.7981 ± 0.0145	-
80% relevanter Positionsanzahl = 12	0.7955 ± 0.0135	-
80% relevanter Positionsanzahl = 16	0.7989 ± 0.0093	-
80% relevanter Positionsanzahl = 20	0.7936 ± 0.0115	-

Abbildung 17: Kleines Datensatz-Set: ursprüngliches Label auf beliebiger Position im Ranking

	letter_adapted_reduced_number_of_instances.earff 26 classes, 16 attributes, 8978 instances)	segment_adapted.earff
original LPCforSOS	0.8796 ± 0.0106	0.9680 ± 0.0178
Originally instance label is taken into account in all experiments		
one first predicted element better than all after it	0.8796 ± 0.0106	0.9680 ± 0.0178
two first predicted elements better than all after it	0.8494 ± 0.0108	0.9688 ± 0.0134 !!!
three first predicted elements better than all after it	0.8437 ± 0.0121	0.9628 ± 0.0141
four first predicted elements better than all after it	0.8382 ± 0.0121	0.9593 ± 0.0187
six first predicted elements better than all after it	0.8316 ± 0.0115	0.9593 ± 0.0192
eight first predicted elements better than all after it	0.8302 ± 0.0098	-
twelve first predicted elements better than all after it	0.8323 ± 0.0135	-
sixteen first predicted elements better than all after it	0.8315 ± 0.0080	-
twenty first predicted elements better than all after it	0.8283 ± 0.0120	-
RANDOM TAKEN PAIRWISE PREFERENCES		
20,00%	0.7202 ± 0.0237	0.9238 ± 0.0200 0.9281 ± 0.0252
40,00%	0.7687 ± 0.0181	0.9446 ± 0.0124 0.9442 ± 0.0176
60,00%	0.7868 ± 0.0093	0.9541 ± 0.0171 0.9528 ± 0.0224
80,00%	0.7976 ± 0.0119	0.9550 ± 0.0139 0.9563 ± 0.0135
20 % of one first predicted element better than all after it	0.7675 ± 0.0119	0.9082 ± 0.0150 0.9156 ± 0.0185
20 % of two first predicted elements better than all after it	0.7283 ± 0.0161	0.9264 ± 0.0121 0.9195 ± 0.0257
20 % of three first predicted elements better than all after it	0.7300 ± 0.0116	0.9294 ± 0.0181 0.9316 ± 0.0203
20 % of four first predicted elements better than all after it	0.7222 ± 0.0159	0.9229 ± 0.0292 0.9190 ± 0.0161
20 % of six first predicted elements better than all after it	0.7266 ± 0.0135	0.9212 ± 0.0243 0.9290 ± 0.0127
20 % of eight first predicted elements better than all after it	0.7239 ± 0.0147	-
20 % of twelve first predicted elements better than all after it	0.7258 ± 0.0120	-
20 % of sixteen first predicted elements better than all after it	0.7216 ± 0.0123	-
20 % of twenty first predicted elements better than all after it	0.7284 ± 0.0148	-
40 % of one first predicted element better than all after it	0.8061 ± 0.0135	0.9398 ± 0.0186 0.9312 ± 0.0160
40 % of two first predicted elements better than all after it	0.7780 ± 0.0098	0.9424 ± 0.0207 0.9411 ± 0.0217
40 % of three first predicted elements better than all after it	0.7720 ± 0.0109	0.9468 ± 0.0235 0.9364 ± 0.0159
40 % of four first predicted elements better than all after it	0.7716 ± 0.0127	0.9459 ± 0.0218 0.9429 ± 0.0147
40 % of six first predicted elements better than all after it	0.7649 ± 0.0066	0.9433 ± 0.0126 0.9506 ± 0.0126
40 % of eight first predicted elements better than all after it	0.7678 ± 0.0151	-
40 % of twelve first predicted elements better than all after it	0.7686 ± 0.0152	-
40 % of sixteen first predicted elements better than all after it	0.7608 ± 0.0178	-
40 % of twenty first predicted elements better than all after it	0.7668 ± 0.0126	-
60 % of one first predicted element better than all after it	0.8328 ± 0.0144	0.9416 ± 0.0141 0.9407 ± 0.0182
60 % of two first predicted elements better than all after it	0.8021 ± 0.0064	0.9541 ± 0.0126 0.9498 ± 0.0159
60 % of three first predicted elements better than all after it	0.7916 ± 0.0126	0.9506 ± 0.0148 0.9498 ± 0.0153
60 % of four first predicted elements better than all after it	0.7922 ± 0.0110	0.9472 ± 0.0193 0.9476 ± 0.0152
60 % of six first predicted elements better than all after it	0.7908 ± 0.0104	0.9476 ± 0.0215 0.9494 ± 0.0193
60 % of eight first predicted elements better than all after it	0.7858 ± 0.0147	-
60 % of twelve first predicted elements better than all after it	0.7868 ± 0.0155	-
60 % of sixteen first predicted elements better than all after it	0.7902 ± 0.0136	-
60 % of twenty first predicted elements better than all after it	0.7843 ± 0.0140	-
80 % of one first predicted element better than all after it	0.8494 ± 0.0121	0.9506 ± 0.0150 0.9463 ± 0.0146
80 % of two first predicted elements better than all after it	0.8137 ± 0.0133	0.9494 ± 0.0196 0.9537 ± 0.0160
80 % of three first predicted elements better than all after it	0.8088 ± 0.0150	0.9537 ± 0.0144 0.9528 ± 0.0094
80 % of four first predicted elements better than all after it	0.8022 ± 0.0105	0.9567 ± 0.0138 0.9563 ± 0.0198
80 % of six first predicted elements better than all after it	-	0.9528 ± 0.0149 0.9481 ± 0.0133
80 % of eight first predicted elements better than all after it	-	-
80 % of twelve first predicted elements better than all after it	-	-
80 % of sixteen first predicted elements better than all after it	-	-
80 % of twenty first predicted elements better than all after it	-	-

Abbildung 18: Kleines Datensatz-Set: ursprüngliches Label an erster Position im Ranking (Teil 1/2)

	letter_reduziert.earff 26 Klassen, 17 Attribute 8978 Instanzen	segment.earff	mfeat-fourier.earff
originales LPCforSOS	0.8796±0.0106	0.9680±0.0178	0.7910±0.0270
Ursprüngliches Label auf erster Position im Ranking			
Prozentsatz der relevanten Positionsanzahl ab der zweiten Position			
10% relevanter Positionsanzahl >= 2			
iterative Schritte = 1	0.8729±0.0107 0.3210±0.0303 0.03712	0.9684±0.0183 0.0420±0.0220 0.00582	0.7915±0.0275 0.3200±0.0401 0.04388
iterative Schritte = 2	0.8719±0.0119 0.3205±0.0303 0.03930	0.9684±0.0183 0.0424±0.0225 0.00582	0.7915±0.0275 0.3195±0.0397 0.04394
iterative Schritte = 3	0.8700±0.0100 0.3224±0.0278 0.03989	0.9684±0.0183 0.0424±0.0225 0.00582	0.7915±0.0275 0.3200±0.0401 0.04394
30% relevanter Positionsanzahl >= 2			
iterative Schritte = 1	0.8613±0.0123 0.3464±0.0311 0.03712	0.9675±0.0150 0.0455±0.0199 0.00582	0.7910±0.0312 0.3285±0.0540 0.04388
iterative Schritte = 2	0.8580±0.0081 0.3576±0.0293 0.04297	0.9632±0.0141 0.0502±0.0172 0.00622	0.7875±0.0254 0.3285±0.0539 0.03736
iterative Schritte = 3	0.8597±0.0104 0.3459±0.0345 0.04571	0.9710±0.0115 0.0416±0.0161 0.00594	0.7940±0.0316 0.3235±0.0488 0.04301
50% relevanter Positionsanzahl >= 2			
iterative Schritte = 1	0.8549±0.0110 0.3644±0.0322 0.03712	0.9662±0.0158 0.0420±0.0190 0.00582	0.7920±0.0305 0.3215±0.0498 0.04388
iterative Schritte = 2	0.8482±0.0118 0.3849±0.0379 0.04551	0.9680±0.0150 0.0407±0.0200 0.00620	0.7805±0.0273 0.3475±0.0419 0.03863
iterative Schritte = 3	0.8495±0.0089 0.3699±0.0361 0.04938	0.9697±0.0154 0.0390±0.0193 0.00628	0.7950±0.0296 0.3295±0.0584 0.04042
70% relevanter Positionsanzahl >= 2			
iterative Schritte = 1	0.8514±0.0099 0.3760±0.0232 0.03712	0.9701±0.0159 0.0398±0.0182 0.00582	0.7900±0.0363 0.3290±0.0596 0.04388
iterative Schritte = 2	0.8464±0.0148 0.3822±0.0368 0.04766	0.9701±0.0138 0.0429±0.0175 0.00639	0.7865±0.0362 0.3280±0.0580 0.04144
iterative Schritte = 3	0.8460±0.0074 0.3839±0.0295 0.05219	0.9680±0.0133 0.0407±0.0155 0.00666	0.7915±0.0273 0.3130±0.0427 0.03605
(75%, 25%) relevanter Positionsanzahl >= 3			
iterative Schritte = 1	0.8493±0.0121 0.3965±0.0314 0.03712	0.9671±0.0139 0.0472±0.0192 0.00582	0.7850±0.0306 0.3270±0.0420 0.04388
iterative Schritte = 2	0.8482±0.0112 0.3818±0.0276 0.04858	0.9693±0.0109 0.0424±0.0153 0.00603	0.7620±0.0219 0.3720±0.0397 0.04083
iterative Schritte = 3	0.8464±0.0117 0.3807±0.0352 0.05262	0.9697±0.0171 0.0411±0.0216 0.00654	0.7735±0.0286 0.3650±0.0526 0.03703
90% relevanter Positionsanzahl >= 2			
iterative Schritte = 1	0.8461±0.0113 0.3901±0.0396 0.03712	0.9688±0.0132 0.0403±0.0197 0.00582	0.7915±0.0286 0.3195±0.0506 0.04388
iterative Schritte = 2	0.8471±0.0088 0.3938±0.0346 0.04860	0.9701±0.0128 0.0390±0.0136 0.00661	0.7870±0.0339 0.3300±0.0566 0.04141
iterative Schritte = 3	0.8435±0.0098 0.3909±0.0374 0.05271	0.9671±0.0153 0.0429±0.0204 0.00668	0.7920±0.0287 0.3230±0.0458 0.03938
Ranking ignorieren bei falscher Vorhersage erster Position			
Vorhersage erster Position			
relevante Positionsanzahl = 2			
iterative Schritte = 1	0.8490±0.0133 0.3956±0.0312 0.03712	0.9680±0.0143 0.0407±0.0196 0.00582	0.7655±0.0218 0.3660±0.0481 0.04388
iterative Schritte = 2	0.8479±0.0108 0.4012±0.0317 0.05048	0.9675±0.0180 0.0429±0.0231 0.00680	0.7715±0.0203 0.3595±0.0556 0.04022
iterative Schritte = 3	0.8437±0.0121 0.3961±0.0303 0.05482	0.9675±0.0193 0.0416±0.0235 0.00718	0.7740±0.0293 0.3695±0.0492 0.03868
relevante Positionsanzahl = 3			
iterative Schritte = 1	0.8479±0.0147 0.4243±0.0390 0.03712	0.9632±0.0137 0.0494±0.0165 0.00582	0.7710±0.0306 0.3930±0.0528 0.04388
iterative Schritte = 2	0.8401±0.0130 0.4416±0.0416 0.05114	0.9649±0.0140 0.0485±0.0169 0.00632	0.7735±0.0331 0.3770±0.0540 0.04227
iterative Schritte = 3	0.8356±0.0109 0.4524±0.0344 0.05592	0.9662±0.0131 0.0463±0.0142 0.00670	0.7720±0.0157 0.3840±0.0395 0.03859
relevante Positionsanzahl = 4			
iterative Schritte = 1	0.8390±0.0121 0.4415±0.0333 0.03712	0.9623±0.0193 0.0506±0.0243 0.00582	0.7565±0.0187 0.3960±0.0387 0.04388
iterative Schritte = 2	0.8361±0.0103 0.4502±0.0332 0.05119	0.9593±0.0158 0.0554±0.0184 0.00637	0.7690±0.0353 0.3670±0.0530 0.04116
iterative Schritte = 3	0.8309±0.0150 0.4689±0.0311 0.05648	0.9593±0.0133 0.0541±0.0160 0.00662	0.7540±0.0390 0.3925±0.0606 0.03700
relevante Positionsanzahl = 5			
iterative Schritte = 1	0.8329±0.0146 0.4487±0.0365 0.03712	0.9589±0.0201 0.0541±0.0212 0.00582	0.7595±0.0240 0.3900±0.0478 0.04388
iterative Schritte = 2	0.8317±0.0134 0.4654±0.0303 0.05173	0.9610±0.0176 0.0494±0.0187 0.00654	0.7575±0.0277 0.3810±0.0572 0.03738
iterative Schritte = 3	0.8337±0.0162 0.4811±0.0264 0.05655	0.9593±0.0149 0.0532±0.0162 0.00695	0.7520±0.0288 0.3905±0.0414 0.03303
relevante Positionsanzahl = 6			
iterative Schritte = 1	0.8321±0.0108 0.4543±0.0315 0.03712	0.9597±0.0191 0.0528±0.0201 0.00582	0.7410±0.0265 0.4095±0.0431 0.04388
iterative Schritte = 2	0.8312±0.0156 0.4741±0.0251 0.05172	0.9576±0.0178 0.0524±0.0181 0.00658	0.7460±0.0335 0.4045±0.0595 0.03933
iterative Schritte = 3	0.8311±0.0141 0.4790±0.0334 0.05682	0.9597±0.0145 0.0515±0.0166 0.00700	0.7550±0.0301 0.3975±0.0535 0.03548
relevante Positionsanzahl = 7			
iterative Schritte = 1	0.8350±0.0114 0.4449±0.0389 0.03712	-	0.7495±0.0276 0.3880±0.0498 0.04388
iterative Schritte = 2	0.8311±0.0070 0.4665±0.0259 0.05171	-	0.7610±0.0192 0.3925±0.0366 0.03763
iterative Schritte = 3	0.8296±0.0110 0.4835±0.0314 0.05665	-	0.7690±0.0318 0.3780±0.0509 0.03488
relevante Positionsanzahl = 8			
iterative Schritte = 1	0.8295±0.0108 0.4543±0.0330 0.03712	-	0.7550±0.0224 0.3825±0.0400 0.04388
iterative Schritte = 2	0.8305±0.0067 0.4664±0.0326 0.05183	-	0.7590±0.0213 0.3795±0.0514 0.03805
iterative Schritte = 3	0.8290±0.0106 0.4828±0.0416 0.05673	-	0.7595±0.0245 0.3825±0.0445 0.03562
relevante Positionsanzahl = 12			
iterative Schritte = 1	0.8333±0.0109 0.4395±0.0261 0.03712	-	0.7545±0.0276 0.3920±0.0438 0.04388
iterative Schritte = 2	0.8268±0.0101 0.4692±0.0278 0.05152	-	0.7560±0.0364 0.3840±0.0719 0.03791
iterative Schritte = 3	0.8260±0.0081 0.4743±0.0281 0.05620	-	0.7560±0.0287 0.3950±0.0539 0.03529
relevante Positionsanzahl = 16			
iterative Schritte = 1	0.8321±0.0119 0.4353±0.0284 0.03712	-	-
iterative Schritte = 2	0.8273±0.0078 0.4488±0.0313 0.05199	-	-
iterative Schritte = 3	0.8273±0.0076 0.4723±0.0312 0.05677	-	-
relevante Positionsanzahl = 20			
iterative Schritte = 1	0.8304±0.0142 0.4398±0.0322 0.03712	-	-
iterative Schritte = 2	0.8307±0.0086 0.4521±0.0288 0.05243	-	-
iterative Schritte = 3	0.8304±0.0093 0.4650±0.0341 0.05722	-	-

Abbildung 19: Kleines Datensatz-Set: ursprüngliches Label an erster Position im Ranking (Teil 2/2)

