FRIEDRICH-ALEXANDER
UNIVERSITÄT
ERLANGEN-NÜRNBERG
TECHNISCHE FAKULTÄT

Department Informatik

Technical Reports / ISSN 2191-5008

Andreas Kaufmann, Ann Barcomb and Dirk Riehle

Using Students as a Distributed Coding Team for Validation through Intercoder Agreement

Technical Report CS-2016-01

April 2016

Please cite as:
Andreas Kaufmann, Ann Barcomb and Dirk Riehle, “Using Students as a Distributed Coding Team for Validation through Intercoder Agreement,” Friedrich-Alexander-Universität Erlangen-Nürnberg, Dept. of Computer Science, Technical Reports, CS-2016-01, April 2016.

Friedrich-Alexander-Universität Erlangen-Nürnberg
Department Informatik

Martensstr. 3 · 91058 Erlangen · Germany

www.cs.fau.de

Using Students as a Distributed Coding Team for Validation through Intercoder Agreement

Andreas Kaufmann, Ann Barcomb and Dirk Riehle
Computer Networks and Communication Systems
Dept. of Computer Science, University of Erlangen, Germany
andreas.kaufmann@fau.de, ann@barcomb.org, dirk@riehle.com

***Abstract*—In qualitative research, results often emerge through an analysis process called coding. A common measure of validity of theories built through qualitative research is the agreement between different people coding the same materials. High intercoder agreement indicates that the findings are derived from the data as opposed to being relative results based on the original researcher's bias. However, measuring such intercoder agreement incurs the high cost of having additional researchers perform seemingly redundant work. In this paper we present first results on a novel method of using students for validating theories. We find that intercoder agreement between a large number of students is almost as good as the intercoder agreement between two professionals working on the same materials.**

***Index Terms*—Qualitative Data Analysis, Theory Triangulation, Intercoder Agreement, Distributed Coding, Collective Coding**

I. Introduction

In this article we demonstrate how a distributed coding team of students can be used as an effective method for strengthening the validity of a qualitative study while at the same time providing an engaging and challenging learning experience for the participating students.

One common method of assessing the validity of qualitative research is showing that the generated theory holds up when investigated from different angles. This can be demonstrated through different forms of triangulation [1].

In this article we focus on *theory triangulation* which requires that additional perspectives should be considered from individuals from outside the field of study of the principle investigator, or - if they are professionals of the same discipline - have different status positions. At least the latter is the case when using students. The former is dependent on the broadness of the definition of the field of study. The participants are mostly students of computer science or international information system. The research area of volunteering in open source communities is however not part of their normal curriculum, and they are thus not experts in the field of study with regard to the data that they were required to analyse during the exercise.

Our experiment took place in the winter term 2015/2016, when we incorporated a series of qualitative analysis exercises into our elective research methods course 'Nailing Your Thesis' at the Friedrich Alexander Universität Erlangen-Nürnberg. The function of these exercises was to provide students with concrete experience in performing qualitative data analysis (QDA), while exploring the possibility of using students to increase the reliability of own qualitative research.

We created a distributed coding team of 41 students, five of whom had some previous experience with qualitative data analysis in a classroom setting, and the remainder of whom had no prior experience in the analysis technique.

The contributions of this article are twofold:

- A novel method for ensuring quality in theory building
- An experiment to provide preliminary data about how effective this method is, and how well it can be integrated with our teaching goals in a course on research methods

The remainder of this article is structured as follows: chapter II presents the design of our exercise as well as our validation strategy. In chapter III we then present the results of our experiment in terms of the achieved agreement scores as well as how the exercise met our teaching goals and how useful the results can be considered with the purpose of strengthening the validation. We then discuss limitations of our study and draw a conclusion in chapters IV and V.

II. Method

A. Data Analysis

We selected the material for analysis from a multi-case study which was being analyzed at the time of the class. The study contained four cases investigating the phenomenon of episodic volunteering in open source communities. The computer-assisted qualitative analysis tool MAXQDA was used.

The initial coding of the material was done by the second author, employing theoretical thematic analysis [2].

