Using Machine Learning to Support Qualitative Coding in Social Science: Shifting The Focus to Ambiguity

NAN-CHEN CHEN, University of Washington, USA
MARGARET DROUHARD, University of Washington, USA
RAFAL KOCIELNIK, University of Washington, USA
JINA SUH, University of Washington, USA
CECILIA R. ARAGON, University of Washington, USA

Machine learning (ML) has become increasingly influential to human society, yet the primary advancements and applications of ML are driven by research in only a few computational disciplines. Even applications that affect or analyze human behaviors and social structures are often developed with limited input from experts outside of computational fields. Social scientists—experts trained to examine and explain the complexity of human behavior and interactions in the world—have considerable expertise to contribute to the development of ML applications for human-generated data, and their analytic practices could benefit from more human-centered ML methods. Although a few researchers have highlighted some gaps between ML and social sciences [51, 57, 70], most discussions only focus on quantitative methods. Yet many social science disciplines rely heavily on qualitative methods to distill patterns that are challenging to discover through quantitative data. One common analysis method for qualitative data is *qualitative coding*. In this work, we highlight three challenges of applying ML to qualitative coding. Additionally, we utilize our experience of designing a visual analytics tool for collaborative qualitative coding to demonstrate the potential in using ML to support qualitative coding by shifting the focus to identifying ambiguity. We illustrate dimensions of ambiguity and discuss the relationship between disagreement and ambiguity. Finally, we propose three research directions to ground ML applications for social science as part of the progression toward *human-centered* machine learning.

CCS Concepts: • **Human-centered computing → Empirical studies in HCI**; *User studies*;

Additional Key Words and Phrases: social scientists, qualitative coding, machine learning, ambiguity, human-centered machine learning, computational social science

ACM Reference Format:
Nan-Chen Chen, Margaret Drouhard, Rafal Kocielnik, Jina Suh, and Cecilia R. Aragon. 2018. Using Machine Learning to Support Qualitative Coding in Social Science: Shifting The Focus to Ambiguity. *ACM Trans. Interact. Intell. Syst.* 9, 4, Article 39 (March 2018), 21 pages. https://doi.org/0000001.0000001

1 INTRODUCTION

Machine learning (ML) is one of the fastest growing fields in computer science (CS), and many disciplines—physics, biology, finance, and others—have adopted ML methods to analyze data or otherwise support their domains of research. At the same time, digital traces of human social

Authors' addresses: Nan-Chen Chen, University of Washington, Seattle, WA, 98195, USA, nanchen@uw.edu; Margaret Drouhard, University of Washington, Seattle, WA, 98195, USA, mdrouhar@uw.edu; Rafal Kocielnik, University of Washington, Seattle, WA, 98195, USA, rkoc@uw.edu; Jina Suh, University of Washington, Seattle, WA, 98195, USA, jinasuh@uw.edu; Cecilia R. Aragon, University of Washington, Seattle, WA, 98195, USA, aragon@uw.edu.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.
© 2018 Copyright held by the owner/author(s). Publication rights licensed to the Association for Computing Machinery.

2160-6455/2018/3-ART39 $15.00
https://doi.org/0000001.0000001

interactions are proliferating, and increasing numbers of researchers in social science are working to study human behaviors through social media and similar datasets (e.g., [59]). However, social data are complex and require detailed examinations. Researchers have pointed out that the limited data for minority populations may cause ML models to learn patterns or make inferences based primarily or exclusively on majority group traits, which can lead to exacerbation of stereotypes or unjust discrimination practices [3, 8, 11, 27, 32, 35, 45, 61, 69]. Therefore, it is important to carefully integrate ML methods with social science practice, and to draw from the expertise of trained social scientists, in order to make claims about human behaviors.

The field of *computational social science* has emerged to support analysis of ever larger datasets. Nevertheless, progress in applying computational methods like ML to social science research has been relatively slow compared to fields like biology [38]. Furthermore, much of the work using ML techniques to make claims about human behavior derives from research in computational fields such as computer science, statistics, and even physics, with limited involvement from social scientists [13, 24, 69, 72]. Some researchers in computational social science are troubled by this phenomenon because researchers with traditional computational backgrounds often lack training in social science methods and have limited experience examining complex social data [24].

A number of existing discussions have tried to explain the gaps between social science and ML in computer science. One suggested difference is that data analysis in social science is theory-driven such that the data is drawn from a population with a presumed distribution (e.g., Poisson distribution, Gaussian distribution, multinomial distributions). On the contrary, classical machine learning problems, such as image recognition, are usually data-driven such that the data is assumed to be drawn from an independent and identically distributed set where the goal is to find the most optimal model that fits the data. Computer scientists often design and tweak models directly from raw data, independent of theoretical presumptions of data distributions [51, 57]. Although greater attention to alignment with theory may account for some of the gaps in ML as quantitative analysis method for social science, other misalignments remain to be addressed. In fact, some social science analysis methods, such as grounded theory methods, start from no theory and construct theories iteratively based on data. This type of qualitative analysis is commonly used in analyzing qualitative data, such as interview transcripts [4] and social media data [68]. Therefore, in order to better utilize ML for social science analysis, understanding the gaps between ML and qualitative methods in social science is critical.

