Commentarii informaticae didacticae | 13

Artikel erschienen in:

Jörg Desel, Simone Opel, Juliane Siegeris (Hrsg.)

Hochschuldidaktik Informatik HDI 2021

9. Fachtagung des GI-Fachbereichs Informatik und Ausbildung/Didaktik der Informatik 15.–16. September 2021 in Dortmund

(Commentarii informaticae didacticae (CID) ; 13)

2023 – 299 S.
ISBN 978-3-86956-548-4
DOI https://doi.org/10.25932/publishup-56507

Empfohlene Zitation:
Karsten Weicker: Peer-Review als Katalysator im Lernprozess, In: Hochschuldidaktik Informatik HDI 2021, Jörg Desel, Simone Opel, Juliane Siegeris (Hrsg.), Potsdam, Universitätsverlag Potsdam, 2023, S. 257–277.
DOI https://doi.org/10.25932/publishup-61602

Universitätsverlag Potsdam

Peer-Review als Katalysator im Lernprozess

Karsten Weicker[1]

Abstract: Peer-Reviews werden seit geraumer Zeit in unterschiedlichen Lehrszenarien eingesetzt. In diesem Paper wird untersucht, inwieweit das Peer-Review die Auseinandersetzung mit den Inhalten eines Grundlagenmoduls in einem präsenzfreien Lehrszenario befördern kann. Dabei scheint in den Ergebnissen die Qualität der selbst erstellten Reviews einer der wichtigsten Einflussfaktoren für den Lernerfolg zu sein, während Experten-Feedback und weitere Faktoren deutlich untergeordnet erscheinen. Die Fähigkeit ausführliche Peer-Reviews zu verfassen geht einher mit dem Erwerb von fachlicher Kompetenz bzw. entsprechenden fachlichen Vorkenntnissen.

Keywords: Distanzlehre; Feedback; Diskussionskultur; Peer-Review; Lernerfolg

1 Einleitung

In der ersten Welle der Covid-19-Pandemie im Frühjahr 2020 wurden die Lehrenden der Hochschule unmittelbar in Szenarien der Fernlehre katapultiert, was eine ganze Reihe gewagter Lehrexperimente nach sich gezogen hat. Dieser Beitrag beschreibt eines dieser Experimente und die daraus gezogenen Einsichten.

Der reine Vorlesungsanteil einer Lehrveranstaltung kann leicht in ein Distanzformat überführt werden, können doch mit relativ geringen Bordmitteln Screencasts produziert und so publiziert werden, dass diese asynchron den Studierenden zur Verfügung stehen. Dies kann als ein Schritt in Richtung Flipped Classroom [We21] sogar mit einem didaktischen Mehrwert einhergehen.

1 HTWK Leipzig, Fakultät Informatik und Medien, Gustav-Freytag-Str. 42A, 04277 Leipzig, karsten.weicker@htwk-leipzig.de https://orcid.org/0000-0003-1077-2509

Distanzformate für begleitende Seminare/Übungen sind schwieriger zu gestalten, haben sie doch häufig das Ziel der studentischen Reflexion und dienen dem Abgleich des eigenen Verständnisses von Konzepten, Aufgaben und Lösungswegen mit dem der Mitstudierenden. Als Präsenzformate stehen hierfür beispielsweise Votierübungen mit Vorrechnen der Lösungen durch Studierende an der Tafel oder kollaborative Lerngruppen [We07; We20] zur Verfügung. In jedem Fall geht das Seminar einher mit einem hohen Grad an Diskussionskultur. Im Frühjahr 2020 war der Ausgangspunkt die infrastrukturell begründete Befürchtung, dass sich synchrone Formate nicht schnell realisieren lassen. Folglich wurde ein Alternativkonzept entworfen, in welchem Peer-Reviews die Rolle der Seminare übernehmen.

Dabei versuchen wir aussagekräftige Argumente bezüglich der folgenden Thesen zu bekommen:

F1 Durch den Einsatz von Peer-Reviews können bei Studierenden eine tiefe Auseinandersetzung mit Inhalten und damit auch gute Prüfungsergebnisse erreicht werden.

F2 Experten-Feedback der Lehrkräfte auf studentische Abgaben ist essenziell für den Lernprozess und die erfolgreiche Prüfungsvorbereitung.

F3 Werden die schwierigeren, spät im Semester gestellten Aufgaben bearbeitet und abgegeben, wirkt sich dies positiv auf die Prüfungsergebnisse aus.

F4 Gute Programmierkompetenzen im Sinne des Computational Thinking wirken positiv auf den Lernprozess im betrachteten Fach „Algorithmen und Datenstrukturen“.

F5 Die Erstellung ausführlicher Peer-Reviews geht einher mit dem Erwerb/-Vorhandensein von hoher Fachkompetenz.

2 Peer-Reviews in der Lehre

Peer-Reviews werden schon länger in Lehrveranstaltungen der Informatik eingesetzt. Ein starker Fokus liegt dabei auf der Programmierausbildung, wobei ganze Programme [RFT09; Li11] oder auch Concept-Maps für den Entwurf objektorientierter Programme [Tu10] begutachtet werden. Turner et al. [Tu10] berichten, dass Peer-Reviews das Lernen von Konzepten in größeren Zusammenhängen verbessern. Reily et al. [RFT09] zeigen, dass das Peer-Review von

den Studierenden angenommen wird und dass die Tätigkeit des Begutachtens die Lernleistung der Feedbackgebenden signifikant verbessert. Auch Li et al. [Li11] berichten von verbesserten Ergebnissen bei den Studierenden, die am Peer-Review teilnehmen.

Allgemein eignet sich das Peer-Review auch in der Informatik für schriftliche Arbeiten, beispielsweise in Oberseminaren [We07]. Gehringer [Ge01] hat es in acht verschiedenen Modulen der Informatik eingesetzt, in welchen die Studierenden das Konzept als hilfreich eingeschätzt haben. Liu et al. [Li01] ließen HTML-Dokumente in einem mehrstufigen Prozess begutachten und überarbeiten; sie berichten zwar von einer hohen Akzeptanz seitens der Studierenden, schließen allerdings aus ihren Daten, dass effektive Reviews keine ausreichende Vorbereitung für eine gute Note in der Prüfung darstellen. Machanick [Ma05] hat das Peer-Review für studentische Abgaben zu vorlesungsbegleitenden Tests eingesetzt und ebenfalls nur eine schwache Korrelation zur Abschlussprüfung identifiziert.

