

Automatische Generierung von Übungsgruppen auf Basis der Nutzung von Online-Ressourcen

Laura Steinert, Tobias Hecking, H. Ulrich Hoppe

Abteilung für Informatik und angewandte Kognitionswissenschaft
Universität Duisburg-Essen
Lotharstr. 63/65, 47048 Duisburg
{steinert, hecking, hoppe}@collide.info

Abstract: Die Arbeit in Kleingruppen ist ein wichtiger Bestandteil aktueller didaktischer Ansätze und zentraler Gegenstand der Forschung zu CSCL. In diesem Kontext kommt der Gruppenzusammenstellung ein entscheidender Einfluss auf die Qualität der Gruppenarbeit und des Lernens in der Gruppe zu. Um in kooperativen Lernumgebungen möglichst gut arbeitende Gruppen bilden zu können, sind Einschätzungen zu den einzelnen Lernenden erforderlich. Dieser Beitrag stellt eine Methode vor, Gruppen auf Basis der Nutzung von Online-Ressourcen derart zu bilden, dass jede Gruppe über einen Experten für einen bestimmte Themenbereich verfügt. Die Gruppenzuordnung bedient sich dabei graphentheoretischer Verfahren. Dieser Ansatz benötigt keinerlei explizite Angaben über die Lernenden.

1 Einleitung

In Lernkontexten ist es häufig förderlich, Übungsgruppen zu bilden. Aus Zeitgründen bzw. auf Grund fehlender Informationen zu den Lernenden wird sich jedoch häufig dafür entschieden, die Lernenden selbstständig Gruppen bilden zu lassen. Solche Gruppen haben allerdings häufig den Nachteil, dass sie nur aus Freunden bestehen. Dies fördert die Tendenz, Fehlverhalten einzelner zu decken. Ferner bestehen diese Gruppen oft aus Lernenden mit ähnlichen Fähigkeiten [O+04]. In zufällig gebildete Gruppen hingegen sind die Teilnehmer nicht zwangsläufig aufeinander abgestimmt. Diesen Nachteil weisen vom Lehrenden gebildete Gruppen im Allgemeinen nicht auf. Mittels vorgegebener Gruppen kann die Lernerinteraktion untereinander verstärkt werden, was zu besseren Lernerfolgen führt [Dei05].

Um adäquate Gruppen zu bilden, können Algorithmen eingesetzt werden, die jedoch Informationen über das Lernverhalten oder den Wissensstand der Lernenden benötigen. Oftmals stehen in Onlinekursen mit (anonymen) Teilnehmern jedoch wenige derartige Informationen zur Verfügung. Um trotzdem Annahmen über den Wissensstand eines Lernenden zu machen, können die Zugriffsprotokolle der Lernmaterialien insbesondere in Kursen mit vielen Lernressourcen genutzt werden. Dieses Papier präsentiert ein System, das auf Grundlage solcher Protokolle Gruppen bildet. Dazu werden zunächst existierende Algorithmen vorgestellt. Anschließend wird die Methode vorgestellt, mit

der Lernende an Hand der Ressourcenbenutzung Gruppen zugeteilt werden. Schlussendlich werden erste Ergebnisse präsentiert sowie zukünftige Arbeiten umrissen.

2 Hintergrund

Die Zusammenstellung effektiver Lerngruppen ist seit langem Gegenstand der Forschung, wobei die Heterogenität des Wissensstandes innerhalb der Gruppen ein wesentlicher Faktor ist. Eine weit verbreitete Methode für die Zusammenstellung von Lerngruppen ist das *Jigsaw-Puzzle* ([Aro78]). Hierbei werden Lernende in Gruppen eingeteilt, wobei jede Gruppe eine andere Aufgabe lösen soll. Anschließend werden die Gruppen derart neu arrangiert, dass sich in jeder Gruppe mindestens ein Mitglied jeder der vorherigen Gruppen befindet. Bei der Bewältigung weiterer Aufgaben nimmt jeder Lernende dadurch eine Expertenrolle für ein Thema ein. Auch die in dieser Publikation vorgestellte Arbeit ist von Gruppen mit Mitgliedern verschiedener Expertisen inspiriert.

