Qualitative Coding: An Approach to Assess Inter-Rater Reliability

Ms. Anne Marguerite McAlister, The Ohio State University

Anne is an undergraduate student at The Ohio State University studying chemical engineering.

Dennis M Lee, Clemson University

Dennis M. Lee is a doctoral student in the Engineering and Science Education Department and Graduate Research Assistant in the office of the Associate Dean for Undergraduate Studies in the College of Engineering, Computing, and Applied Sciences at Clemson University. He received his BA and MS in bacteriology from the University of Wisconsin, Madison. Prior to his studies at Clemson University, he taught introductory biology at Tri-County Technical College in Pendleton, SC. His research interests include the development of researcher identity and epistemic cognition in undergraduate STEM students.

Ms. Katherine M Ehlert, Clemson University

Katherine M. Ehlert is a doctoral student in the Engineering and Science Education department in the College of Engineering, Computing, and Applied Sciences at Clemson University. She earned her BS in Mechanical Engineering from Case Western Reserve University and her MS in Mechanical Engineering focusing on Biomechanics from Cornell University. Prior to her enrollment at Clemson, Katherine worked as a Biomedical Engineering consultant in Philadelphia, PA. Her research interests include identity development through co and extra-curricular experiences for engineering students.

Dr. Rachel Louis Kajfez, Ohio State University

Dr. Rachel Louis Kajfez is an Assistant Professor in the Department of Engineering Education at The Ohio State University. She earned her B.S. and M.S. degrees in Civil Engineering from Ohio State and earned her Ph.D. in Engineering Education from Virginia Tech. Her research interests focus on the intersection between motivation and identity of undergraduate and graduate students, first-year engineering programs, mixed methods research, and innovative approaches to teaching.

Dr. Courtney June Faber, University of Tennessee, Knoxville

Courtney is a Lecturer and Research Assistant Professor in the College of Engineering Honors Program at the University of Tennessee. She completed her Ph.D. in Engineering & Science Education at Clemson University. Prior to her Ph.D. work, she received her B.S. in Bioengineering at Clemson University and her M.S. in Biomedical Engineering at Cornell University. Courtney's research interests include epistemic cognition in the context of problem solving, and researcher identity.

Dr. Marian S. Kennedy, Clemson University

Marian Kennedy is an Associate Professor within the Department of Materials Science & Engineering at Clemson University. Her research group focused on the mechanical and tribological characterization of thin films. She also contributes to the engineering education community through research related to undergraduate research programs and navigational capital needed for graduate school.

Qualitative Coding:
An Approach to Assess Inter-Rater Reliability

Abstract

When using qualitative coding techniques, establishing inter-rater reliability (IRR) is a recognized method of ensuring the trustworthiness of the study when multiple researchers are involved with coding. However, the process of manually determining IRR is not always fully explained within manuscripts or books. This is especially true if specialized qualitative coding software is being used since these software packages are often able to automatically calculate IRR providing little explanation on the methods used. Methods of coding without commercial software vary greatly including using non-specialized word processing or spreadsheet software and marking transcripts by hand using colored highlighters, pens, and even sticky notes. This array of coding approaches has led to a variety of techniques for calculating IRR. It is important that these techniques be shared, since IRR calculation is only automatic when using specialized coding software. This study summarizes a possible approach to establishing IRR for studies when researchers use word or spreadsheet processing software (e.g., Microsoft Word® and Excel®). Additionally, the authors provide their recommendations or "tricks of the trade" for future teams interested in calculating IRR between members of a coding team without specialized software.

Introduction

To ensure that the data collected within qualitative and quantitative research is correctly interpreted by a research team and can be used to build new insight, it is imperative that data analysis is conducted using best practices (Lincoln & Guba, 1985). These best practices should include methods to safeguard the trustworthiness and quality of the research. Trustworthiness gauges how well the evidence presented supports the value of the results, while quality measures of how likely systematic error and bias have been prevented through the design of the study (Lincoln & Guba, 1985). Given the interpretive nature of qualitative research methodologies, there are no standardized methods to ensure rigor across all types of qualitative studies. However, multiple frameworks have been established to guide qualitative researchers as they design rigorous research studies. Some frameworks, like Lincoln and Guba's trustworthiness framework (Lincoln & Guba, 1985), place emphasis on evaluating research quality at the end of a study, while Walther, Sochacka, & Kellam's (2013) Q3 (Quality in Qualitative research) framework emphasizes ensuring quality throughout the research process. Regardless of whether a continuous process or ad-hoc approach is taken by researchers to ensure rigor, all qualitative quality frameworks seek to mitigate interpretive bias of a single researcher.