Während manche der vorstehenden Ansätze sowohl im Unterricht als auch als Hausaufgabe einsetzbar sind, seien hier noch Peer-Reviews erwähnt, die ausschließlich als Mittel der Interaktion in der Präsenzlehre dienen [NNP19]. Diesbezügliche Untersuchungen haben keine Relevanz für die Ergebnisse des vorliegenden Papers.

Unabhängig von der Informatik gibt es zahlreiche Studien und Veröffentlichungen zum Einsatz von Peer-Reviews in der Lehre. Topping [To98] berichtet, dass die Mehrheit der von ihm betrachteten Studien dem Peer-Review eine hohe Zuverlässigkeit attestiert, womit eine hohe inhaltliche Akzeptanz des Feedbacks unter den Peers gemeint ist. Das Literaturreview von Dochy et al. [DSS99] fasst 63 Studien hinsichtlich Fairness, inkonsistenter Gutachten und positiver Effekte für den Lernprozess zusammen. Unter den jüngeren Veröffentlichungen haben Nicol et al. [NTB14] den Einsatz der Technik in einem Ingenieursmodul mit Fragebögen begleitet und herausgefunden, dass das Peer-Review die Studierenden auf vielfache Weise aktiviert. Mulder et al. [MPB14] haben noch tiefgreifender durch mehrere Fragebögen untersucht, wie sich die Haltung der Studierenden durch das Peer Review verändert, wobei die Sinnhaftigkeit des Peer-Reviews in den vier betrachteten Fachgebieten teilweise radikal unterschiedlich beurteilt wurde.

3 Peer-Review statt Seminar

Im Sommersemester 2020 unterliegt die Lehrveranstaltung „Algorithmen und Datenstrukturen“ den folgenden Rahmenbedingungen:

- 4 Semesterwochenstunden (SWS) Vorlesung sind als 46 kleinteilige, thematisch abgeschlossene, vorproduzierte Screencasts aufbereitet,
- die Prüfungsvorleistung ist in der Prüfungsordnung vage als „Beleg und Präsentation“ formuliert und somit flexibel auslegbar,
- es gibt 12 Übungsblätter mit jeweils 4–5 Aufgaben (vor allem das händische Anwenden von Algorithmen, aber auch Laufzeitbetrachtungen, Analysen zu Sonderfällen und Transferaufgaben) und
- es gibt 3 Programmieraufgaben, von denen jeder Studierende mindestens eine so lösen muss, dass die unbekannten Testfälle erfüllt werden.

Anstatt des bisherigen Seminarkonzepts (Vorbereitung auf mindestens 70 % der Übungsaufgaben, mindestens einmaliges Vorrechnen an der Tafel) kommt ein asynchrones Peer-Review für die Auseinandersetzung mit unterschiedlichen Lösungen zum Einsatz.

Für jedes Übungsblatt müssen die Studierenden ihre Lösung entweder direkt am PC erstellen oder die handschriftliche Lösung per Scan/Foto digitalisieren und im Lernmanagementsystem Opal abgeben. (Opal ist eine Weiterentwicklung des Lernmanagementsystems OLAT durch das Bildungsportal Sachsen.) Nach Ablauf der Abgabefrist sind Abgaben von drei anderen Studierenden einsehbar und müssen innerhalb 72 Stunden kommentiert werden. Hierfür steht ein eher prototypischer Peer-Review-Baustein zur Verfügung, der immer nur eine andere studentische Abgabe zum Review zuordnet – erst nach deren Begutachtung kann man sich eine weitere Abgabe zuordnen lassen.

Kommentare zur eigenen Abgabe sind nur dann einsehbar, wenn selbst mindestens drei Abgaben von anderen Studierenden begutachtet und kommentiert wurden.

Jede ernsthaft bearbeitete Aufgabe eines Übungsblatts ergibt – ungeachtet der Korrektheit des Lösungsversuchs – einen Punkt für die Prüfungszulassung, wenn für das Übungsblatt drei Reviews für Abgaben von Mitstudierenden erstellt wurden. Es müssen insgesamt 34 Punkte über alle 12 Übungsblätter hinweg erreicht werden. Letztlich wurden für die Prüfungszulassung auch die Punkte einer freiwilligen Probeklausur wenige Wochen vor dem Ende der Vorlesungszeit berücksichtigt, was allerdings anfangs nicht so angekündigt

war und ein Entgegenkommen für die teilweise extremen und andersartigen Anforderungen der Pandemie darstellte.

Stichprobenartige Experten-Reviews der Lehrkräfte ergänzen die studentischen Peer-Reviews. Wegen der unterschiedlich hohen anderweitigen Lehrbelastung des involvierten Lehrpersonals hat die Hälfte der Studierenden Experten-Reviews zu einem Großteil der Aufgaben erhalten, während die andere Hälfte nur die ursprünglich geplanten, stichprobenartigen Rückmeldungen erhielt. Bei den Experten-Reviews handelt es sich in der Regel um Einzeiler pro Aufgabe, die die Korrektheit feststellen oder auf grundsätzliche Probleme in Lösungsweg oder Darstellung verweisen. Die Experten-Reviews sind ebenfalls nur dann einsehbar, wenn drei eigene Reviews erstellt werden.

Im Sommersemester 2020 stand für Rückfragen und den Austausch unter Studierenden bzw. zwischen Studierenden und Lehrkräften ein Forum im Lernmanagementsystem sowie alle ca. drei Wochen eine synchrone Fragestunde per Videokonferenz zur Verfügung. Die Prüfung konnte im Sommer 2020 als Klausur in Präsenz durchgeführt werden.

4 Studentische Einschätzung

In den Freitextkommentaren der studentischen Lehrevaluation werden die Peer-Reviews neun Mal in den Kategorien „Was hat Ihren Lernerfolg negativ beeinflusst?“ und „Was hätte in dieser Lehrveranstaltung besser gemacht werden können?“ erwähnt. Demgegenüber stehen fünf Nennungen bei „Was hat Ihren Lernerfolg positiv beeinflusst?“ und „Was hat Ihnen an dieser Lehrveranstaltung besonders gut gefallen?“.