In der Forschung zu computergestütztem Gruppenlernen existieren verschiedene Algorithmen zur Unterstützung der Gruppenbildung. Dabei wird auf algorithmischer Basis versucht, die Diversität innerhalb der Lerngruppen zu maximieren, wobei hierfür unterschiedliche Kriterien zu Rate gezogen werden. Eine generelle Formulierung von Regeln zur Gruppenbildung auf Basis von Kompetenzprofilen von Lernenden ist in [Hop95] formal beschrieben. Neuere Ansätze zur automatischen Gruppengenerierung beziehen neben der Wissensmodellierung auch den sozialen Kontext eines Lerners mit ein ([Hub10], [B+13]). [O+09] und [K+13] geben einen Überblick über weitere Methoden, die hier nicht weiter beschrieben werden.

Bei den meisten in der Literatur beschriebenen Methoden ist die Modellierung der Lernerprofile sehr datenintensiv. Ferner erfordern sie einen erheblichen manuellen Aufwand (s. bspw. [K+13], [AGT13]) oder aber spezielle Softwareinfrastrukturen, die das kontinuierliche Bewerten von Lernenden bewerkstelligen (s. bspw. [SKJ06]). Im Kontext von *Massive Open Online Courses* (MOOCs) ist dies jedoch nicht immer möglich. Dies liegt zum einen an der potentiell unbegrenzten Teilnehmerzahl, so dass der Zeitaufwand zu groß wird. Zum anderen sind in MOOCs die Lernenden weitgehend anonym, so dass es nur sehr schwer möglich ist, die nötigen Daten zur Kompetenz- und Vorliebenermittlung zu sammeln. Somit sind viele existierende Algorithmen nicht auf MOOCs anwendbar. MOOCs bieten den Lernenden jedoch im Allgemeinen zahlreiche Online-Ressourcen an, zu denen oft Zugriffsprotokolle existieren.

3 Methode

Die im Folgenden vorgestellte Methode benutzt zur Generierung von Übungsgruppen Zugriffsprotokolle von Lehrmaterialien. Dadurch ist es möglich, Lernerprofile trotz der Restriktionen von Onlinekursen zu erstellen. Der Methode liegt ein optimistischer Lernermodellierungsansatz zu Grunde. Hierbei wird angenommen, dass sich ein Lerner mit der Ressource befasst hat, auf die er zugegriffen hat. Mittels dieses Ansatzes

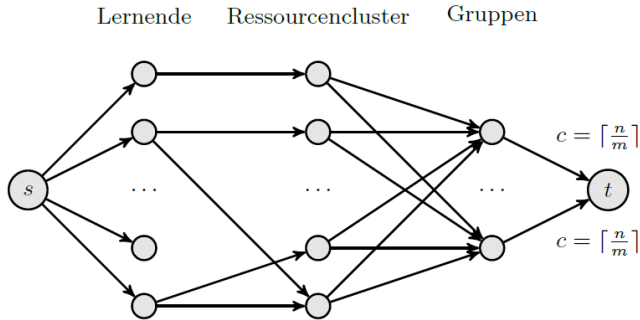


Abbildung 1: Beispielflussgraph - falls keine Kantenkapazität c angegeben ist, beträgt diese 1

können die Interessen bzw. der Wissensstand der Lernenden näherungsweise bestimmt werden. Der Ressourcenzugriff von Lernenden in einer gegebenen Zeitspanne wird als ein bipartites Netzwerk modelliert. Hierbei wird ein Lernender mit den Ressourcen verbunden, die er in dem Zeitfenster benutzt hat. Seiteneffekte, wie das zufällige Öffnen einer Ressource, können herausgefiltert werden, indem Ressourcenzugriffsmuster von Lernenden mittels Clusteringalgorithmen generalisiert werden. Dies geschieht auf Basis von Bicliquen. Für eine detaillierte Beschreibung des Clusteringalgorithmus wird auf [H+14] verwiesen. Basierend auf diesem geclusterten Graphen kann ähnlich des in [AMO93] beschriebenen *Problem of Representatives* ein Flussgraph erstellt werden (Abb. 1).