While coding, the second author iteratively developed a codebook with the names, descriptions, rules of use and examples of the codes identified in the material [3], [4]. The codebook was modified collaboratively to reduce error and bias [5]. The first and second author discussed and revised the codebook at two points: first, after the second author had coded a subset of the material and intercoder agreement had been calculated, and second, prior to the codebook being used in the class. The codebook was also modified according to student input following classroom discussion.

Overall, besides providing a testable measurement to strengthen the validity of the original research, these collaborative revisions improved the quality of the codebook by removing ambiguities and ensuring a clear understanding for investigators previously unfamiliar with the research.

B. Intercoder Agreement

The first author coded four documents from the first and second case before the initial coding of the third case was started. The documents being coded by both researchers were three interviews and one mailing list transcript.

The intercoder agreement between the first and second author was calculated to provide a baseline of the similarity which could be expected between two experienced researchers working with this material.

Intercoder agreement was calculated using software developed by the first author which operates on XML exports from MAXQDA 11.

Agreement was calculated as the harmonic mean of precision (equation 3) and recall (equation 2). This metric which is commonly used in the field of information retrieval is called the f-measure (equation 1) (also known as F_1 score or F-score).

$$F_1 = 2 \cdot \frac{precision \cdot recall}{precision + recall} \tag{1}$$

Recall measures how many of the codings performed by the researcher were replicated (see equation 2) whereas precision measures how many of the text segments coded by the second coder were coded using the same code by the researcher (see equation 3).

$$Recall = \frac{true\ positives}{true\ positives + false\ negatives} \tag{2}$$

$$Precision = \frac{true\ positives}{true\ positives + false\ positives} \tag{3}$$

This measure not only rewards agreement on applying the same code to a text segment through a high recall score, but also punishes applying a code where the original researcher did not apply a code, through a low precision. In the latter aspect, the f-measure differs from simple agreement which would weigh agreeing not to apply a code in one specific instance (i.e. on sentence, or paragraph level) equally compared to agreeing to apply a code in one instance. Because the frequency of *not* applying one code to a specific text segment is usually far higher than that for applying one code to a text segment, this leads to higher agreement values even for arguably low effort codings.

For example, consider comparing a document where a particular code has been applied to 20% of the text segments to one that has not been coded at all. Simple agreement would still count the remaining 80% of text segments as agreed (true positives). The f-measure, however, in this case would simply yield 0 agreement.

In this aspect, the f-measure resembles the kappa statistic [6], which is often applied as an agreement metric for precisely this reason [7]. The kappa statistic relativizes the observed agreement by factoring in the agreement by chance, meaning a corpus with the large majority of segments not coded has a high probability of agreement by chance. The f-measure however is completely independent on the number of elements that have been coded in a given corpus.

The magnitude of the agreement has to be interpreted the context of the number of categories (codes) that the coder was tasked to apply [8], since a large number of codes introduces a bias concerning the ability to consider all parts of the codebook with equal importance. While frequently cited classifications for interpreting intercoder agreement measured through Cohen's kappa exist [9], the boundaries for these benchmarks are by nature arbitrary, as it is explicitly stated by the authors.

Further it has to be considered, that such agreement measure is a measure of frequency of exact agreement,

not approximate agreement. With polytomous nominal data however, some pair of categories may be more similar than another, meaning some disagreement could be considered worse than another [10]. This weakness is also more prevalent with an increasing number of categories.

We utilized the following rough classification which uses no less arbitrary boundaries for each category but is based on our experience with this specific set of data:

- Agreement below 0.3 is considered to be low.
- Agreement between 0.3 and 0.4 is considered to be acceptable.
- Agreement above 0.4 is considered to be good.

This categorization was also used for the evaluation of student coding exercises, the sum of which contributed 20% to their final grade of this course, with another 20% being contributed by the other two types of exercises (summarizing and reviewing), and the remaining 40% were derived from the quality and frequency of their participation in in-class discussions.

The unit of coding considered for the calculation of the intercoder agreement was not set to a fixed text element such as a paragraph or a sentence. Rather, a code was considered to be applied in agreement if the coded text segments overlapped. A single large coding of a student was, however, not counted multiple times if it spanned multiple smaller codings by the researcher. All but one were then considered to be not identified by the student and consequently lowered their recall score. This was implemented to prevent a skew towards rewarding unnecessarily large units of coding.