Häufig wird der hohe Arbeitsaufwand angeführt (4×). Andere Teilnehmer bezeichnen die Reviews als „nicht hilfreich, eher nervig“, hinterfragen den Lerneffekt („Was mir bis jetzt unverständlich geblieben ist, ist der Lerneffekt dabei“, „hat mir beim Lernen nicht wirklich geholfen“, „wozu die studentische Kontrolle?“) und zeigen Unzufriedenheit mit den erhaltenen Reviews („Die Reviews fand ich bis zum Schluss eher mittelmäßig, da die Bearbeitung meist eher schlecht ausfiel und man als Leistungsstärkerer selten selber etwas lernen konnte“, „Reviews hatten wahrscheinlich nicht genau den Effekt, den sie haben hätten sollten. Viele schrieben nur ’habe alles genauso’, obwohl eine Aufgabe komplett fehlte“).

Zwei Kommentaren wägen Nutzen und hohen Aufwand ab („mit den Reviews weiß ich nicht recht, gab zwar schon Fälle, wo ich es hilfreich fand, aber oft war es dann auch noch mehr Arbeit“ vs. „viel Arbeit, aber auch wirklich effektiv und wirkungsvoll“).

Zwei positive Anmerkungen zum Peer-Review würdigen die Auseinandersetzung mit anderen Abgaben („Die Peer-Reviews sind sehr hilfreich, da man sich noch einmal mit den Aufgaben beschäftigt und Lösungen von anderen sehen kann“, „das Peer-Review hat bei mir manchen Aha-Effekt ausgelöst“).

Bezüglich der Durchschnittswerte der Lehrevaluation bewegt sich die Lehrveranstaltung im Bereich der beiden vorherigen Evaluationen (Tabelle 1). Es haben sich 42 von 106 Teilnehmern beteiligt. Ferner gibt die Mehrheit an, dass in dieser Lehrveranstaltung der Lernerfolg durch die Umstellung auf das digitale Lehrformat positiv beeinflusst wurde (20 % „sehr positiv“, 37,1 % „eher positiv“). Nur 14,3 % sehen die Umstellung „eher negativ“, die restlichen 28,6 % nahmen keine Auswirkung auf den Lernerfolg wahr.

Tab. 1: Ergebnisse der studentischen Lehrevaluation – 2020 mit Peer-Reviews und 2016/19 mit der klassischen Durchführung der Seminare/Übungen.

	2016	**2019**	**2020**
Insgesamt bewerte ich diese Vorlesung mit der Note …	1,8	1,5	1,5
Diese Lehrveranstaltung fördert mein Interesse an dem Thema (1 = stimme voll zu, 5 = stimme gar nicht zu)	2,0	1,9	1,9
Ich habe in dieser Vorlesung ein tiefes Verständnis für den Stoff gewonnen (1 = stimme voll zu, 5 = stimme gar nicht zu)	2,0	1,8	1,9

5 Daten & Methodik

Anhand der anonymisierten Daten aus der Lehrveranstaltung „Algorithmen und Datenstrukturen“ wird im Weiteren überprüft, als wie stichhaltig sich die Thesen F1–F5 in der beschriebenen Lehrsituation erwiesen haben.

5.1 Erhobene Daten

Für jeden Teilnehmer liegen die folgenden Daten vor, welche die Basis für die Untersuchungen in den Abschnitten 6 und 7 liefern.

P: Punktzahl der Prüfungszulassung, die durch Übungsaufgaben und Reviews erreicht wurde [5–58, Median: 37]

ΔB: Spanne der bearbeiteten Übungsblätter $\#_{letztes} - \#_{erstes} + 1$ [1–12, Median: 10]

#oR: Anzahl der selbst bearbeiteten Blätter, für die keine Reviews anderer Abgaben erstellt wurden [0–5, Median: 1]

#R: Anzahl der erstellten Reviews [3–58, Median: 26,5]

QR: Durchschnittswert der Qualität der erstellten Reviews – diese wurden im Nachhinein manuell in drei Klassen eingeteilt (1 = oberflächlich, 2 = akzeptabel, 3 = spezifisch/detailliert) [1–2,424, Median: 1,418]

E: Teilnehmer war in der Gruppe mit Experten-Feedback [0/1]

Pr✓: Nummer der ersten erfolgreich bewältigten Programmieraufgabe [1–3, Median: 1]

#Pr: Anzahl der bearbeiteten Programmieraufgaben [1–3, Median: 1]

Z: Erhalt der Prüfungszulassung [0/1]

N: Note in der Prüfung, falls teilgenommen [1,0; 1,3; 1,7;...; 4,0; 5,0; Median: 2,0]

Im Abschnitt 8 werden diese Daten zusätzlich in Beziehung gesetzt zur Qualität der studentischen Abgaben im Übungsbetrieb – einer Größe, die nicht systematisch während des Semesters erhoben wurde. Daher wird dort auf eine Maßzahl zurückgegriffen, die für Studierende mit Experten-Feedback ($E = 1$) bezüglich der Übungsblätter 3–6 und 8 vorliegt:

QA: Qualität der Abgaben als Punkte von fünf Übungsblättern (pro Aufgabe: korrekt = 1 Punkt, kleinere Mängel = 0,5, falsch = 0) [2 – 24,5, Median: 17,5]

Insgesamt liegen Daten von 107 Studierenden vor. In die meisten Untersuchungen gehen allerdings nur die Daten der 91 Studierenden ein, die direkt im Sommersemester 2020 die Prüfungszulassung erworben und an der ersten Prüfung nach der Lehrveranstaltung teilgenommen haben, da nur von diesen Studierenden eine Note im Modul vorliegt. Die Informationen zur Qualität der Abgaben (QA) liegen nur von 47 der 91 Studierenden vor. Vollständig

Unberücksichtigt bleiben 26 Prüflinge mit einer älteren Prüfungszulassung, da sie nicht den Peer-Review absolviert haben.

5.2 Methodik

Diese Untersuchung zielt wesentlich darauf ab, festzustellen, ob sich in den Daten Zusammenhänge zwischen den verschiedenen Faktoren des Lehr-/Lernverhaltens und dem Abschluss des Moduls erkennen lassen. Dabei wird als Indikator für den Erfolg die Note der Präsenzklausur als objektives Kriterium herangezogen – auch wenn Prüfungsangst und die beschränkte Kompetenzmessung durch Klausuren seine Aussagekraft mindern.