Eine gute Gruppe sollte in Analogie zum Jigsaw-Puzzle möglichst einen Repräsentanten (Experten) aus jedem Ressourcencluster (Themengebiet) enthalten. Basierend auf [AMO93] kann dieses Problem gelöst werden, indem für diesen Flussgraphen ein maximaler Fluss berechnet wird. Auf Grund der Kantenkapazität von 1 von Kanten zwischen der Quelle und den Lernendenknoten kann jeder Lernende höchstens Experte für ein Thema werden. Dies stellt sicher, dass kein Lernender durch mehrere Expertenrollen überfordert ist. Das Kantengewicht zwischen Ressourcencluster- und Gruppenknoten bewirkt, dass pro Gruppe maximal ein Experte pro Ressourcencluster existiert. Auch wenn in einer manuell angefertigten Gruppenzuordnung für eine Ressource mehrere Experten existieren können, kann in einem allgemeinen Flussgraphen keine Minimalanforderung an den maximalen Fluss gestellt werden. Eine Kantenkapazität größer 1 könnte dazu führen, dass eine Gruppe gebildet wird, die nur aus Experten des gleichen Ressourcenclusters besteht. Die Kapazität von $\lfloor n/m \rfloor$ der Kanten zwischen den Gruppenknoten und der Senke schließlich sorgt dafür, dass jeder Gruppe maximal $\lfloor n/m \rfloor$ Lernende zugeordnet werden, so dass die Gruppengrößen ähnlich sind. Hierbei steht n für die Anzahl der Lernenden, m für die vorzugebende Gruppenanzahl.

Die Berechnung des maximalen Flusses alleine erzeugt hierbei keine Zuordnung der Lernenden auf Gruppen. Vielmehr zeigt der maximale Fluss an, welcher Lernende Experte für welches Ressourcencluster wird. Im ersten Schritt der Zuordnung werden für jedes Ressourcencluster die Lernenden ermittelt, die mit diesem Ressourcenclusterknoten mittels einer Kante mit positivem Fluss verbunden sind. Dadurch sind nur die Flüsse zwischen Lernenden- und Ressourcenknoten für die Gruppenzuordnung von

Interesse. Die restlichen Kanten sind lediglich nötig, damit der maximale Fluss die durch die Anwendung gegebenen Bedingungen erfüllt.

Die durch den maximalen Fluss festgelegten Experten werden nun derart auf die verschiedenen Gruppen verteilt, dass immer die Gruppe mit den wenigsten Teilnehmern präferiert wird – bei mehreren Möglichkeiten wird die Gruppe mit der kleinsten ID gewählt. Der Algorithmus garantiert zudem, dass jeder Gruppe maximal ein Experte pro Ressourcencluster zugeteilt wird. Lernende, die im ersten Durchlauf keiner Gruppe zugewiesen werden, weil sie kein Experte sind oder es zu viele Experten für eine Ressource gibt, werden in einem zweiten Durchlauf analog zum ersten Schritt gleichmäßig auf die Gruppen verteilt. Dieser textuell beschriebene Gruppenbildungsalgorithmus findet sich in Algorithmus 1 in Pseudocode.

Algorithmus 1 : Gruppenzuordnung mit Hilfe eines maximalen Flusses

```

Input :  $m$  – Gruppenanzahl
Input :  $G$  – Flussgraph
Output : Gruppenzuordnung
initialisiere  $m$  leere Gruppen;
Fluss  $f$  = maximalerFluss( $G$ );
for  $i \leftarrow 1$  to |Ressourcencluster| do
    //  $L$  = Menge der Experten für Ressourcencluster  $i$ 
     $L(i) = \{ \text{Lernender } l \mid \exists \text{ Kante } [l, i] \text{ mit Fluss } f > 0 \}$ ;
    for  $j \leftarrow 1$  to | $L(i)$ | do
        | weise  $j$  der kleinsten Gruppen mit der kleinsten ID zu;
    end
end
 $L' = \{ \text{Lernender } l \mid \text{noch keiner Gruppe zugewiesen} \}$ ;
for  $i \leftarrow 1$  to | $L'$ | do
    | weise  $i$  der kleinsten Gruppen mit der kleinsten ID zu;
end