In this paper, by leveraging our experiences of building a visual analytics tool for collaborative qualitative coding, we highlight a few intrinsic tensions between social science practice and machine learning analysis in the context of qualitative coding. We then propose shifting the focus of ML predictions to identify ambiguity. We conduct a Mechanical Turk study to understand different types of ambiguity. Finally, we describe three future research directions and connect with existing work in each direction. In highlighting these research opportunities, we intend to stimulate further research toward supporting social scientists' use of ML for both quantitative and qualitative methods. We anticipate that more human-centered, interpretable ML methods have the potential to transform social science research. Furthermore, findings from social science studies that use ML methods may bolster the theoretical foundations for ML applications built upon social data, in addition to promoting more human-centered, socially useful methods overall.

2 BACKGROUND

In this section, we define the scope of the terms 'ML' and 'social sciences' in this paper. Then, we illustrate how ML is used in social science, followed by a summary of existing discussion on the gaps between ML and social science. Next, we provide a brief overview of qualitative methods in social science, and specifically describe the process of *qualitative coding* and *grounded theory methods*.

2.1 Definition of Machine Learning in This Paper

ML as a scientific field primarily focuses on developing algorithms and techniques to construct representations (i.e., learn models) from data. Common topics in ML include supervised learning, unsupervised learning, and reinforcement learning. Supervised learning algorithms, such as support vector machine (SVM) and decision tree, aim to build a model based on a labeled dataset (i.e., data with *ground truth* labels) for predicting unlabeled or even unseen data. Unsupervised learning methods, like k-means and hierarchical clustering, try to represent a dataset without ground truth labels. Reinforcement learning works under dynamic environments where model construction is based not only on data but also on feedback from the environment. *Data mining* is the science of extracting information from large, sometimes raw, data sets and can leverage techniques from other disciplines such as statistics, ML, data management, pattern recognition, etc [26]. There are four types of tasks within data mining. Exploratory data analysis is an interactive and iterative process of exploring the data without any clear structure or goal (e.g., principal component analysis). Descriptive modeling is a way to describe or generate data (e.g., clustering, density estimation). Predictive modeling is a task that builds a model that predicts a variable given a set of other variables extracted from data (e.g., classification, regression). Lastly, pattern recognition or detection is a way to find regions in the space of data that differ from the rest (e.g., association rule learning). While ML and data mining are easily conflated, it is important to remember that the goal of data mining is to support the task of understanding and discovering of unknown structures in a given data through the use of various techniques, and ML is one of many sources of methods that provides the technical basis for data mining [75]. In this paper, we use the term ML to refer to the techniques used for modeling, exploration, and pattern recognition without necessarily connecting these techniques to how they are used in data mining.

2.2 Definition of Social Science in This Paper

In this paper, we refer to ‘social science’ as any discipline that studies human behaviors and social phenomena. This definition includes traditional social science fields such as sociology, political science, economics, anthropology, organization study, social psychology, etc. We also consider more recently emerged disciplines or research areas like computer-supported cooperative work (CSCW), social computing, and information science as social science since these fields partly inherit social science traditions and often rely on theories or methods from traditional social science fields.

2.3 Use of Machine Learning in Social Science

2.3.1 Computational Methods in Social Science. Computational or statistical methods have been widely used in social science since Adolphe Quetelet (1749-1827), one of the most influential social statisticians and the inventor of the notion of the “average man," applied the principles and methods of physical sciences to social sciences [46]. The study of social science events typically require some type of basic descriptive statistics (e.g., central tendency, dispersion) and inferential statistics (e.g., estimation of confidence interval, significance testing). Regression analysis (e.g., linear regression, analysis of variance) is used to model the relation between two or more variables for prediction and forecasting [50].

ML methods, influenced heavily by statistics, have been gaining popularity in some social science disciplines [63] and are used for predictive and descriptive analysis of data. With computational and ML methods, social scientists can process and distill large amounts of data with numerous variables to infer causality or relations among latent variables or to predict outcomes for unseen data.

Dimensionality reduction techniques (e.g., factor analysis, principal component analysis) can infer the relations between latent structures [74]. Structural equation modeling is a technique that allows development of a new variable from an existing set of variables [7]. Latent dirichlet allocation is a generative probabilistic modeling technique used for topic modeling that reduces the data to a probability distribution of “latent" topics discovered within the data [5]. K-means clustering, although published more than 60 years ago, is still widely used among ML practitioners and social scientists to separate large data space into subspaces that are manageable and meaningful [33].

2.3.2 Bridging the Gap between Machine Learning and Social Science. Although ML methods used in computational social science share the same foundations as statistical methods used in traditional social science, some ML researchers in computational social science have indicated that there are several fundamental conflicts between ML and social science [51, 57, 70]. For example, the goal of ML is to predict behavior on unobserved data (A changes with B when everything else is held constant, or correlation), whereas the goal of social science is to understand and explain the whys and hows of observed phenomena (A influences B when all else is equal, or causation). While causality has been a fundamental concern in social science, ML has traditionally not focused on causality. Instead, ML puts emphasis on fine-tuning models and parameters to achieve high prediction accuracy, or other metrics like precision and recall. Settles suggested that social science is deductive and hypothesis-driven, while machine learning is inductive and data-driven [57]. Similarly, Rudin highlighted that a social scientist may form a hypothesis first and collect the data specifically curated to test the hypothesis. A ML scientist will collect a large set of data with few pre-defined goals and usually assume that the data is independent and identically distributed [51]. In addition, in a recent book chapter, Wallach pointed out several differences between social science and computational social science in computer science [70]. For instance, social scientists usually make claims based on multiple sources of information. However, most modern ML algorithms only work under single-source settings. Thus, directly applying such a method to social science problems is challenging. Furthermore, many ML methods are concerned merely about feasibility and efficiency of the model. Results with only efficiency concerns without taking interpretability or accountability into account can be risky for making social science claims.