Die Faktoren des Lehr-/Lernverhaltens ergeben sich oft direkt aus den erhobenen Metriken. So kann beispielsweise aus einer großen Spanne der bearbeiteten Übungsblätter (ΔB) in Verbindung mit einer hohen Punktzahl (P) ein hoher Grad an Disziplin oder ein großes Interesse am Fach abgeleitet werden. Wurden jedoch zu den selbst bearbeiteten Übungsblättern keine Reviews (#oR) angefertigt, spricht dies für eine gewisse Nachlässigkeit, da weder Feedback noch Punkte resultieren. Ein anderes Beispiel sind die Programmieraufgaben: Mehrere Versuche (#Pr) und später Erfolg (Pr✓) legen Probleme im Computational Thinking oder der Programmierkompetenz nahe. Auf diese und weitere Faktoren wird im Rahmen der Analyse genauer eingegangen.

Grundsätzlich ist bei der Auswahl der Indikatoren maßgebend, dass sie keine direkte Aussage zur Korrektheit typischer Prüfungs-/Übungsaufgaben enthalten. Dies soll die objektive Beurteilung der verschiedenen Lehr-/Lernfaktoren und des didaktischen Konzepts ermöglichen und verfälschende Tendenzen in den Indikatoren verringern. Aus diesem Grund bleiben auch die Punkte aus dem Ergebnis der Probeklausur unberücksichtigt und der Indikator P enthält nur die Punkte aus den Peer-Reviews.

Um auch komplexe Einflüsse aus der Kombination mehrerer Indikatoren zu erfassen, kommen neben klassischen Methoden der Statistik auch zwei Algorithmen des Machine Learning zum Einsatz.

6 Analyse der Daten

Zum besseren Verständnis der Datenbasis und als erster Analyseschritt werden die Daten mit einfachen statistischen Methoden sowie einem Clustering-Algorithmus untersucht.

6.1 Korrelationen

Tabelle 2 zeigt die Korrelationswerte zwischen den verschiedenen Rohdaten. So besteht beispielsweise eine moderate Korrelation zwischen der Anzahl der bearbeiteten Programmieraufgaben und der ersten erfolgreichen Abgabe.

Ein großes Korrelations-Cluster mit vornehmlich moderaten Korrelationen spiegelt die enge Verzahnung der Anzahl der Reviews #R, den spät bearbeiteten Übungsblättern ΔB, der erreichten Gesamtpunktzahl P und der Zulassung Z wider. Aus diesen Attributen weist allerdings einzig die Punktzahl eine schwache (negative) Korrelation zur Note in der Prüfung auf, d. h. große Punktzahlen gehen einher mit kleinen, sprich: besseren, Notenwerten. Die einzige weitere nennenswerte, allerdings ebenfalls schwache negative Korrelation besteht zwischen der Note und der Qualität der Reviews – dies stützt schwach die These F1.

6.2 Daten-Cluster

Um stärker das Zusammenwirken aller erfassten Attribute zu berücksichtigen, haben wir mit dem Expectation-Maximization-(EM-)Algorithmus [RN20, S. 737 ff.] alle Studierenden in Cluster eingeteilt und die Studierenden mit ähnlichen Werten in den Rohdaten zusammengefasst. Dies soll einen Einblick in typische Vorgehensweisen der Studierenden sowie die resultierenden Noten in der Prüfung geben. Tabelle 3 zeigt die sieben ermittelten Cluster mit ihren Zentroiden und der Notenverteilung.

Die meisten Noten „sehr gut“ gehören zum Cluster 4 und gehen einher mit vielen bearbeiteten Übungsblättern, Punkten und Reviews; die Reviews sind hochwertig und werden zuverlässig erbracht; auch die Programmierfertigkeit ist gut.

Tab. 2: Korrelationswerte zwischen den Attributen der erfassten Rohdaten

	N	**E**	**Z**	**#R**	**QR**	**P**	**ΔB**	**#oR**	**Pr✓**	**#Pr**
N	1									
E	0,036	1								
Z	Div/0	0,141	1							
#R	−0,195	−0,085	**0,530**	1						
QR	−0,419	−0,166	0,141	0,246	1					
P	−0,446	−0,039	**0,620**	**0,874**	0,366	1				
ΔB	−0,086	0,143	0,589	**0,680**	0,134	**0,734**	1			
#oR	0,177	0,236	0,018	−0,500	−0,354	−0,397	0,076	1		
Pr✓	0,149	−0,001	Div/0	−0,029	−0,281	−0,165	0,043	0,052	1	
#Pr	0,003	0,140	0,115	0,220	−0,065	0,176	0,198	−0,029	**0,617**	1

Tab. 3: Cluster der Rohdaten als Ergebnis des EM-Algorithmus. Auffallend große (und im unteren Teil auch kleine) Werte sind in jeder Zeile markiert

	1	**2**	**3**	**4**	**5**	**6**	**7**
Note: sehr gut	1,186	1,803	2,620	**12,381**	1,185	**6,387**	2,437
gut	1,094	**15,219**	**14,923**	2,368	**4,113**	2,022	3,261
befriedigend	1,554	1,069	1,243	1,847	**5,999**	1,076	**8,214**
ausreichend	**8,267**	1,080	2,042	1,241	2,584	**3,287**	1,501
ungenügend	1,089	1,040	2,183	1,114	**4,028**	**3,013**	1,534
Anteil E	0,165	**0,919**	**0,097**	0,229	0,884	**0,906**	0,514
#R	27,159	28,401	26,393	**30,425**	24,436	**20,287**	27,672
QR	1,362	1,601	1,640	**1,858**	**1,222**	**1,250**	1,368
P	34,527	**43,917**	39,289	**47,308**	34,339	**32,576**	40,690
ΔB	9,743	10,277	**9,139**	**10,789**	9,532	10,055	9,810
#oR	1,622	0,674	0,649	**0,288**	1,058	**2,917**	**0,389**
Pr✓	2,165	1,334	1,549	**1,173**	**1,076**	**2,356**	**2,881**
#Pr	1,432	1,660	1,616	1,352	**1,000**	**2,284**	**2,723**

Cluster 2 und 3 repräsentieren vornehmlich Studierende mit der Note „gut“. Beide Cluster haben ein auffallend ähnliches Profil mit hoher Punktzahl und eher ausführlichen Reviews – der einzige ausgesprochen markante, große Unterschied ist das Experten-Feedback. Dies ist ein starker Hinweis darauf, dass F2 falsch sein könnte und der Einfluss des Experten-Feedbacks auf die Note nur marginal ist.

Cluster 7 vereinigt Studierende, für welche die Programmieraufgabe eine große Herausforderung darstellt, die aber dennoch bei den Reviews Zuverlässigkeit bewiesen haben und zumeist mit der Prüfungsnote „befriedigend“ belohnt wurden.