	letter.earff	mfeat-fourier.earff	oh0_wc.earff
originales LPCforSOS	0.9168±0.0077 0.1706±0.0162	0.7910±0.0270 0.3185±0.0398	0.7857±0.0420 0.5889±0.1061
Ursprüngliches Label auf beliebiger Position im Ranking			
relevante Positionsanzahl = 1	0.9070±0.0069 0.2336±0.0182	0.7875±0.0347 0.3360±0.0465	0.7708±0.0363 0.6846±0.1196
relevante Positionsanzahl = 2	0.8875±0.0096 0.2700±0.0205	0.7805±0.0315 0.3560±0.0539	0.7329±0.0297 0.7235±0.1142
relevante Positionsanzahl = 3	0.8804±0.0067 0.2828±0.0210	0.7805±0.0232 0.3575±0.0448	0.7070±0.0264 0.8124±0.0662
relevante Positionsanzahl = 4	0.8800±0.0069 0.2835±0.0206	0.7685±0.0267 0.3865±0.0446	0.7039±0.0282 0.8424±0.0995
relevante Positionsanzahl = 6	0.8782±0.0065 0.2865±0.0171	0.7555±0.0306 0.3915±0.0534	0.7000±0.0440 0.9012±0.1093
relevante Positionsanzahl = 8	0.8765±0.0070 0.2827±0.0184	0.7545±0.0340 0.3970±0.0524	0.6950±0.0422 0.9062±0.1174
relevante Positionsanzahl = 12	0.8735±0.0065 0.2844±0.0179	0.7540±0.0342 0.3925±0.0478	0.6940±0.0412 0.9002±0.1095
relevante Positionsanzahl = 16	0.8737±0.0072 0.2839±0.0170	-	-
relevante Positionsanzahl = 20	0.8741±0.0071 0.2816±0.0163	-	-
Zufällig genommene paarweise Präferenzen			
20,00%	0.7860±0.0081 0.4873±0.0242	0.6805±0.0347 0.5450±0.0702	0.5403±0.0462 1.3719±0.1296
40,00%	0.8189±0.0117 0.4092±0.0240	0.6955±0.0373 0.4945±0.0482	0.5843±0.0560 1.2779±0.1338
60,00%	0.8344±0.0087 0.3685±0.0132	0.7280±0.0185 0.4540±0.0418	0.6012±0.0440 1.2260±0.1463
80,00%	0.8484±0.0081 0.3418±0.0249	0.7350±0.0179 0.4460±0.0543	0.6501±0.0435 1.0216±0.1396
20% relevanter Positionsanzahl = 1	0.8144±0.0083 0.5146±0.0253	0.6805±0.0184 0.6040±0.0453	0.5663±0.0439 1.2802±0.1347
20% relevanter Positionsanzahl = 2	0.7920±0.0085 0.5502±0.0288	0.6885±0.0339 0.5320±0.0734	0.5783±0.0415 1.1135±0.1407
20% relevanter Positionsanzahl = 3	0.7918±0.0087 0.5090±0.0307	0.6635±0.0267 0.5675±0.0746	0.5474±0.0369 1.2518±0.2309
20% relevanter Positionsanzahl = 4	0.7836±0.0111 0.5216±0.0230	0.6960±0.0346 0.4985±0.0308	0.5813±0.0361 1.2231±0.1687
20% relevanter Positionsanzahl = 6	0.7848±0.0065 0.5217±0.0237	0.6720±0.0400 0.5405±0.0763	0.5294±0.0278 1.3898±0.1344
20% relevanter Positionsanzahl = 8	0.7952±0.0094 0.4840±0.0250	0.6735±0.0236 0.5375±0.0458	0.5473±0.0339 1.3866±0.1182
20% relevanter Positionsanzahl = 12	0.7874±0.0078 0.4943±0.0339	0.6715±0.0215 0.5660±0.0688	0.5334±0.0646 1.3896±0.1911
20% relevanter Positionsanzahl = 16	0.7848±0.0115 0.4877±0.0225	-	-
20% relevanter Positionsanzahl = 20	0.7868±0.0131 0.4884±0.0348	-	-
40% relevanter Positionsanzahl = 1	0.8536±0.0087 0.3828±0.0253	0.7430±0.0349 0.4360±0.0627	0.6901±0.0532 0.8721±0.1033
40% relevanter Positionsanzahl = 2	0.8254±0.0073 0.4174±0.0302	0.7390±0.0277 0.4325±0.0508	0.6192±0.0385 0.9488±0.1316
40% relevanter Positionsanzahl = 3	0.8295±0.0091 0.4055±0.0296	0.7200±0.0160 0.4740±0.0519	0.6311±0.0461 1.0028±0.1542
40% relevanter Positionsanzahl = 4	0.8262±0.0094 0.4118±0.0235	0.7220±0.0366 0.4605±0.0637	0.6052±0.0572 1.1045±0.1662
40% relevanter Positionsanzahl = 6	0.8162±0.0070 0.4222±0.0270	0.7165±0.0342 0.4790±0.0537	0.5703±0.0479 1.2748±0.1795
40% relevanter Positionsanzahl = 8	0.8291±0.0092 0.3968±0.0241	0.7085±0.0292 0.4785±0.0682	0.5933±0.0535 1.2958±0.1723
40% relevanter Positionsanzahl = 12	0.8226±0.0076 0.4023±0.0267	0.7085±0.0398 0.4700±0.0390	0.5813±0.0413 1.2710±0.1198
40% relevanter Positionsanzahl = 16	0.8217±0.0102 0.3989±0.0277	-	-
40% relevanter Positionsanzahl = 20	0.8210±0.0093 0.4068±0.0265	-	-
60% relevanter Positionsanzahl = 1	0.8678±0.0081 0.3284±0.0250	0.7670±0.0310 0.3770±0.0532	0.7269±0.0383 0.7585±0.1073
60% relevanter Positionsanzahl = 2	0.8443±0.0092 0.3668±0.0326	0.7565±0.0376 0.4060±0.0696	0.6691±0.0430 0.8641±0.1212
60% relevanter Positionsanzahl = 3	0.8498±0.0077 0.3510±0.0239	0.7190±0.0244 0.4555±0.0504	0.6561±0.0289 0.9521±0.0883
60% relevanter Positionsanzahl = 4	0.8397±0.0099 0.3713±0.0171	0.7410±0.0274 0.4305±0.0451	0.6421±0.0408 1.0625±0.1423
60% relevanter Positionsanzahl = 6	0.8368±0.0078 0.3788±0.0268	0.7275±0.0225 0.4495±0.0529	0.5923±0.0358 1.2191±0.1498
60% relevanter Positionsanzahl = 8	0.8448±0.0086 0.3542±0.0230	0.7330±0.0253 0.4380±0.0406	0.6162±0.0339 1.1902±0.1650
60% relevanter Positionsanzahl = 12	0.8385±0.0058 0.3544±0.0172	0.7245±0.0318 0.4355±0.0387	0.6113±0.0580 1.1880±0.1842
60% relevanter Positionsanzahl = 16	0.8382±0.0098 0.3584±0.0180	-	-
60% relevanter Positionsanzahl = 20	0.8371±0.0074 0.3643±0.0193	-	-
80% relevanter Positionsanzahl = 1	0.8751±0.0074 0.3034±0.0219	0.7685±0.0327 0.3955±0.0695	0.7309±0.0505 0.7594±0.1376
80% relevanter Positionsanzahl = 2	0.8553±0.0074 0.3344±0.0241	0.7630±0.0163 0.3990±0.0284	0.6740±0.0466 0.8322±0.1127
80% relevanter Positionsanzahl = 3	0.8590±0.0066 0.3274±0.0260	0.7225±0.0366 0.4765±0.0680	0.6500±0.0494 0.9271±0.1281
80% relevanter Positionsanzahl = 4	0.8548±0.0071 0.3362±0.0207	0.7345±0.0251 0.4550±0.0457	0.6730±0.0280 0.9411±0.1080
80% relevanter Positionsanzahl = 6	0.8463±0.0103 0.3574±0.0262	0.7300±0.0277 0.4440±0.0457	0.6502±0.0471 1.0435±0.1343
80% relevanter Positionsanzahl = 8	0.8535±0.0079 0.3332±0.0210	0.7260±0.0310 0.4385±0.0454	0.6401±0.0411 1.0555±0.1305
80% relevanter Positionsanzahl = 12	0.8479±0.0068 0.3357±0.0167	0.7255±0.0520 0.4410±0.0710	0.6442±0.0433 1.0705±0.1412
80% relevanter Positionsanzahl = 16	0.8488±0.0061 0.3364±0.0157	-	-
80% relevanter Positionsanzahl = 20	0.8439±0.0072 0.3421±0.0195	-	-
Ursprüngliches Label auf erster Position im Ranking			
relevante Positionsanzahl = 1	0.9168±0.0077 0.1706±0.0162 0.02622	0.7910±0.0270 0.3185±0.0398 0.04388	0.7857±0.0420 0.5889±0.1061 0.08729
relevante Positionsanzahl = 2	0.8886±0.0081 0.2485±0.0155 0.02622	0.7715±0.0241 0.3575±0.0484 0.04388	0.7309±0.0397 0.7135±0.1339 0.08729
relevante Positionsanzahl = 3	0.8842±0.0083 0.2668±0.0223 0.02622	0.7625±0.0202 0.3820±0.0397 0.04388	0.7149±0.0391 0.7825±0.1093 0.08729
relevante Positionsanzahl = 4	0.8824±0.0071 0.2736±0.0218 0.02622	0.7590±0.0247 0.3990±0.0364 0.04388	0.7140±0.0536 0.8273±0.1094 0.08729
relevante Positionsanzahl = 6	0.8826±0.0072 0.2752±0.0201 0.02622	0.7535±0.0323 0.3875±0.0535 0.04388	0.7130±0.0534 0.8892±0.1175 0.08729
relevante Positionsanzahl = 8	0.8795±0.0080 0.2750±0.0207 0.02622	0.7605±0.0266 0.3875±0.0462 0.04388	0.7090±0.0500 0.8922±0.1321 0.08729
relevante Positionsanzahl = 12	0.8763±0.0061 0.2760±0.0164 0.02622	0.7600±0.0220 0.3835±0.0419 0.04388	0.7090±0.0495 0.8922±0.1263 0.08729
relevante Positionsanzahl = 16	0.8756±0.0084 0.2770±0.0157 0.02622	-	-
relevante Positionsanzahl = 20	0.8776±0.0075 0.2746±0.0171 0.02622	-	-
Zufällig genommene paarweise Präferenzen			
20,00%	0.7859±0.0079 0.4790±0.0279 0.02622	0.6690±0.0390 0.5620±0.0667 0.04388	0.5313±0.0450 1.3975±0.1492 0.08729
40,00%	0.8205±0.0099 0.3998±0.0210 0.02622	0.7155±0.0325 0.4600±0.0563 0.04388	0.5822±0.0506 1.2612±0.1631 0.08729
60,00%	0.8370±0.0055 0.3606±0.0156 0.02622	0.7140±0.0397 0.4685±0.0621 0.04388	0.5933±0.0472 1.2439±0.1607 0.08729
80,00%	0.8453±0.0047 0.3384±0.0155 0.02622	0.7245±0.0226 0.4375±0.0426 0.04388	0.6382±0.0522 1.0645±0.1445 0.08729
20% relevanter Positionsanzahl = 1	0.8212±0.0105 0.4622±0.0279 0.02622	0.6900±0.0337 0.5430±0.0870 0.04388	0.5545±0.0584 1.2887±0.1925 0.08729
20% relevanter Positionsanzahl = 2	0.7867±0.0101 0.5265±0.0258 0.02622	0.6830±0.0363 0.5245±0.0807 0.04388	0.5694±0.0498 1.1741±0.1731 0.08729
20% relevanter Positionsanzahl = 3	0.7912±0.0092 0.5066±0.0213 0.02622	0.6605±0.0402 0.5780±0.0900 0.04388	0.5384±0.0794 1.2593±0.1676 0.08729
20% relevanter Positionsanzahl = 4	0.7896±0.0092 0.5000±0.0274 0.02622	0.6800±0.0421 0.5340±0.0776 0.04388	0.5644±0.0339 1.2351±0.1303 0.08729
20% relevanter Positionsanzahl = 6	0.7850±0.0060 0.5197±0.0244 0.02622	0.6865±0.0315 0.5395±0.0557 0.04388	0.5543±0.0410 1.3261±0.1563 0.08729
20% relevanter Positionsanzahl = 8	0.7896±0.0086 0.4859±0.0310 0.02622	0.6780±0.0326 0.5500±0.0365 0.04388	0.5184±0.0481 1.4384±0.1571 0.08729
20% relevanter Positionsanzahl = 12	0.7866±0.0100 0.4985±0.0291 0.02622	0.6475±0.0274 0.5785±0.0520 0.04388	0.5334±0.0412 1.3897±0.1483 0.08729
20% relevanter Positionsanzahl = 16	0.7853±0.0124 0.4916±0.0319 0.02622	-	-
20% relevanter Positionsanzahl = 20	0.7870±0.0099 0.4793±0.0286 0.02622	-	-
40% relevanter Positionsanzahl = 1	0.8556±0.0077 0.3350±0.0226 0.02622	0.7320±0.0404 0.4665±0.0842 0.04388	0.7070±0.0676 0.8242±0.1769 0.08729
40% relevanter Positionsanzahl = 2	0.8287±0.0082 0.4060±0.0283 0.02622	0.7295±0.0264 0.4280±0.0571 0.04388	0.6363±0.0606 0.9398±0.1622 0.08729
40% relevanter Positionsanzahl = 3	0.8318±0.0093 0.3983±0.0354 0.02622	0.7170±0.0270 0.4580±0.0539 0.04388	0.6163±0.0497 1.0147±0.1521 0.08729
40% relevanter Positionsanzahl = 4	0.8303±0.0061 0.3995±0.0164 0.02622	0.7100±0.0314 0.4725±0.0617 0.04388	0.6143±0.0416 1.0715±0.1646 0.08729

40% relevanter Positionsanzahl = 6	0.8228±0.0113 0.4126±0.0300 0.02622	0.7235±0.0417 0.4560±0.0698 0.04388	0.5942±0.0491 1.2511±0.1686 0.08729
40% relevanter Positionsanzahl = 8	0.8273±0.0109 0.3980±0.0238 0.02622	0.7060±0.0301 0.4735±0.0580 0.04388	0.5833±0.0361 1.2668±0.1631 0.08729
40% relevanter Positionsanzahl = 12	0.8230±0.0073 0.3908±0.0235 0.02622	0.7160±0.0368 0.4800±0.0684 0.04388	0.5983±0.0437 1.2698±0.1840 0.08729
40% relevanter Positionsanzahl = 16	0.8229±0.0069 0.3944±0.0241 0.02622	-	-
40% relevanter Positionsanzahl = 20	0.8189±0.0108 0.4001±0.0239 0.02622	-	-
60% relevanter Positionsanzahl = 1	0.8757±0.0095 0.2834±0.0256 0.02622	0.7475±0.0293 0.3970±0.0525 0.04388	0.7229±0.0487 0.7604±0.1546 0.08729
60% relevanter Positionsanzahl = 2	0.8497±0.0093 0.3341±0.0252 0.02622	0.7335±0.0236 0.4265±0.0411 0.04388	0.6472±0.0531 0.8193±0.1209 0.08729
60% relevanter Positionsanzahl = 3	0.8515±0.0089 0.3359±0.0211 0.02622	0.7280±0.0228 0.4440±0.0571 0.04388	0.6193±0.0535 0.9767±0.1328 0.08729
60% relevanter Positionsanzahl = 4	0.8454±0.0088 0.3544±0.0252 0.02622	0.7230±0.0165 0.4435±0.0527 0.04388	0.6251±0.0251 1.0676±0.1676 0.08729
60% relevanter Positionsanzahl = 6	0.8372±0.0118 0.3742±0.0267 0.02622	0.7165±0.0359 0.4550±0.0537 0.04388	0.6082±0.0596 1.1713±0.1772 0.08729
60% relevanter Positionsanzahl = 8	0.8449±0.0093 0.3468±0.0185 0.02622	0.7225±0.0273 0.4675±0.0666 0.04388	0.6023±0.0500 1.2140±0.1794 0.08729
60% relevanter Positionsanzahl = 12	0.8401±0.0081 0.3539±0.0193 0.02622	0.7250±0.0332 0.4525±0.0565 0.04388	0.6133±0.0331 1.2310±0.1532 0.08729
60% relevanter Positionsanzahl = 16	0.8401±0.0073 0.3563±0.0259 0.02622	-	-
60% relevanter Positionsanzahl = 20	0.8390±0.0067 0.3576±0.0149 0.02622	-	-
Prozentsatz der relevanten Positionsanzahl ab der zweiten Position			
20% relevanter Positionsanzahl >= 2			
iterative Schritte = 1			
iterative Schritte = 2			
iterative Schritte = 3			
40% relevanter Positionsanzahl >= 2			
iterative Schritte = 1			
iterative Schritte = 2			
iterative Schritte = 3			
60% relevanter Positionsanzahl >= 2			
iterative Schritte = 1			
iterative Schritte = 2			
iterative Schritte = 3			
80% relevanter Positionsanzahl >= 2			
iterative Schritte = 1			
iterative Schritte = 2			
iterative Schritte = 3			
Ranking ignorieren bei falscher Vorhersage erster Position			
relevante Positionsanzahl = 2			
iterative Schritte = 1			
iterative Schritte = 2			
iterative Schritte = 3			
relevante Positionsanzahl = 3			
iterative Schritte = 1			
iterative Schritte = 2			
iterative Schritte = 3			
relevante Positionsanzahl = 4			
iterative Schritte = 1			
iterative Schritte = 2			
iterative Schritte = 3			
relevante Positionsanzahl = 6			
iterative Schritte = 1			
iterative Schritte = 2			
iterative Schritte = 3			
relevante Positionsanzahl = 8			
iterative Schritte = 1			
iterative Schritte = 2			
iterative Schritte = 3			
relevante Positionsanzahl = 12			
iterative Schritte = 1			
iterative Schritte = 2			
iterative Schritte = 3			
relevante Positionsanzahl = 16			
iterative Schritte = 1			
iterative Schritte = 2			
iterative Schritte = 3			
relevante Positionsanzahl = 20			
iterative Schritte = 1			
iterative Schritte = 2			
iterative Schritte = 3			

Abbildung 20: J48: letter, mfeat-fourier, oh0.wc

	oh5_wc.earff	oh10_wc.earff	oh15_wc.earff
originales LPCforSOS	0.7995±0.0516 0.5788±0.1710	0.7343±0.0375 0.7314±0.1850	0.7634±0.0383 0.6982±0.1973
Ursprüngliches Label auf beliebiger Position im Ranking			
relevante Positionsanzahl = 1	0.7919±0.0536 0.6954±0.2331	0.7229±0.0328 0.8457±0.1879	0.7426±0.0456 0.8239±0.2070
relevante Positionsanzahl = 2	0.7570±0.0582 0.7511±0.2680	0.6952±0.0263 0.8810±0.1740	0.7109±0.0448 0.8600±0.2030
relevante Positionsanzahl = 3	0.7625±0.0601 0.7216±0.2373	0.6790±0.0341 0.9362±0.1539	0.6747±0.0501 0.9664±0.2483
relevante Positionsanzahl = 4	0.7548±0.0662 0.7326±0.2443	0.7019±0.0341 0.9095±0.1504	0.6671±0.0345 0.10287±0.2163
relevante Positionsanzahl = 6	0.7494±0.0585 0.7544±0.2405	0.6971±0.0279 0.9724±0.1679	0.6506±0.0494 1.0616±0.2424
relevante Positionsanzahl = 8	0.7472±0.0514 0.7500±0.2061	0.6943±0.0342 0.9619±0.1696	0.6452±0.0473 1.0703±0.2460
relevante Positionsanzahl = 12	0.7461±0.0583 0.7511±0.2209	0.6943±0.0342 0.9610±0.1671	0.6451±0.0502 1.0659±0.2328
relevante Positionsanzahl = 16	-	-	-
relevante Positionsanzahl = 20	-	-	-
Zufällig genommene paarweise Präferenzen			
20,00%	0.5718±0.0588 1.1442±0.1631	0.5562±0.0562 1.4686±0.2711	0.4544±0.0575 1.7242±0.2924
40,00%	0.6132±0.0570 1.0410±0.2477	0.5686±0.0480 1.4086±0.2091	0.5103±0.0570 1.5863±0.2622
60,00%	0.6437±0.0546 0.9813±0.2085	0.5781±0.0447 1.3162±0.1656	0.5082±0.0722 1.5028±0.2834
80,00%	0.7003±0.0785 0.8524±0.2255	0.6438±0.0283 1.0762±0.1797	0.5848±0.0522 1.2414±0.2336
20% relevanter Positionsanzahl = 1	0.5685±0.0402 1.2401±0.2001	0.5419±0.0300 1.3495±0.1675	0.5026±0.0772 1.5283±0.3225
20% relevanter Positionsanzahl = 2	0.6241±0.0456 1.0780±0.2436	0.5771±0.0354 1.1943±0.1474	0.5597±0.0554 1.2983±0.2710
20% relevanter Positionsanzahl = 3	0.5370±0.0642 1.2641±0.2142	0.5686±0.0245 1.2238±0.1952	0.5257±0.0558 1.3925±0.3113
20% relevanter Positionsanzahl = 4	0.6033±0.0583 1.1411±0.1982	0.5838±0.0233 1.2343±0.1886	0.5059±0.0521 1.4429±0.2762
20% relevanter Positionsanzahl = 6	0.5773±0.0603 1.1573±0.2147	0.5562±0.0497 1.3886±0.2101	0.4784±0.0721 1.6395±0.3807
20% relevanter Positionsanzahl = 8	0.5241±0.0767 1.2715±0.1864	0.5114±0.0560 1.4924±0.2226	0.4611±0.0782 1.7100±0.2847
20% relevanter Positionsanzahl = 12	0.5260±0.0752 1.2958±0.2445	0.5590±0.0233 1.4352±0.1529	0.4480±0.0382 1.7725±0.2495
20% relevanter Positionsanzahl = 16	-	-	-
20% relevanter Positionsanzahl = 20	-	-	-
40% relevanter Positionsanzahl = 1	0.7177±0.0741 0.8981±0.2916	0.6524±0.0369 1.0610±0.1920	0.6550±0.0620 1.0409±0.2787
40% relevanter Positionsanzahl = 2	0.7135±0.0586 0.8730±0.2642	0.6419±0.0296 1.0638±0.1742	0.6495±0.0669 1.0255±0.2470
40% relevanter Positionsanzahl = 3	0.6732±0.0459 0.9200±0.2424	0.6181±0.0314 1.0905±0.1538	0.5849±0.0651 1.1655±0.2667
40% relevanter Positionsanzahl = 4	0.6851±0.0587 0.9285±0.2304	0.6171±0.0493 1.1429±0.1631	0.5236±0.0557 1.3013±0.2284
40% relevanter Positionsanzahl = 6	0.6405±0.0364 0.9841±0.2031	0.5952±0.0384 1.2810±0.2042	0.5365±0.0589 1.4045±0.2720
40% relevanter Positionsanzahl = 8	0.6328±0.0548 1.0202±0.2150	0.6095±0.0374 1.2990±0.1831	0.4993±0.0574 1.5997±0.2892
40% relevanter Positionsanzahl = 12	0.6437±0.0655 1.0366±0.2429	0.5857±0.0177 1.3562±0.1702	0.4763±0.0731 1.6194±0.3165
40% relevanter Positionsanzahl = 16	-	-	-
40% relevanter Positionsanzahl = 20	-	-	-
60% relevanter Positionsanzahl = 1	0.7407±0.0586 0.8414±0.2571	0.6752±0.0389 0.9610±0.1908	0.6693±0.0663 1.0298±0.2722
60% relevanter Positionsanzahl = 2	0.7134±0.0529 0.8416±0.2421	0.6581±0.0464 0.9724±0.1904	0.6550±0.0517 0.9968±0.2055
60% relevanter Positionsanzahl = 3	0.7145±0.0656 0.8731±0.2666	0.6371±0.0311 1.0600±0.1374	0.6199±0.0475 1.0671±0.2408
60% relevanter Positionsanzahl = 4	0.6938±0.0566 0.9111±0.2371	0.6343±0.0506 1.1038±0.1964	0.5684±0.0595 1.2194±0.2580
60% relevanter Positionsanzahl = 6	0.6851±0.0584 0.9112±0.2175	0.6248±0.0386 1.1857±0.1546	0.5387±0.0735 1.3749±0.2723
60% relevanter Positionsanzahl = 8	0.6753±0.0615 0.9145±0.2374	0.6029±0.0509 1.2381±0.2247	0.5289±0.0762 1.4858±0.3089
60% relevanter Positionsanzahl = 12	0.6633±0.0635 0.9221±0.2147	0.6029±0.0292 1.2857±0.1600	0.5169±0.0550 1.5217±0.3150
60% relevanter Positionsanzahl = 16	-	-	-
60% relevanter Positionsanzahl = 20	-	-	-
80% relevanter Positionsanzahl = 1	0.7549±0.0534 0.7814±0.2316	0.6762±0.0369 0.9819±0.1734	0.6900±0.0469 0.8919±0.1956
80% relevanter Positionsanzahl = 2	0.7233±0.0546 0.8329±0.2429	0.6752±0.0439 0.9410±0.1847	0.6572±0.0456 0.9850±0.2192
80% relevanter Positionsanzahl = 3	0.7243±0.0602 0.8427±0.2744	0.6467±0.0215 0.2086±0.1411	0.6473±0.0547 1.0397±0.2482
80% relevanter Positionsanzahl = 4	0.6971±0.0707 0.8753±0.2520	0.6629±0.0391 1.0429±0.1861	0.6079±0.0619 1.1130±0.2271
80% relevanter Positionsanzahl = 6	0.7156±0.0399 0.8468±0.1704	0.6552±0.0487 1.0895±0.2099	0.5826±0.0563 1.2226±0.2914
80% relevanter Positionsanzahl = 8	0.7178±0.0652 0.8361±0.2183	0.6657±0.0342 1.0514±0.1497	0.5750±0.0554 1.2402±0.2560
80% relevanter Positionsanzahl = 12	0.7091±0.0512 0.8633±0.2252	0.6371±0.0337 1.1171±0.1458	0.5411±0.0533 1.2948±0.2133
80% relevanter Positionsanzahl = 16	-	-	-
80% relevanter Positionsanzahl = 20	-	-	-
Ursprüngliches Label auf erster Position im Ranking			
relevante Positionsanzahl = 1	0.7995±0.0516 0.5788±0.1710 0.05543	0.7343±0.0375 0.7314±0.1850 0.07481	0.7634±0.0383 0.6982±0.1973 0.05598
relevante Positionsanzahl = 2	0.7505±0.0516 0.7096±0.2245 0.05543	0.6990±0.0311 0.8638±0.1606 0.07481	0.7000±0.0442 0.8885±0.2150 0.05598
relevante Positionsanzahl = 3	0.7658±0.0544 0.6997±0.2241 0.05543	0.6810±0.0365 0.9181±0.1641 0.07481	0.6605±0.0438 0.9686±0.2298 0.05598
relevante Positionsanzahl = 4	0.7527±0.0517 0.7402±0.2171 0.05543	0.6971±0.0272 0.9076±0.1563 0.07481	0.6572±0.0294 1.0474±0.2033 0.05598
relevante Positionsanzahl = 6	0.7516±0.0600 0.7533±0.2291 0.05543	0.6905±0.0260 0.9848±0.1753 0.07481	0.6517±0.0405 1.0704±0.2382 0.05598
relevante Positionsanzahl = 8	0.7494±0.0404 0.7434±0.2031 0.05543	0.6924±0.0276 0.9629±0.1665 0.07481	0.6397±0.0525 1.0889±0.2523 0.05598
relevante Positionsanzahl = 12	0.7516±0.0529 0.7326±0.2210 0.05543	0.6924±0.0256 0.9629±0.1608 0.07481	0.6342±0.0458 1.0966±0.2423 0.05598
relevante Positionsanzahl = 16	-	-	-
relevante Positionsanzahl = 20	-	-	-
Zufällig genommene paarweise Präferenzen			
20,00%	0.5446±0.0629 1.2414±0.2531 0.05543	0.5695±0.0301 1.4171±0.1869 0.07481	0.4731±0.0566 1.5944±0.3453 0.05598
40,00%	0.5990±0.0588 1.0537±0.1923 0.05543	0.5867±0.0437 1.3448±0.2124 0.07481	0.5015±0.0465 1.5976±0.3014 0.05598
60,00%	0.6568±0.0613 0.9667±0.2089 0.05543	0.5924±0.0440 1.3029±0.2105 0.07481	0.5136±0.0695 1.5207±0.2773 0.05598
80,00%	0.7057±0.0675 0.8449±0.2337 0.05543	0.6429±0.0424 1.0905±0.1499 0.07481	0.5640±0.0774 1.2796±0.2976 0.05598
20% relevanter Positionsanzahl = 1	0.5686±0.0550 1.1930±0.2182 0.05543	0.5571±0.0691 1.3152±0.3200 0.07481	0.4886±0.0480 1.6520±0.2555 0.05598
20% relevanter Positionsanzahl = 2	0.6416±0.0530 1.0442±0.2071 0.05543	0.5629±0.0546 1.2257±0.1954 0.07481	0.5268±0.0884 1.3211±0.3031 0.05598
20% relevanter Positionsanzahl = 3	0.5969±0.0647 1.1333±0.2049 0.05543	0.5581±0.0374 1.2590±0.1539 0.07481	0.4939±0.0404 1.4365±0.2499 0.05598
20% relevanter Positionsanzahl = 4	0.6046±0.0777 1.0766±0.2181 0.05543	0.5695±0.0584 1.2552±0.1945 0.07481	0.5179±0.0641 1.4112±0.2811 0.05598
20% relevanter Positionsanzahl = 6	0.5914±0.0486 1.1029±0.1817 0.05543	0.5629±0.0514 1.3695±0.2146 0.07481	0.4740±0.0912 1.6349±0.3301 0.05598
20% relevanter Positionsanzahl = 8	0.5336±0.0886 1.2577±0.2536 0.05543	0.5457±0.0495 1.4505±0.2214 0.07481	0.4534±0.0688 1.7221±0.2984 0.05598
20% relevanter Positionsanzahl = 12	0.5860±0.0810 1.1748±0.2486 0.05543	0.5286±0.0600 1.5162±0.3054 0.07481	0.4578±0.0791 1.7619±0.3701 0.05598
20% relevanter Positionsanzahl = 16	-	-	-
20% relevanter Positionsanzahl = 20	-	-	-
40% relevanter Positionsanzahl = 1	0.7134±0.0722 0.9037±0.2718 0.05543	0.6524±0.0504 0.9933±0.1948 0.07481	0.6505±0.0824 0.9838±0.2484 0.05598
40% relevanter Positionsanzahl = 2	0.6752±0.0536 0.9224±0.2587 0.05543	0.6210±0.0502 1.0305±0.2243 0.07481	0.6342±0.0521 1.0737±0.2537 0.05598
40% relevanter Positionsanzahl = 3	0.6808±0.0358 0.9448±0.2175 0.05543	0.6314±0.0263 1.0333±0.1505 0.07481	0.5772±0.0554 1.1688±0.2326 0.05598
40% relevanter Positionsanzahl = 4	0.6743±0.0460 0.9405±0.2107 0.05543	0.5981±0.0343 1.1800±0.1889 0.07481	0.5552±0.0590 1.1833±0.2623 0.05598