With increasing demand and interest in computational social science, many efforts have been made to bridge the gap between ML and social science. There has been a shift in focus to reframe ML in causal inference in order to discover natural experiments hidden in large data or to use machine learning to predict counterfactual relations [2, 55, 66]. Some recent computational social science research places strong focus on the explanation of phenomena (why and how), and while research on single data source is still prevalent, there have been discussions on potential computational challenges and opportunities to go beyond a single data source in order to answer meaningful social science questions [70]. Wallach suggests several ways for computer scientists and social scientists to collaborate and to advance computational social science: (1) understand each other by engaging in meaningful discussions and attending each other’s conferences, (2) create high-quality publication venues for interdisciplinary work, and (3) create interdisciplinary degree programs to train the next generation of computational social scientists [70].

2.4 Qualitative Methods in Social Science and Qualitative Coding

In addition to quantitative methods (e.g., statistical, computational,ML), social scientists use qualitative methods, such as observation, interviews, and case studies, as their primary research means or mixed with other quantitative methods. Qualitative methods have a long history in the humanities. Since the early 20th century, fields like anthropology and sociology have established the importance of qualitative inquiry [18]. For instance, a common approach to collect data in anthropology is *ethnography*. In ethnographic studies, anthropologists go into a foreign society (called a *field site*) for a long period of time and use a set of qualitative methods to collect data for understanding the culture of the society: researchers observe people and their interactions on the site, and interview people to collect their explanations of their behaviors. In the digital era, such an ethnographic approach is often deployed to study people's cyber behaviors and online social phenomena. Researchers can go on a virtual site to observe users' posting behaviors or collect a set of historical posts to distill interaction patterns.

A common way to analyze qualitative data is *qualitative coding*. Qualitative coding, or simply *coding*, is one of the major techniques used in qualitative analysis among social scientists [60]. In general, coding refers to the process of assigning descriptive or inferential labels to chunks of data, which may assist concept or theory development [42, 44, 64]. Coding is usually a very labor-intensive and time-consuming task [60, 77]. It requires researchers to examine their data in detail, find relevant or potential points of interest, and assign labels. As the size of datasets grows significantly in the era of big data, manually coding the entire dataset in detail is not feasible for social scientists. As a result, social scientists can only sample and code a small part of their data. Since a large portion will remain under-explored, researchers may not be able to resolve inconsistencies in their theories and may not even recognize if some analysis is missing or incomplete.

2.4.1 The need for qualitative coding. Coding is a process of arranging qualitative data in a systematic order by segregating, grouping and linking it in order to facilitate formulation of meaning and explanation. Such analysis is often used to search for patterns in the data by organizing and grouping similarly coded data into categories based on commonly shared characteristics [53]. Even though some qualitative data—such as tweets—have metadata, organizing data based on these metadata is often not enough to achieve researchers' analysis goals. Thus, coding is necessary for researchers to create structure and impose it on the data to determine how best to organize the information and facilitate its interpretation for their purposes [39].

2.4.2 Grounded Theory. Grounded theory (GT) is one of the most well-known set of approaches to deal with code organization and theory development. As Charmaz articulated in one of the most referenced GT textbooks [12], "grounded theory methods consist of systematic, yet flexible guidelines for collecting and analyzing qualitative data to construct theories from the data themselves." It is important to note that this broad definition encompasses a number of GT approaches such as classical Glaserian GT [23] and Strauss's approach [29]. Here we describe the steps involved in one of the most recent variations of GT, called a constructionist GT as presented in Charmaz's introduction to grounded theory [12]. Despite differences, other GT approaches share a majority of the steps described here.

Analysis steps under GT are not carried out sequentially since insights or realizations of analytic connections can happen any time during the research process [12]. In GT, coding provides an analytic skeleton and links connecting data with developing emergent theory. The analysis starts with an initial line-by-line, unrestricted coding of the data termed *open coding*. The aim is to produce concepts that seem to fit the data. The line-by-line focus is meant to prompt studying the data closely and trigger conceptualizing emerging ideas. At this stage, codes are entirely provisional and prone to change.

The second step is *focused coding* in which a researcher works with initial codes that indicate analytic significance. This permits for separation, sorting and synthesizing large amounts of data and also accelerates the analytic pace [62]. While performing focused coding, it is also possible to engage in an optional step of *axial coding*, which involves coding dimensions of a category and exploring the relationship between that category and other categories and subcategories [22]. Extended notes, also known as memos, are an an important tool for comparing data, exploring ideas about codes, and directing further data-gathering. Such memos often form the core of the analysis and become a record for how it was developed [12].

The third step involves a process of *theoretical sampling*, which is a strategy in GT to obtain further selective data and to refine and fill out major categories; this step ultimately leads to the theoretical saturation [29]. Arriving at theoretical saturation of major categories (that is, no new properties or dimensions have emerged from continued coding and comparison [30]) often becomes a criterion for stopping further data collection.

The final step of *theoretical coding* is a sophisticated level of coding that uses the codes selected during focused coding to integrate and solidify the analysis in a theoretical structure. In the end, one of the theoretical codes will be chosen for the study [29]. Even though theoretical coding is presented here as the last step, several theoretical code may emerge during analysis in practice [12]. As Charmaz advocated, using emergent theoretical codes keeps the analysis creative and fresh [12].