Spannend ist das Cluster 6, dessen Studierende in der Programmieraufgabe eine Herausforderung sahen und viel Experten-Feedback erhielten, aber nur das Notwendigste bzgl. Punkten, Reviews und Reviewqualität geleistet haben. Die resultierenden Noten sind entweder „sehr gut“ oder „ausreichend“/„ungenügend“.

Studierende in Cluster 5 sind gute Programmierer, aber eher oberflächliche Reviewer; dafür erzielten sie vornehmlich Noten im Spektrum „gut“ bis „ungenügend“.

Cluster 1 wird durch die vorherrschenden Note „ausreichend“, wenig Experten-Feedback, viele versäumte Reviews und späte Erledigung der Programmieraufgabe bestimmt.

6.3 Hauptkomponentenanalyse

Schon bei der Betrachtung der Korrelationen hat sich angedeutet, dass verschiedene Attribute einen starken Zusammenhang aufweisen. Daher wird hier nochmals tiefgehender untersucht, welche kombinierten Vektoren die Punktwolke ohne Betrachtung der Prüfungsnote aufspannen. Tabelle 4 zeigt die Ergebnisse.

So können über die fünf wichtigsten Hauptkomponenten definierende Verhaltensweisen in der Menge der Studierenden mit abnehmender Bedeutung aufgezeigt werden:

Nachlässigkeit (PC1): wenig Punkte, wenig Reviews und viele fehlende Reviews

Mängel im Computational Thinking (PC2): Erfolg bei später Programmieraufgabe, mehrere Programmieraufgaben

Tab. 4: Die fünf wichtigsten Hauptkomponenten in den Daten ohne Note und Zulassung.

	PC1	**PC2**	**PC3**	**PC4**	**PC5**
Experten-Feedback E	0,1511	0,1951	0,6538	0,2657	−0,6247
Anzahl Reviews #R	−0,5265	0,0882	0,0606	0,2506	0,2008
Qualität Reviews QR	−0,3089	−0,2742	−0,1825	−0,6563	−0,5484
Punkte P	−0,5511	0,0118	0,1525	0,0418	0,0029
Spanne Blätter ΔB	−0,2886	0,1972	0,5668	−0,4287	0,3593
Blätter ohne Review #oR	**0,4497**	0,0335	0,2763	−0,4702	0,2236
erfolgr. Prog. Pr✓	0,0711	0,6487	−0,2768	−0,1406	0,0917
Anzahl Prog. #Pr	−0,1013	0,6465	−0,1952	−0,0973	−0,2848
Wichtigkeit (sdv)	1,6822	1,3379	1,1274	0,8906	0,8105

Wertschätzung von Experten-Feedback (PC3): regelmäßiges Experten-Feedback, auch späte Blätter bearbeitet

Reviews ohne Tiefgang (PC4): niedrige Reviewqualität, wenig fehlende Reviews, wenig bearbeitete Blätter

Suchende (PC5): kein regelmäßiges Experten-Feedback, geringe Reviewqualität, auch späte Blätter bearbeitet

Erstaunlicherweise ist die Reviewqualität im Gegensatz zur hohen Korrelation zur Note ein untergeordneter bestimmender Faktor in den restlichen Daten (erst in PC4 und PC5).

Möchte man den typischen Verhaltensweisen Cluster zuordnen, so repräsentiert

- für PC1 Cluster 6 das untere Ende und Cluster 4 das obere Ende des Spektrums,
- für PC2 Cluster 7 unten und Cluster 5 oben,
- für PC3 Cluster 2/6 unten und Cluster 3 oben,
- für PC4 Cluster 7 unten und (jeweils mit Abstrichen) Cluster 3/4 oben sowie
- für PC5 kein Cluster komplett die Enden – am ehesten Cluster 1 das untere Ende.

7 Vorhersagekraft des Peer-Reviews

Die reine Datenanalyse liefert bereits einige Hinweise, dass F1 positiv beantwortet werden kann. Um insbesondere auch für F2, F3 und F4 komplexere Abhängigkeiten der Note von den verschiedenen Eingangsdaten zu analysieren, soll in diesem Abschnitt über zwei Machine-Learning-Modelle erfasst werden, wie genau eine wenigstens gute Note ($\leq 2{,}3$) prognostiziert werden kann und welche Abhängigkeiten die Modelle dafür benutzen.

Abbildung 1 zeigt einen Entscheidungsbaum, welcher 82 der 91 Studierenden richtig einordnet. Der Entscheidungsbaum wurde mit dem C4.5-Algorithmus in der Implementation J48graft pruned [We99] schrittweise durch Einfügen von Entscheidungsknoten erstellt und durch Ausdünnen (pruning) schlank gehalten. Die Prognosefähigkeit kann durch die 10-fache Kreuzvalidierung ermessen werden, welche mit 10 ähnlichen Modellen auf Basis von jeweils 90 % der Daten eine korrekte Klassifikation von 68,13 % erreicht.

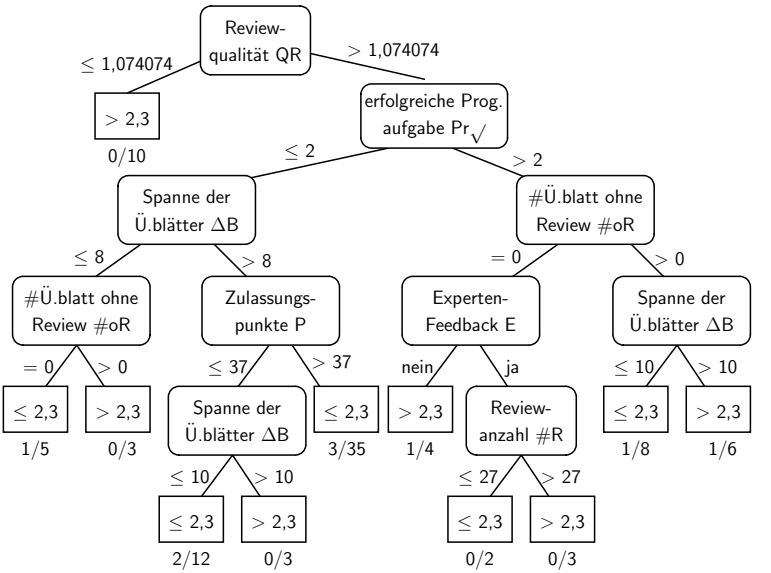

Abb. 1: Entscheidungsbaum (J48graft pruned) zur Prognose, ob wenigstens eine Note 2,3 erreicht wird. An jedem Blatt ist die #Fehlklassifikationen/#Gesamtklassifikationen annotiert.