40% relevanter Positionsanzahl = 6	0.6514±0.0578 0.9885±0.2289 0.05543	0.5924±0.0251 1.2619±0.1379 0.07481	0.5312±0.0716 1.3860±0.3168 0.05598
40% relevanter Positionsanzahl = 8	0.6448±0.0522 0.9994±0.2079 0.05543	0.5867±0.0613 1.3562±0.2424 0.07481	0.5059±0.0623 1.5885±0.3129 0.05598
40% relevanter Positionsanzahl = 12	0.6404±0.0463 1.0290±0.2449 0.05543	0.5714±0.0602 1.4219±0.2500 0.07481	0.5059±0.0620 1.5886±0.3502 0.05598
40% relevanter Positionsanzahl = 16	-	-	-
40% relevanter Positionsanzahl = 20	-	-	-
60% relevanter Positionsanzahl = 1	0.7482±0.0716 0.7676±0.2778 0.05543	0.6686±0.0525 0.9057±0.1706 0.07481	0.7031±0.0511 0.8580±0.1984 0.05598
60% relevanter Positionsanzahl = 2	0.7156±0.0516 0.8567±0.2507 0.05543	0.6514±0.0459 0.9600±0.2029 0.07481	0.6693±0.0551 0.9717±0.2227 0.05598
60% relevanter Positionsanzahl = 3	0.7199±0.0500 0.8307±0.2477 0.05543	0.6248±0.0293 1.0486±0.1816 0.07481	0.6078±0.0673 1.1197±0.2737 0.05598
60% relevanter Positionsanzahl = 4	0.6894±0.0594 0.9025±0.2475 0.05543	0.6286±0.0439 1.0829±0.1881 0.07481	0.5859±0.0577 1.1996±0.2387 0.05598
60% relevanter Positionsanzahl = 6	0.6688±0.0771 0.9319±0.2418 0.05543	0.6238±0.0402 1.1829±0.1892 0.07481	0.5519±0.0658 1.3840±0.3105 0.05598
60% relevanter Positionsanzahl = 8	0.6676±0.0471 0.9222±0.2222 0.05543	0.6210±0.0434 1.2400±0.2234 0.07481	0.5321±0.0564 1.4638±0.2690 0.05598
60% relevanter Positionsanzahl = 12	0.6678±0.0449 0.9446±0.1682 0.05543	0.6019±0.0348 1.2371±0.1728 0.07481	0.5333±0.0455 1.5297±0.2898 0.05598
60% relevanter Positionsanzahl = 16	-	-	-
60% relevanter Positionsanzahl = 20	-	-	-
Prozentsatz der relevanten Positionsanzahl ab der zweiten Position			
20% relevanter Positionsanzahl >= 2			
iterative Schritte = 1	0.7439±0.0553 0.7574±0.1907 0.05543	0.6762±0.0330 0.8610±0.1619 0.07481	0.7000±0.0578 0.8448±0.2277 0.05598
iterative Schritte = 2	0.7189±0.0647 0.8043±0.2348 0.05543	0.6714±0.0402 0.9029±0.1530 0.07481	0.6462±0.0552 0.9884±0.2458 0.05598
iterative Schritte = 3	0.7363±0.0471 0.8426±0.2391 0.05543	0.6495±0.0329 1.0171±0.1809 0.07481	0.6352±0.0636 1.0539±0.2602 0.05598
40% relevanter Positionsanzahl >= 2			
iterative Schritte = 1	0.7308±0.0642 0.8177±0.2524 0.05543	0.6429±0.0471 1.0429±0.1697 0.07481	0.6265±0.0494 1.1033±0.2412 0.05598
iterative Schritte = 2	0.7101±0.0464 0.8328±0.1896 0.05543	0.6400±0.0321 0.9090±0.1714 0.07481	0.5836±0.0575 1.2064±0.2614 0.05598
iterative Schritte = 3	0.7112±0.0489 0.8361±0.2202 0.05543	0.6352±0.0575 1.0705±0.1938 0.07481	0.5782±0.0579 1.2599±0.2747 0.05598
80% relevanter Positionsanzahl = 12	0.6960±0.0443 0.8631±0.1883 0.05543	0.6286±0.0381 1.1010±0.1610 0.07481	0.5629±0.0618 1.2643±0.3030 0.05598
80% relevanter Positionsanzahl = 16	-	-	-
80% relevanter Positionsanzahl = 20	-	-	-
Ranking ignorieren bei falscher Vorhersage erster Position			
relevante Positionsanzahl = 2			
iterative Schritte = 1	0.7908±0.0446 0.6137±0.1970 0.05543	0.7190±0.0267 0.7629±0.1566 0.07481	0.7558±0.0317 0.7156±0.1466 0.05598
iterative Schritte = 2	0.7766±0.0446 0.6475±0.1777 0.06342	0.7190±0.0263 0.7600±0.1545 0.08264	0.7613±0.0435 0.6959±0.1829 0.06224
iterative Schritte = 3	0.7799±0.0387 0.6353±0.1662 0.06604	0.7248±0.0507 0.7714±0.1716 0.08670	0.7502±0.0323 0.7320±0.1826 0.06656
40% relevanter Positionsanzahl >= 2			
iterative Schritte = 1	0.7723±0.0620 0.6529±0.2045 0.05543	0.6990±0.0374 0.8333±0.1780 0.07481	0.7229±0.0500 0.7911±0.1856 0.05598
iterative Schritte = 2	0.7744±0.0526 0.6746±0.2209 0.06995	0.7000±0.0435 0.8038±0.1885 0.08772	0.7328±0.0454 0.8053±0.1693 0.07210
iterative Schritte = 3	0.7668±0.0491 0.6976±0.2000 0.07536	0.6981±0.0388 0.8333±0.1811 0.09312	0.7218±0.0498 0.7812±0.1648 0.08028
60% relevanter Positionsanzahl >= 2			
iterative Schritte = 1	0.7745±0.0448 0.6747±0.2029 0.05543	0.6952±0.0424 0.8390±0.1819 0.07481	0.7164±0.0335 0.8064±0.2012 0.05598
iterative Schritte = 2	0.7603±0.0537 0.7062±0.2130 0.07334	0.7000±0.0347 0.8276±0.1919 0.08989	0.7208±0.0469 0.8195±0.2074 0.07581
iterative Schritte = 3	0.7603±0.0396 0.6855±0.1828 0.07996	0.6943±0.0287 0.8533±0.1684 0.09738	0.7174±0.0642 0.8088±0.1939 0.08746
80% relevanter Positionsanzahl >= 2			
iterative Schritte = 1	0.7548±0.0715 0.7085±0.2370 0.05543	0.6962±0.0425 0.8438±0.1834 0.07481	0.7131±0.0326 0.8328±0.2023 0.05598
iterative Schritte = 2	0.7570±0.0623 0.7010±0.2586 0.07685	0.6933±0.0385 0.8514±0.1663 0.09560	0.7109±0.0483 0.8262±0.2131 0.08378
iterative Schritte = 3	0.7472±0.0503 0.7378±0.2003 0.08557	0.6838±0.0413 0.8400±0.1948 0.10278	0.7065±0.0428 0.8514±0.2034 0.09180
Ranking ignorieren bei falscher Vorhersage erster Position			
relevante Positionsanzahl = 2			
iterative Schritte = 1	0.7504±0.0546 0.7194±0.2317 0.05543	0.6924±0.0245 0.8562±0.1428 0.07481	0.6989±0.0402 0.8941±0.2240 0.05598
iterative Schritte = 2	0.7570±0.0490 0.7238±0.2274 0.07582	0.6838±0.0425 0.8771±0.1806 0.09317	0.7000±0.0424 0.8525±0.2058 0.08196
iterative Schritte = 3	0.7527±0.0519 0.7368±0.2350 0.08230	0.6829±0.0316 0.8619±0.1742 0.09932	0.6955±0.0431 0.8668±0.2206 0.09066
relevante Positionsanzahl = 3			
iterative Schritte = 1	0.7690±0.0381 0.7041±0.2076 0.05543	0.6848±0.0337 0.9095±0.1800 0.07481	0.6583±0.0453 0.9797±0.2503 0.05598
iterative Schritte = 2	0.7570±0.0538 0.7282±0.2564 0.07740	0.6867±0.0431 0.9057±0.1895 0.09497	0.6747±0.0488 0.9708±0.2614 0.08439
iterative Schritte = 3	0.7527±0.0556 0.7346±0.2348 0.08537	0.6857±0.0324 0.8981±0.1692 0.10201	0.6704±0.0317 0.9663±0.2162 0.09387
relevante Positionsanzahl = 4			
iterative Schritte = 1	0.7603±0.0430 0.7368±0.2038 0.05543	0.6829±0.0316 0.9143±0.1677 0.07481	0.6594±0.0431 0.10364±0.2414 0.05598
iterative Schritte = 2	0.7635±0.0469 0.7359±0.2310 0.07716	0.6771±0.0370 0.9333±0.1451 0.09391	0.6681±0.0355 1.0210±0.2001 0.08439
iterative Schritte = 3	0.7537±0.0510 0.7598±0.2202 0.08400	0.6752±0.0339 0.9352±0.1458 0.10186	0.6506±0.0323 1.0430±0.2081 0.09561
relevante Positionsanzahl = 6			
iterative Schritte = 1	0.7668±0.0552 0.7249±0.2092 0.05543	0.6905±0.0277 0.9781±0.1632 0.07481	0.6408±0.0455 1.0682±0.2506 0.05598
iterative Schritte = 2	0.7624±0.0492 0.7315±0.2058 0.07546	0.6867±0.0339 0.9857±0.1562 0.09354	0.6561±0.0584 1.0363±0.2474 0.08646
iterative Schritte = 3	0.7690±0.0517 0.7305±0.2214 0.08254	0.6810±0.0311 0.9781±0.1629 0.10056	0.6593±0.0391 1.0595±0.2565 0.09772
relevante Positionsanzahl = 8			
iterative Schritte = 1	0.7690±0.0511 0.7217±0.1955 0.05543	0.6848±0.0300 0.9686±0.1555 0.07481	0.6375±0.0371 1.0868±0.2363 0.05598
iterative Schritte = 2	0.7657±0.0463 0.7512±0.2058 0.07637	0.6771±0.0334 0.9648±0.1380 0.09243	0.6496±0.0513 1.0846±0.2424 0.08768
iterative Schritte = 3	0.7592±0.0522 0.7434±0.1942 0.08371	0.6829±0.0230 0.9562±0.1215 0.09904	0.6452±0.0431 1.1076±0.2407 0.09711
relevante Positionsanzahl = 12			
iterative Schritte = 1	0.7602±0.0553 0.7282±0.2042 0.05543	0.6838±0.0298 0.9733±0.1529 0.07481	0.6430±0.0375 1.0747±0.2201 0.05598
iterative Schritte = 2	0.7613±0.0433 0.7413±0.1969 0.07649	0.6790±0.0341 0.9657±0.1438 0.09227	0.6485±0.0455 1.0823±0.2257 0.08804
iterative Schritte = 3	0.7646±0.0447 0.7293±0.1936 0.08331	0.6876±0.0304 0.9514±0.1424 0.09908	0.6430±0.0419 1.0779±0.2264 0.09756
relevante Positionsanzahl = 16			
iterative Schritte = 1	-	-	-
iterative Schritte = 2	-	-	-
iterative Schritte = 3	-	-	-
relevante Positionsanzahl = 20			
iterative Schritte = 1	-	-	-
iterative Schritte = 2	-	-	-
iterative Schritte = 3	-	-	-

Abbildung 21: J48: oh5.wc, oh10.wc, oh15.wc

	optdigits.earff	re0_wc.earff	segment.earff
originales LPCforSOS	0.9457±0.0151 0.0822±0.0258	0.7700±0.0306 0.4136±0.0719	0.9680±0.0178 0.0437±0.0222
Ursprüngliches Label auf beliebiger Position im Ranking			
relevante Positionsanzahl = 1	0.9431±0.0103 0.0881±0.0165	0.7679±0.0266 0.5632±0.1032	0.9671±0.0159 0.0498±0.0249
relevante Positionsanzahl = 2	0.9249±0.0139 0.1187±0.0261	0.7474±0.0303 0.4867±0.0741	0.9645±0.0139 0.0459±0.0171
relevante Positionsanzahl = 3	0.9299±0.0051 0.1119±0.0114	0.7394±0.0309 0.5021±0.0744	0.9623±0.0188 0.0519±0.0209
relevante Positionsanzahl = 4	0.9256±0.0081 0.1155±0.0164	0.7480±0.0354 0.4801±0.0913	0.9602±0.0219 0.0519±0.0236
relevante Positionsanzahl = 6	0.9238±0.0114 0.1176±0.0179	0.7454±0.0382 0.5087±0.0872	0.9580±0.0196 0.0558±0.0194
relevante Positionsanzahl = 8	0.9221±0.0094 0.1256±0.0215	0.7440±0.0374 0.4908±0.0742	-
relevante Positionsanzahl = 12	0.9208±0.0115 0.1251±0.0217	0.7440±0.0372 0.4914±0.0714	-
relevante Positionsanzahl = 16	-	-	-
relevante Positionsanzahl = 20	-	-	-
Zufällig genommene paarweise Präferenzen			
20,00%	0.8635±0.0216 0.2078±0.0272	0.6443±0.0403 0.8537±0.1584	0.9203±0.0260 0.1026±0.0356
40,00%	0.8950±0.0086 0.1676±0.0166	0.6802±0.0399 0.7347±0.1430	0.9433±0.0190 0.0736±0.0226
60,00%	0.8938±0.0142 0.1641±0.0231	0.7008±0.0342 0.6077±0.1315	0.9528±0.0145 0.0671±0.0234
80,00%	0.9073±0.0115 0.1418±0.0200	0.7321±0.0310 0.5512±0.1128	0.9485±0.0153 0.0649±0.0196
20% relevanter Positionsanzahl = 1	0.8751±0.0145 0.2130±0.0370	0.6502±0.0332 0.9635±0.2558	0.9177±0.0128 0.1264±0.0231
20% relevanter Positionsanzahl = 2	0.8801±0.0050 0.1867±0.0271	0.6542±0.0390 0.8424±0.2174	0.9052±0.0240 0.1247±0.0304
20% relevanter Positionsanzahl = 3	0.8546±0.0142 0.2270±0.0313	0.6689±0.0346 0.7739±0.1621	0.9294±0.0199 0.0935±0.0310
20% relevanter Positionsanzahl = 4	0.8724±0.0085 0.2071±0.0249	0.6516±0.0518 0.7600±0.1710	0.9156±0.0163 0.1117±0.0230
20% relevanter Positionsanzahl = 6	0.8703±0.0122 0.2059±0.0250	0.6496±0.0414 0.8052±0.1812	0.9290±0.0129 0.0887±0.0173
20% relevanter Positionsanzahl = 8	0.8726±0.0140 0.2016±0.0261	0.6410±0.0348 0.8018±0.1664	-
20% relevanter Positionsanzahl = 12	0.8665±0.0205 0.2105±0.0387	0.6537±0.0424 0.8323±0.1638	-
20% relevanter Positionsanzahl = 16	-	-	-
20% relevanter Positionsanzahl = 20	-	-	-
40% relevanter Positionsanzahl = 1	0.9109±0.0127 0.1434±0.0209	0.6829±0.0357 0.8032±0.1851	0.9381±0.0243 0.0831±0.0327
40% relevanter Positionsanzahl = 2	0.8954±0.0109 0.1730±0.0214	0.6935±0.0489 0.6795±0.1781	0.9455±0.0223 0.0758±0.0311
40% relevanter Positionsanzahl = 3	0.8959±0.0128 0.1687±0.0246	0.6875±0.0383 0.6643±0.1079	0.9433±0.0202 0.0749±0.0222
40% relevanter Positionsanzahl = 4	0.8893±0.0126 0.1692±0.0321	0.7028±0.0357 0.6304±0.1112	0.9368±0.0163 0.0827±0.0203
40% relevanter Positionsanzahl = 6	0.8915±0.0116 0.1687±0.0245	0.6755±0.0395 0.7028±0.1693	0.9355±0.0224 0.0814±0.0248
40% relevanter Positionsanzahl = 8	0.8849±0.0111 0.1747±0.0179	0.6829±0.0538 0.7128±0.1999	-
40% relevanter Positionsanzahl = 12	0.8868±0.0142 0.1676±0.0248	0.6955±0.0391 0.7002±0.1529	-
40% relevanter Positionsanzahl = 16	-	-	-
40% relevanter Positionsanzahl = 20	-	-	-
60% relevanter Positionsanzahl = 1	0.9242±0.0122 0.1155±0.0226	0.7208±0.0306 0.6576±0.1626	0.9455±0.0151 0.0797±0.0301
60% relevanter Positionsanzahl = 2	0.9176±0.0107 0.1295±0.0190	0.7141±0.0434 0.5844±0.1349	0.9515±0.0134 0.0649±0.0187
60% relevanter Positionsanzahl = 3	0.8998±0.0096 0.1578±0.0230	0.7148±0.0428 0.5905±0.1071	0.9459±0.0205 0.0701±0.0315
60% relevanter Positionsanzahl = 4	0.9066±0.0112 0.1425±0.0187	0.7254±0.0391 0.5838±0.0881	0.9429±0.0170 0.0762±0.0232
60% relevanter Positionsanzahl = 6	0.9062±0.0127 0.1468±0.0164	0.6988±0.0290 0.6163±0.1293	0.9511±0.0146 0.0641±0.0202
60% relevanter Positionsanzahl = 8	0.8996±0.0127 0.1559±0.0244	0.7021±0.0352 0.6117±0.1144	-
60% relevanter Positionsanzahl = 12	0.8972±0.0091 0.1596±0.0214	0.7154±0.0402 0.5745±0.1038	-
60% relevanter Positionsanzahl = 16	-	-	-
60% relevanter Positionsanzahl = 20	-	-	-
80% relevanter Positionsanzahl = 1	0.9281±0.0096 0.1164±0.0223	0.7247±0.0369 0.6383±0.1515	0.9528±0.0145 0.0710±0.0205
80% relevanter Positionsanzahl = 2	0.9133±0.0072 0.1324±0.0148	0.7175±0.0381 0.5452±0.1026	0.9554±0.0151 0.0597±0.0220
80% relevanter Positionsanzahl = 3	0.9126±0.0138 0.1333±0.0236	0.7268±0.0533 0.5518±0.0971	0.9463±0.0151 0.0688±0.0189
80% relevanter Positionsanzahl = 4	0.9078±0.0080 0.1377±0.0186	0.7361±0.0571 0.5484±0.1257	0.9502±0.0170 0.0662±0.0204
80% relevanter Positionsanzahl = 6	0.9096±0.0097 0.1399±0.0171	0.7168±0.0334 0.5818±0.1301	0.9554±0.0173 0.0615±0.0238
80% relevanter Positionsanzahl = 8	0.9046±0.0078 0.1418±0.0149	0.7068±0.0457 0.5891±0.1137	-
80% relevanter Positionsanzahl = 12	0.8959±0.0159 0.1630±0.0254	0.7214±0.0333 0.5625±0.1093	-
80% relevanter Positionsanzahl = 16	-	-	-
80% relevanter Positionsanzahl = 20	-	-	-
Ursprüngliches Label auf erster Position im Ranking			
relevante Positionsanzahl = 1	0.9457±0.0151 0.0822±0.0258 0.01107	0.7700±0.0306 0.4136±0.0719 0.07971	0.9680±0.0178 0.0437±0.0222 0.00582
relevante Positionsanzahl = 2	0.9306±0.0106 0.1091±0.0226 0.01107	0.7460±0.0264 0.4735±0.0767 0.07971	0.9688±0.0134 0.0403±0.0189 0.00582
relevante Positionsanzahl = 3	0.9295±0.0048 0.1121±0.0129 0.01107	0.7307±0.0345 0.5074±0.0870 0.07971	0.9628±0.0141 0.0494±0.0160 0.00582
relevante Positionsanzahl = 4	0.9272±0.0097 0.1096±0.0172 0.01107	0.7553±0.0275 0.4834±0.0918 0.07971	0.9593±0.0187 0.0545±0.0233 0.00582
relevante Positionsanzahl = 6	0.9269±0.0138 0.1135±0.0220 0.01107	0.7520±0.0203 0.4980±0.0918 0.07971	0.9593±0.0192 0.0537±0.0208 0.00582
relevante Positionsanzahl = 8	0.9238±0.0078 0.1201±0.0175 0.01107	0.7414±0.0276 0.4920±0.0802 0.07971	-
relevante Positionsanzahl = 12	0.9244±0.0111 0.1192±0.0207 0.01107	0.7434±0.0253 0.4914±0.0747 0.07971	-
relevante Positionsanzahl = 16	-	-	-
relevante Positionsanzahl = 20	-	-	-
Zufällig genommene paarweise Präferenzen			
20,00%	0.8617±0.0133 0.2119±0.0264 0.01107	0.6430±0.0496 0.8584±0.1956 0.07971	0.9169±0.0345 0.1134±0.0456 0.00582
40,00%	0.8838±0.0155 0.1833±0.0240 0.01107	0.6696±0.0340 0.7408±0.1515 0.07971	0.9498±0.0177 0.0675±0.0240 0.00582
60,00%	0.8964±0.0091 0.1546±0.0200 0.01107	0.7068±0.0265 0.5784±0.0975 0.07971	0.9485±0.0177 0.0693±0.0198 0.00582
80,00%	0.9101±0.0116 0.1416±0.0271 0.01107	0.7161±0.0407 0.5613±0.0995 0.07971	0.9498±0.0101 0.0680±0.0137 0.00582
20% relevanter Positionsanzahl = 1	0.8658±0.0132 0.2247±0.0314 0.01107	0.6163±0.0348 0.9095±0.1974 0.07971	0.9190±0.0206 0.1134±0.0270 0.00582
20% relevanter Positionsanzahl = 2	0.8769±0.0077 0.1932±0.0199 0.01107	0.6376±0.0275 0.8251±0.1967 0.07971	0.9195±0.0298 0.1074±0.0377 0.00582
20% relevanter Positionsanzahl = 3	0.8596±0.0161 0.2173±0.0357 0.01107	0.6477±0.0589 0.7819±0.1855 0.07971	0.9264±0.0293 0.0965±0.0425 0.00582
20% relevanter Positionsanzahl = 4	0.8714±0.0122 0.2020±0.0281 0.01107	0.6184±0.0351 0.8285±0.2021 0.07971	0.9273±0.0219 0.0965±0.0254 0.00582
20% relevanter Positionsanzahl = 6	0.8705±0.0155 0.2039±0.0281 0.01107	0.6383±0.0299 0.7985±0.1549 0.07971	0.9221±0.0222 0.0991±0.0287 0.00582
20% relevanter Positionsanzahl = 8	0.8616±0.0131 0.2174±0.0233 0.01107	0.6516±0.0525 0.7713±0.2003 0.07971	-
20% relevanter Positionsanzahl = 12	0.8676±0.0138 0.2032±0.0214 0.01107	0.6536±0.0429 0.8351±0.1749 0.07971	-
20% relevanter Positionsanzahl = 16	-	-	-
20% relevanter Positionsanzahl = 20	-	-	-
40% relevanter Positionsanzahl = 1	0.9066±0.0102 0.1436±0.0181 0.01107	0.6962±0.0371 0.6157±0.1556 0.07971	0.9342±0.0165 0.0848±0.0303 0.00582
40% relevanter Positionsanzahl = 2	0.9055±0.0113 0.1520±0.0244 0.01107	0.6995±0.0423 0.6104±0.1650 0.07971	0.9468±0.0169 0.0710±0.0185 0.00582
40% relevanter Positionsanzahl = 3	0.8982±0.0173 0.1578±0.0271 0.01107	0.6968±0.0402 0.6463±0.1561 0.07971	0.9437±0.0214 0.0736±0.0264 0.00582
40% relevanter Positionsanzahl = 4	0.8938±0.0115 0.1610±0.0207 0.01107	0.6722±0.0364 0.6749±0.1261 0.07971	0.9476±0.0191 0.0662±0.0204 0.00582