A common analysis practice within qualitative research is coding. Coding is an iterative process that seeks to identify "a word or short phrase that captures and signals what is going on in a piece of data in a way that links it to some more general analysis issue" (Rossman & Rallis, 2012, p. 282). Codes are used to link data to conceptual frameworks and other broader concepts. Later in analysis, researchers will move beyond this coding to identify themes in their data and attach significance to their findings by offering explanations, drawing conclusions, and extrapolating examples (Creswell, 2013; Rossman & Rallis, 2012).

To prevent codes from seeming abstract or vague, researchers will often develop a codebook that describes each code with a concrete definition and example quote from the data (Creswell, 2014). This codebook can then be used by multiple researchers within the project or future researchers conducting similar studies. It is common to have multiple coders code the same data set or split large data sets between multiple coders. Walther et al. (2013) suggested IRR as a means to "mitigate interpretative bias" and ensure a "continuous dialogue between researchers to maintain consistency of the coding" (p. 650). Miles and Huberman (1994) suggest that an IRR of 80% agreement between coders on 95% of the codes is sufficient agreement among multiple coders(Miles & Huberman, 1994)(Miles & Huberman, 1994)(Miles & Huberman, 1994)(Miles & Huberman, 1994)(Miles & Huberman, 1994)(Miles & Huberman, 1994)(Miles & Huberman, 1994)(Miles & Huberman, 1994)(Miles & Huberman, 1994).

For this paper, we used the formula described in Miles and Huberman (1994):

$$\text{reliability} = \frac{\text{number of agreements}}{\text{number of agreements} + \text{disagreements}}$$

This calculation is but one method to measure consistency between coders. Other common measures are Cohen's Kappa (1960), Scott's Pi (1955), or Krippendorff's Alpha (1980) and have been used increasingly in well-respected communication journals ((Lovejoy, Watson, Lacy, & Riffe, 2016). All three additional measures consider how two coders agree in the same passage of text and then compares it to an expected percentage of agreement due to two random allocations of codes. To determine Kappa, Pi, or Alpha, we would determine the value for each code comparing each pair of coders, as well as all three coders together and then combine the code/coder pair specific values into an average across the board. Our large codebook (64 unique codes), made calculating Kappa, Pi, or Alpha difficult for two reasons (1) we often used a unique code only once in a transcript and (2) determining 4*the number of unique codes in a transcript (three pairs plus the three coders together) required excessive effort without rewarding us with additional insights. Instead, we considered reliability measured in aggregate (depicted above) for each transcript (total number of agreements between each pair of coders or the triplet).

This paper will focus on one method for calculating IRR for studies where common word (Microsoft Word®) and data processing (Microsoft Excel®) software is used.

Background

To highlight our approach to calculating IRR, we will use examples from our recent collaborative research project focused on understanding undergraduate engineering students' development of identity and beliefs about knowledge during research experiences (Faber et al., 2016). In this research project, the team used open-ended surveys to probe students' perceptions about their undergraduate research experience, their view of themselves as researchers, and their beliefs about research. The open-ended responses were processed where a document was created for each of the 154 survey participants. Specifically, participant responses were imported into separate Microsoft Word® files. This paper highlights the IRR technique used by three coders on the project to code the 154 documents.

Our Systematic Method for Calculating IRR

Throughout the coding process, coders worked from a codebook. As explained above, the codebook, an example of which can be seen in Table 1, lists the codes along with their definitions and provides examples of what each code should be and should not be.

The codebook for this research was developed through open coding techniques. Open coding consists of selecting sections of text of interest and coding them with a key word generated from the data itself opposed to using a predefined set of categories or codes (Patton, 2002; Creswell, 2014).