Das Modell kann durch Reviewqualität und die frühe Bearbeitung der Programmieraufgabe bereits 10 schlechte und 46 gute Noten richtig zuordnen. Im

linken Teilbaum unterhalb der zweiten Ebene können über die Bedingungen „wenig Übungsblätter, aber fehlende Reviews“ sowie „viele Übungsblätter bei wenig Punkten“ noch sechs der insgesamt 12 schlechten Noten im Teilbaum richtig klassifiziert werden. Der rechte Teilbaum unterhalb der zweiten Ebene ist schwerlich interpretierbar, lässt aber bezüglich des Experten-Feedbacks vermuten, dass gewissenhafte Studierende (Ebene 3) mit Schwierigkeiten im Computational Thinking (Ebene 2) von Experten-Feedback ggf. mehr profitieren können als von eigenen Reviews.

Ein mit dem Alternating Decision Tree (AD-Tree) erstelltes Modell in Abbildung 2 kann zwar lediglich 76 der 91 Klausurergebnisse richtig einordnen, hat aber in der Kreuzvalidierung mit 69,23 % Trefferquote eine leicht bessere Vorhersagekraft.

```
(1)reviewquality < 1,079: 1,315
(1)reviewquality >= 1,079: -0,175
|  (9)reviewquality < 1,225: -0,71
|  (9)reviewquality >= 1,225: 0,12
|  |  (10)reviewquality < 1,289: 0,728
|  |  (10)reviewquality >= 1,289: -0,129
(2)zulassungspunkte < 37,5: 0,426
(2)zulassungspunkte >= 37,5: -0,331
|  (5)reviewzahl < 27,5: -0,591
|  |  (7)reviewquality < 1,155: 0,703
|  |  (7)reviewquality >= 1,155: -0,836
|  (5)reviewzahl >= 27,5: 0,155
|  |  (6)zulassungspunkte < 43,5: 0,901
|  |  (6)zulassungspunkte >= 43,5:-0,363
(3)ersteerfolgreicheproga < 2,5: -0,26
(3)ersteerfolgreicheproga >= 2,5: 0,477
|  (8)reviewquality < 1,328: 0,539
|  (8)reviewquality >= 1,328: -0,26
(4)reviewquality < 1,877: 0,135
(4)reviewquality >= 1,877: -0,868
```

Abb. 2: AD-Tree für die Vorhersage der Note $\leq 2{,}3$. Summen kleiner als Schwellwert $-0{,}253$ entsprechen einer besseren Note .

Bemerkenswert ist an diesem Modell, dass es ausschließlich die Qualität der Reviews, die Anzahl der Reviews, die erreichten Punkte bei der Zulassung und die erste erfolgreich bewältigte Programmieraufgabe benutzt. Dies unterstreicht den Einfluss, den eine gute Auseinandersetzung mit anderen Lösungen auf den Erfolg haben kann. Die Einflussfaktoren entsprechen vornehmlich den Hauptkomponenten PC1, PC4 und PC2.

Das Ergebnis des Modells ist in tabellarischer Form in Tabelle 5 aufbereitet. Überragende Reviewqualität bzw. schlechte Reviewqualität (in Verbindung mit gutem Computational Thinking) ist dort ein guter Gradmesser für eine gute Note. Im mittleren Bereich der Reviewqualität scheint gutes Computational Thinking positiv und das Weglassen vieler Aufgaben (viele Reviews, aber mittelmäßig viele bis wenig Punkte) negativ zu wirken.

Tab. 5: Einordnung der Studierenden im AD-Tree, wobei + für die Noten „sehr gut“ und „gut“ steht und – für eine schlechtere Note. Bei den Zulassungspunkten unterscheiden wir die Bereiche hoch (↑), mittel (→) und niedrig (↓), bei der Anzahl der Reviews niedrig (⇓) und hoch (⇑).

	Prog.aufgabe	**früh**				**spät**			
	Punkte	↑	→	↑ / →	↓	↑	→	↑ / →	↓
	#Reviews	⇑	⇑	⇓	⇑/⇓	⇑	⇑	⇓	⇑/⇓
Reviewqualität	$< 1{,}079$	–	–	–	–	–	–	–	–
	$1{,}079 - 1{,}155$	+	+	+	+	+	–	–	–
	$1{,}155 - 1{,}225$	+	+	+	+	+	–	+	–
	$1{,}225 - 1{,}289$	–	–	+	–	–	–	–	–
	$1{,}289 - 1{,}328$	+	–	+	–	–	–	+	–
	$1{,}328 - 1{,}877$	+	–	+	–	+	–	+	–
	$\geq 1{,}877$	+	+	+	+	+	–	+	+

8 Unabhängigkeit der Review-Fähigkeit

Der Einfluss einzelner didaktischer Maßnahmen auf den Lernprozess lässt sich naturgemäß nur schwer isolieren. So stellt sich auch im Kontext dieser Untersuchung die Frage, ob nicht einfach die Review-Fähigkeit an der fachlichen Kompetenz der Studierenden hängt. Anders formuliert: Falls nur die sowieso guten Studierenden in der Lage sind, ausführliche Reviews zu schreiben, könnte ein hohe Korrelation zwischen der Qualität des Reviews und der Note in der Klausur fälschlicherweise zugunsten der Methode des Peer-Reviews interpretiert werden.

Dies lässt sich im ausschließlichen Kontext der betrachteten Lehrveranstaltung nur schwer beantworten, da sich der Zugewinn an Wissen und Kompetenzen in der Veranstaltung kaum von grundsätzlichen kognitiven Vorteilen trennen lässt. Daher ist die These F5 so formuliert, dass sie bewusst beide Richtungen des Einflusses umfasst. Trotzdem soll im Weiteren genauer differenziert werden. Es sei an dieser Stelle auch daran erinnert, dass das Peer-Review keine klassische Korrektur von Lösungsaufgaben darstellt, die in jedem Fall an Fachwissen gekoppelt wäre. Stattdessen soll ein Abgleich mit der eigenen Abgabe stattfinden („habe ich auch so“, „ist gut erklärt“, „ich verstehe deine Argumentation nicht“ oder „in dem Teilschritt ist ein Fehler enthalten“), welcher deutlich niederschwelliger durchführbar ist.