40% relevanter Positionsanzahl = 6	0.8920±0.0128 0.1721±0.0255 0.01107	0.6390±0.0507 0.7819±0.1326 0.07971	0.9476±0.0132 0.0701±0.0166 0.00582
40% relevanter Positionsanzahl = 8	0.8911±0.0145 0.1649±0.0284 0.01107	0.6789±0.0340 0.6810±0.1625 0.07971	-
40% relevanter Positionsanzahl = 12	0.8947±0.0215 0.1639±0.0384 0.01107	0.6663±0.0304 0.7394±0.1738 0.07971	-
40% relevanter Positionsanzahl = 16	-	-	-
40% relevanter Positionsanzahl = 20	-	-	-
60% relevanter Positionsanzahl = 1	0.9208±0.0088 0.1260±0.0211 0.01107	0.7314±0.0374 0.5326±0.0717 0.07971	0.9459±0.0160 0.0706±0.0223 0.00582
60% relevanter Positionsanzahl = 2	0.9077±0.0143 0.1383±0.0246 0.01107	0.6975±0.0281 0.5679±0.1136 0.07971	0.9476±0.0176 0.0706±0.0210 0.00582
60% relevanter Positionsanzahl = 3	0.9085±0.0142 0.1420±0.0237 0.01107	0.7187±0.0490 0.5898±0.1340 0.07971	0.9463±0.0174 0.0732±0.0245 0.00582
60% relevanter Positionsanzahl = 4	0.9025±0.0128 0.1582±0.0215 0.01107	0.7127±0.0326 0.5712±0.1143 0.07971	0.9494±0.0084 0.0688±0.0135 0.00582
60% relevanter Positionsanzahl = 6	0.9041±0.0122 0.1502±0.0257 0.01107	0.6848±0.0475 0.6317±0.0947 0.07971	0.9476±0.0233 0.0710±0.0342 0.00582
60% relevanter Positionsanzahl = 8	0.8961±0.0146 0.1593±0.0226 0.01107	0.7088±0.0329 0.5924±0.1094 0.07971	-
60% relevanter Positionsanzahl = 12	0.8943±0.0100 0.1680±0.0200 0.01107	0.6982±0.0310 0.6010±0.0992 0.07971	-
60% relevanter Positionsanzahl = 16	-	-	-
60% relevanter Positionsanzahl = 20	-	-	-
80% relevanter Positionsanzahl = 1	0.9367±0.0111 0.0938±0.0183 0.01107	0.7261±0.0333 0.5306±0.0864 0.07971	0.9567±0.0104 0.0567±0.0190 0.00582
80% relevanter Positionsanzahl = 2	0.9190±0.0083 0.1308±0.0156 0.01107	0.7247±0.0383 0.5466±0.0943 0.07971	0.9515±0.0175 0.0654±0.0254 0.00582
80% relevanter Positionsanzahl = 3	0.9110±0.0088 0.1388±0.0189 0.01107	0.7228±0.0308 0.5559±0.0949 0.07971	0.9524±0.0145 0.0658±0.0170 0.00582
80% relevanter Positionsanzahl = 4	0.9034±0.0099 0.1464±0.0235 0.01107	0.7148±0.0295 0.5606±0.1152 0.07971	0.9532±0.0172 0.0610±0.0221 0.00582
80% relevanter Positionsanzahl = 6	0.9020±0.0116 0.1516±0.0203 0.01107	0.7128±0.0389 0.5773±0.1229 0.07971	0.9498±0.0139 0.0675±0.0163 0.00582
80% relevanter Positionsanzahl = 8	0.9107±0.0121 0.1383±0.0229 0.01107	0.7227±0.0387 0.5685±0.0975 0.07971	-
80% relevanter Positionsanzahl = 12	0.9064±0.0149 0.1384±0.0242 0.01107	0.6941±0.0322 0.6091±0.1002 0.07971	-
80% relevanter Positionsanzahl = 16	-	-	-
80% relevanter Positionsanzahl = 20	-	-	-
Prozentsatz der relevanten Positionsanzahl ab der zweiten Position			
20% relevanter Positionsanzahl >= 2			
iterative Schritte = 1	0.9418±0.0087 0.0895±0.0170 0.01107	0.7626±0.0348 0.4362±0.0718 0.07971	0.9693±0.0161 0.0416±0.0190 0.00582
iterative Schritte = 2	0.9427±0.0083 0.0899±0.0173 0.01074	0.7573±0.0262 0.4496±0.0660 0.08370	0.9688±0.0150 0.0411±0.0204 0.00615
iterative Schritte = 3	0.9384±0.0111 0.0940±0.0224 0.01154	0.7653±0.0247 0.4343±0.0788 0.08407	0.9688±0.0106 0.0403±0.0136 0.00610
40% relevanter Positionsanzahl >= 2			
iterative Schritte = 1	0.9407±0.0084 0.0895±0.0150 0.01107	0.7606±0.0233 0.4429±0.0842 0.07971	0.9654±0.0124 0.0450±0.0168 0.00582
iterative Schritte = 2	0.9358±0.0130 0.0984±0.0240 0.01175	0.7633±0.0263 0.4549±0.0815 0.08344	0.9662±0.0163 0.0446±0.0215 0.00598
iterative Schritte = 3	0.9331±0.0096 0.0979±0.0175 0.01206	0.7659±0.0311 0.4555±0.0956 0.08520	0.9658±0.0154 0.0442±0.0219 0.00636
60% relevanter Positionsanzahl >= 2			
iterative Schritte = 1	0.9342±0.0099 0.0973±0.0141 0.01107	0.7513±0.0280 0.4635±0.0833 0.07971	0.9701±0.0162 0.0403±0.0210 0.00582
iterative Schritte = 2	0.9333±0.0065 0.1011±0.0159 0.01206	0.7553±0.0263 0.4549±0.1021 0.08529	0.9675±0.0171 0.0429±0.0219 0.00589
iterative Schritte = 3	0.9326±0.0068 0.1075±0.0151 0.01240	0.7527±0.0415 0.4774±0.1013 0.08852	0.9671±0.0145 0.0420±0.0183 0.00683
80% relevanter Positionsanzahl >= 2			
iterative Schritte = 1	0.9365±0.0119 0.0970±0.0226 0.01107	0.7553±0.0314 0.4694±0.1000 0.07971	0.9662±0.0194 0.0433±0.0213 0.00582
iterative Schritte = 2	0.9338±0.0087 0.1011±0.0168 0.01249	0.7553±0.0309 0.4821±0.1003 0.08569	0.9680±0.0151 0.0416±0.0210 0.00651
iterative Schritte = 3	0.9304±0.0090 0.1066±0.0152 0.01337	0.7447±0.0324 0.4681±0.0900 0.08931	0.9693±0.0143 0.0420±0.0187 0.00639
Ranking ignorieren bei falscher Vorhersage erster Position			
relevante Positionsanzahl = 2			
iterative Schritte = 1	0.9310±0.0112 0.1087±0.0224 0.01107	0.7500±0.0431 0.4648±0.0820 0.07971	0.9680±0.0143 0.0407±0.0196 0.00582
iterative Schritte = 2	0.9310±0.0089 0.1110±0.0183 0.01176	0.7493±0.0381 0.4714±0.0812 0.08379	0.9675±0.0180 0.0429±0.0231 0.00680
iterative Schritte = 3	0.9310±0.0121 0.1100±0.0187 0.01210	0.7394±0.0297 0.4828±0.0807 0.09037	0.9675±0.0193 0.0416±0.0235 0.00718
relevante Positionsanzahl = 3			
iterative Schritte = 1	0.9286±0.0061 0.1100±0.0123 0.01107	0.7420±0.0378 0.4947±0.0813 0.07971	0.9632±0.0137 0.0494±0.0165 0.00582
iterative Schritte = 2	0.9281±0.0089 0.1139±0.0196 0.01246	0.7413±0.0407 0.4963±0.0802 0.08344	0.9649±0.0140 0.0485±0.0169 0.00632
iterative Schritte = 3	0.9278±0.0089 0.1146±0.0187 0.01284	0.7400±0.0343 0.4988±0.0883 0.08599	0.9662±0.0131 0.0463±0.0142 0.00670
relevante Positionsanzahl = 4			
iterative Schritte = 1	0.9251±0.0098 0.1101±0.0168 0.01107	0.7434±0.0384 0.4886±0.0974 0.07971	0.9623±0.0193 0.0506±0.0243 0.00582
iterative Schritte = 2	0.9272±0.0089 0.1110±0.0171 0.01218	0.7527±0.0319 0.4927±0.0737 0.08340	0.9593±0.0158 0.0554±0.0184 0.00637
iterative Schritte = 3	0.9265±0.0105 0.1112±0.0183 0.01246	0.7507±0.0360 0.4914±0.0828 0.08478	0.9593±0.0133 0.0541±0.0160 0.00662
relevante Positionsanzahl = 6			
iterative Schritte = 1	0.9238±0.0110 0.1174±0.0205 0.01107	0.7440±0.0262 0.4914±0.0801 0.07971	0.9597±0.0191 0.0528±0.0201 0.00582
iterative Schritte = 2	0.9263±0.0096 0.1162±0.0199 0.01181	0.7487±0.0235 0.5073±0.0840 0.08266	0.9576±0.0178 0.0524±0.0181 0.00658
iterative Schritte = 3	0.9256±0.0119 0.1198±0.0229 0.01211	0.7473±0.0230 0.5087±0.0689 0.08397	0.9597±0.0145 0.0515±0.0166 0.00700
relevante Positionsanzahl = 8			
iterative Schritte = 1	0.9222±0.0061 0.1210±0.0148 0.01107	0.7387±0.0312 0.5040±0.0927 0.07971	-
iterative Schritte = 2	0.9205±0.0134 0.1247±0.0229 0.01179	0.7514±0.0293 0.4934±0.0780 0.08189	-
iterative Schritte = 3	0.9176±0.0086 0.1263±0.0175 0.01186	0.7554±0.0280 0.4980±0.0960 0.08335	-
relevante Positionsanzahl = 12			
iterative Schritte = 1	0.9210±0.0090 0.1208±0.0187 0.01107	0.7401±0.0312 0.5027±0.0896 0.07971	-
iterative Schritte = 2	0.9231±0.0081 0.1219±0.0231 0.01158	0.7514±0.0302 0.4966±0.0912 0.08174	-
iterative Schritte = 3	0.9226±0.0063 0.1214±0.0165 0.01161	0.7460±0.0375 0.5133±0.0918 0.08360	-
relevante Positionsanzahl = 16			
iterative Schritte = 1	-	-	-
iterative Schritte = 2	-	-	-
iterative Schritte = 3	-	-	-
relevante Positionsanzahl = 20			
iterative Schritte = 1	-	-	-
iterative Schritte = 2	-	-	-
iterative Schritte = 3	-	-	-

Abbildung 22: J48: optdigits, re0.wc, segment

	tr11_wc.earff	tr12_wc.earff	tr21_wc.earff
originales LPCforSOS	0.8066±0.0438 0.3535±0.1770	0.7738±0.0794 0.3023±0.1548	0.8421±0.0628 0.1818±0.0704
Ursprüngliches Label auf beliebiger Position im Ranking			
relevante Positionsanzahl = 1	0.7993±0.0694 0.3972±0.2272	0.7925±0.0771 0.3409±0.1946	0.8539±0.0535 0.2024±0.0595
relevante Positionsanzahl = 2	0.7487±0.0535 0.4525±0.1816	0.7988±0.0667 0.2900±0.1656	0.8600±0.0428 0.1759±0.0581
relevante Positionsanzahl = 3	0.7775±0.0624 0.4456±0.1984	0.8021±0.0945 0.2864±0.1936	0.8569±0.0445 0.1881±0.0638
relevante Positionsanzahl = 4	0.7752±0.0558 0.4283±0.1693	0.7735±0.0668 0.3181±0.1637	0.8599±0.0449 0.1850±0.0660
relevante Positionsanzahl = 6	0.7559±0.0238 0.4692±0.1618	0.7703±0.0626 0.3406±0.1673	0.8599±0.0449 0.1938±0.0709
relevante Positionsanzahl = 8	0.7535±0.0293 0.4477±0.1649	0.7703±0.0626 0.3406±0.1673	-
relevante Positionsanzahl = 12	-	-	-
relevante Positionsanzahl = 16	-	-	-
relevante Positionsanzahl = 20	-	-	-
Zufällig genommene paarweise Präferenzen			
20,00%	0.6089±0.0816 0.7473±0.1894	0.5934±0.1220 0.6266±0.2159	0.7881±0.0742 0.2955±0.1004
40,00%	0.7028±0.0679 0.5682±0.2334	0.7037±0.1259 0.4456±0.2324	0.7911±0.0646 0.3404±0.1002
60,00%	0.7078±0.0386 0.5148±0.1476	0.7451±0.1092 0.3783±0.2230	0.8092±0.0495 0.2652±0.0729
80,00%	0.6980±0.0590 0.5274±0.1979	0.7609±0.1076 0.3728±0.1969	0.7852±0.0667 0.3072±0.0841
20% relevanter Positionsanzahl = 1	0.5962±0.0903 1.1570±0.3910	0.5593±0.0960 1.0734±0.3923	0.7348±0.0730 0.4860±0.1805
20% relevanter Positionsanzahl = 2	0.6932±0.0613 0.6530±0.1828	0.5393±0.0913 0.9074±0.2782	0.6994±0.0770 0.5209±0.1792
20% relevanter Positionsanzahl = 3	0.6424±0.0635 0.6823±0.2126	0.4917±0.0958 1.0000±0.2494	0.7468±0.0818 0.3810±0.1252
20% relevanter Positionsanzahl = 4	0.6394±0.0819 0.7480±0.2672	0.6161±0.0834 0.6875±0.2023	0.6746±0.0862 0.4901±0.1746
20% relevanter Positionsanzahl = 6	0.6377±0.0735 0.6643±0.1929	0.6229±0.1369 0.7118±0.2941	0.7466±0.0801 0.3550±0.1444
20% relevanter Positionsanzahl = 8	0.6692±0.0513 0.6623±0.1397	0.6294±0.1237 0.6403±0.2858	-
20% relevanter Positionsanzahl = 12	-	-	-
20% relevanter Positionsanzahl = 16	-	-	-
20% relevanter Positionsanzahl = 20	-	-	-
40% relevanter Positionsanzahl = 1	0.7051±0.0789 0.7578±0.2915	0.6485±0.0974 0.6770±0.2467	0.7824±0.0796 0.3818±0.1763
40% relevanter Positionsanzahl = 2	0.7220±0.0398 0.5396±0.1735	0.7382±0.1099 0.4562±0.1970	0.7885±0.0599 0.3336±0.0958
40% relevanter Positionsanzahl = 3	0.7316±0.0493 0.5635±0.1821	0.7254±0.0778 0.4401±0.1829	0.7528±0.0810 0.3273±0.1426
40% relevanter Positionsanzahl = 4	0.7075±0.0377 0.5899±0.1644	0.7288±0.1006 0.4524±0.2049	0.7531±0.0764 0.3508±0.1348
40% relevanter Positionsanzahl = 6	0.7079±0.0649 0.5607±0.1829	0.7380±0.0462 0.4364±0.1688	0.7971±0.0535 0.2684±0.0974
40% relevanter Positionsanzahl = 8	0.7028±0.0269 0.5735±0.1738	0.7291±0.0874 0.4428±0.1579	-
40% relevanter Positionsanzahl = 12	-	-	-
40% relevanter Positionsanzahl = 16	-	-	-
40% relevanter Positionsanzahl = 20	-	-	-
60% relevanter Positionsanzahl = 1	0.7242±0.0720 0.6614±0.2699	0.7161±0.0874 0.4430±0.1658	0.8063±0.0700 0.3100±0.1689
60% relevanter Positionsanzahl = 2	0.7293±0.0648 0.5254±0.1793	0.7671±0.0698 0.3537±0.1759	0.7612±0.0608 0.3131±0.0926
60% relevanter Positionsanzahl = 3	0.7126±0.0624 0.5681±0.2511	0.7480±0.1136 0.4243±0.2754	0.8061±0.0633 0.2565±0.0903
60% relevanter Positionsanzahl = 4	0.7441±0.0777 0.5153±0.1882	0.7571±0.0689 0.3317±0.1372	0.7705±0.0535 0.3373±0.1026
60% relevanter Positionsanzahl = 6	0.7487±0.0416 0.4891±0.1720	0.7638±0.0691 0.3597±0.1540	0.7974±0.0499 0.2740±0.0636
60% relevanter Positionsanzahl = 8	0.7147±0.0574 0.5326±0.2270	0.7385±0.0968 0.4016±0.1515	-
60% relevanter Positionsanzahl = 12	-	-	-
60% relevanter Positionsanzahl = 16	-	-	-
60% relevanter Positionsanzahl = 20	-	-	-
80% relevanter Positionsanzahl = 1	0.7365±0.0408 0.5663±0.1661	0.7224±0.0895 0.4070±0.1952	0.8214±0.0458 0.2680±0.1163
80% relevanter Positionsanzahl = 2	0.7171±0.0599 0.5567±0.2106	0.7707±0.1080 0.3466±0.2108	0.8004±0.0661 0.2831±0.1194
80% relevanter Positionsanzahl = 3	0.7315±0.0692 0.5279±0.2170	0.7478±0.0666 0.3887±0.1433	0.8061±0.0859 0.2626±0.0991
80% relevanter Positionsanzahl = 4	0.7463±0.0716 0.4864±0.2027	0.7514±0.0927 0.3661±0.1605	0.7911±0.0854 0.2716±0.1054
80% relevanter Positionsanzahl = 6	0.6980±0.0559 0.5226±0.1607	0.7481±0.1048 0.3697±0.1924	0.7891±0.0871 0.3060±0.1292
80% relevanter Positionsanzahl = 8	0.7435±0.0600 0.4918±0.2219	0.7580±0.1033 0.4289±0.2631	-
80% relevanter Positionsanzahl = 12	-	-	-
80% relevanter Positionsanzahl = 16	-	-	-
80% relevanter Positionsanzahl = 20	-	-	-
Ursprüngliches Label auf erster Position im Ranking			
relevante Positionsanzahl = 1	0.8066±0.0438 0.3535±0.1770 0.02925	0.7738±0.0794 0.3023±0.1548 0.01952	0.8421±0.0628 0.1818±0.0704 0.021
relevante Positionsanzahl = 2	0.7487±0.0454 0.4380±0.1470 0.02925	0.7859±0.0777 0.2839±0.1481 0.01952	0.8807±0.0362 0.1520±0.0394 0.021
relevante Positionsanzahl = 3	0.7631±0.0624 0.4405±0.1873 0.02925	0.7958±0.0744 0.2770±0.1544 0.01952	0.8600±0.0449 0.1789±0.0505 0.021
relevante Positionsanzahl = 4	0.7633±0.0637 0.4522±0.1755 0.02925	0.7830±0.0923 0.3056±0.1805 0.01952	0.8600±0.0449 0.1819±0.0580 0.021
relevante Positionsanzahl = 6	0.7392±0.0424 0.4808±0.1708 0.02925	0.7830±0.0828 0.3185±0.1682 0.01952	0.8600±0.0449 0.1906±0.0681 0.021
relevante Positionsanzahl = 8	0.7416±0.0338 0.4523±0.1603 0.02925	0.7830±0.0828 0.3185±0.1682 0.01952	-
relevante Positionsanzahl = 12	-	-	-
relevante Positionsanzahl = 16	-	-	-
relevante Positionsanzahl = 20	-	-	-
Zufällig genommene paarweise Präferenzen			
20,00%	0.6399±0.0640 0.6894±0.1956 0.02925	0.6327±0.1589 0.6248±0.2945 0.01952	0.7763±0.0541 0.2980±0.0877 0.021
40,00%	0.6812±0.0427 0.5736±0.1640 0.02925	0.7228±0.1080 0.4431±0.1767 0.01952	0.7941±0.0491 0.2922±0.0774 0.021
60,00%	0.7243±0.0689 0.5470±0.1967 0.02925	0.7448±0.0963 0.3629±0.1691 0.01952	0.7709±0.0547 0.3187±0.0513 0.021
80,00%	0.7467±0.0626 0.5074±0.1594 0.02925	0.7574±0.0906 0.3767±0.1433 0.01952	0.8005±0.0621 0.2800±0.0960 0.021
20% relevanter Positionsanzahl = 1	0.5846±0.1097 1.2087±0.2375 0.02925	0.5171±0.1144 1.2019±0.3500 0.01952	0.7913±0.0497 0.4086±0.1545 0.021
20% relevanter Positionsanzahl = 2	0.6470±0.0663 0.7209±0.2093 0.02925	0.5239±0.0816 0.9204±0.2355 0.01952	0.7047±0.0846 0.5189±0.1992 0.021
20% relevanter Positionsanzahl = 3	0.6784±0.0724 0.6900±0.2548 0.02925	0.5460±0.0662 0.8315±0.1955 0.01952	0.7527±0.0703 0.3816±0.1220 0.021
20% relevanter Positionsanzahl = 4	0.6785±0.0562 0.6456±0.1310 0.02925	0.6864±0.0956 0.5250±0.1875 0.01952	0.7319±0.0844 0.3669±0.1239 0.021
20% relevanter Positionsanzahl = 6	0.6713±0.0447 0.6599±0.1465 0.02925	0.6358±0.1170 0.7154±0.2878 0.01952	0.7821±0.0791 0.3406±0.1751 0.021
20% relevanter Positionsanzahl = 8	0.6301±0.0843 0.6750±0.1774 0.02925	0.5684±0.1134 0.7537±0.2134 0.01952	-
20% relevanter Positionsanzahl = 12	-	-	-
20% relevanter Positionsanzahl = 16	-	-	-
20% relevanter Positionsanzahl = 20	-	-	-
40% relevanter Positionsanzahl = 1	0.7049±0.0827 0.6996±0.2640 0.02925	0.6935±0.1079 0.5941±0.1936 0.01952	0.7792±0.0949 0.4053±0.1474 0.021
40% relevanter Positionsanzahl = 2	0.7023±0.0829 0.5596±0.2163 0.02925	0.7062±0.0876 0.4498±0.1864 0.01952	0.7467±0.0777 0.3577±0.1226 0.021
40% relevanter Positionsanzahl = 3	0.7195±0.0724 0.5788±0.2550 0.02925	0.7225±0.0915 0.4680±0.2192 0.01952	0.8059±0.0620 0.2956±0.1021 0.021
40% relevanter Positionsanzahl = 4	0.6977±0.0684 0.5571±0.2012 0.02925	0.7281±0.1026 0.4379±0.1816 0.01952	0.7971±0.0638 0.3012±0.1071 0.021