This way of measurement was chosen mainly because of its convenience in processing large amounts of homework submissions provided by the over 40 students each week.

C. Exercise Design

Students were expected to complete a weekly qualitative analysis exercise for four consecutive weeks as part of a course on research methods. These assignments will be referred to as exercises 1, 2A, 2B, and 2. The assignments were accompanied by a lecture on qualitative research techniques. While our teaching objectives included the complete research process, this particular exercise focused exclusively on applying codes from a fixed codebook to a set of documents that was part of the original research. Students were not required to create their own code system, neither did they gather their own data.

Prior to the 4 weeks of coding exercises we conducted a survey about previous qualitative research experience. Forty students completed the questionnaire. Of these, four declined to participate in the research. The material generated by these four students and two students who completed the assignment but did not complete the survey are excluded from further analysis.

An overview of the exercise timeline is presented in figure 1.

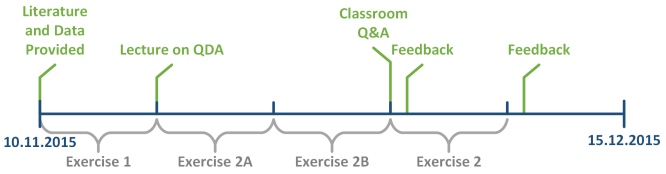

Fig. 1. Number of Coded Text Segments

In the first week, students were given exercise 1, which was not part of the research validation and which was intended to introduce them to MAXQDA, qualitative research and codebooks. The the type of material differed in that it was always part of the literature review and consisted mostly of scientific papers, whitepapers, book chapters and similar material.

Another key difference compared to the later exercise is a heavy reduction of the number of codes in the codebook. Within the first exercise only three codes were supposed to be coded by the students, which were easily understandable for non-expert coders. The codes to be applied were ”context”, ”problem” and ”solution”, which was part of an effort to abstract patterns from a large set of research data.

This first part of the coding exercise was purposefully *not* preceded by a an in-depth instruction, or lecture, on QDA. Instead students were provided with pointers to instructional material and were told to code as they understood it to be useful. As seen in figure 1 an in-depth lecture followed after the first week. We deliberately require students to investigate possible solutions on their own to strengthen the learning effect for the students, as they had first hand experiences with the issues that typically arise many of which can be addressed by following a well described time-tested methodology.

We did not calculate any intercoder agreement measures here, but the results were evaluated manually to determine if the codings seemed sensible and on this the students’ work for this week was graded on a nominal

scale of [0, 1, 2, 3] where a zero was given for no effort, a one for minimal effort, a two for good work and a three for exceptional work.

We provided individual feedback on the coding style where appropriate after the first exercise, but only for few students this was necessary.

After the first week, a lecture on QDA was given, and the full version of the codebook was introduced, and the new material for exercise 2A was distributed. The codebook at this stage consisted of 96 codes that could be applied which were structured under 5 core categories which had a total of 8 sub categories. The scope of the codebook was one of the major challenges students, as well as the co-investigating researcher, faced.

Also during this lecture, students were introduced to background information on the case study. They were explained the coding style—such as how to address duplication stemming from quoted text in emails—and introduced to the research question. Following the lecture they were provided with the material for exercise 2A, namely an interview which was not among the documents already recoded by the second author. The nine interviews were distributed using balanced incomplete block design.

Exercise 2B consisted of three supplemental documents:

- one "small" document where the second author identified fewer than 10 text segments to be coded
- one "medium" document where she found 10–20 codings
- one"large" document with more than 20 codings

Each student received a unique combination of documents. In order to ensure that all documents would be coded at least once, students were divided into two groups: those who had submitted the previous assignment and those who had not. All 41 students who completed the survey completed all four coding exercises. One document from each category (small, medium and large) was assigned to each student in the first group randomly but with the constraint that each document in a category was assigned with equal frequency. Distribution of the documents was continued in the same manner with the second group of students after the allocation of documents to the first group.