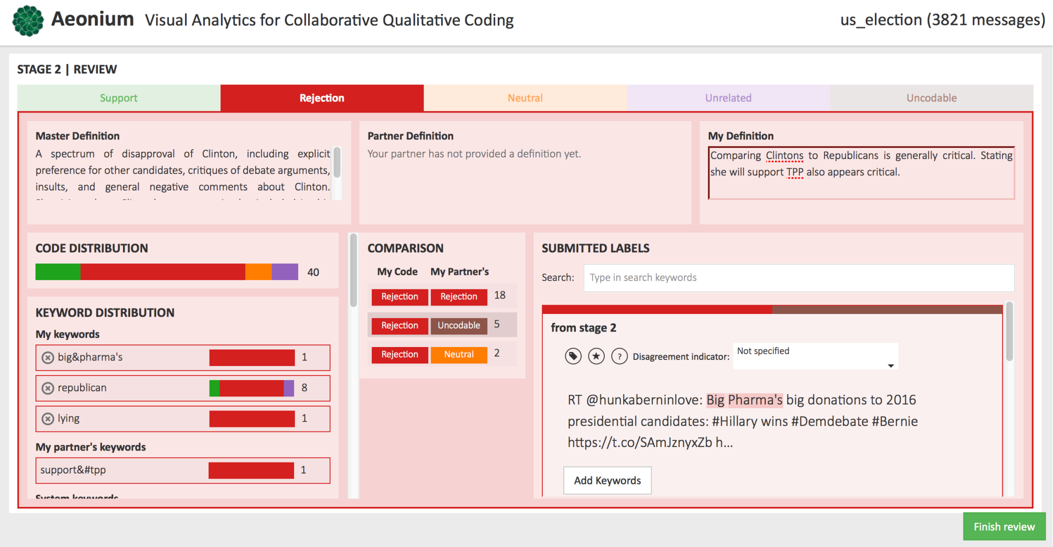
As there are many variations in how to conduct GT approaches [47], individual researchers may use the method differently. Sometimes, the output from the analysis may be an under-developed theory that captures only key theoretical ideas. Also, although GT evolved as a method of theory construction, not everyone who uses its strategies intends to develop a theory [12].

2.5 Challenges in Adopting Machine Learning Approaches in Qualitative Coding

Although the application of ML in quantitative/computational social sciences have been increasingly popular in the recent years, only a relatively small number of references exist for ML-supported applications that facilitate qualitative analysis of large datasets using fully or semi-automated techniques. For example, Yan et al. proposed using natural language processing (NLP) and ML to generate initial codes and then ask humans to correct the codes. Other work utilizes NLP to derive potential codes and/or learn models [15, 16, 25, 40, 64, 77]. Recently, Muller et al. proposed new research directions for combining grounded theory and ML methods [43]. While low accuracy has been considered the primary limitation of such automated approaches, we outline three other challenges for ML applications in qualitative coding. We do not intend to present a comprehensive list of challenges, but rather highlight some of the fundamental complications in applying ML in qualitative analysis, to encourage further research on mitigating them.

These challenges were identified during the design and implementation of *Aeonium* (Fig 1), a visual analytics tool for collaborative qualitative coding that highlight ambiguity [20]. The details of tool design are out of scope for this paper, but here we will summarize three key challenges discovered from our formative studies during the initial phases of design of Aeonium as well as the expert reviews for evaluating the tool. These formative studies included conceptual analyses drawn from the authors' experience with qualitative coding in CSCW research, as well as interviews with five qualitative researchers in CSCW and social science fields. The expert reviews were conducted with four other graduate-level qualitative researchers in CSCW and information science who did not participate in the formative study. We invited them to test the tool in an one-hour think-aloud session. More details of the studies can be found in our introduction to Aeonium [20].

2.5.1 Lack of understanding between disciplines may deteriorate trust. One of the core issues limiting the application of ML in qualitative analysis is that people who use qualitative methods are generally not trained in ML techniques. Due to the complexity of selecting features, building models, and tuning parameters, it may be difficult for non-experts in ML to construct useful models.



A spectrum of disapproval of Clinton, including explicit preference for other candidates, critiques of debate arguments, insults, and general negative comments about Clinton.

Your partner has not provided a definition yet.

Comparing Clintons to Republicans is generally critical. Stating she will support TPP also appears critical.

My Code	My Partner's	
Rejection	Rejection	18
Rejection	Uncodable	5
Rejection	Neutral	2

RT @hunkaberninlove: Big Pharma's big donations to 2016 presidential candidates: #Hillary wins #Demdebate #Bernie https://t.co/SAmJznyxZb h...

Fig. 1. Aeonium Review Interface. The design of the review interface highlights disagreement between coders (the COMPARISON panel) and allows coders to give feedback to the disagreement (the SUBMITTED LABELS panel).

Conversely, an ML expert might be able to train an accurate classifier using codes labeled by social scientists, but that individual likely would not have the social science training to consider issues that may be critical to analysis. For example, since few ML experts are trained in social science, they may not have contextual information to engineer good features, nor to adequately address issues such as overfitting. Social scientists are usually interested in sophisticated social phenomena, so their codes often capture nuanced understanding of the data. These conceptualizations are often difficult to characterize using de-contextualized features such as counts, keywords or even semantic features. Sometimes, even social scientists do not have a clearly articulated vocabulary for some concepts of interest. Understandably, then, it may be quite difficult for researchers who do not have background in social science to engineer good features and find ways to distinguish these concepts using ML. As a result, even if an ML expert can construct a model ostensibly capable of labeling data with qualitative codes, social scientists may not have trust in the system. One of the Aeonium expert review participants told us, “I am not sure if I can believe ML output. I don’t think keywords (referring to dictionary features) are enough to represent the concept I am looking for.”