40% relevanter Positionsanzahl = 6	0.6931±0.0804 0.5704±0.1835 0.02925	0.7131±0.1007 0.4743±0.2175 0.01952	0.7856±0.0658 0.3068±0.1313 0.021
40% relevanter Positionsanzahl = 8	0.6959±0.0569 0.5560±0.1092 0.02925	0.7382±0.0573 0.4470±0.1291 0.01952	-
40% relevanter Positionsanzahl = 12	-	-	-
40% relevanter Positionsanzahl = 16	-	-	-
40% relevanter Positionsanzahl = 20	-	-	-
60% relevanter Positionsanzahl = 1	0.7171±0.0647 0.6001±0.2117 0.02925	0.7542±0.0776 0.4172±0.1841 0.01952	0.7940±0.0580 0.2956±0.1161 0.021
60% relevanter Positionsanzahl = 2	0.7246±0.0554 0.5587±0.2102 0.02925	0.7763±0.1280 0.3536±0.2255 0.01952	0.7914±0.0686 0.3102±0.1167 0.021
60% relevanter Positionsanzahl = 3	0.7243±0.0468 0.5034±0.1602 0.02925	0.7546±0.0836 0.3982±0.1674 0.01952	0.8040±0.0881 0.2734±0.1005 0.021
60% relevanter Positionsanzahl = 4	0.7024±0.0802 0.5494±0.1958 0.02925	0.7735±0.0757 0.3441±0.1811 0.01952	0.7914±0.0340 0.2923±0.0961 0.021
60% relevanter Positionsanzahl = 6	0.7294±0.0463 0.5247±0.1400 0.02925	0.7381±0.0611 0.3984±0.1382 0.01952	0.7971±0.0769 0.2807±0.0965 0.021
60% relevanter Positionsanzahl = 8	0.7679±0.0436 0.4772±0.1709 0.02925	0.7223±0.0786 0.3955±0.1559 0.01952	-
60% relevanter Positionsanzahl = 12	-	-	-
60% relevanter Positionsanzahl = 16	-	-	-
60% relevanter Positionsanzahl = 20	-	-	-
80% relevanter Positionsanzahl = 1	0.7412±0.0502 0.5521±0.2140 0.02925	0.7573±0.0832 0.3702±0.1395 0.01952	0.8095±0.0401 0.2864±0.1092 0.021
80% relevanter Positionsanzahl = 2	0.7150±0.0631 0.5393±0.1412 0.02925	0.7796±0.0682 0.3223±0.1268 0.01952	0.8094±0.0506 0.2440±0.0865 0.021
80% relevanter Positionsanzahl = 3	0.7219±0.0616 0.5210±0.2089 0.02925	0.7830±0.0676 0.3637±0.1500 0.01952	0.7765±0.0463 0.3041±0.0578 0.021
80% relevanter Positionsanzahl = 4	0.7249±0.0543 0.5179±0.1751 0.02925	0.7446±0.0545 0.3796±0.1248 0.01952	0.8362±0.0472 0.2294±0.0782 0.021
80% relevanter Positionsanzahl = 6	0.7293±0.0396 0.4865±0.1566 0.02925	0.7358±0.0960 0.3976±0.1762 0.01952	0.7585±0.0635 0.3307±0.0771 0.021
80% relevanter Positionsanzahl = 8	0.7466±0.0596 0.4858±0.1417 0.02925	0.7867±0.0774 0.3306±0.1721 0.01952	-
80% relevanter Positionsanzahl = 12	-	-	-
80% relevanter Positionsanzahl = 16	-	-	-
80% relevanter Positionsanzahl = 20	-	-	-
Prozentsatz der relevanten Positionsanzahl ab der zweiten Position			
20% relevanter Positionsanzahl >= 2			
iterative Schritte = 1	0.7992±0.0507 0.3585±0.1821 0.02925	0.8056±0.0626 0.2800±0.1500 0.01952	0.8421±0.0628 0.1818±0.0704 0.021
iterative Schritte = 2	0.7969±0.0530 0.3657±0.1808 0.03005	0.8022±0.0708 0.2867±0.1439 0.02006	0.8360±0.0676 0.1879±0.0800 0.021
iterative Schritte = 3	0.8042±0.0402 0.3631±0.1631 0.03050	0.7769±0.0792 0.3184±0.1580 0.02188	0.8360±0.0676 0.1879±0.0800 0.021
40% relevanter Positionsanzahl >= 2			
iterative Schritte = 1	0.7968±0.0474 0.3705±0.1837 0.02925	0.7928±0.0608 0.2768±0.1331 0.01952	0.8483±0.0548 0.1874±0.0721 0.021
iterative Schritte = 2	0.7993±0.0516 0.3729±0.1669 0.03019	0.8183±0.0827 0.2384±0.1454 0.02005	0.8808±0.0443 0.1520±0.0622 0.023
iterative Schritte = 3	0.7945±0.0540 0.3729±0.1771 0.02907	0.7955±0.0837 0.2969±0.1633 0.02556	0.8574±0.0664 0.1695±0.0732 0.022
60% relevanter Positionsanzahl >= 2			
iterative Schritte = 1	0.7847±0.0599 0.3899±0.1748 0.02925	0.7924±0.0531 0.3029±0.1305 0.01952	0.8688±0.0540 0.1640±0.0631 0.021
iterative Schritte = 2	0.7872±0.0580 0.3969±0.1656 0.03032	0.8020±0.0928 0.3127±0.1598 0.02361	0.8777±0.0475 0.1522±0.0597 0.023
iterative Schritte = 3	0.8066±0.0475 0.3510±0.1670 0.03014	0.8051±0.0735 0.2838±0.1453 0.02283	0.8597±0.0762 0.1673±0.0848 0.024
80% relevanter Positionsanzahl >= 2			
iterative Schritte = 1	0.8042±0.0455 0.3656±0.1726 0.02925	0.7991±0.0686 0.2835±0.1499 0.01952	0.8750±0.0541 0.1696±0.0875 0.021
iterative Schritte = 2	0.7969±0.0504 0.3777±0.1836 0.02952	0.8084±0.0637 0.2802±0.1491 0.02254	0.8750±0.0711 0.1578±0.0784 0.023
iterative Schritte = 3	0.7920±0.0593 0.3899±0.1910 0.03050	0.7989±0.0727 0.3029±0.1241 0.02354	0.8570±0.0372 0.1935±0.0869 0.024
Ranking ignorieren bei falscher Vorhersage erster Position			
relevante Positionsanzahl = 2			
iterative Schritte = 1	0.7486±0.0539 0.4380±0.1634 0.02979	0.7957±0.0799 0.2708±0.1493 0.01952	0.8748±0.0354 0.1639±0.0487 0.021
iterative Schritte = 2	0.7437±0.0629 0.4598±0.2011 0.03945	0.7990±0.0880 0.2771±0.1638 0.02556	0.8807±0.0383 0.1701±0.0592 0.023
iterative Schritte = 3	0.7509±0.0551 0.4552±0.1810 0.04169	0.8052±0.0854 0.2676±0.1598 0.02698	0.8747±0.0403 0.1761±0.0600 0.023
relevante Positionsanzahl = 3			
iterative Schritte = 1	0.7413±0.0610 0.4768±0.1909 0.02979	0.8053±0.0732 0.2739±0.1627 0.01952	0.8629±0.0429 0.1759±0.0546 0.021
iterative Schritte = 2	0.7509±0.0498 0.4624±0.1718 0.03703	0.8084±0.0995 0.2741±0.1944 0.02378	0.8749±0.0497 0.1610±0.0559 0.025
iterative Schritte = 3	0.7387±0.0635 0.4795±0.1809 0.03927	0.8117±0.1019 0.2740±0.1964 0.02437	0.8599±0.0449 0.1789±0.0487 0.025
relevante Positionsanzahl = 4			
iterative Schritte = 1	0.7510±0.0426 0.4672±0.1360 0.02979	0.7993±0.0752 0.2765±0.1597 0.01952	0.8659±0.0451 0.1760±0.0610 0.021
iterative Schritte = 2	0.7463±0.0576 0.4721±0.1709 0.03609	0.7895±0.0842 0.2959±0.1804 0.02414	0.8808±0.0498 0.1610±0.0645 0.025
iterative Schritte = 3	0.7462±0.0653 0.4382±0.1646 0.03623	0.7926±0.0831 0.3121±0.1960 0.02496	0.8659±0.0451 0.1759±0.0530 0.025
relevante Positionsanzahl = 6			
iterative Schritte = 1	0.7319±0.0365 0.4595±0.1417 0.02979	0.7863±0.0816 0.3088±0.1612 0.01952	0.8599±0.0449 0.1760±0.0597 0.021
iterative Schritte = 2	0.7318±0.0425 0.4546±0.1526 0.03784	0.7864±0.0797 0.2959±0.1775 0.02484	0.8749±0.0497 0.1668±0.0696 0.025
iterative Schritte = 3	0.7270±0.0448 0.4760±0.1407 0.03963	0.7799±0.0866 0.3183±0.1812 0.02532	0.8599±0.0449 0.1789±0.0538 0.025
relevante Positionsanzahl = 8			
iterative Schritte = 1	0.7392±0.0273 0.4304±0.1338 0.02979	0.7863±0.0816 0.3088±0.1612 0.01952	-
iterative Schritte = 2	0.7343±0.0384 0.4426±0.1615 0.03784	0.7864±0.0797 0.2959±0.1775 0.02484	-
iterative Schritte = 3	0.7365±0.0456 0.4597±0.1762 0.03856	0.7799±0.0866 0.3183±0.1812 0.02532	-
relevante Positionsanzahl = 12			
iterative Schritte = 1	-	-	-
iterative Schritte = 2	-	-	-
iterative Schritte = 3	-	-	-
relevante Positionsanzahl = 16			
iterative Schritte = 1	-	-	-
iterative Schritte = 2	-	-	-
iterative Schritte = 3	-	-	-
relevante Positionsanzahl = 20			
iterative Schritte = 1	-	-	-
iterative Schritte = 2	-	-	-
iterative Schritte = 3	-	-	-

Abbildung 23: J48: tr11.wc, tr12.wc, tr21.wc

	tr23_wc.earff	tr31_wc.earff	tr41_wc.earff
originales LPCforSOS	0.8833±0.0774 0.2036±0.1310	0.9417±0.0385 0.0982±0.0792	0.9010±0.0237 0.2038±0.0688
Ursprüngliches Label auf beliebiger Position im Ranking			
relevante Positionsanzahl = 1	0.8633±0.0921 0.2138±0.1272	0.9342±0.0391 0.1187±0.0837	0.8885±0.0276 0.2424±0.0750
relevante Positionsanzahl = 2	0.8440±0.0947 0.2669±0.1770	0.9342±0.0390 0.1165±0.0802	0.8793±0.0211 0.2561±0.0633
relevante Positionsanzahl = 3	0.8388±0.0914 0.2579±0.1558	0.9288±0.0306 0.1230±0.0695	0.8804±0.0230 0.2528±0.0652
relevante Positionsanzahl = 4	0.8438±0.0877 0.2824±0.1588	0.9331±0.0308 0.1294±0.0732	0.8827±0.0205 0.2518±0.0741
relevante Positionsanzahl = 6	0.8490±0.0964 0.2576±0.1543	0.9342±0.0291 0.1241±0.0750	0.8815±0.0250 0.2495±0.0782
relevante Positionsanzahl = 8	-	-	0.8838±0.0278 0.2460±0.0771
relevante Positionsanzahl = 12	-	-	0.8815±0.0245 0.2506±0.0710
relevante Positionsanzahl = 16	-	-	-
relevante Positionsanzahl = 20	-	-	-
Zufällig genommene paarweise Präferenzen			
20,00%	0.6905±0.1037 0.6264±0.1996	0.8316±0.0628 0.3000±0.1334	0.7985±0.0358 0.4837±0.1165
40,00%	0.7355±0.1202 0.4543±0.2543	0.8760±0.0385 0.2233±0.0933	0.8315±0.0258 0.3552±0.0786
60,00%	0.7750±0.0843 0.3814±0.1739	0.8792±0.0461 0.2113±0.1046	0.8394±0.0320 0.3620±0.0896
80,00%	0.7502±0.0863 0.3948±0.2018	0.8932±0.0389 0.1887±0.0931	0.8644±0.0259 0.2778±0.0873
20% relevanter Positionsanzahl = 1	0.6967±0.0737 0.6690±0.1762	0.8468±0.0462 0.2729±0.1142	0.7187±0.0481 0.7772±0.2334
20% relevanter Positionsanzahl = 2	0.5638±0.0980 0.9312±0.3113	0.8478±0.0522 0.2730±0.1177	0.7961±0.0419 0.5202±0.1311
20% relevanter Positionsanzahl = 3	0.6429±0.0913 0.7605±0.2853	0.8360±0.0453 0.2805±0.1019	0.7848±0.0317 0.5622±0.1377
20% relevanter Positionsanzahl = 4	0.5648±0.1611 0.8621±0.1777	0.8036±0.0471 0.3528±0.1418	0.8007±0.0387 0.4474±0.0872
20% relevanter Positionsanzahl = 6	0.6369±0.0835 0.7148±0.1870	0.8587±0.0579 0.2437±0.1223	0.7814±0.0322 0.5041±0.1256
20% relevanter Positionsanzahl = 8	-	-	0.7849±0.0532 0.5291±0.1746
20% relevanter Positionsanzahl = 12	-	-	0.7551±0.0499 0.4953±0.1115
20% relevanter Positionsanzahl = 16	-	-	-
20% relevanter Positionsanzahl = 20	-	-	-
40% relevanter Positionsanzahl = 1	0.7595±0.0819 0.5333±0.2003	0.8835±0.0433 0.2103±0.0976	0.8475±0.0420 0.3493±0.1079
40% relevanter Positionsanzahl = 2	0.7307±0.1098 0.4983±0.2416	0.8791±0.0508 0.2071±0.1148	0.8212±0.0355 0.3699±0.0994
40% relevanter Positionsanzahl = 3	0.7107±0.0744 0.5724±0.1927	0.9018±0.0442 0.1845±0.0925	0.8394±0.0357 0.3563±0.1047
40% relevanter Positionsanzahl = 4	0.6814±0.0745 0.5769±0.1573	0.8587±0.0595 0.2437±0.1192	0.8485±0.0221 0.3119±0.0941
40% relevanter Positionsanzahl = 6	0.7107±0.0342 0.5131±0.1628	0.8781±0.0453 0.2190±0.0875	0.8303±0.0360 0.3597±0.1042
40% relevanter Positionsanzahl = 8	-	-	0.8554±0.0293 0.3449±0.0699
40% relevanter Positionsanzahl = 12	-	-	0.8235±0.0168 0.3949±0.0976
40% relevanter Positionsanzahl = 16	-	-	-
40% relevanter Positionsanzahl = 20	-	-	-
60% relevanter Positionsanzahl = 1	0.7695±0.1119 0.3707±0.2129	0.9126±0.0432 0.1564±0.0965	0.8873±0.0251 0.2777±0.1070
60% relevanter Positionsanzahl = 2	0.7648±0.0682 0.4350±0.0988	0.8792±0.0576 0.2113±0.1252	0.8542±0.0246 0.3050±0.0786
60% relevanter Positionsanzahl = 3	0.8098±0.0842 0.3657±0.1714	0.8813±0.0436 0.2007±0.0890	0.8531±0.0303 0.3108±0.0898
60% relevanter Positionsanzahl = 4	0.7850±0.1106 0.3607±0.2082	0.9051±0.0485 0.1715±0.1033	0.8519±0.0342 0.3234±0.0909
60% relevanter Positionsanzahl = 6	0.7850±0.0592 0.3902±0.1466	0.8900±0.0409 0.2039±0.1048	0.8451±0.0220 0.3438±0.0810
60% relevanter Positionsanzahl = 8	-	-	0.8428±0.0257 0.3245±0.0708
60% relevanter Positionsanzahl = 12	-	-	0.8486±0.0377 0.3311±0.0985
60% relevanter Positionsanzahl = 16	-	-	-
60% relevanter Positionsanzahl = 20	-	-	-
80% relevanter Positionsanzahl = 1	0.8529±0.0433 0.2650±0.1158	0.9158±0.0353 0.1490±0.0783	0.8850±0.0262 0.2527±0.0781
80% relevanter Positionsanzahl = 2	0.7752±0.1152 0.3805±0.2266	0.8964±0.0546 0.1822±0.1064	0.8610±0.0169 0.3086±0.0653
80% relevanter Positionsanzahl = 3	0.7552±0.0931 0.4450±0.2113	0.8932±0.0423 0.1856±0.1126	0.8474±0.0327 0.2892±0.0832
80% relevanter Positionsanzahl = 4	0.7748±0.0873 0.4105±0.1715	0.9007±0.0452 0.1758±0.1021	0.8656±0.0355 0.2835±0.1063
80% relevanter Positionsanzahl = 6	0.7605±0.0711 0.3843±0.1657	0.9126±0.0411 0.1575±0.0926	0.8554±0.0269 0.2983±0.0889
80% relevanter Positionsanzahl = 8	-	-	0.8701±0.0307 0.2676±0.0786
80% relevanter Positionsanzahl = 12	-	-	0.8679±0.0182 0.2789±0.0672
80% relevanter Positionsanzahl = 16	-	-	-
80% relevanter Positionsanzahl = 20	-	-	-
Ursprüngliches Label auf erster Position im Ranking			
relevante Positionsanzahl = 1	0.8833±0.0774 0.2036±0.1310 0.01143	0.9417±0.0385 0.0982±0.0792 0.01414	0.9010±0.0237 0.2038±0.0688 0.023
relevante Positionsanzahl = 2	0.8293±0.0885 0.2717±0.1719 0.01143	0.9353±0.0348 0.1154±0.0803 0.01414	0.8884±0.0286 0.2345±0.0763 0.023
relevante Positionsanzahl = 3	0.8440±0.0875 0.2967±0.1704 0.01143	0.9418±0.0345 0.1100±0.0760 0.01414	0.8781±0.0235 0.2449±0.0503 0.023
relevante Positionsanzahl = 4	0.8488±0.0753 0.2821±0.1640 0.01143	0.9385±0.0298 0.1230±0.0739 0.01414	0.8838±0.0239 0.2414±0.0621 0.023
relevante Positionsanzahl = 6	0.8443±0.0947 0.2574±0.1595 0.01143	0.9353±0.0337 0.1208±0.0774 0.01414	0.8883±0.0204 0.2357±0.0629 0.023
relevante Positionsanzahl = 8	-	-	0.8884±0.0227 0.2300±0.0595 0.023
relevante Positionsanzahl = 12	-	-	0.8873±0.0193 0.2357±0.0543 0.023
relevante Positionsanzahl = 16	-	-	-
relevante Positionsanzahl = 20	-	-	-
Zufällig genommene paarweise Präferenzen			
20,00%	0.6333±0.1268 0.6581±0.2444 0.01143	0.8263±0.0642 0.2977±0.1353 0.01414	0.7541±0.0496 0.5123±0.1405 0.023
40,00%	0.7402±0.0797 0.5036±0.1386 0.01143	0.8630±0.0415 0.2330±0.1033 0.01414	0.8247±0.0545 0.3881±0.1078 0.023
60,00%	0.7645±0.0663 0.4107±0.2026 0.01143	0.8932±0.0435 0.1932±0.1065 0.01414	0.8452±0.0194 0.3574±0.0767 0.023
80,00%	0.7655±0.0850 0.3610±0.1572 0.01143	0.9008±0.0476 0.1812±0.1013 0.01414	0.8543±0.0361 0.2982±0.0931 0.023
20% relevanter Positionsanzahl = 1	0.6538±0.1094 0.8000±0.2220 0.01143	0.8457±0.0451 0.2696±0.0995 0.01414	0.7403±0.0474 0.7501±0.1557 0.023
20% relevanter Positionsanzahl = 2	0.6529±0.1284 0.8010±0.2638 0.01143	0.8382±0.0615 0.2966±0.1272 0.01414	0.7619±0.0449 0.5522±0.1394 0.023
20% relevanter Positionsanzahl = 3	0.6655±0.0995 0.7055±0.1924 0.01143	0.8393±0.0430 0.2825±0.1041 0.01414	0.7608±0.0473 0.5350±0.1416 0.023
20% relevanter Positionsanzahl = 4	0.6426±0.1341 0.7486±0.2922 0.01143	0.8436±0.0523 0.2815±0.1384 0.01414	0.7928±0.0335 0.4280±0.1248 0.023
20% relevanter Positionsanzahl = 6	0.6171±0.1563 0.7490±0.2500 0.01143	0.8284±0.0612 0.2998±0.1332 0.01414	0.7586±0.0503 0.5088±0.1244 0.023
20% relevanter Positionsanzahl = 8	-	-	0.7597±0.0523 0.5611±0.1728 0.023
20% relevanter Positionsanzahl = 12	-	-	0.7482±0.0527 0.5146±0.1194 0.023
20% relevanter Positionsanzahl = 16	-	-	-
20% relevanter Positionsanzahl = 20	-	-	-
40% relevanter Positionsanzahl = 1	0.7505±0.1039 0.4690±0.1767 0.01143	0.8932±0.0431 0.1855±0.0922 0.01414	0.8144±0.0528 0.3858±0.1625 0.023
40% relevanter Positionsanzahl = 2	0.6662±0.0602 0.6321±0.1442 0.01143	0.8846±0.0549 0.2059±0.1047 0.01414	0.8349±0.0433 0.3277±0.0830 0.023
40% relevanter Positionsanzahl = 3	0.7052±0.0688 0.6036±0.1509 0.01143	0.8857±0.0506 0.2049±0.1160 0.01414	0.8223±0.0305 0.3837±0.0876 0.023
40% relevanter Positionsanzahl = 4	0.7310±0.0969 0.5317±0.2136 0.01143	0.8803±0.0348 0.2264±0.0857 0.01414	0.8463±0.0234 0.3199±0.0909 0.023