Initially, six completed survey documents from the set of 154 were selected at random and coded inductively by each of the three researchers to establish a codebook. The process of developing the codebook through open coding was iterative and involved numerous conversations among the coders and other members of the research team. The creation of a codebook was an initial step towards establishing IRR between coders within a research team by allowing each coder to compare their work to the established definitions. The goal of this process was to develop a codebook that could be used reliably by all members of the research team to consistently analyze the remaining open-ended survey documents and eliminate inconsistencies due to who was coding. More information about our coding process and the development of the codebook can be found in a recent paper by the researchers (Kajfez, McAlister, Faber, Ehlert, Lee, Benson, & Kennedy, 2017).

Table 1: Excerpt of codebook used to define codes

Code Name	Code Definition	Code Includes	Code Excludes
Experimenting	Student discussion of performing experiments or the use of verified data.	Includes performing, doing experiments or testing things. Trial and error, testing hypothesis, troubleshooting	Not planning experiments, collecting or analyzing data
Student Interest	Discussion of student's interest in subject matter of research or research itself	Interest coming directly from the student, desire to know more about subject, also includes expression of passion or enthusiasm	Interest from outside the individual like from family, friends, or mentors/peers
Tedious	Research taking too long, being slow or dull: tiresome or monotonous.	Expressing that the student finds the process of "doing research" to be tedious	Putting time and effort into research

A single participant’s responses (i.e., one document) were read by three coders, who tagged phases that they thought were linked with codes in the codebook using the comment function in the review tab of Microsoft Word®. The coder would write the code name in the comment, as shown in Figure 1.

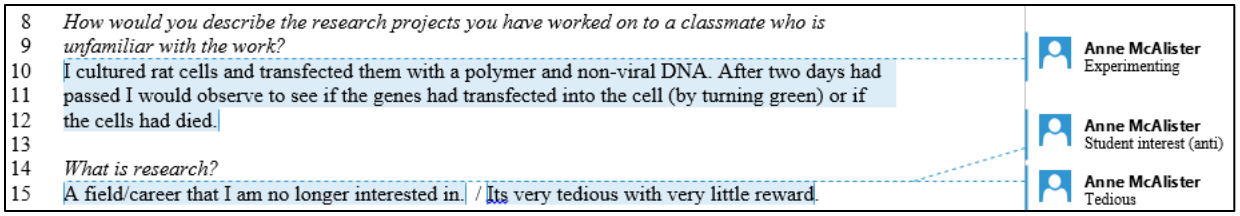

Figure 1: Selecting and commenting text in a Microsoft Word® document

Once the entire document for a single participant was coded, a macro was used to extract the comments and create a table in a separate Microsoft Word® document. Comments were used rather than highlighting or coloring the sections of text, so that they could then be extracted from the document using a macro. It is important to select enough text within the survey with the comment so sufficient context is provided when the section of text is extracted from the longer document.

A macro is a customizable function in Microsoft Word® that combines commands into a single process. The process described in this paper uses a macro developed by Fredborg (2013) and can be found at http://www.thedoctools.com/downloads/basComments_Extract.bas. To set up the macro, we selected the macros button in Microsoft Word® and selected "view macros" then "edit.' A Visual Basic for Applications window appeared, and the macro code was cut and pasted into the window. The original script creates a table that has columns for:

- page and line number (corresponding to the page and line number from which the excerpt of text is extracted in a single transcript)
- code name (the comment)
- text (the section of text that corresponds to the code)
- name of the coder
- the date the code was created

Table 2: Example of a coding table created using a Microsoft Word® macro to extract code data

Page	**Code**	**Text**	**Coder**	**Date**
Page 1 Line 10	Experimenting	I cultured rat cells and transfected them with a polymer and non-viral DNA. After two days had passed I would observe to see if the genes had transfected into the cell (by turning green) or if the cells had died.	Coder 1	05-Jul-2016
Page 1 Line 15	Student interest (anti)	A field/career that I am no longer interested in.	Coder 1	05-Jul-2016
Page 1 Line 15	Tedious	Its very tedious with very little reward	Coder 1	05-Jul-2016

Table 2 displays a section of a table of extracted codes produced within our work. These columns and data listed can be modified by revising the macro code within the Visual Basic for Applications window. The table created by each coder for each participant as a result of running the macro was then copied into a Microsoft Excel® document. This allows the codes of each

individual coder to be compared to each other. Microsoft Excel® was more efficient than Microsoft Word® as a tool for comparing the codes and phrases. Cells, rows, or columns could be easily moved and sorted for more efficient review.