2.5.2 Building a learnable model is not the primary goal. In some cases, ML experts’ limited understanding of social science values and methods can hamper effective collaborations. However, some inherent tensions or conflicts between goals of ML and social science could also cause complications. For example, the goal of performance optimization of ML models may be in conflict with goals of qualitative coding. To build a strong classifier, we usually need predefined categories and a large quantity of corresponding labeled data (for supervised learning), or the distributions of the dataset must have some distinct separation (for unsupervised learning). However, most often, neither of these is the case for qualitative coding. As coding requires significant manual analysis, it is difficult to label sufficient data for strong ML results. In the open coding stage, scientists often do not have a priori categories, but rather, categories gradually emerge from careful analysis of the data. Even in closed coding, in which categories have usually been determined, the definition of each category may still need to evolve or shift as more of the dataset is explored. While social scientists may sometimes want to label as much data as possible, their ultimate goal is not to build a machine learnable model, but instead to discover patterns in the data or to answer particular research questions. Labeling sufficient data to train a strong ML classifier may not have utility for these researchers. If new patterns are not emerging from qualitative coding, social scientists may determine that "saturation" has been reached, and coding more data in the dataset will not contribute significantly to the analysis.

2.5.3 Fundamental differences between qualitative and quantitative methods. Additionally, ML usually performs better on categories that have more instances, but codes with numerous instances might not be the most meaningful to social scientists. In quantitative analysis, data points that appear very few times may be considered noise, but from a qualitative analysis perspective, quantity of instances is not always reflective of significance. Since it is very hard for any ML method to capture categories that have sparse instances in the dataset, social scientists may prefer to manually code the raw data rather than spend time trying to tune the models. Aside from the considerations of utility of coding effort, even highly accurate ML models may not be very informative or reliable from a social scientist's point of view. Most ML methods function as black boxes and do not offer explanations or interpretable results. Without knowing how prediction results are derived, social scientists may be reluctant to adopt ML in their analytical practices.

In addition, although some computer-assisted qualitative data analysis software (CAQDAS), such as NVivo, ATLAS.ti, and MaxQDA, have been developed and used, there are debates on the potential negative impact of computer assistance on qualitative research [73]. For instance, when people do not have sufficient understanding and experience in the methods they use, they might be more easily influenced by the results the software suggested [21]. If people start to depend on ML and statistical methods in their qualitative analysis, we need to be careful about unintended consequences of over-reliance.

3 SHIFTING THE FOCUS OF ML: A CASE STUDY IN IDENTIFYING AMBIGUITY IN COLLABORATIVE CODING

In this section, we begin by extending our earlier discussion of the challenges in adopting ML in qualitative coding, and we describe our approach to bridging the gap between ML and qualitative coding through drawing out ambiguity in collaborative coding. Then we discuss dimensions of ambiguity and highlight results from an MTurk study for testing the relation between disagreement and ambiguity.

3.1 Emphasizing Ambiguity in Collaborative Coding

The challenges from the results of our formative studies of Aeonium, outlined in section 2.5, indicated that it might not be possible or useful to build ML models to predict codes for data instances, so we shifted our focus to identifying ambiguity in the context of collaborative qualitative coding. In qualitative coding, it is common to have multiple coders working collaboratively to label data with codes, dividing the labor and helping to ensure quality of the codes. However, since coding is inherently subjective, inconsistencies between coders necessarily arise. Our interviewees in the formative studies expressed interest in exploring inconsistencies between coders, as these disagreements may indicate ambiguity in either the code definitions or particular data points that collaborators need to negotiate or clarify. In addition, ambiguity can signal potential nuances in data that are interesting to social scientists. Moreover, collaborative qualitative coding frequently reveals unanticipated complexities in the dataset that could require articulation of new codes or reconsideration of previously labeled data.

The inherent ambiguity of qualitative data derives from the fact that it is generated by humans with diverse experiences, priorities, and ways of communicating. In our case, we focused on social media data, which are particularly subject to ambiguity due to the style of informal text communication. In social media, people condense their ideas into brief statements and often presume shared context with an audience based on the time and venue for posts. The abridged structure of such posts enables only limited clarification of context within the data. After completing our initial iteration of the Aeonium tool, we conducted an expert review with graduate-level qualitative researchers who rely on qualitative analysis as a core method in their research. In addition to evaluating Aeonium, participants in this review helped distill our analysis of the core challenges in applying ML to qualitative analysis. During the review, experts indicated that qualitative researchers may interpret such data quite differently depending upon their own experiences. As one expert noted, "Coding is a process to help coders get familiar with the data and the domain, and inconsistency sometimes comes from information asymmetry. In other words, one collaborator may know something that the other person doesn't know. By pointing out inconsistency, it can help resolve these differences in background knowledge." For qualitative coding, additional ambiguity may be introduced through unclear boundaries of code definitions. Another qualitative researcher noted, "Coding always involves some analytical messiness. [Codes] are never defined enough." In other words, regardless of how carefully researchers consider and articulate codes, new insights may arise in the data that confound interpretations or do not fall neatly into already defined code categories.