40% relevanter Positionsanzahl = 6	0.7448±0.1069 0.4788±0.1939 0.01143	0.8640±0.0510 0.2417±0.1130 0.01414	0.8281±0.0330 0.3575±0.1074 0.023
40% relevanter Positionsanzahl = 8	-	-	0.8223±0.0278 0.3813±0.0858 0.023
40% relevanter Positionsanzahl = 12	-	-	0.8155±0.0367 0.3939±0.0876 0.023
40% relevanter Positionsanzahl = 16	-	-	-
40% relevanter Positionsanzahl = 20	-	-	-
60% relevanter Positionsanzahl = 1	0.7700±0.0963 0.3898±0.2221 0.01143	0.9072±0.0534 0.1533±0.1065 0.01414	0.8576±0.0324 0.2811±0.0848 0.023
60% relevanter Positionsanzahl = 2	0.7700±0.0760 0.4152±0.1380 0.01143	0.8846±0.0532 0.1855±0.1187 0.01414	0.8587±0.0248 0.3040±0.0648 0.023
60% relevanter Positionsanzahl = 3	0.8093±0.0954 0.3560±0.1847 0.01143	0.8781±0.0388 0.2124±0.0972 0.01414	0.8383±0.0405 0.3188±0.0735 0.023
60% relevanter Positionsanzahl = 4	0.7314±0.0995 0.4679±0.1801 0.01143	0.8954±0.0385 0.1985±0.0967 0.01414	0.8599±0.0428 0.2823±0.1066 0.023
60% relevanter Positionsanzahl = 6	0.7660±0.1077 0.4093±0.1841 0.01143	0.9008±0.0550 0.1855±0.1146 0.01414	0.8554±0.0291 0.3199±0.0989 0.023
60% relevanter Positionsanzahl = 8	-	-	0.8497±0.0277 0.3392±0.0669 0.023
60% relevanter Positionsanzahl = 12	-	-	0.8406±0.0357 0.3289±0.0998 0.023
60% relevanter Positionsanzahl = 16	-	-	-
60% relevanter Positionsanzahl = 20	-	-	-
80% relevanter Positionsanzahl = 1	0.8481±0.0599 0.2979±0.1435 0.01143	0.9094±0.0295 0.1629±0.0828 0.01414	0.8736±0.0261 0.2425±0.0763 0.023
80% relevanter Positionsanzahl = 2	0.7986±0.0794 0.3333±0.1597 0.01143	0.9202±0.0472 0.1412±0.0960 0.01414	0.8405±0.0227 0.2972±0.0712 0.023
80% relevanter Positionsanzahl = 3	0.8088±0.0562 0.3521±0.1756 0.01143	0.9116±0.0443 0.1628±0.1007 0.01414	0.8621±0.0366 0.2858±0.0943 0.023
80% relevanter Positionsanzahl = 4	0.7555±0.1176 0.4002±0.1872 0.01143	0.9084±0.0553 0.1650±0.1148 0.01414	0.8622±0.0248 0.2915±0.0955 0.023
80% relevanter Positionsanzahl = 6	0.7307±0.1231 0.4052±0.1835 0.01143	0.9083±0.0329 0.1607±0.0818 0.01414	0.8554±0.0221 0.2879±0.0949 0.023
80% relevanter Positionsanzahl = 8	-	-	0.8508±0.0280 0.2824±0.0779 0.023
80% relevanter Positionsanzahl = 12	-	-	0.8473±0.0286 0.3086±0.0803 0.023
80% relevanter Positionsanzahl = 16	-	-	-
80% relevanter Positionsanzahl = 20	-	-	-
Prozentsatz der relevanten Positionsanzahl ab der zweiten Position			
20% relevanter Positionsanzahl >= 2			
iterative Schritte = 1	0.8783±0.0675 0.2286±0.1054 0.01143	0.9471±0.0254 0.0972±0.0616 0.01414	0.9021±0.0168 0.2084±0.0672 0.023
iterative Schritte = 2	0.8881±0.0639 0.2086±0.1160 0.01143	0.9471±0.0292 0.1004±0.0623 0.01384	0.8998±0.0267 0.2038±0.0728 0.024
iterative Schritte = 3	0.8838±0.0647 0.2126±0.1172 0.01361	0.9568±0.0278 0.0820±0.0579 0.01482	0.8953±0.0252 0.2174±0.0737 0.025
40% relevanter Positionsanzahl >= 2			
iterative Schritte = 1	0.8638±0.0839 0.2521±0.1631 0.01143	0.9471±0.0301 0.1079±0.0760 0.01414	0.8998±0.0246 0.2038±0.0688 0.023
iterative Schritte = 2	0.8395±0.1066 0.2714±0.1830 0.01116	0.9450±0.0326 0.1068±0.0805 0.01582	0.8987±0.0260 0.2060±0.0713 0.023
iterative Schritte = 3	0.8636±0.0880 0.2524±0.1599 0.01416	0.9417±0.0394 0.0982±0.0795 0.01398	0.8850±0.0225 0.2243±0.0626 0.027
60% relevanter Positionsanzahl >= 2			
iterative Schritte = 1	0.8393±0.0882 0.2812±0.1892 0.01143	0.9428±0.0345 0.1036±0.0716 0.01414	0.8918±0.0205 0.2300±0.0546 0.023
iterative Schritte = 2	0.8588±0.0866 0.2667±0.1696 0.01443	0.9406±0.0382 0.1025±0.0772 0.01396	0.8850±0.0286 0.2278±0.0689 0.027
iterative Schritte = 3	0.8833±0.0707 0.2133±0.1218 0.01270	0.9428±0.0391 0.0971±0.0800 0.01418	0.8987±0.0260 0.2038±0.0688 0.023
80% relevanter Positionsanzahl >= 2			
iterative Schritte = 1	0.8531±0.0803 0.2674±0.1735 0.01143	0.9396±0.0374 0.1058±0.0710 0.01414	0.8827±0.0163 0.2448±0.0589 0.023
iterative Schritte = 2	0.8881±0.0599 0.1988±0.1170 0.01197	0.9460±0.0294 0.0982±0.0694 0.01546	0.8827±0.0301 0.2334±0.0737 0.029
iterative Schritte = 3	0.8931±0.0666 0.1936±0.1313 0.01233	0.9396±0.0379 0.1003±0.0781 0.01394	0.8770±0.0253 0.2528±0.0745 0.031
Ranking ignorieren bei falscher Vorhersage erster Position			
relevante Positionsanzahl = 2			
iterative Schritte = 1	0.8198±0.1041 0.2762±0.1876 0.01143	0.9385±0.0348 0.1089±0.0764 0.01414	0.8884±0.0207 0.2288±0.0752 0.023
iterative Schritte = 2	0.8345±0.1074 0.2617±0.1853 0.01688	0.9385±0.0435 0.1079±0.0830 0.01456	0.8884±0.0225 0.2322±0.0694 0.028
iterative Schritte = 3	0.8198±0.1133 0.2812±0.1929 0.01743	0.9363±0.0441 0.1111±0.0834 0.01550	0.8884±0.0188 0.2345±0.0774 0.030
relevante Positionsanzahl = 3			
iterative Schritte = 1	0.8436±0.0767 0.2924±0.1437 0.01143	0.9353±0.0392 0.1187±0.0789 0.01414	0.8804±0.0241 0.2380±0.0549 0.023
iterative Schritte = 2	0.8438±0.0824 0.2924±0.1405 0.01388	0.9331±0.0416 0.1284±0.0871 0.01546	0.8747±0.0229 0.2471±0.0559 0.029
iterative Schritte = 3	0.8436±0.0672 0.2874±0.1319 0.01506	0.9417±0.0343 0.1166±0.0815 0.01594	0.8815±0.0217 0.2471±0.0501 0.031
relevante Positionsanzahl = 4			
iterative Schritte = 1	0.8331±0.0625 0.2831±0.1317 0.01143	0.9406±0.0314 0.1219±0.0775 0.01414	0.8884±0.0232 0.2300±0.0615 0.023
iterative Schritte = 2	0.8388±0.0739 0.3069±0.1441 0.01388	0.9385±0.0283 0.1155±0.0695 0.01648	0.8701±0.0193 0.2392±0.0671 0.028
iterative Schritte = 3	0.8483±0.0756 0.2976±0.1579 0.01525	0.9353±0.0251 0.1176±0.0653 0.01650	0.8815±0.0230 0.2278±0.0725 0.030
relevante Positionsanzahl = 6			
iterative Schritte = 1	0.8433±0.0601 0.2483±0.1186 0.01143	0.9353±0.0341 0.1187±0.0789 0.01414	0.8906±0.0173 0.2255±0.0597 0.023
iterative Schritte = 2	0.8436±0.0699 0.2826±0.1260 0.01416	0.9342±0.0300 0.1165±0.0740 0.01606	0.8838±0.0151 0.2471±0.0587 0.028
iterative Schritte = 3	0.8340±0.0805 0.3021±0.1430 0.01506	0.9277±0.0311 0.1273±0.0715 0.01614	0.8804±0.0187 0.2460±0.0691 0.030
relevante Positionsanzahl = 8			
iterative Schritte = 1	-	-	0.8918±0.0173 0.2267±0.0524 0.023
iterative Schritte = 2	-	-	0.8861±0.0180 0.2335±0.0521 0.029
iterative Schritte = 3	-	-	0.8815±0.0212 0.2426±0.0581 0.030
relevante Positionsanzahl = 12			
iterative Schritte = 1	-	-	0.8918±0.0179 0.2244±0.0519 0.023
iterative Schritte = 2	-	-	0.8861±0.0187 0.2335±0.0540 0.029
iterative Schritte = 3	-	-	0.8838±0.0216 0.2426±0.0607 0.031
relevante Positionsanzahl = 16			
iterative Schritte = 1	-	-	-
iterative Schritte = 2	-	-	-
iterative Schritte = 3	-	-	-
relevante Positionsanzahl = 20			
iterative Schritte = 1	-	-	-
iterative Schritte = 2	-	-	-
iterative Schritte = 3	-	-	-

Abbildung 24: J48: tr23.wc, tr31.wc, tr41.wc

originale LPCforSOS	0.9029±0.0415 0.1928±0.0884
Ursprüngliches Label auf beliebiger Position im Ranking	
relevante Positionsanzahl = 1	0.8957±0.0457 0.2217±0.0891
relevante Positionsanzahl = 2	0.8754±0.0490 0.2406±0.1195
relevante Positionsanzahl = 3	0.8928±0.0436 0.2319±0.1106
relevante Positionsanzahl = 4	0.8739±0.0565 0.2652±0.1188
relevante Positionsanzahl = 6	0.8725±0.0457 0.2884±0.1147
relevante Positionsanzahl = 8	0.8739±0.0454 0.2870±0.1186
relevante Positionsanzahl = 12	0.8768±0.0441 0.2841±0.1174
relevante Positionsanzahl = 16	-
relevante Positionsanzahl = 20	-
Zufällig genommene paarweise Präferenzen	
20,00%	0.7435±0.0440 0.5638±0.1348
40,00%	0.7928±0.0395 0.4391±0.1141
60,00%	0.8319±0.0566 0.3449±0.1128
80,00%	0.8435±0.0377 0.3391±0.1077
20% relevanter Positionsanzahl = 1	0.7145±0.0490 0.9594±0.1851
20% relevanter Positionsanzahl = 2	0.7754±0.0555 0.5609±0.2001
20% relevanter Positionsanzahl = 3	0.7348±0.0635 0.6232±0.1652
20% relevanter Positionsanzahl = 4	0.7812±0.0516 0.4739±0.1264
20% relevanter Positionsanzahl = 6	0.7478±0.0464 0.5754±0.1519
20% relevanter Positionsanzahl = 8	0.7594±0.0562 0.5594±0.1600
20% relevanter Positionsanzahl = 12	0.7725±0.0477 0.5522±0.1959
20% relevanter Positionsanzahl = 16	-
20% relevanter Positionsanzahl = 20	-
40% relevanter Positionsanzahl = 1	0.8290±0.0419 0.4058±0.0728
40% relevanter Positionsanzahl = 2	0.8159±0.0615 0.4043±0.1570
40% relevanter Positionsanzahl = 3	0.8232±0.0541 0.3551±0.1232
40% relevanter Positionsanzahl = 4	0.8145±0.0530 0.3913±0.1115
40% relevanter Positionsanzahl = 6	0.8072±0.0304 0.4145±0.1163
40% relevanter Positionsanzahl = 8	0.8087±0.0651 0.4217±0.1451
40% relevanter Positionsanzahl = 12	0.8130±0.0585 0.4130±0.1256
40% relevanter Positionsanzahl = 16	-
40% relevanter Positionsanzahl = 20	-
60% relevanter Positionsanzahl = 1	0.8594±0.0494 0.2870±0.1253
60% relevanter Positionsanzahl = 2	0.8319±0.0390 0.3391±0.1247
60% relevanter Positionsanzahl = 3	0.8449±0.0587 0.3159±0.1309
60% relevanter Positionsanzahl = 4	0.8391±0.0422 0.3464±0.1184
60% relevanter Positionsanzahl = 6	0.8420±0.0603 0.3725±0.1509
60% relevanter Positionsanzahl = 8	0.8246±0.0544 0.3768±0.1253
60% relevanter Positionsanzahl = 12	0.8333±0.0482 0.3826±0.1049
60% relevanter Positionsanzahl = 16	-
60% relevanter Positionsanzahl = 20	-
80% relevanter Positionsanzahl = 1	0.8507±0.0389 0.3145±0.0972
80% relevanter Positionsanzahl = 2	0.8652±0.0485 0.2812±0.1108
80% relevanter Positionsanzahl = 3	0.8551±0.0449 0.3087±0.1298
80% relevanter Positionsanzahl = 4	0.8391±0.0483 0.3609±0.1345
80% relevanter Positionsanzahl = 6	0.8275±0.0487 0.3855±0.1341
80% relevanter Positionsanzahl = 8	0.8420±0.0442 0.3348±0.1239
80% relevanter Positionsanzahl = 12	0.8478±0.0441 0.3420±0.0877
80% relevanter Positionsanzahl = 16	-
80% relevanter Positionsanzahl = 20	-
Ursprüngliches Label auf erster Position im Ranking	
relevante Positionsanzahl = 1	0.9029±0.0415 0.1928±0.0884 0.02254
relevante Positionsanzahl = 2	0.8725±0.0557 0.2478±0.1147 0.02254
relevante Positionsanzahl = 3	0.8826±0.0386 0.2449±0.0990 0.02254
relevante Positionsanzahl = 4	0.8725±0.0484 0.2739±0.1091 0.02254
relevante Positionsanzahl = 6	0.8696±0.0425 0.2913±0.1149 0.02254
relevante Positionsanzahl = 8	0.8667±0.0360 0.3058±0.1017 0.02254
relevante Positionsanzahl = 12	0.8681±0.0358 0.3043±0.1014 0.02254
relevante Positionsanzahl = 16	-
relevante Positionsanzahl = 20	-
Zufällig genommene paarweise Präferenzen	
20,00%	0.7493±0.0472 0.5638±0.1622 0.02254
40,00%	0.8014±0.0405 0.4449±0.1232 0.02254
60,00%	0.8261±0.0420 0.3652±0.0918 0.02254
80,00%	0.8391±0.0442 0.3536±0.0922 0.02254
20% relevanter Positionsanzahl = 1	0.6986±0.0679 0.8783±0.2560 0.02254
20% relevanter Positionsanzahl = 2	0.7551±0.0465 0.5696±0.1545 0.02254
20% relevanter Positionsanzahl = 3	0.7536±0.0598 0.6188±0.2011 0.02254
20% relevanter Positionsanzahl = 4	0.7942±0.0462 0.4652±0.1326 0.02254
20% relevanter Positionsanzahl = 6	0.7609±0.0551 0.5203±0.1373 0.02254
20% relevanter Positionsanzahl = 8	0.7522±0.0520 0.5551±0.1383 0.02254
20% relevanter Positionsanzahl = 12	0.7435±0.0645 0.5928±0.1687 0.02254
20% relevanter Positionsanzahl = 16	-
20% relevanter Positionsanzahl = 20	-
40% relevanter Positionsanzahl = 1	0.8174±0.0569 0.3754±0.1224 0.02254
40% relevanter Positionsanzahl = 2	0.8116±0.0514 0.4014±0.1526 0.02254
40% relevanter Positionsanzahl = 3	0.8058±0.0591 0.4043±0.1528 0.02254
40% relevanter Positionsanzahl = 4	0.8246±0.0508 0.3609±0.1412 0.02254

40% relevanter Positionsanzahl = 6	0.8246±0.0524 0.3899±0.1395 0.02254
40% relevanter Positionsanzahl = 8	0.8275±0.0340 0.4029±0.1088 0.02254
40% relevanter Positionsanzahl = 12	0.8072±0.0420 0.4406±0.1287 0.02254
40% relevanter Positionsanzahl = 16	-
40% relevanter Positionsanzahl = 20	-
60% relevanter Positionsanzahl = 1	0.8290±0.0634 0.3333±0.1479 0.02254
60% relevanter Positionsanzahl = 2	0.8420±0.0469 0.3261±0.0953 0.02254
60% relevanter Positionsanzahl = 3	0.8362±0.0490 0.3522±0.1491 0.02254
60% relevanter Positionsanzahl = 4	0.8304±0.0490 0.3435±0.1130 0.02254
60% relevanter Positionsanzahl = 6	0.8246±0.0460 0.3971±0.1410 0.02254
60% relevanter Positionsanzahl = 8	0.8391±0.0720 0.3362±0.1226 0.02254
60% relevanter Positionsanzahl = 12	0.8261±0.0389 0.3855±0.0944 0.02254
60% relevanter Positionsanzahl = 16	-
60% relevanter Positionsanzahl = 20	-
80% relevanter Positionsanzahl = 1	0.8681±0.0307 0.2739±0.1026 0.02254
80% relevanter Positionsanzahl = 2	0.8420±0.0596 0.3130±0.1096 0.02254
80% relevanter Positionsanzahl = 3	0.8348±0.0527 0.3464±0.1407 0.02254
80% relevanter Positionsanzahl = 4	0.8493±0.0473 0.3101±0.1218 0.02254
80% relevanter Positionsanzahl = 6	0.8391±0.0504 0.3551±0.1194 0.02254
80% relevanter Positionsanzahl = 8	0.8377±0.0541 0.3609±0.1192 0.02254
80% relevanter Positionsanzahl = 12	0.8435±0.0409 0.3638±0.1149 0.02254
80% relevanter Positionsanzahl = 16	-
80% relevanter Positionsanzahl = 20	-
Prozentsatz der relevanten Positionsanzahl ab der zweiten Position	
20% relevanter Positionsanzahl >= 2	
iterative Schritte = 1	0.8913±0.0528 0.2391±0.1112 0.02254
iterative Schritte = 2	0.8942±0.0515 0.2420±0.1202 0.02181
iterative Schritte = 3	0.8942±0.0580 0.2203±0.1173 0.02114
40% relevanter Positionsanzahl >= 2	
iterative Schritte = 1	0.8855±0.0386 0.2174±0.0946 0.02254
iterative Schritte = 2	0.8870±0.0497 0.2188±0.1108 0.02246
iterative Schritte = 3	0.8855±0.0504 0.2188±0.1173 0.02222
60% relevanter Positionsanzahl >= 2	
iterative Schritte = 1	0.8826±0.0512 0.2304±0.1052 0.02254
iterative Schritte = 2	0.8797±0.0445 0.2580±0.1057 0.02101
iterative Schritte = 3	0.8884±0.0542 0.2217±0.1117 0.02227
80% relevanter Positionsanzahl >= 2	
iterative Schritte = 1	0.8696±0.0618 0.2667±0.1163 0.02254
iterative Schritte = 2	0.8826±0.0478 0.2377±0.0926 0.02294
iterative Schritte = 3	0.8797±0.0361 0.2464±0.0961 0.02404
Ranking ignorieren bei falscher Vorhersage erster Position	
relevante Positionsanzahl = 2	
iterative Schritte = 1	0.8725±0.0561 0.2551±0.1180 0.02254
iterative Schritte = 2	0.8638±0.0436 0.2652±0.1100 0.02367
iterative Schritte = 3	0.8652±0.0454 0.2609±0.1132 0.02404
relevante Positionsanzahl = 3	
iterative Schritte = 1	0.8797±0.0400 0.2420±0.0921 0.02254
iterative Schritte = 2	0.8870±0.0419 0.2406±0.1055 0.02713
iterative Schritte = 3	0.8739±0.0468 0.2551±0.1083 0.02657
relevante Positionsanzahl = 4	
iterative Schritte = 1	0.8754±0.0450 0.2623±0.1015 0.02254
iterative Schritte = 2	0.8681±0.0555 0.2797±0.1136 0.02640
iterative Schritte = 3	0.8551±0.0463 0.2928±0.0832 0.02683
relevante Positionsanzahl = 6	
iterative Schritte = 1	0.8667±0.0484 0.2754±0.1218 0.02254
iterative Schritte = 2	0.8580±0.0568 0.2928±0.1317 0.02520
iterative Schritte = 3	0.8696±0.0558 0.2754±0.1303 0.02571
relevante Positionsanzahl = 8	
iterative Schritte = 1	0.8580±0.0409 0.2928±0.1016 0.02254
iterative Schritte = 2	0.8609±0.0642 0.3014±0.1461 0.02439
iterative Schritte = 3	0.8638±0.0609 0.2841±0.1387 0.02474
relevante Positionsanzahl = 12	
iterative Schritte = 1	0.8580±0.0409 0.2942±0.1053 0.02254
iterative Schritte = 2	0.8623±0.0636 0.2942±0.1426 0.02447
iterative Schritte = 3	0.8623±0.0649 0.2899±0.1308 0.02479
relevante Positionsanzahl = 16	
iterative Schritte = 1	-
iterative Schritte = 2	-
iterative Schritte = 3	-
relevante Positionsanzahl = 20	
iterative Schritte = 1	-
iterative Schritte = 2	-
iterative Schritte = 3	-