Agreements and disagreements between coders were each tallied for each survey participant by directly comparing the codes applied to the same (or similar) excerpts. As each coder might not have included the same amount of context in their coding, we considered phrases including most of the same excerpt and the same code to be an agreement between coders. IRR was calculated as the number of agreed codes over the total number of codes in the document. This formula gives the fractional percent of codes that agree, making it easy to compare to our desired 80-90% agreement. The overall IRR was calculated (the number of codes that all three coders agreed on divided by the number of total coded sections) as well as the IRR between each individual pair of coders (the number of codes that both coders agreed on divided by the total number of coded sections). Two IRR values were determined between each set of two coders: (1) the number of times Coder 1 agreed with Coder 2 divided by the total number of codes used by Coder 1, and (2) the number of times Coder 2 agreed with Coder 1 divided by the total number of codes used by Coder 2. It is important to check both ways because these numbers may vary greatly due to the total number of codes applied by each coder as shown in Table 3.

Table 3: Sample of comparison table (codes that were not in agreement are shaded)

Page/ Line	**Code**	**Text**	**Coder**	**Date**	**Agreement?**
Page1 Line #16	A Process	systematic investigation	Coder 2	28-Jun-16	**Agreements: 16 Coder 1 total codes: 18 (IRR = 89%) Coder 2 total codes: 20 (IRR = 80%)**
Page1 Line #35	Acquisition of Knowledge	learning about new things	Coder 2	28-Jun-16	Full sentence is "interest in learning about new things" which is Having Curiosity--not acquisition of knowledge
Page1 Line #44	Application of Established Knowledge/ Theory	my major	Coder 2	29-Jun-16	

Results

To begin to determine our IRR after establishing the initial codebook, twelve participant documents were coded, and the initial IRR was calculated for each survey response. The IRR was found to be 40-60 % for this initial coding. The large amount of disagreement was due to varying interpretations of the codebook. Before coding more surveys, the coders went back and reexamined the codebook. Together they narrowed the definitions of some codes and added new codes with specific definitions to ensure the use of the codes would be consistent.

Each of the three coders independently reviewed five more surveys (for a total of 15 surveys) to increase their familiarity with the data and to test the established codebook on a variety of different surveys to see if additions or amendments were necessary. When the three coders reviewed another survey together to check coding consistency, they found that the IRR was still below 80%. Upon review of the second iteration of IRR, the researchers found that the analysis of the transcripts resulted in a small number of codes, exaggerating the effects of a single disagreement on IRR. For example, if a short survey response only required 8 codes and 2 were not in agreement, the IRR would be only 75%. As a result, subsequent surveys were selected for length to minimize the effects of small numbers of disagreements. Additional modifications were made to the names and definitions of individual codes within the codebook based on discussions about coding disagreements revealed by the IRR results. The coders performed several more iterations of coding surveys together to check IRR and discussed coding disagreements.

Through this process, the three coders were able to check coding agreement on a total of 24 surveys and consistently got 80-90% IRR. Once the IRR was consistently at least 80% on 95% of the codes, the codebook and coders were ready to code additional pieces of the data set that contains surveys taken by undergraduate students (n = 154) at five different universities.

Conclusion and Future Work

The preliminary coding of each of the first 24 surveys by all researchers was a time consuming but necessary initial step to safeguard coding trustworthiness and quality during the early stages of data analysis. However, it would be impractical to code all 154 surveys of the dataset in this manner. Since the IRR has confirmed consistent coding between researchers, our future coding procedure will be changed to increase the efficiency and speed of coding while integrating evaluations to maintain consistency between coders.

To increase efficiency and speed while maintain consistency, coding the remaining surveys in the dataset will be completed using a round-robin format. Going forward, the responses from participants to the open-ended items will be initially coded by one of the three coders. A second coder will then check the codes of the initial coder and tag any places where this second reviewer thinks that the code should be amended. These tags will also contain reasoning by Coder 2 for each change. Finally, the third coder will evaluate the coded and tagged surveys in order to moderate any disagreements between coders. The role for Coders 1, 2 and 3 will rotate between surveys so that each person codes, tags or moderates every transcript. This method ensures that each survey is looked at by all three coders with varying levels of specificity.