After exercise 2B was submitted, there was an in-class homework review session where students were invited to ask questions about coding and to make suggestions for improving the clarity of the codebook. This resulted in two modifications to the codebook.

Closely after this class session students were provided with their agreement score from the previous two weeks (exercise 2A and 2B) individually. With this information students were now also aware how many codes had been assigned by the researcher within the documents assigned to him or her. Giving an estimate of the number of codes an experienced coder found in a document may help novice coders improve their coding and create better alignment with an existing coding [11].

For exercise 2, students were given the opportunity to recode documents 2A and 2B based on what they had learned from the coding experience, the homework review session, the individual feedback, and the knowledge how many coded text segments could potentially lead to full agreement.

III. Results

A. Validation

For the proposition of using a larger group of non-experts, in our case students, to strengthen the interrater reliability through a distributed coding team to be viable, the quality of the non-expert coding has to be considered acceptable.

One of the clues we examined besides the final agreement measure is the number of coded text segments selected by students compared to the original researchers. This provides a rough indicator if the coding granularity was likely to be similar.

In the first part of the experiment containing data from the multiple case study on episodic volunteering (exercise 2A and 2B) we did not provide any target numbers for how many text segments have been coded in the gold-standard coding. Figure 2 illustrates how many text segments were coded by the students compared to the expert-coders. Each dot represents one document being coded.

Although there is significant variance, the linear trendline of the student's coding suggests that the average number of codes applied in a document of the collective coding performed by the large group of non-expert coders matches the expert-coder almost perfectly, independent of how many codes have been identified by the expert coder. Although the data averages around the expert-coder, fewer students apply too many codes, but if they do they are further off target than students applying too few codes.

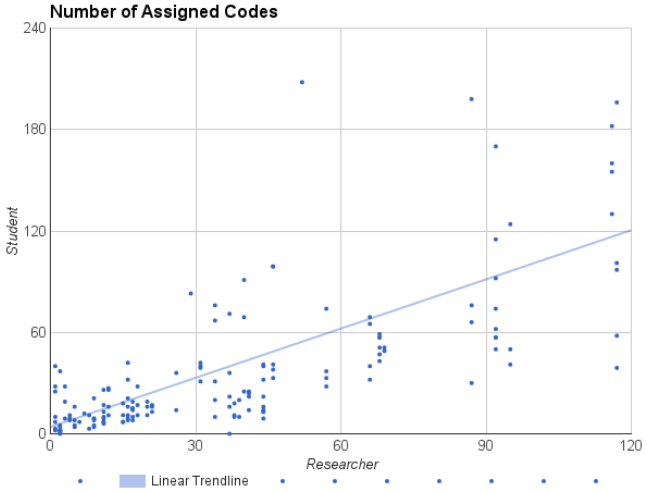

Fig. 2. Number of Coded Text Segments

In the second half of the experiment we followed Conway's advice on providing target numbers of how many codings were found by the expert coders [11]. Naturally after we supplied this information the average number of codings moved closer to our gold-standard coding, although not as much as anticipated. Even after knowing for certain that at a number of codings that were found by the expert-coders were not identified, many students struggled to finding additional text segments that made sense to be coded for them using one of the predefined codes. For students who coded too many text segments similar difficulties could be observed in their efforts of identifying and removing codings that were not in agreement.

The deviations of the number of coded text segments coded by students before and after the re-coding is documented in figure 3.

It is noticeable that the deviation in percentage is higher for documents with few codes assigned. This volatility is also reflected in the agreement metric, and is cause for us to consider a minimum number of codings of around 15 for each document within the next iteration of the class, along with considering the type of documents. Disregarding a few outliers the number of coded text segments moved closer towards the researcher's number of codings during the re-coding of the material in exercise 2 which is unsurprising given the incentive to code in a similar fashion for a good evaluation.