Abbildung 25: J48: tr45.wc

	fbis_wc.earff	ohscal_wc.earff	re1_wc.earff
originales LPCforSOS	0.8408±0.0219 0.2793±0.0724	0.7721±0.0115 0.4221±0.0373	0.8582±0.0210 0.3777±0.1026
Ursprüngliches Label auf beliebiger Position im Ranking			
relevante Positionsanzahl = 1	0.8408±0.0215 0.2802±0.0723	0.7721±0.0115 0.4221±0.0373	0.8600±0.0214 0.3831±0.1070
relevante Positionsanzahl = 2	0.8209±0.0293 0.3573±0.0833	0.7574±0.0129 0.4676±0.0391	0.8316±0.0278 0.5062±0.1494
relevante Positionsanzahl = 3	0.8043±0.0296 0.3939±0.0904	0.7466±0.0127 0.4887±0.0439	0.8117±0.0360 0.6040±0.1542
relevante Positionsanzahl = 4	0.8010±0.0340 0.4060±0.0809	0.7429±0.0137 0.4980±0.0399	0.7954±0.0305 0.7235±0.1706
relevante Positionsanzahl = 6	0.7816±0.0376 0.4284±0.0993	0.7357±0.0116 0.5079±0.0375	0.7761±0.0256 0.8556±0.2188
relevante Positionsanzahl = 8	0.7710±0.0355 0.4556±0.0945	0.7340±0.0141 0.5133±0.0472	0.7532±0.0224 0.9527±0.1904
relevante Positionsanzahl = 12	0.7702±0.0315 0.4787±0.1063	0.7336±0.0143 0.5167±0.0496	0.7327±0.0285 1.0474±0.2553
relevante Positionsanzahl = 16	0.7726±0.0323 0.4751±0.1099	-	0.7248±0.0297 1.0788±0.2615
relevante Positionsanzahl = 20	-	-	0.7260±0.0283 1.0764±0.2664
Zufällig genommene paarweise Präferenzen			
20,00%	0.7145±0.0369 0.5940±0.0867	0.7006±0.0115 0.6035±0.0459	0.6204±0.0530 1.7190±0.3690
40,00%	0.7446±0.0356 0.5279±0.1183	0.7156±0.0165 0.5552±0.0464	0.6590±0.0514 1.4602±0.3464
60,00%	0.7519±0.0307 0.5124±0.0904	0.7238±0.0146 0.5461±0.0451	0.6922±0.0357 1.3148±0.3045
80,00%	0.7613±0.0317 0.4986±0.0878	0.7242±0.0090 0.5365±0.0412	0.6958±0.0217 1.3092±0.3317
20% relevanter Positionsanzahl = 1	0.7393±0.0297 0.5449±0.0969	0.7287±0.0077 0.5346±0.0312	0.6547±0.0478 1.2742±0.2427
20% relevanter Positionsanzahl = 2	0.7625±0.0297 0.4624±0.0841	0.7242±0.0093 0.5383±0.0392	0.6705±0.0510 1.2140±0.2259
20% relevanter Positionsanzahl = 3	0.7507±0.0341 0.5014±0.0970	0.7122±0.0132 0.5687±0.0447	0.6578±0.0390 1.3270±0.2669
20% relevanter Positionsanzahl = 4	0.7487±0.0371 0.5027±0.0728	0.7132±0.0162 0.5783±0.0516	0.6590±0.0449 1.4102±0.2765
20% relevanter Positionsanzahl = 6	0.7373±0.0397 0.5644±0.1006	0.7047±0.0120 0.5885±0.0354	0.6524±0.0424 1.5267±0.2762
20% relevanter Positionsanzahl = 8	0.7442±0.0320 0.5595±0.0930	0.7057±0.0134 0.5916±0.0426	0.6337±0.0467 1.6141±0.3068
20% relevanter Positionsanzahl = 12	0.7279±0.0391 0.5746±0.1104	0.6998±0.0116 0.5994±0.0424	0.6385±0.0411 1.7021±0.3922
20% relevanter Positionsanzahl = 16	0.7296±0.0447 0.5688±0.1090	-	0.6349±0.0372 1.7576±0.3949
20% relevanter Positionsanzahl = 20	-	-	0.6252±0.0443 1.7310±0.4144
40% relevanter Positionsanzahl = 1	0.7735±0.0206 0.4644±0.1000	0.7532±0.0106 0.4692±0.0345	0.7568±0.0317 0.7544±0.1836
40% relevanter Positionsanzahl = 2	0.7864±0.0238 0.3918±0.0714	0.7369±0.0112 0.5052±0.0395	0.7362±0.0398 0.8853±0.2070
40% relevanter Positionsanzahl = 3	0.7804±0.0379 0.4373±0.0900	0.7294±0.0127 0.5330±0.0400	0.7344±0.0445 0.9595±0.1749
40% relevanter Positionsanzahl = 4	0.7674±0.0383 0.4486±0.0844	0.7249±0.0158 0.5400±0.0476	0.7121±0.0466 1.0644±0.2361
40% relevanter Positionsanzahl = 6	0.7600±0.0321 0.4885±0.0867	0.7187±0.0105 0.5501±0.0335	0.6958±0.0464 1.2304±0.2290
40% relevanter Positionsanzahl = 8	0.7499±0.0328 0.5148±0.0931	0.7213±0.0124 0.5583±0.0372	0.6838±0.0298 1.2858±0.2250
40% relevanter Positionsanzahl = 12	0.7483±0.0306 0.5355±0.0894	0.7207±0.0160 0.5572±0.0413	0.6596±0.0397 1.4457±0.2764
40% relevanter Positionsanzahl = 16	0.7442±0.0286 0.5291±0.0961	-	0.6626±0.0368 1.4554±0.3460
40% relevanter Positionsanzahl = 20	-	-	0.6693±0.0452 1.4504±0.3260
60% relevanter Positionsanzahl = 1	0.7893±0.0309 0.4039±0.0992	0.7637±0.0080 0.4478±0.0309	0.7839±0.0383 0.6360±0.1761
60% relevanter Positionsanzahl = 2	0.7970±0.0276 0.3764±0.0829	0.7442±0.0085 0.4824±0.0304	0.7755±0.0357 0.7458±0.2066
60% relevanter Positionsanzahl = 3	0.7897±0.0266 0.4235±0.0792	0.7337±0.0118 0.5093±0.0453	0.7616±0.0371 0.7983±0.1747
60% relevanter Positionsanzahl = 4	0.7698±0.0287 0.4519±0.0794	0.7307±0.0108 0.5190±0.0387	0.7411±0.0388 0.9413±0.2045
60% relevanter Positionsanzahl = 6	0.7698±0.0369 0.4653±0.0967	0.7259±0.0148 0.5386±0.0405	0.7260±0.0270 1.0734±0.1805
60% relevanter Positionsanzahl = 8	0.7462±0.0351 0.4986±0.0798	0.7251±0.0105 0.5460±0.0423	0.7031±0.0358 1.1851±0.2011
60% relevanter Positionsanzahl = 12	0.7617±0.0398 0.5010±0.1259	0.7228±0.0152 0.5420±0.0476	0.6946±0.0304 1.2924±0.2892
60% relevanter Positionsanzahl = 16	0.7458±0.0397 0.5376±0.1069	-	0.6868±0.0396 1.3249±0.3149
60% relevanter Positionsanzahl = 20	-	-	0.6844±0.0379 1.3503±0.3233
80% relevanter Positionsanzahl = 1	0.8059±0.0305 0.3662±0.0749	0.7690±0.0156 0.4356±0.0333	0.8202±0.0283 0.5274±0.1368
80% relevanter Positionsanzahl = 2	0.8055±0.0256 0.3633±0.0909	0.7491±0.0129 0.4792±0.0408	0.7827±0.0330 0.6794±0.1621
80% relevanter Positionsanzahl = 3	0.7897±0.0375 0.4158±0.1001	0.7371±0.0124 0.4995±0.0352	0.7682±0.0323 0.7645±0.1921
80% relevanter Positionsanzahl = 4	0.7771±0.0206 0.4548±0.0871	0.7374±0.0120 0.5112±0.0381	0.7562±0.0322 0.8630±0.1681
80% relevanter Positionsanzahl = 6	0.7722±0.0343 0.4698±0.0879	0.7318±0.0143 0.5309±0.0468	0.7351±0.0364 1.0336±0.2038
80% relevanter Positionsanzahl = 8	0.7560±0.0340 0.4986±0.1024	0.7273±0.0134 0.5291±0.0470	0.7079±0.0467 1.1253±0.2275
80% relevanter Positionsanzahl = 12	0.7621±0.0324 0.4999±0.0915	0.7240±0.0146 0.5385±0.0508	0.7031±0.0417 1.2556±0.3008
80% relevanter Positionsanzahl = 16	0.7535±0.0350 0.5076±0.0819	-	0.7103±0.0388 1.2260±0.2893
80% relevanter Positionsanzahl = 20	-	-	0.7013±0.0320 1.2531±0.2842
Ursprüngliches Label auf erster Position im Ranking			
relevante Positionsanzahl = 1	0.8408±0.0219 0.2793±0.0724	0.7721±0.0115 0.4221±0.0373	0.8582±0.0210 0.3777±0.1026
relevante Positionsanzahl = 2	0.8213±0.0295 0.3569±0.0836	0.7574±0.0129 0.4676±0.0391	0.8310±0.0280 0.5080±0.1479
relevante Positionsanzahl = 3	0.8035±0.0289 0.3947±0.0898	0.7466±0.0127 0.4887±0.0439	0.8129±0.0368 0.6040±0.1575
relevante Positionsanzahl = 4	0.8010±0.0340 0.4060±0.0809	0.7429±0.0137 0.4980±0.0399	0.7942±0.0315 0.7241±0.1705
relevante Positionsanzahl = 6	0.7820±0.0383 0.4272±0.0994	0.7357±0.0116 0.5079±0.0375	0.7755±0.0256 0.8568±0.2165
relevante Positionsanzahl = 8	0.7710±0.0355 0.4556±0.0945	0.7340±0.0141 0.5133±0.0472	0.7532±0.0226 0.9521±0.1910
relevante Positionsanzahl = 12	0.7702±0.0315 0.4787±0.1063	0.7336±0.0143 0.5167±0.0496	0.7321±0.0280 1.0468±0.2553
relevante Positionsanzahl = 16	0.7726±0.0323 0.4751±0.1099	-	0.7248±0.0293 1.0782±0.2596
relevante Positionsanzahl = 20	-	-	0.7260±0.0279 1.0770±0.2635
Zufällig genommene paarweise Präferenzen			
20,00%	0.7231±0.0361 0.5822±0.1140	0.7061±0.0112 0.5845±0.0413	0.6143±0.0446 1.7407±0.4046
40,00%	0.7341±0.0362 0.5530±0.0962	0.7188±0.0113 0.5596±0.0381	0.6711±0.0432 1.4680±0.3145
60,00%	0.7560±0.0325 0.5108±0.0907	0.7245±0.0130 0.5426±0.0421	0.6838±0.0443 1.3546±0.3153
80,00%	0.7548±0.0358 0.5128±0.1063	0.7249±0.0120 0.5404±0.0397	0.6940±0.0411 1.2845±0.3167
20% relevanter Positionsanzahl = 1	0.7479±0.0311 0.5376±0.1084	0.7259±0.0130 0.5390±0.0396	0.6620±0.0554 1.2886±0.2782
20% relevanter Positionsanzahl = 2	0.7597±0.0395 0.4657±0.0727	0.7225±0.0125 0.5473±0.0321	0.6711±0.0447 1.1967±0.2308
20% relevanter Positionsanzahl = 3	0.7576±0.0371 0.4775±0.0843	0.7106±0.0167 0.5685±0.0445	0.6699±0.0336 1.3059±0.2500
20% relevanter Positionsanzahl = 4	0.7450±0.0240 0.5347±0.0941	0.7106±0.0059 0.5700±0.0339	0.6470±0.0384 1.4180±0.2605
20% relevanter Positionsanzahl = 6	0.7320±0.0370 0.5554±0.0920	0.7069±0.0102 0.5906±0.0410	0.6415±0.0403 1.5388±0.2905
20% relevanter Positionsanzahl = 8	0.7361±0.0272 0.5494±0.0886	0.6999±0.0079 0.5979±0.0355	0.6313±0.0442 1.6153±0.3167
20% relevanter Positionsanzahl = 12	0.7450±0.0302 0.5623±0.0957	0.6998±0.0127 0.5962±0.0422	0.6354±0.0467 1.6841±0.3429
20% relevanter Positionsanzahl = 16	0.7251±0.0318 0.5713±0.0963	-	0.6253±0.0324 1.7219±0.3481
20% relevanter Positionsanzahl = 20	-	-	0.6282±0.0384 1.7557±0.4127
40% relevanter Positionsanzahl = 1	0.7739±0.0280 0.4831±0.0756	0.7507±0.0147 0.4708±0.0369	0.7525±0.0474 0.8080±0.2136
40% relevanter Positionsanzahl = 2	0.7865±0.0302 0.3942±0.0942	0.7411±0.0148 0.5033±0.0360	0.7417±0.0390 0.8810±0.2032
40% relevanter Positionsanzahl = 3	0.7682±0.0304 0.4332±0.0757	0.7321±0.0107 0.5278±0.0409	0.7254±0.0403 0.9721±0.2006
40% relevanter Positionsanzahl = 4	0.7653±0.0311 0.4771±0.0866	0.7320±0.0125 0.5310±0.0364	0.7193±0.0430 1.0705±0.1951

40% relevanter Positionsanzahl = 6	0.7677±0.0302 0.4994±0.0793	0.7227±0.0130 0.5481±0.0415	0.6880±0.0483 1.1850±0.1660
40% relevanter Positionsanzahl = 8	0.7535±0.0385 0.5031±0.0992	0.7179±0.0138 0.5493±0.0438	0.6813±0.0418 1.3064±0.2805
40% relevanter Positionsanzahl = 12	0.7389±0.0292 0.5493±0.0965	0.7159±0.0136 0.5579±0.0359	0.6687±0.0365 1.4469±0.3291
40% relevanter Positionsanzahl = 16	0.7393±0.0322 0.5429±0.1062	-	0.6663±0.0409 1.4799±0.3742
40% relevanter Positionsanzahl = 20	-	-	0.6747±0.0448 1.4281±0.2976
60% relevanter Positionsanzahl = 1	0.7779±0.0404 0.4344±0.0924	0.7661±0.0136 0.4458±0.0342	0.8051±0.0268 0.6010±0.1500
60% relevanter Positionsanzahl = 2	0.7937±0.0319 0.3833±0.0800	0.7462±0.0126 0.4904±0.0393	0.7694±0.0393 0.7205±0.1442
60% relevanter Positionsanzahl = 3	0.7808±0.0271 0.4150±0.0847	0.7364±0.0153 0.5159±0.0422	0.7513±0.0432 0.8562±0.1824
60% relevanter Positionsanzahl = 4	0.7803±0.0308 0.4389±0.0821	0.7329±0.0117 0.5189±0.0418	0.7417±0.0391 0.9340±0.1916
60% relevanter Positionsanzahl = 6	0.7641±0.0368 0.4759±0.0871	0.7210±0.0113 0.5436±0.0423	0.7266±0.0394 1.1078±0.1969
60% relevanter Positionsanzahl = 8	0.7466±0.0361 0.5242±0.1132	0.7242±0.0102 0.5374±0.0374	0.7031±0.0408 1.1899±0.2536
60% relevanter Positionsanzahl = 12	0.7503±0.0340 0.5148±0.0947	0.7233±0.0153 0.5455±0.0445	0.6928±0.0364 1.3148±0.2934
60% relevanter Positionsanzahl = 16	0.7511±0.0349 0.5100±0.0944	-	0.6855±0.0331 1.3696±0.3234
60% relevanter Positionsanzahl = 20	-	-	0.6862±0.0341 1.3811±0.3151
80% relevanter Positionsanzahl = 1	0.8003±0.0334 0.3910±0.0733	0.7709±0.0071 0.4324±0.0298	0.8165±0.0276 0.5316±0.1537
80% relevanter Positionsanzahl = 2	0.8080±0.0302 0.3682±0.0845	0.7520±0.0136 0.4696±0.0409	0.7779±0.0250 0.6903±0.1735
80% relevanter Positionsanzahl = 3	0.7913±0.0271 0.3902±0.0780	0.7424±0.0143 0.4959±0.0439	0.7676±0.0306 0.7748±0.1640
80% relevanter Positionsanzahl = 4	0.7807±0.0285 0.4454±0.0763	0.7345±0.0139 0.5124±0.0401	0.7568±0.0442 0.8827±0.2075
80% relevanter Positionsanzahl = 6	0.7682±0.0373 0.4621±0.0837	0.7271±0.0167 0.5353±0.0433	0.7326±0.0373 1.0119±0.1900
80% relevanter Positionsanzahl = 8	0.7596±0.0355 0.4815±0.0949	0.7289±0.0164 0.5360±0.0527	0.7158±0.0372 1.1501±0.2085
80% relevanter Positionsanzahl = 12	0.7588±0.0369 0.4990±0.0820	0.7208±0.0160 0.5452±0.0513	0.7103±0.0315 1.2478±0.2782
80% relevanter Positionsanzahl = 16	0.7527±0.0344 0.5116±0.0962	-	0.6874±0.0343 1.2809±0.3092
80% relevanter Positionsanzahl = 20	-	-	0.7013±0.0326 1.2912±0.3004
Prozentsatz der relevanten Positionsanzahl ab der zweiten Position			
20% relevanter Positionsanzahl >= 2			
iterative Schritte = 1	0.8380±0.0228 0.2769±0.0546	0.7728±0.0133 0.4264±0.0392	0.8558±0.0200 0.3910±0.1064
iterative Schritte = 2	-	-	-
40% relevanter Positionsanzahl >= 2			
iterative Schritte = 1	0.8368±0.0229 0.2846±0.0783	0.7692±0.0098 0.4307±0.0322	0.8497±0.0263 0.4030±0.1234
iterative Schritte = 2	0.8368±0.0226 0.2826±0.0718	-	-
60% relevanter Positionsanzahl >= 2			
iterative Schritte = 1	0.8376±0.0222 0.2907±0.0650	0.7669±0.0108 0.4348±0.0341	0.8503±0.0199 0.3964±0.1077
iterative Schritte = 2	-	-	-
80% relevanter Positionsanzahl >= 2			
iterative Schritte = 1	0.8335±0.0237 0.3037±0.0650	0.7656±0.0122 0.4419±0.0381	0.8455±0.0261 0.4326±0.1266
iterative Schritte = 2	-	-	0.8425±0.0252 0.4364±0.1489
Ranking ignorieren bei falscher Vorhersage erster Position			
relevante Positionsanzahl = 2			
iterative Schritte = 1	0.8201±0.0282 0.3585±0.0826	0.7574±0.0129 0.4676±0.0391	0.8310±0.0280 0.5068±0.1468
iterative Schritte = 2	0.8205±0.0242 0.3346±0.0758	0.7578±0.0144 0.4571±0.0430	0.8364±0.0337 0.5002±0.1386
iterative Schritte = 3	0.8205±0.0242 0.3342±0.0756	0.7599±0.0143 0.4547±0.0425	0.8364±0.0337 0.4996±0.1409
relevante Positionsanzahl = 3			
iterative Schritte = 1	0.8035±0.0289 0.3951±0.0905	0.7466±0.0127 0.4887±0.0439	0.8129±0.0368 0.6046±0.1553
iterative Schritte = 2	0.8112±0.0278 0.3801±0.0958	0.7497±0.0128 0.4788±0.0418	0.8135±0.0341 0.5949±0.1484
iterative Schritte = 3	0.8100±0.0289 0.3817±0.0954	0.7489±0.0137 0.4795±0.0421	0.8123±0.0340 0.5979±0.1458
relevante Positionsanzahl = 4			
iterative Schritte = 1	0.8014±0.0341 0.4060±0.0811	0.7429±0.0137 0.4980±0.0399	0.7948±0.0318 0.7229±0.1707
iterative Schritte = 2	0.7994±0.0277 0.3983±0.0796	0.7424±0.0138 0.4979±0.0433	0.8027±0.0292 0.6909±0.1672
iterative Schritte = 3	0.7998±0.0274 0.3967±0.0762	0.7433±0.0142 0.4963±0.0413	0.8021±0.0286 0.6927±0.1633
relevante Positionsanzahl = 6			
iterative Schritte = 1	0.7799±0.0385 0.4292±0.0997	0.7357±0.0116 0.5079±0.0375	0.7767±0.0255 0.8550±0.2164
iterative Schritte = 2	0.7901±0.0289 0.4231±0.0897	0.7354±0.0125 0.5146±0.0440	0.7827±0.0255 0.8224±0.2066
iterative Schritte = 3	0.7860±0.0325 0.4243±0.0880	0.7350±0.0111 0.5162±0.0407	0.7815±0.0255 0.8225±0.2065
relevante Positionsanzahl = 8			
iterative Schritte = 1	0.7718±0.0356 0.4548±0.0959	0.7340±0.0141 0.5133±0.0472	0.7538±0.0225 0.9521±0.1910
iterative Schritte = 2	0.7726±0.0384 0.4564±0.1024	0.7338±0.0143 0.5165±0.0485	0.7640±0.0245 0.8743±0.1794
iterative Schritte = 3	0.7751±0.0380 0.4511±0.1086	0.7345±0.0139 0.5151±0.0464	0.7670±0.0263 0.8810±0.1860
relevante Positionsanzahl = 12			
iterative Schritte = 1	0.7694±0.0315 0.4808±0.1058	0.7336±0.0143 0.5167±0.0496	0.7315±0.0269 1.0492±0.2563
iterative Schritte = 2	0.7682±0.0357 0.4795±0.0968	0.7302±0.0151 0.5193±0.0525	0.7417±0.0277 1.0191±0.2358
iterative Schritte = 3	0.7698±0.0334 0.4706±0.1019	0.7309±0.0145 0.5168±0.0504	0.7411±0.0267 1.0197±0.2417
relevante Positionsanzahl = 16			
iterative Schritte = 1	0.7734±0.0320 0.4743±0.1107	-	0.7236±0.0284 1.0794±0.2595
iterative Schritte = 2	0.7665±0.0372 0.4852±0.1125	-	0.7278±0.0261 1.0836±0.2547
iterative Schritte = 3	0.7698±0.0351 0.4828±0.1097	-	0.7266±0.0253 1.0891±0.2595
relevante Positionsanzahl = 20			
iterative Schritte = 1	-	-	0.7236±0.0259 1.0842±0.2615
iterative Schritte = 2	-	-	0.7290±0.0276 1.0806±0.2510
iterative Schritte = 3	-	-	0.7302±0.0287 1.0836±0.2560