Qualitative researchers indicated in our formative studies that it would be a valuable contribution for ML to help identify points of probable inconsistency and ambiguity. They did not necessarily intend to resolve all inconsistencies or come to agreement about each data point, but they could often gain more insight analyzing ambiguous data in greater depth than they could analyzing data for which coders' initial interpretations agree. The inconsistent or unclear data may help researchers disambiguate related concepts or themes. Moreover, by analyzing their own disagreements and interpretations, they can build more complete understandings of the data, relying on the diversity of experiences, background knowledge, and analytical styles of all members of a research team. If an ML model is able to successfully identify ambiguous data, qualitative researchers may use it to focus their energy on data that are more likely to extend or better articulate their insights around a dataset.

Based on our interviews with qualitative researchers, as well as our own experiences coding qualitatively, we articulated dimensions of ambiguity and proxies for ambiguity that may be valuable for the purposes of drawing out ambiguous data. One of these proxies is described in detail in our initial work with the Aeonium system [20], but we have explored the dimensions further through studies using Amazon Mechanical Turk (Mturk) in order to articulate them more clearly for future ML or visual analytics applications.

3.2 Dimensions of Ambiguity

Two dimensions of ambiguity seem most significant to us: *data ambiguity* and *human subjectivity.* Data ambiguity refers to ambiguity inherent in data that usually arises from a lack of context in particular data points. One form of data ambiguity, which we call "semantic ambiguity," has limited context due to the structure or formulation of expression. For instance, someone may clearly express an attitude or opinion, but not state explicitly who or what the target of the opinion is, leaving the overall opinion unclear. Alternatively, statements may contain multiple, sometimes contradictory sentiments. Adjusting the granularity of interpretation can help mitigate inconsistencies of interpretation for these cases, but sometimes inconsistencies cannot be fully resolved.

The second dimension of ambiguity, human subjectivity, results from different levels of understanding or sets of experiences among collaborating researchers, leading to distinct interpretations. Some of the significant components of human subjectivity are exposure to different kinds of discourse and confidence in knowledge about particular topics. "Unknown unknowns" may be the most problematic of these inconsistencies, since people are often unaware of gaps in their own knowledge about particular topics and may be confident in faulty interpretations. It should be noted that these dimensions are not mutually independent, and the line between data ambiguity and human subjectivity is not clear-cut. However, each dimension may contribute differently to the ambiguity and inconsistency between coders of qualitative data.

In consideration of these dimensions, the clearest proxy for ambiguity that we identified is disagreement between coders. Disagreement may indicate human subjectivity of interpretation but could also signify that multiple reasonable interpretations are possible due to data ambiguity. Other proxies might include comments or annotations around data to explain coding choices or time spent analyzing data. In Aeonium, we also provided mechanisms for explicit flagging of ambiguity based on user interpretation and for feedback about ambiguity after identification of inconsistency between coders (Fig. 1). However, the proxy on which we most relied was coder disagreement.

When designing Aeonium, we leveraged our understandings of ambiguity—particularly the proxy of coder disagreement—to draw out data points that are most likely to be inconsistently coded between two collaborators. Our approach is similar to the Query by Committee (QBC) sampling method in active learning [56]. The Aeonium system takes coded data from a pair of coders, and then trains individual SVM models for two coders respectively, using keyword features extracted from the data set (i.e., bag-of-words n-gram features) and user-defined keywords as additional features. Next, the system uses the trained model to label uncoded data, and it identifies data points that are inconsistent between partners but for which the model has high confidence in the label applied. The system also includes a review interface that highlights data points (which are tweets in our current design) for which coders disagreed, and the interface allows coders to review disagreement in the codes applied. Our study participants indicated that these functionalities helped draw their attention to details or other context that they did not consider when initially coding the tweets. They also reported that the features highlighting ambiguity and disagreement helped them develop insight about the dataset and reflect more on their own interpretations in coding.

Our approach to identify ambiguous data points using a QBC-style method is still preliminary, and we have not yet performed a longitudinal study with users to examine how such an approach can benefit coders in a long run. However, the positive feedback we received from expert users indicate that this direction is promising. We believe that using ML methods to help draw out ambiguous data is one path toward a better ML support for qualitative coding and that this approach merits further research. As an initial validation study for this sampling approach and the proxy of coder disagreement for ambiguity, we examined the relationship between disagreement and ambiguity utilizing MTurk.

3.3 MTurk Study on The Relation between Disagreement and Ambiguity

We designed our MTurk study to assess whether and to what extent coder disagreement correlates with perceived ambiguity of data. The dataset for this study is drawn from a collection of tweets collected in February 2016 focused on Hillary Clinton during one of the Democratic primary debates. "Master coders" for the dataset refer to members of our research team with experience in qualitative coding who open-coded the tweets and came up with a set of five mutually exclusive codes to be applied to each tweet. The master coders then proceeded to closed coding, independently labeling tweets with a code and reviewing these initial codes together to make code decisions and identify ambiguity. Master coders identified ambiguity in part through disagreement between coders, but in some cases, master coders determined that a tweet was ambiguous even though the initial codes independently assigned by master coders were in agreement. In our MTurk study, we asked MTurkers to go through the same process of closed coding and then individually make determinations about the degree of ambiguity in a particular tweet. We sought to evaluate whether the master coder determinations of ambiguity extended to a more general perception of ambiguity by others interpreting the same data. We expected that at least some sources of ambiguity are inherent in the data itself, and thus might be identified even when there is agreement among coders in the codes applied. Consequently, we expected that the ambiguity is predictable, meaning that tweets deemed ambiguous by one set of coders are more likely to be considered ambiguous by other coders as well. We thus formed our first hypothesis:

H1: Tweets identified as ambiguous by master coders will more likely be rated as ambiguous by the MTurk coders, as compared to the tweets deemed unambiguous by master coders.