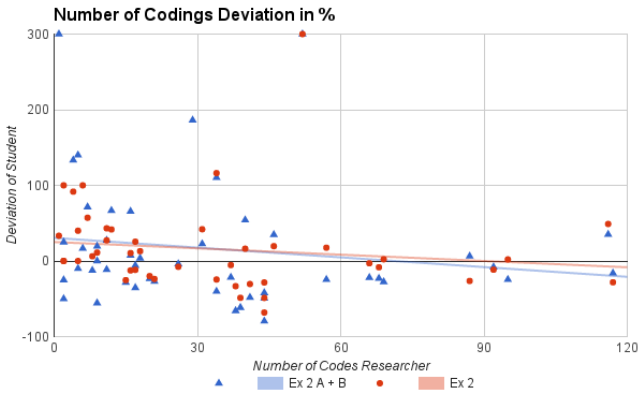

Fig. 3. Number of Coded Text Segments

The number of coded text segments alone obviously does not provide a good picture of the potential quality that collective coding may or may not have.

In table I we present the intercoder agreement based on our information retrieval metrics. The results show that using students as a distributed interrater team who are rated on their collective coding comes close to the agreement that was achieved between the two researchers, which was was rooted in a much deeper common understanding of the research topic.

TABLE I
INTERCODER AGREEMENT BY EXERCISE.

	Recall	Precision	f-measure
Researcher	0.371	0.416	0.386
Students Ex. 2A	0.323	0.421	0.357
Students Ex. 2B	0.303	0.139	0.303
Students Ex. 2	0.374	0.386	0.374

The agreement among the two researchers already showed very clearly that a high degree of agreement within the coding is highly dependent on the type of data that is being analyzed. Even though a shared understanding of the phenomenon may exist the analysis patterns may be very different for different types of artifacts. For instance, the interviews that were coded by the second researcher reached an average agreement greater than 0.4 (f-measure). One of the interrated documents, however, was a mailing list, for which the coding style diverged significantly leading to an agreement of only 0.23. This phenomenon could be observed during the analysis of the student's results as well.

Whereas the number of codings for these documents was not vastly different (see figure 2) among students and compared to the researcher, the document type did have a very significant impact on the agreement score. We determined that this effect could only partially be

accounted for by the fact that document types other than interviews, which were typically 5–10 pages long and contained many codings, were more a hit-or-miss for the agreement score, when in extreme cases a student could only find one coding, essentially reducing the recall component of our metric to a binary 1 or 0. This issue should have been eliminated by the number of students participating in the study. Instead we draw the conclusion that coding style was simply very different when coding something other than an interview transcript, since the interviews were conducted with the research questions in mind and therefore align better to the research directive.

Other than the mailing lists there were also a few (web-based) user interfaces of tools like issue-trackers that were used within the community, which were apparently also harder to code than interview transcripts, or at least achieved a lower agreement score.

The theory that the document size is less important here than the document type is also supported by the fact that there were other types of relatively shorter documents for which the average agreement was consistently higher than for the interviews. These were the documents describing the code of conduct within the different communities, and field notes written by the researcher during and after an interview. Our assumption is that this was because the documents were already well-structured, easing the coding process for non-expert coders.

Figure 4 illustrates how the average agreement of all students and all documents varied by document type. The data also shows that the re-coding (exercise 2) after the classroom discussion and a little more experience in coding did not improve the results for tooling user interfaces or mailing lists. In fact the average intercoder agreement for mailing lists even decreased. In contrast to this the most improvement was seen for interviews, followed by field notes and the codes of conduct.

Within these five categories, interview transcripts were the most frequent document type. Besides these, there was also one scientific paper and one slide deck from a presentation. Since these were the only representatives in each respective category they are omitted from the figure. While students struggled with the slides, the scientific paper reached high agreement scores. The high scores for the paper support the assumption that more structured documents are simply easier to code, however the slides fared less favorably, even though they are also highly structured. The issue may be that without a speaker to provide context, non-expert coders unfamiliar with the

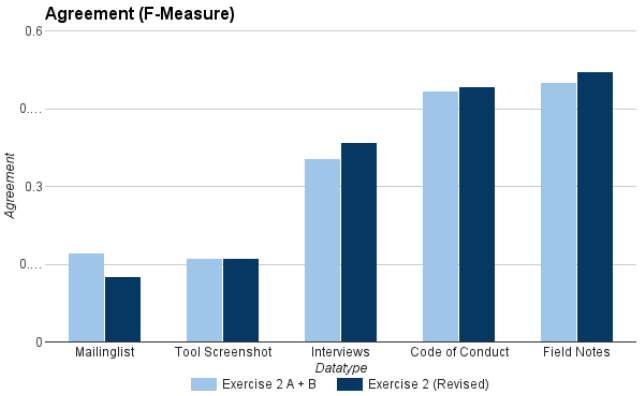

Fig. 4. Agreement by Document Type

field may not fully understand the topic, whereas the experienced researcher is able to.