Using Microsoft Word® and Excel® to code and determine IRR allowed us to effectively establish a codebook and ensure coding consistency among multiple researchers prior to dividing and coding all 154 transcripts as described above without the use of specialized coding software. We hope the description of our process can help other researchers implement this method to accelerate or standardize IRR practices in their qualitative studies contributing to the quality of their work.

Tricks of the Trade

In keeping with the tricks of the trade nature of this paper, we provide the following recommendations for those who are interested in using the IRR process described in this paper in their research.

1. Select more than just the single word or phrase when commenting a code. Include enough context that when the text is extracted, the text still makes sense.
2. Code the same phrase with multiple codes when appropriate rather than trying to pick out the most salient code. This will increase agreement across users and provide complex relationships between codes to be explored during analysis.
3. Provide narrow definitions and specific examples in the codebook. This will reduce future changes to the codebook, and thus, additional rounds of coding. The less interpreting of the codebook that an individual coder can do, the higher the IRR should be.
4. The extracted tables of many participants can be compiled into one Microsoft Excel® document. This combined table allows all the data to be sorted by code. Comparing all the excerpts that have been coded a certain way can help narrow the definition of a code, and can make it easier to see what does and does not belong.
5. When calculating IRR, use a participant response that has a sufficient number of codes. If you have few codes to compare, your IRR can be skewed easily by one agreement or disagreement. Use a longer transcript to reduce this error.
6. Calculate an IRR between two coders both ways, because if one coder has applied more codes than the other, the IRR can vary greatly. If all of Coder 1's codes agree with Coder 2, but Coder 2 used twice as many codes as Coder 1, Coder 1 could have an IRR of 100%, while Coder 2 would have 50%.

These recommendations along with the process documented in this paper should provide researchers a method for establishing IRR when specialized coding software is not being used.

Acknowledgements

This material is based in part upon work supported by the National Science Foundation under Grant Number EEC-1531641. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

References

Benson, L. C., Kennedy, M. S., Ehlert, K. M., Vargas, P. M. D., Faber, C. J., Kajfez, R. L., & McAlister, A. M. (2016). Understanding undergraduate engineering researchers and how they learn. In *Frontiers in Education Conference (FIE), 2016 IEEE* (pp. 1–5). IEEE.

Cohen, J. (1960). A coefficient of agreement for nominal scales. *Educational and Psychological Measurement*, (20), 37–46.

Creswell, J. W. (2013). *Qualitative inquiry and research design: Choosing among five approaches* (Third Edit). Los Agneles: Sage Publications.

Fredborg, L. (2013). No Ti. Retrieved January 23, 2017, from http://www.thedoctools.com/downloads/basComments_Extract.bas

Krippendorff, K. (1980). *Content analysis: An introduction to its methodology* (2nd ed). Thousand Oaks, CA: Sage Publications.

Lincoln, Y. S., & Guba, E. G. (1985). Naturalistic Inquiry. Newbury Park, CA: Sage Publications.

Lovejoy, J., Watson, B. R., Lacy, S., & Riffe, D. (2016). Three Decades of Reliability in Communication Content Analyses: Reporting of Reliability Statistics and Coefficient Levels in Three Top Journals. *Journalism & Mass Communication Quarterly*, *93.4 (2016*, 1135–1159.

Miles, M. B., & Huberman, A. M. (1994). *Qualitative data analysis: An expanded sourcebook* (Second Edi). Sage Publications.

Rossman, G. B., & Rallis, S. F. (2011). *Learning in the field: An introducation to qualitative research* (Third edit). book, Los Angeles: Sage Publications.

Scott, W. A. (1955). Reliability of content analysis: The case of nominal scale coding. *Public Opinion Quarterly*, (19), 321–325.

Walther, J., Sochacka, N. W., & Kellam, N. N. (2013). Quality in interpretive engineering education research: Reflections on an example study. *Journal of Engineering Education*, *102*(4), 626–659. http://doi.org/10.1002/jee.20029