Die Unabhängigkeit der Review-Fähigkeit wird anhand der im Experten-Review vorhandenen Beurteilung der studentischen Abgaben (QA) begutachtet, die für 47 Studierende vorliegt.

Tabelle 6 zeigt die Korrelationen der Review- und Abgabequalität untereinander und mit der Note. Dabei wird deutlich, dass die höchste Korrelation zwischen der Abgabequalität und der Note vorliegt. Zwischen Reviewqualität und Abgabequalität kann lediglich eine moderate Korrelation beobachtet werden.

Tab. 6: Paarweise Korrelationen der Klausurnote, Review- und Abgabequalität

	Note	**Reviewqualität**
Reviewqualität	−0,480	
Abgabequalität	−0,690	−0,545

Um einen besseren Einblick in die Beziehung der beiden Qualitätsattribute zueinander zu bekommen, sind in Abbildung 3 die beiden Werte gegeneinander abgetragen und zusätzlich durch Symbole den Notenwerten zugeordnet. Der Abbildung lassen sich folgende Beobachtungen entnehmen, die wir nachfolgend genauer interpretieren:

1. Es gibt keine Studierenden mit hoher Reviewqualität und niedriger Qualität der eigenen Abgaben.
2. Für Studierende mit hoher Abgabequalität scheint die Notenverteilung zwischen „gut“ und „sehr gut“ unabhängig von der Reviewqualität zu sein.

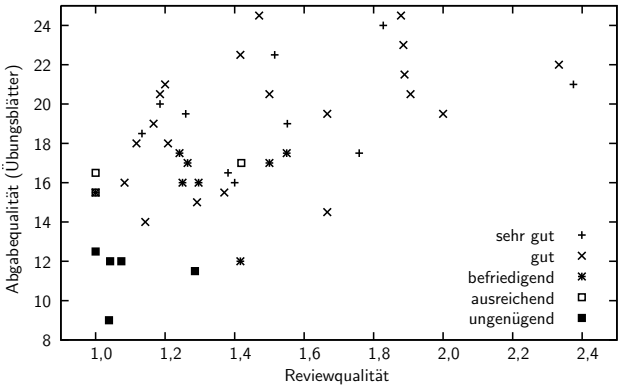

Abb. 3: Qualität der Reviews und Abgaben für die 47 betrachteten Studierenden.

3. Für Studierende mit geringerer Abgabequalität scheint es ansatzweise ein Notengefälle entlang der Reviewqualität zu geben.

Dabei stützt (1) eindeutig die These F5, wobei die Kausalität offen bleibt – in jedem Fall fertigen schlechtere Studierende keine ausführlichen Reviews an, ohne sich dadurch auch fachlich zu verbessern.

Der unterschiedliche Einfluss der Reviewqualität in (2) und (3) wurde für die entsprechenden Teilmengen als Korrelationen in Tabelle 7 genauer analysiert. Dies zeigt, dass sich die Reviewqualität für die schlechteren Studierenden mit einer moderaten Korrelation zur Note deutlich vom Wert der besseren Studierenden abhebt. Dazu kommt noch ein wesentlich geringerer Korrelationswert zwischen Abgabe- und Reviewqualität. Dies legt die Schlussfolgerung nahe, dass die Reviewqualität durchaus ein Faktor im Lernprozess darstellt und auch die fachliche Entwicklung des Feedback-Gebenden unterstützt.

9 Ergebnisse

Die These **F1**, dass das Peer-Review in der Lehre ein Mechanismus für die tiefe Auseinandersetzung mit Inhalten sein kann, wird verschiedentlich gestützt: Die Qualität der Reviews (QR) ist die einzige nicht-triviale Korrelation zur Note, die Notenverteilung in den drei Clustern mit den qualitativ hochwertigsten Reviews weist überdurchschnittlich viele Noten „gut“ und „sehr gut“ auf und die Prognosemodelle zeigen, dass durchgängig schlechte Reviewqualität mit

Tab. 7: Korrelationen in Abhängigkeit von der Abgabequalität

(a) Teilnehmer mit Abgabequalität ≥ 18

	Note	**Reviewqualität**
Reviewqualität	−0,048	
Abgabequalität	0,042	0,447

(b) Teilnehmer mit Abgabequalität < 18

	Note	**Reviewqualität**
Reviewqualität	−0,542	
Abgabequalität	−0,506	0,354

schlechten Noten einhergeht. Während die Hauptkomponentenanalyse zeigt, dass die Qualität der Reviews wenig mit anderen Merkmalen zusammenhängt, zeigt das Modell des AD-Trees, dass QR kein alleiniger Indikator sein kann, da sie beispielsweise bei vielen Reviews in Verknüpfung mit mäßiger Punktzahl eher negativ zu wirken scheint.

Die These **F2**, dass sich Experten-Feedback (E) auf studentische Abgaben positiv auswirkt, bleibt ungeklärt. So zeigt zwar das Prognosemodell J48, dass es Situationen gibt, in denen Studierende davon profitieren können, der Faktor spielt jedoch keine Rolle im AD-Tree-Modell und in den Korrelationen. Auch bei der Clusteranalyse belegen die Cluster 2 und 3 sowie Cluster 6 einen höchstens untergeordneten Einfluss von (E).

Der positive Einfluss von der Bearbeitung später Übungsaufgaben (ΔB, These **F3**) scheint in schwacher Form gegeben zu sein: So zeigen die Cluster mit späten Übungsaufgaben (2, 4, 6) einen hohen Anteil sehr guter und guter Noten, auch wenn das J48-Modell eine starke Abhängigkeit von anderen Faktoren nahe legt. Die Korrelationen können diesen Einfluss zwar nicht direkt belegen, aber ΔB ist mit P stark korreliert und P hat im AD-Tree-Modell einen hohen Einfluss bei ordentlicher Reviewqualität QR. Das unscharfe Bild rührt vermutlich auch von den gemischten Motivationen, die letzten Übungsaufgaben zu bearbeiten: Interesse am Lehrstoff vs. letzte Möglichkeit zur Prüfungszulassung.

Selbst in einem Fach wie „Algorithmen und Datenstrukturen“ scheint der Einfluss von Vorkenntnissen im Computational Thinking (Pr✓, These **F4**) schwächer ausgeprägt zu sein, als üblicherweise angenommen wird: Es liegt

keine Korrelation zur Note vor und auch Cluster 5 zeigt, dass die Studierenden mit gutem Computational Thinking und oberflächlichen Reviews (QR) vornehmlich Noten im mittelmäßigen Bereich und die höchsten Durchfallquoten aufweisen. Die unterschiedliche Anzahl der „+“ im linken und rechten Block von Tabelle 5 zeigt hingegen deutlich, dass es einen Einfluss gibt, wobei die Struktur des J48-Baums mit Pr✓ in der zweiten Ebene zeigt, dass der Faktor stark mit anderen Faktoren interagiert.