While the improvement through the recoding of the material provided for exercise 2 (A + B) was less noticeable in certain document types, the average improvement was measurable across all sizes of document (in terms of number of codes being applied), the differences being most significant for documents with fewer than 15 coded text segments though (see figure 5).

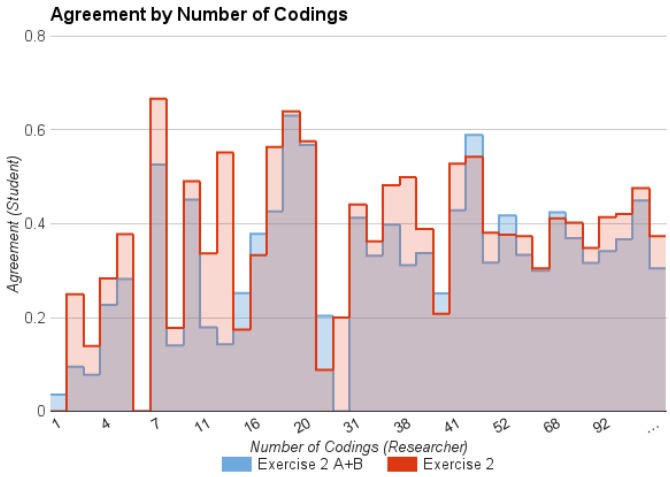

Fig. 5. Agreement by Document Coding Count

From a researcher's perspective this form of validation provided additional benefits in identifying ambiguities in the codebook, which regardless of any agreement score increases the reproducibility of the analysis. The preparation of the codebook for use by non-experts however was time consuming.

The statistical results are encouraging since the average agreement of the distributed coding team is only slightly below our benchmark.

B. Learning

As our distributed coding team was conceived as part of a course, it was also important to ensure learning objectives were achieved. We employed two methods to create an educational environment: repetition and extending the qualitative research methods to include analysis.

Students repeated the coding process multiple times. Moving from exercise 1 to exercise 2, they were able to rely on their experience with MAXQDA and coding. Exercise 2 was designed iteratively, so that students were able to apply their increasing knowledge of the codebook and coding in addition to personalized feedback to improve their work.

The learning effect for the ability to code closer to how an expert-coder would code is documented by the increased agreement values comparing exercise 2A and 2B to 2 (see table I and figure 4).

In addition to applying codes to several types of documents, students were expected to write a short paper addressing one of the research questions after the conclusion of the coding, in order to understand how analysis is performed from the low-level work of applying codes. These analysis documents were also graded on a rough nominal scale of 0 to 3.

In future classes we would reduce the number of different document types, or at least take special consideration about their distribution to students, their grading, or provide additional guidance on how to code these types of documents.

The modus of letting the students code on their own after a lecture on QDA, then providing quantitative feedback and having a discussion in the class following the homework submission worked out well. Students were highly engaged and appreciated the instant feedback. Qualitative feedback of the homework would likely have improved the learning effect, but was too time consuming for the teaching team, considering the high number of participating students. The automatic calculation of the intercoder agreement values and informing each student of their score proved to be an adequate tradeoff. A challenge in the execution of this in-class exercise concerned the automatic evaluation of more than 40 submissions each week was solved by implementing our own software for parsing and analysing the XML export of MAXQDA. For a smoother execution of an exercise as described in this article, with less administrative overhead, a new tool is currently in development.

IV. Limitations

Because of the grading scheme, we can expect that students were motivated to align their responses with the original researcher. Response bias was mitigated by providing students with no information about the original coder's work aside from the examples in the codebook, and for exercise 2 the number of coded segments in each document.