```

Beispiel

Als Beispiel sei der in Abbildung 2 dargestellte Flussgraph gegeben. Hier zeigt sich, dass die Wahl der Experten für jede Gruppe keinesfalls trivial ist. Wie zu sehen ist, könnte der Lernende l_1 Experte für c_1 und c_2 werden, da er mit beiden verbunden ist. Würde er Experte für c_2 werden, gäbe es keinen Experten für c_1 . Wird er jedoch, wie in der Abbildung dargestellt, Experte für das Cluster c_1 , kann der Lernende l_2 Experte für c_2 werden. Die Berechnung des maximalen Flusses versucht, so viele Experten wie möglich zuzuweisen. Ferner fällt auf, dass der Lernende l_5 kein Experte ist, obwohl er dem Ressourcencluster c_3 zugeordnet wurde. Dies liegt daran, dass pro Ressourcencluster maximal so viele Experten ausgewiesen werden können, wie Gruppen existieren. Da bereits zwei Experten für c_3 existieren, kann l_5 kein Experte für c_3 werden. Der Algorithmus generiert für dieses Beispiel die Gruppen $\{l_1, l_3, l_5\}$ und $\{l_2, l_4\}$.

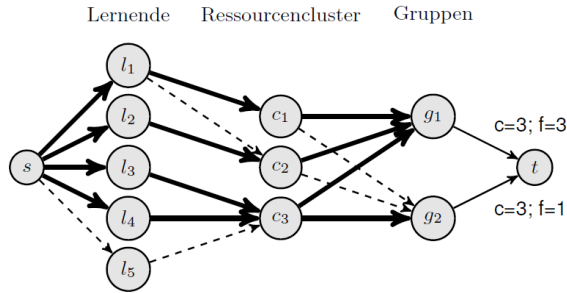


Abbildung 2: Ein Flussgraph mit eingezeichnetem maximalen Fluss f – fett gezeichnete Kanten haben einen Fluss von 1, gestrichelte Kanten einen Fluss von 0

4 Ergebnisse und zukünftige Arbeiten

Das beschriebene Verfahren wurde mit Daten des Kurses *Gestaltung interaktiver Lehr- und Lernsysteme* der Universität Duisburg-Essen evaluiert. Dieser Kurs bestand aus 44 Teilnehmern aus den Master Programmen *Angewandte Informatik*, *Angewandte Kognitions- und Medienwissenschaften* sowie einem internationalen Studienprogramm *Computer Engineering*. In diesem Kurs wurden sowohl Vorlesungen, als auch Übungen angeboten. Inspiriert von aktuellen MOOC Ansätzen wurden den Studierenden mehr Materialien im Learning Management System Moodle bereit gestellt als in anderen Kursen. Durch die meisten Übungsaufgaben wurden die angebotenen Ressourcen im Laufe des Semesters zudem von den Studenten selbst ergänzt.

Basierend auf den Ressourcenzugriffen wurden die Studenten retrospektiv jede Woche in fünf Gruppen eingeteilt. Diese wurden hinsichtlich der benutzten Ressourcen und der Durchmischung der Gruppen im Vergleich zur Vorwoche mit den tatsächlich gebildeten Gruppen verglichen. Wie erwartet zeigten die realen Gruppen wenig Durchmischung, d.h. dass die gleichen Studenten immer wieder gemeinsam Gruppen bildeten. Die algorithmisch gebildeten Gruppen hingegen wiesen eine deutlich stärkere Durchmischung auf. Bei der Ressourcenbenutzung zeigte sich, dass diese bei Studenten verschiedener realer Gruppen ähnlich war.

Nur in einem der betrachteten Zeiträume wurden die Studenten einer Vielzahl verschiedener Ressourcenclustern zugeordnet, so dass nur dieser Zeitraum für die Evaluation des Gruppenbildungsalgorithmus verwendet werden kann. Die Mitglieder der realen Gruppen zeigten in diesem Zeitraum, dass sich Mitglieder einer Gruppe eher die gleichen Ressourcen ansahen. Dieses Verhalten kann jedoch auch auf die Aufgabenstellung zurückzuführen sein. Die algorithmisch gebildeten Gruppen hingegen weisen eine deutlich gesteigerte Diversität hinsichtlich der genutzten Ressourcen innerhalb der Gruppen auf. Insgesamt wurden die benutzten Ressourcen für diesen interessanten Zeitraum in acht Cluster aufgeteilt. Die algorithmisch gebildeten fünf Gruppen hatten eine durchschnittliche Größe von 11,4 Studenten, wovon im Schnitt 4,2 Experten waren. Im Vergleich dazu wurden in der Realität sechs Gruppen mit einer