We also wanted to investigate if the disagreement among master coders is predictive of disagreement among other group of coders. Such higher disagreement may stem from the fact that the specific tweets contain conflicting or hard to interpret information pieces that are likely to lead to different interpretations and disagreement among other coders. Therefore, we formed our second hypothesis:

H2: Tweets for which master coders' labels disagreed will likely show more disagreement among the MTurk coders, as compared to the tweets for which master coders' labels agreed.

Participants. Participants for the study were 41 MTurk master workers (a distinction that Amazon makes based on overall performance on MTurk tasks). All of the participants, according to MTurk's criteria limitations, had also completed high school in the U.S., so we anticipate that they had some degree of familiarity with U.S. electoral politics. Voluntarily provided demographic data indicates that participants ranged in age from 20 years to 60+ years, with the majority aged 39 or younger. 14 participants stated they fell in age range 20-29, 19 fell in the range 30-39, 4 in the range 40-49, 2 in the range 50-59, and 2 in the range 60+. Most also indicated at least some university education, with 3 having completed a master's degree, 20 having completed a bachelor's (4-year) degree, 6 an associate's (2-year) degree, and 7 more having completed some college. 5 indicated having earned a high school diploma, and none identified having received graduate degrees other than master's degrees. The racial makeup of participants was more heavily skewed, with 37 identifying as White, 2 as Black or African American, 3 as Latino or Hispanic and 1 as Asian. None identified as American Indian or Alaska Native, Native Hawaiian or Pacific Islander, or any other racial/ethnic identity. 20 participants identified as male, 20 as female, and 1 as genderqueer. Postal codes provided voluntarily reveal a variety of locations within the U.S., most along the east and west coasts, with many also in the southeast and midwest.

Study Design and Procedure. Our study considered within-group measures for all participants. Participants were asked to label 45 tweets with a code (choosing from one of the five codes defined by master coders) and also to label that tweet as either ambiguous or unambiguous. Detailed code definitions and the definitions for 'ambiguous' and 'unambiguous' data are provided in Appendix A. After a brief set of training exercises for participants to practice coding tweets, participants labeled 15 tweets each from the following three groups of data:

(1) Data for which master coders' initial codes **disagreed**, which master coders determined to be **ambiguous** after reviewing together

(2) Data for which master coders' initial codes **agreed**, which master coders determined to be **ambiguous** after reviewing together
(3) Data for which master coders' initial codes **agreed**, which master coders determined to be **unambiguous** after reviewing together

We did not consider data for which master coders' initial codes disagreed but which master coders determined to be unambiguous for two reasons: there were insufficient data points in the group, and most of these cases resulted from coder inattention. To avoid ordering effects in the study, tweets were presented to participants in random order.

Data analysis. For testing our first hypothesis, we focused our analysis on every coded tweet. We had 15 tweets from each of the three groups, a total of 45 per participant, or a grand total of 1845 coded tweets for all 41 participants. Therefore, we transformed the data into 1845 rows as shown in Table 1. Each row represents a coding instance with data corresponding to the tweet id (1–45), the group the tweet belonged to (1,2 or 3), the categorical ambiguity rating (0-unambiguous, 1-ambiguous) and a unique id of the participant who coded this tweet. To analyze such data and take into account that the tweet codings were not independent (multiple tweets were coded by the same participant), we used a mixed-effects logistic regression model. In the model, we used ambiguity rating of individual coded tweets as a predicted outcome (hence the logistic model), the group the tweet belonged to as fixed effect, and participant id as well as tweet id as random effects to account for dependency among codings. We obtained p-values for the model using the Wald test.

To test the second hypothesis, we calculated the level of agreement across coders for each individual tweet (this is different than the explicit ambiguity rating we investigate in H1). For this analysis we represented the data per unique individual tweet, not individual coding of a tweet as in analysis for the first hypothesis. Consequently we had a total of 45 rows in this representation, with codes assigned by individual participants as 41 columns. We defined a measure of code agreement for an individual tweet as a ratio of the count of the most frequent category given for a tweet divided by the total number of codes for this tweet (fixed at 41 as each tweet has been coded by all the participants). We further normalized this measure based on the total number of available categories

Table 1. Example format of transformed MTurk study result data

row_id	**participant_id**	**group_id**	**tweet_id**	**labeled_code**	**is_ambiguous**
1	1	1	1	Not Understandable	1
2	2	1	1	Support	1
			...		
41	41	1	1	Support	0
42	1	1	2	Not Understandable	1
			...		
82	41	1	2	Neutral	1
			...		
616	1	2	16	Neutral	1
			...		
1845	41	3	45	Rejection/Criticism	0

(5 in our case), so that it takes a value from 0 to 1, where 0 represents a uniform distribution of all possible codes per category and 1 represents the situation when a tweet is coded under the same category by all the coders (e.g. if a tweet has been coded as Support by 38 coders, as Unrelated by 2 coders, as Rejection/Criticism by 1 coder, and never coded as belonging to the 2 remaining categories, the Agreement would be ≈ 0.91; if the codes distribution was: Support-14, Neutral-10, Rejection/Criticism-12, Not understandable-3, and Unrelated-2, the Agreement would be ≈ 0.18). With such organization we could apply a simple one-way ANOVA with the group the tweet belonged to as the independent variable and the code agreement level for a tweet as the dependent variable. Post-hoc comparisons (conservative Bonferroni correction) were conducted to compare the groups.