In a distributed coding team, it can be expected that some participants will be more motivated to complete the work well. We addressed the problem of slacking through the document distribution system described in section II-C, which ensured that multiple students processed key documents and each document in a category had an equal probability of being coded by a diligent student. An increase in student numbers, or a selection of a smaller subset of documents could further mitigate this issue.

A further limitation in the evaluation of the student's result is the lack of flexibility in choosing different agreement metrics. Although recall and precision are accepted standard metrics for information retrieval we would have liked to run a series of different agreement metrics and compare them to one another. This was not possible due to technical limitations on the data export the students generated from their project, and handed in as their homework assignments. Also, the ability to choose different granularities of the unit of coding (sentence or paragraph for instance) would have been desirable.

Having been restricted in the evaluation criteria also prevented us from applying existing research that indicates which level of agreement may be considered good, adequate or poor, and instead forced us to rely on data points based on our own experience.

For the learning objectives a coding exercise in the form that we presented in this article neglects the crucial aspect of creating a codesystem, writing memos, creating and defining new codes, deleting obsolete ones and restructuring the hierarchy. Also the gathering of the data was not performed by the students.

All of these aspects were fixed in our environment. The exercise was focused exclusively on the aspect of analysing material with regard to an already existing codebook. Although this does not affect our validation purposes this does have an effect on the learning objectives. A different exercise that would consider more of the complete research process is, however, not possible anyway due to time constraints that can be considered

adequate for a course with an intended total work load of 125 to 150 hours for an average student.

Within the scope of this article we only analyzed the aggregated the individual agreement values of students in the distributed coding team. We see significant additional potential in combining the student's results to build one collective coding result which would then be compared with the researcher's coding.

V. Conclusion

Creating a distributed coding team from a class of students taking an elective research methods course enabled us to demonstrate the reproducibility of our coding. With our automated grading system, validation required a little additional researcher time, compared to using a second researcher to completely recode the material.

The exercise also contributed to our teaching objectives by giving students hands-on experience with qualitative research techniques. Through iteration students were able to take advantage of their previous experience to apply techniques more effectively.

References

[1] L. A. Guion, D. Diehl, and D. McDonald, "Triangulation: Establishing the validity of qualitative studies," 2002.

[2] V. Braun and V. Clarke, "Using thematic analysis in psychology," *Qualitative research in psychology*, vol. 3, no. 2, pp. 77–101, 2006.

[3] G. Guest, A. Bunce, and L. Johnson, "How many interviews are enough? an experiment with data saturation and variability," *Field methods*, vol. 18, no. 1, pp. 59–82, 2006.

[4] K. M. MacQueen, E. McLellan, K. Kay, and B. Milstein, "Codebook development for team-based qualitative analysis," *Cultural anthropology methods*, vol. 10, no. 2, pp. 31–36, 1998.

[5] D. J. Hruschka, D. Schwartz, D. C. S. John, E. Picone-Decaro, R. A. Jenkins, and J. W. Carey, "Reliability in coding open-ended data: Lessons learned from hiv behavioral research," *Field Methods*, vol. 16, no. 3, pp. 307–331, 2004.

[6] J. Cohen *et al.*, "A coefficient of agreement for nominal scales," *Educational and psychological measurement*, vol. 20, no. 1, pp. 37–46, 1960.

[7] G. Hripcsak and A. S. Rothschild, "Agreement, the f-measure, and reliability in information retrieval," *Journal of the American Medical Informatics Association*, vol. 12, no. 3, pp. 296–298, 2005.

[8] J. Sim and C. C. Wright, "The kappa statistic in reliability studies: use, interpretation, and sample size requirements," *Physical therapy*, vol. 85, no. 3, pp. 257–268, 2005.

[9] J. R. Landis and G. G. Koch, "The measurement of observer agreement for categorical data," *biometrics*, pp. 159–174, 1977.

[10] M. Maclure and W. C. Willett, "Misinterpretation and misuse of the kappa statistic," *American journal of epidemiology*, vol. 126, no. 2, pp. 161–169, 1987.

[11] D. Conway, "Methods for collecting large-scale non-expert text coding," *Available at SSRN 2260437*, 2013.