Durchschnittsgröße von 6,7 Studenten gebildet¹. Basierend auf der Ressourcenbenutzung konnten im Schnitt 3,2 Studenten pro Gruppe retrospektiv als Experten ausgewiesen werden. Wären diese Gruppen jedoch in der Realität gebildet worden, hätten sich auf Grund der Aufgabenstellung sicher andere Ressourcennutzungen gebildet. Trotzdem zeigt dieses Beispiel, dass die Methode prinzipiell funktioniert.

Zukünftige Arbeiten könnten andere Clustermethoden evaluieren, um die Aussagekraft der Ergebnisse zu stärken. Zudem wäre die praktische Anwendung des Systems denkbar, so dass das Lernverhalten und die Zufriedenheit der Studenten aus selbst gebildeten mit dem von Studenten aus algorithmisch generierten Gruppen verglichen werden können. Der bisherige Clusteransatz clustert zudem Ressourcen nicht explizit thematisch, so dass Lernende weniger Experten für ein Thema werden, als dies im Jigsaw-Puzzle passiert. Vielmehr werden sie Experten für eine Gruppe von Ressourcen, die inhaltlich nicht zusammenhängen müssen. Zudem kann ein Themenbereich von mehreren Ressourcen abgedeckt sein, die jedoch nicht zwangsläufig im selben Cluster zu finden sind. Daher könnten in zukünftigen Arbeiten die Ressourcen thematisch mittels der Verwendung von Metadaten oder Ähnlichkeitsmaßen geclustert werden.

Literaturverzeichnis

- [AGT13] R. Agrawal, B. Golshan und E. Terzi. Forming Beneficial Teams of Students in Massive Online Classes. ACM, Under Submission (2013).
- [AMO93] R. Ahuja, T. Magnani und J. Orlin. Network Flows: Theory, Algorithms, and Applications. Prentice Hall, Upper Saddle River, NJ, USA, 1993.
- [Aro78] E. Aronson. The Jigsaw Classroom. Sage, 1978.
- [B+13] J. Bais et al. An assistant for group formation in CSDL based on constraint satisfaction. Proceedings of ASAI 2013, S. 12–23, 2013.
- [Dei05] K. Deibel. Team Formation Methods for Increasing Interaction During In-class Group Work. In ACM SIGCSE, 37(3), New York, USA, 2005.
- [Hop95] H. U. Hoppe. The Use of Multiple Student Modeling to Parameterize Group Learning. In Proceedings of AI-ED'95, Washington, DC, 1995.
- [Hub10] R. Hubscher. Assigning Students to Groups Using General and Context-Specific Criteria. IEEE Transactions on Learning Technologies, 3(3), 2010.
- [H+14] T. Hecking et al. Analysis of Dynamic Resource Access Patterns in a Blended Learning Course. In Proceedings of the Fourth International Conference on Learning Analytics and Knowledge, S. 173–182, New York, USA, 2014.
- [K+13] J. Konert et al. GroupAL: ein Algorithmus zur Formation und Qualitätsbewertung von Lerngruppen in E-Learning-Szenarien mittels n-dimensionaler Gütekriterien. In Proceedings of the DeLFI 2013, S. 71–82, 2013.
- [O+04] B. Oakley et al. Turning student groups into effective teams. Journal of Student Centered Learning, S. 9–34, 2004.
- [O+09] A. Ounnas et al. A Framework for Semantic Group Formation in Education. Journal of Educational Technology & Society, 12(4), 2009.
- [SKJ06] L.-K. Soh, N. Khandaker und H. Jiang. Multiagent coalition formation for computer-supported cooperative learning. In Proceedings of the National Conference on Artificial Intelligence, 21:1844–1851, 2006.

¹ Die verschiedenen Gesamtzahlen kommen dadurch zustande, dass die algorithmisch gebildeten Gruppen alle angemeldeten Studenten einbeziehen, die realen Gruppen hingegen nur solche mit Abgaben berücksichtigen.