Abschnitt 8 legt nahe, dass die These **F5** wahr ist – allerdings muss man dies differenziert betrachten. Studierende mit guten Vorkenntnissen ziehen kaum oder wenig Vorteile aus dem Peer-Review, aber die restlichen Studierenden können davon profitieren und sich fachlich entsprechend weiterentwickeln.

10 Fazit und Ausblick

Die Analyse zeigt im Kontext des pandemiebedingt aus der Not geborenen Lehrexperiments, dass ein tiefgründig ausgeführtes, studentisches Peer-Review einen großen Einflussfaktor auf den Lernprozess bildet. Allerdings scheint der A-Priori-Wissenstand mitzubestimmen, welche Studierenden davon profitieren können, wobei der Einfluss bei den besseren Studierenden eher geringer zu sein scheint. Das Experten-Feedback und die Bearbeitung von Übungsaufgaben gegen Ende des Semesters scheinen als Faktoren deutlich untergeordnet zu sein. Zumindest im Falle der Lehrveranstaltung „Algorithmen und Datenstrukturen“ ist das Computational Thinking ein – wenn auch nicht dominierender – Einflussfakor. Inwieweit die Technik des Peer-Reviews generell auch in anderen Lehrsituationen hilfreich sein kann, muss in weiteren Untersuchungen analysiert werden.

Die Analyse wurde mit R^2 und WEKA [Ha09] durchgeführt.

2 R: A Language and Environment for Statistical Computing. https://www.r-project.org/, letzter Zugriff: 16.01.2023

Literaturverzeichnis

[DSS99] Dochy, F.; Segers, M.; Sluijsmans, D.: The use of self-, peer and co-assessment in higher education: A review. Studies in Higher Education 24/3, S. 331–350, 1999.

[Ge01] Gehringer, E. F.: Electronic peer review and peer grading in computer-science courses. In (Walker, H. M.; Mccauley, R. A.; Gersting, J. L.; Russell, I., Hrsg.): Proceedings of the thirty-second SIGCSE technical symposium on Computer Science Education (SIGCSE '01). ACM, New York, S. 139–143, Feb. 2001.

[Ha09] Hall, M. A.; Frank, E.; Holmes, G.; Pfahringer, B.; Reutemann, P.; Witten, I. H.: The WEKA data mining software: an update. SIGKDD Explorations Newsletter 11/1, S. 10–18, 16. Nov. 2009.

[Li01] Liu, E. Z.-F.; Lin, S. S. J.; Chiu, C.-H.; Yuan, S.-M.: Web-based peer review: the learner as both adapter and reviewer. IEEE Transactions on Education 44/3, S. 246–251, 2001.

[Li11] Li, C.; Dong, Z.; Untch, R. H.; Chasteen, M.; Reale, N.: PeerSpace – An Online Collaborative Learning Environment for Computer Science Students. In: 2011 IEEE 11th International Conference on Advanced Learning Technologies. IEEE, Washington, D. C., S. 409–411, Juli 2011.

[Ma05] Machanick, P.: Peer Assessment for Action Learning of Data Structures and Algorithms. In (Young, A.; Tolhurst, D., Hrsg.): Proceedings of the 7th Australasian Conference on Computing Education (ACE '05). Australian Computer Society, Darlinghurst, S. 73–82, 2005.

[MPB14] Mulder, R. A.; Pearce, J. M.; Baik, C.: Peer review in higher education: Student perceptions before and after participation. Active Learning in Higher Education 15/2, S. 157–171, 2014.

[NNP19] Nazir, S.; Naicken, S.; Paterson, J. H.: Teaching Data Structures through Group Based Collaborative Peer Interactions. In (Rahimi, E.; Stikkolorum, D. R., Hrsg.): Proceedings of the 8th Computer Science Education Research Conference (CSERC '19). ACM, New York, S. 98–103, 2019.

[NTB14] Nicol, D.; Thomson, A.; Breslin, C.: Rethinking feedback practices in higher education: a peer review perspective. 39/1, S. 102–122, 2014.

[RFT09] Reily, K.; Finnerty, P. L.; Terveen, L. G.: Two peers are better than one: aggregating peer reviews for computing assignments is surprisingly accurate. In (Teasley, S. D.; Havn, E. C.; Prinz, W.; Lutters, W. G., Hrsg.): Proceedings of the ACM 2009 International Conference on Supporting Group Work (GROUP '09). ACM, New York, S. 115–124, 2009.

[RN20] Russell, S. J.; Norvig, P. Pearson, Hoboken, 2020.

[To98] Topping, K.: Peer Assessment Between Students in Colleges and Universities. Review of Educational Research 68/3, S. 249–276, 1998.

[Tu10] Turner, S. A.; Pérez-Quiñones, M. A.; Edwards, S. H.; Chase, J.: Peer review in CS2: conceptual learning. In (Lewandowski, G.; Wolfman, S. A.; Cortina, T. J.; Walker, E. L., Hrsg.): Proceedings of the 41st ACM Technical Symposium on Computer Science Education (SIGCSE '10). ACM, New York, S. 331–335, 2010.

[We07] Weicker, N.: Zielorientierte Didaktik der Informatik – Kompetenzvermittlung bei engen Zeitvorgaben. In (Schubert, S. E., Hrsg.): Didaktik der Informatik in Theorie und Praxis – INFOS 2007 – 12. GI-Fachtagung Informatik und Schule. Deutsche Gesellschaft für Informatik e.V., Bonn, S. 337–348, 2007.

[We20] Weicker, K.: Teaching cooperative problem solving. In (Mottock, J., Hrsg.): Proceedings of the 4th European Conference on Software Engineering Education (ECSEE '20). ACM, New York, S. 6–11, 2020.

[We21] Werner, J.; Ebel, C.; Spannagel, C.; Bayer, S., Hrsg. 3. Aufl., Gütersloh: Verlag Bertelsmann Stiftung, 2021.

[We99] Webb, G. I.: Decision Tree Grafting From the All Tests But One Partition. In (Dean, T., Hrsg.): The problem of missing values in decision tree grafting. Morgan Kaufmann, San Francisco, S. 702–707, 1999.