Abbildung 26: LibLINEAR: fbis.wc, ohscal.wc, re1.wc

	la1s_wc.earff	la2s_wc.earff	wap_wc.earff
originales LPCforSOS	0.8892±0.0163 0.1588±0.0249	0.8976±0.0164 0.1483±0.0261	0.8506±0.0197 0.4481±0.1506
Ursprüngliches Label auf beliebiger Position im Ranking			
relevante Positionsanzahl = 1	0.8892±0.0163 0.1588±0.0249	0.8976±0.0164 0.1483±0.0261	0.8506±0.0197 0.4481±0.1506
relevante Positionsanzahl = 2	0.8745±0.0164 0.1851±0.0274	0.8803±0.0140 0.1788±0.0249	0.8276±0.0223 0.5346±0.1694
relevante Positionsanzahl = 3	0.8646±0.0152 0.2054±0.0249	0.8741±0.0152 0.2010±0.0226	0.8173±0.0232 0.5769±0.1883
relevante Positionsanzahl = 4	0.8602±0.0130 0.2085±0.0244	0.8706±0.0140 0.2026±0.0321	0.8083±0.0213 0.6199±0.1849
relevante Positionsanzahl = 6	0.8564±0.0172 0.2206±0.0327	0.8680±0.0190 0.2075±0.0284	0.8038±0.0254 0.6968±0.1823
relevante Positionsanzahl = 8	0.8564±0.0172 0.2206±0.0327	0.8680±0.0190 0.2075±0.0284	0.7885±0.0205 0.7359±0.1732
relevante Positionsanzahl = 12	-	-	0.7891±0.0161 0.7526±0.1848
relevante Positionsanzahl = 16	-	-	0.7885±0.0188 0.7615±0.1876
relevante Positionsanzahl = 20	-	-	0.7885±0.0188 0.7609±0.1877
Zufällig genommene paarweise Präferenzen			
20,00%	0.7831±0.0176 0.3533±0.0410	0.7838±0.0200 0.3346±0.0425	0.6141±0.0253 1.6237±0.2487
40,00%	0.8140±0.0124 0.2949±0.0346	0.8361±0.0206 0.2702±0.0327	0.6987±0.0258 1.1667±0.1886
60,00%	0.8308±0.0171 0.2678±0.0355	0.8413±0.0254 0.2517±0.0284	0.7250±0.0225 1.0154±0.2183
80,00%	0.8346±0.0198 0.2434±0.0286	0.8472±0.0164 0.2400±0.0318	0.7526±0.0119 0.9167±0.1813
20% relevanter Positionsanzahl = 1	0.8174±0.0224 0.2768±0.0302	0.8273±0.0240 0.2715±0.0427	0.6273±0.0326 1.4931±0.2318
20% relevanter Positionsanzahl = 2	0.7694±0.0288 0.3689±0.0479	0.7906±0.0284 0.3375±0.0443	0.6179±0.0249 1.4506±0.2268
20% relevanter Positionsanzahl = 3	0.7803±0.0255 0.3371±0.0466	0.8023±0.0148 0.3258±0.0317	0.6173±0.0404 1.5173±0.2385
20% relevanter Positionsanzahl = 4	0.7734±0.0131 0.3829±0.0395	0.7779±0.0300 0.3590±0.0433	0.6250±0.0347 1.4846±0.2073
20% relevanter Positionsanzahl = 6	0.7709±0.0212 0.3661±0.0466	0.8013±0.0166 0.3284±0.0336	0.6115±0.0313 1.6013±0.2360
20% relevanter Positionsanzahl = 8	0.7878±0.0268 0.3542±0.0577	0.7942±0.0137 0.3275±0.0322	0.6154±0.0336 1.5949±0.2658
20% relevanter Positionsanzahl = 12	-	-	0.6192±0.0275 1.6417±0.2380
20% relevanter Positionsanzahl = 16	-	-	0.6199±0.0326 1.5705±0.2552
20% relevanter Positionsanzahl = 20	-	-	0.6263±0.0250 1.6359±0.2782
40% relevanter Positionsanzahl = 1	0.8387±0.0205 0.2291±0.0316	0.8576±0.0157 0.2143±0.0269	0.7244±0.0247 1.0494±0.2144
40% relevanter Positionsanzahl = 2	0.8262±0.0155 0.2537±0.0282	0.8367±0.0232 0.2485±0.0404	0.7179±0.0240 0.9763±0.1935
40% relevanter Positionsanzahl = 3	0.8224±0.0208 0.2787±0.0373	0.8306±0.0193 0.2729±0.0276	0.6974±0.0267 1.0731±0.2213
40% relevanter Positionsanzahl = 4	0.8109±0.0227 0.2899±0.0424	0.8322±0.0175 0.2770±0.0310	0.7071±0.0278 1.0564±0.2389
40% relevanter Positionsanzahl = 6	0.8162±0.0222 0.2909±0.0361	0.8351±0.0130 0.2634±0.0311	0.6923±0.0287 1.1449±0.2247
40% relevanter Positionsanzahl = 8	0.8171±0.0174 0.2880±0.0320	0.8348±0.0179 0.2728±0.0265	0.6981±0.0237 1.1519±0.2192
40% relevanter Positionsanzahl = 12	-	-	0.6981±0.0303 1.1468±0.2146
40% relevanter Positionsanzahl = 16	-	-	0.6955±0.0341 1.1635±0.2262
40% relevanter Positionsanzahl = 20	-	-	0.6981±0.0221 1.1763±0.2201
60% relevanter Positionsanzahl = 1	0.8605±0.0200 0.2069±0.0220	0.8673±0.0175 0.1893±0.0255	0.7744±0.0300 0.8212±0.1982
60% relevanter Positionsanzahl = 2	0.8496±0.0102 0.2278±0.0249	0.8566±0.0200 0.2143±0.0359	0.7596±0.0163 0.8077±0.1955
60% relevanter Positionsanzahl = 3	0.8436±0.0092 0.2306±0.0322	0.8459±0.0236 0.2358±0.0388	0.7429±0.0241 0.8551±0.2166
60% relevanter Positionsanzahl = 4	0.8368±0.0122 0.2515±0.0331	0.8530±0.0155 0.2390±0.0337	0.7462±0.0195 0.8846±0.1996
60% relevanter Positionsanzahl = 6	0.8302±0.0167 0.2631±0.0400	0.8455±0.0256 0.2478±0.0360	0.7301±0.0215 0.9359±0.2564
60% relevanter Positionsanzahl = 8	0.8343±0.0161 0.2615±0.0286	0.8462±0.0233 0.2487±0.0350	0.7295±0.0240 0.9955±0.1995
60% relevanter Positionsanzahl = 12	-	-	0.7346±0.0172 0.9769±0.1846
60% relevanter Positionsanzahl = 16	-	-	0.7340±0.0205 1.0077±0.2144
60% relevanter Positionsanzahl = 20	-	-	0.7314±0.0189 1.0135±0.1794
80% relevanter Positionsanzahl = 1	0.8705±0.0181 0.1922±0.0264	0.8833±0.0224 0.1756±0.0305	0.8026±0.0172 0.6981±0.1732
80% relevanter Positionsanzahl = 2	0.8511±0.0203 0.2153±0.0313	0.8631±0.0166 0.2058±0.0237	0.7769±0.0248 0.7231±0.1937
80% relevanter Positionsanzahl = 3	0.8465±0.0154 0.2306±0.0296	0.8569±0.0192 0.2276±0.0319	0.7577±0.0200 0.7846±0.2242
80% relevanter Positionsanzahl = 4	0.8380±0.0221 0.2540±0.0377	0.8540±0.0150 0.2241±0.0316	0.7647±0.0190 0.8103±0.2208
80% relevanter Positionsanzahl = 6	0.8424±0.0216 0.2487±0.0447	0.8537±0.0195 0.2325±0.0264	0.7660±0.0195 0.8571±0.2076
80% relevanter Positionsanzahl = 8	0.8361±0.0220 0.2659±0.0398	0.8511±0.0142 0.2445±0.0214	0.7603±0.0211 0.8878±0.1858
80% relevanter Positionsanzahl = 12	-	-	0.7545±0.0215 0.9314±0.2115
80% relevanter Positionsanzahl = 16	-	-	0.7487±0.0234 0.9295±0.2174
80% relevanter Positionsanzahl = 20	-	-	0.7449±0.0227 0.9365±0.1857
Ursprüngliches Label auf erster Position im Ranking			
relevante Positionsanzahl = 1	0.8892±0.0163 0.1588±0.0249	0.8976±0.0164 0.1483±0.0261	0.8506±0.0197 0.4481±0.1506
relevante Positionsanzahl = 2	0.8745±0.0164 0.1851±0.0274	0.8803±0.0140 0.1788±0.0249	0.8276±0.0223 0.5346±0.1694
relevante Positionsanzahl = 3	0.8646±0.0152 0.2054±0.0249	0.8741±0.0152 0.2010±0.0226	0.8173±0.0232 0.5769±0.1883
relevante Positionsanzahl = 4	0.8602±0.0130 0.2085±0.0244	0.8706±0.0140 0.2026±0.0321	0.8083±0.0213 0.6199±0.1849
relevante Positionsanzahl = 6	0.8564±0.0172 0.2206±0.0327	0.8680±0.0190 0.2075±0.0284	0.8038±0.0254 0.6968±0.1823
relevante Positionsanzahl = 8	0.8564±0.0172 0.2206±0.0327	0.8680±0.0190 0.2075±0.0284	0.7885±0.0205 0.7359±0.1732
relevante Positionsanzahl = 12	-	-	0.7691±0.0225 0.7219±0.1783
relevante Positionsanzahl = 16	-	-	0.7640±0.0217 0.7248±0.1815
relevante Positionsanzahl = 20	-	-	0.7607±0.0242 0.7309±0.1773
Zufällig genommene paarweise Präferenzen			
20,00%	0.7868±0.0198 0.3411±0.0385	0.7916±0.0250 0.3268±0.0427	0.6122±0.0401 1.6199±0.2877
40,00%	0.8146±0.0241 0.2906±0.0375	0.8296±0.0188 0.2715±0.0319	0.6955±0.0193 1.1808±0.2228
60,00%	0.8337±0.0129 0.2631±0.0256	0.8468±0.0209 0.2488±0.0313	0.7256±0.0269 0.9942±0.1907
80,00%	0.8443±0.0209 0.2462±0.0303	0.8423±0.0192 0.2407±0.0304	0.7532±0.0170 0.9019±0.1954
20% relevanter Positionsanzahl = 1	0.8115±0.0287 0.2731±0.0448	0.8273±0.0197 0.2673±0.0312	0.5942±0.0363 1.6526±0.2052
20% relevanter Positionsanzahl = 2	0.7665±0.0226 0.3661±0.0393	0.7685±0.0302 0.3635±0.0402	0.6353±0.0356 1.4410±0.2461
20% relevanter Positionsanzahl = 3	0.7762±0.0205 0.3589±0.0498	0.7851±0.0258 0.3398±0.0392	0.6135±0.0355 1.4987±0.2228
20% relevanter Positionsanzahl = 4	0.7687±0.0121 0.3655±0.0428	0.7717±0.0269 0.3772±0.0557	0.6231±0.0238 1.4545±0.2097
20% relevanter Positionsanzahl = 6	0.7668±0.0184 0.3396±0.0483	0.8036±0.0201 0.3310±0.0381	0.6154±0.0305 1.5654±0.2161
20% relevanter Positionsanzahl = 8	0.7856±0.0198 0.3452±0.0492	0.7971±0.0164 0.3320±0.0331	0.6199±0.0406 1.6058±0.2626
20% relevanter Positionsanzahl = 12	-	-	0.6160±0.0311 1.6571±0.2590
20% relevanter Positionsanzahl = 16	-	-	0.6295±0.0295 1.5910±0.2851
20% relevanter Positionsanzahl = 20	-	-	0.6071±0.0267 1.6865±0.2774
40% relevanter Positionsanzahl = 1	0.8493±0.0168 0.2082±0.0271	0.8608±0.0130 0.2036±0.0239	0.7147±0.0265 1.0577±0.2014
40% relevanter Positionsanzahl = 2	0.8268±0.0160 0.2600±0.0336	0.8459±0.0221 0.2354±0.0306	0.7199±0.0284 1.0385±0.2270
40% relevanter Positionsanzahl = 3	0.8187±0.0217 0.2784±0.0418	0.8306±0.0175 0.2598±0.0252	0.7122±0.0273 1.0654±0.2243
40% relevanter Positionsanzahl = 4	0.8221±0.0167 0.2831±0.0321	0.8280±0.0216 0.2738±0.0333	0.7032±0.0344 1.0474±0.2399

40% relevanter Positionsanzahl = 6	0.8140±0.0161 0.3055±0.0415	0.8289±0.0168 0.2780±0.0290	0.7038±0.0246 1.1026±0.1835
40% relevanter Positionsanzahl = 8	0.8140±0.0133 0.2996±0.0357	0.8251±0.0229 0.2663±0.0364	0.7000±0.0253 1.1288±0.2325
40% relevanter Positionsanzahl = 12	-	-	0.6840±0.0337 1.1897±0.2134
40% relevanter Positionsanzahl = 16	-	-	0.6968±0.0378 1.1609±0.2548
40% relevanter Positionsanzahl = 20	-	-	0.6962±0.0374 1.1904±0.2379
60% relevanter Positionsanzahl = 1	0.8564±0.0174 0.2041±0.0225	0.8738±0.0231 0.1782±0.0273	0.7647±0.0174 0.8538±0.1998
60% relevanter Positionsanzahl = 2	0.8518±0.0092 0.2313±0.0274	0.8611±0.0268 0.2166±0.0360	0.7558±0.0241 0.8212±0.1871
60% relevanter Positionsanzahl = 3	0.8399±0.0215 0.2462±0.0440	0.8452±0.0198 0.2270±0.0275	0.7455±0.0184 0.8647±0.2027
60% relevanter Positionsanzahl = 4	0.8318±0.0221 0.2587±0.0362	0.8433±0.0216 0.2455±0.0375	0.7372±0.0229 0.8840±0.2031
60% relevanter Positionsanzahl = 6	0.8259±0.0197 0.2746±0.0400	0.8416±0.0233 0.2543±0.0303	0.7333±0.0226 0.9449±0.2107
60% relevanter Positionsanzahl = 8	0.8296±0.0217 0.2668±0.0465	0.8449±0.0174 0.2439±0.0299	0.7321±0.0256 0.9596±0.2269
60% relevanter Positionsanzahl = 12	-	-	0.7256±0.0220 1.0417±0.2202
60% relevanter Positionsanzahl = 16	-	-	0.7385±0.0245 1.0096±0.2109
60% relevanter Positionsanzahl = 20	-	-	0.7295±0.0143 0.9994±0.1820
80% relevanter Positionsanzahl = 1	0.8670±0.0137 0.1954±0.0257	0.8797±0.0192 0.1785±0.0201	0.7917±0.0197 0.7263±0.1940
80% relevanter Positionsanzahl = 2	0.8589±0.0202 0.2128±0.0334	0.8690±0.0129 0.1928±0.0238	0.7795±0.0272 0.7590±0.2089
80% relevanter Positionsanzahl = 3	0.8465±0.0148 0.2306±0.0283	0.8615±0.0141 0.2143±0.0297	0.7641±0.0284 0.7859±0.2200
80% relevanter Positionsanzahl = 4	0.8415±0.0168 0.2434±0.0231	0.8527±0.0201 0.2374±0.0314	0.7635±0.0286 0.8051±0.2231
80% relevanter Positionsanzahl = 6	0.8374±0.0155 0.2519±0.0289	0.8527±0.0219 0.2367±0.0330	0.7577±0.0188 0.8737±0.1926
80% relevanter Positionsanzahl = 8	0.8412±0.0215 0.2528±0.0315	0.8540±0.0217 0.2380±0.0292	0.7506±0.0185 0.9000±0.1814
80% relevanter Positionsanzahl = 12	-	-	0.7494±0.0204 0.9487±0.2275
80% relevanter Positionsanzahl = 16	-	-	0.7506±0.0163 0.9410±0.2185
80% relevanter Positionsanzahl = 20	-	-	0.7455±0.0203 0.9128±0.2028
Prozentsatz der relevanten Positionsanzahl ab der zweiten Position			
20% relevanter Positionsanzahl >= 2			
iterative Schritte = 1	0.8908±0.0144 0.1598±0.0247	0.8972±0.0148 0.1467±0.0259	0.8506±0.0250 0.4583±0.1519
iterative Schritte = 2	0.8901±0.0180 0.1601±0.0248	0.8966±0.0178 0.1473±0.0261	0.8489±0.0237 0.4649±0.1671
40% relevanter Positionsanzahl >= 2			
iterative Schritte = 1	0.8861±0.0137 0.1673±0.0249	0.8940±0.0160 0.1532±0.0262	0.8474±0.0188 0.4955±0.1569
iterative Schritte = 2	0.8870±0.0113 0.1660±0.0190	0.8945±0.0157 0.1526±0.0221	0.8429±0.0207 0.4603±0.1491
60% relevanter Positionsanzahl >= 2			
iterative Schritte = 1	0.8905±0.0173 0.1595±0.0270	0.8989±0.0155 0.1454±0.0277	0.8474±0.0220 0.4628±0.1651
iterative Schritte = 2	0.8886±0.0177 0.1610±0.0247	0.8963±0.0158 0.1499±0.0291	0.8457±0.0215 0.5036±0.1628
80% relevanter Positionsanzahl >= 2			
iterative Schritte = 1	0.8889±0.0163 0.1592±0.0223	0.8982±0.0159 0.1460±0.0269	0.8506±0.0229 0.4660±0.1576
iterative Schritte = 2	0.8886±0.0158 0.1701±0.0280	0.8959±0.0152 0.1580±0.0275	0.8433±0.0280 0.4931±0.1706
Ranking ignorieren bei falscher Vorhersage erster Position			
relevante Positionsanzahl = 2			
iterative Schritte = 1	0.8745±0.0164 0.1851±0.0274	0.8803±0.0140 0.1788±0.0249	0.8276±0.0223 0.5346±0.1694
iterative Schritte = 2	0.8752±0.0142 0.1860±0.0251	0.8810±0.0157 0.1801±0.0222	0.8308±0.0238 0.5417±0.1705
iterative Schritte = 3	0.8752±0.0142 0.1860±0.0251	0.8810±0.0157 0.1808±0.0215	0.8308±0.0238 0.5417±0.1705
relevante Positionsanzahl = 3			
iterative Schritte = 1	0.8646±0.0152 0.2054±0.0249	0.8741±0.0152 0.2010±0.0226	0.8173±0.0232 0.5769±0.1883
iterative Schritte = 2	0.8630±0.0168 0.2069±0.0282	0.8729±0.0184 0.2019±0.0247	0.8167±0.0233 0.5724±0.1786
iterative Schritte = 3	0.8624±0.0164 0.2078±0.0267	0.8725±0.0177 0.2036±0.0257	0.8167±0.0233 0.5718±0.1772
relevante Positionsanzahl = 4			
iterative Schritte = 1	0.8602±0.0130 0.2085±0.0244	0.8706±0.0140 0.2026±0.0321	0.8083±0.0213 0.6199±0.1849
iterative Schritte = 2	0.8586±0.0140 0.2113±0.0275	0.8696±0.0154 0.2088±0.0317	0.8077±0.0229 0.6224±0.1869
iterative Schritte = 3	0.8586±0.0143 0.2097±0.0265	0.8702±0.0149 0.2068±0.0314	0.8090±0.0229 0.6218±0.1874
relevante Positionsanzahl = 6			
iterative Schritte = 1	0.8564±0.0172 0.2206±0.0327	0.8680±0.0190 0.2075±0.0284	0.8038±0.0254 0.6968±0.1823
iterative Schritte = 2	0.8533±0.0166 0.2231±0.0293	0.8670±0.0200 0.2114±0.0315	0.8026±0.0240 0.6859±0.2005
iterative Schritte = 3	0.8533±0.0166 0.2260±0.0284	0.8673±0.0198 0.2117±0.0324	0.8026±0.0256 0.6833±0.1984
relevante Positionsanzahl = 8			
iterative Schritte = 1	0.8564±0.0172 0.2206±0.0327	0.8680±0.0190 0.2075±0.0284	0.7885±0.0205 0.7359±0.1732
iterative Schritte = 2	0.8533±0.0166 0.2231±0.0293	0.8670±0.0200 0.2114±0.0315	0.7942±0.0212 0.7295±0.1836
iterative Schritte = 3	0.8533±0.0166 0.2260±0.0284	0.8673±0.0198 0.2117±0.0324	0.7929±0.0209 0.7295±0.1826
relevante Positionsanzahl = 12			
iterative Schritte = 1	-	-	0.7891±0.0161 0.7526±0.1848
iterative Schritte = 2	-	-	0.7872±0.0174 0.7596±0.1871
iterative Schritte = 3	-	-	0.7885±0.0181 0.7583±0.1866
relevante Positionsanzahl = 16			
iterative Schritte = 1	-	-	0.7885±0.0188 0.7615±0.1876
iterative Schritte = 2	-	-	0.7865±0.0162 0.7667±0.1863
iterative Schritte = 3	-	-	0.7872±0.0162 0.7654±0.1870
relevante Positionsanzahl = 20			
iterative Schritte = 1	-	-	0.7885±0.0188 0.7609±0.1877
iterative Schritte = 2	-	-	0.7859±0.0158 0.7641±0.1856
iterative Schritte = 3	-	-	0.7878±0.0173 0.7641±0.1857

Abbildung 27: LibLINEAR: la1s.wc, la2s.wc, wap.wc