Results. **H1:** Testing of our first hypothesis, on the predictability of the ambiguity rating, revealed that on average 6.93 ($sd = 2.79$) out of 15 tweets were rated as ambiguous in Group 1 (master coders disagree and ambiguous). Similarly, for Group 2 (master coders agreed and ambiguous) the average number of tweets rated as ambiguous was 6.85 ($sd = 2.74$). For both these groups the difference as compared to the average ambiguity ratings of 4.71 ($sd = 2.75$) in Group 3 (master coders agreed and unambiguous) was statistically significant ($\beta = 0.7471$, $SE = 0.3166$, $p < .05$) for Group 2 and ($\beta = 0.7712$, $SE = 0.3161$, $p < .05$) for Group 1 in the mixed-effects logistic regression model. Interpreting our results as odds ratio, a tweet from Group 1 is 2.2 times more likely to be rated as ambiguous as compared to the tweet from Group 3. Similarly, a tweet from Group 2 is 2.1 times more likely to be rated as ambiguous as compared to the tweet from Group 3. The odds were not significantly different between Groups 1 and 2.

This result shows that the ambiguity determination under the same coding scheme seems predictable to other coders, meaning tweets that are deemed ambiguous by one group of coders are likely to also be rated ambiguous by other coders under the same coding scheme.

H2: Testing the second hypothesis, on the predictability of disagreement, revealed that for tweets in Group 1 (master coders disagree and ambiguous) the average agreement among MTurk coders was indeed the lowest, on average 0.38 ($sd = 0.18$). It was significantly lower ($p < 0.01$) than for Group 3 (master coders agreed and unambiguous), where the average agreement was 0.63 ($sd = 0.20$), but not significantly lower than for Group 2 (master coders agreed and ambiguous), where it was 0.48 ($sd = 0.24$).

These results show that whenever master coders disagreed with other master coders on the appropriate label for a tweet, the MTurk coders were also more likely to disagree. This suggests that the disagreement is predictable between different groups of coders under the same coding scheme. Lack of significant difference in agreement between tweets in Group 1 and Group 2 may indicate that the rating of ambiguity might also be meaningful for predicting disagreement. In order to test this observation and estimate how predictive the master coders ambiguity rating is for MTurk coders disagreement, we performed correlation analysis. Please note that in our dataset ambiguity rating is binary (level 1 - combined tweets from groups 1 and 2, level 2 - tweets from group 3) and disagreement rating is continuous. Point-Biseral Correlation is most appropriate in such case. We found a moderate, positive correlation, which was statistically significant ($r_{pb} = 0.403$, $p < 0.01$). This result suggests that ambiguity rating is indeed predictive of coding disagreement.

Qualitative feedback about ambiguity. In addition to the labeling of tweets with a code and ambiguity label, we asked participants to provide open-ended feedback about what factors contributed to the ambiguity of a tweet. Specifically, we asked them to respond to the following prompt: “What are reasons that some tweets were or might be ambiguous? You may answer N/A if no tweets were ambiguous.” Two members of our research team then qualitatively coded the open-ended feedback to identify factors most frequently identified as relating to ambiguity. One of the authors open-coded the responses to develop a codebook of reasons for ambiguity, including responses such as content being unfamiliar to the coder and therefore difficult to evaluate; contradictory or confusing hashtags; coders perceiving potential sarcasm that rendered the message of a tweet ambiguous (See Appendix A.3 for a full list of the codes). The developer of the codebook and another researcher then independently coded each response. We validated our qualitative coding by measuring percentage agreement (A) between coders for each unique code, and using the Cohen's Kappa (κ) measure for Inter-rater Reliability [14]. Many of the factors indicated by participants as contributing to ambiguity aligned with our conceptualizations of the dimensions of ambiguity. Almost 60% of participants noted that limited context in some tweets made them ambiguous (A = 75.6%, $\kappa = 0.517$), and around a quarter pointed out that tweets could be interpreted differently depending on the readers' experiences and background (A = 85.4%, $\kappa = 0.614$). Around half of the participants also identified semantic ambiguity as a source of confusion, noting that the target of expressed opinions was unclear (A = 80.5%, $\kappa = 0.616$). This feedback suggests that our dimensions for ambiguity are generalizable at least to some extent, which can help improve feature engineering and model tuning for ML tools working to identify ambiguity.

3.3.1 Future work on learning disagreement to identify ambiguity. Through our MTurk study, we demonstrate that both ambiguity and disagreement are predictable between groups of coders, and that disagreement or inconsistency can be an indicator of ambiguity in coded data. According to interviewees in our formative studies, ML models that can pinpoint potential ambiguity in a dataset are valuable for qualitative analysis, so research in this area has the potential to make significant contributions. Further work is needed to explore the best ways to leverage disagreement as a proxy for ambiguity in ML models and to create interpretable, human-centered ML tools for qualitative analysis.

4 FUTURE DIRECTIONS TO ENGAGE MACHINE LEARNING AND SOCIAL SCIENCE

In the previous two sections, we summarized the existing discussions on the gaps between ML and quantitative methods in social science. Then we described the challenges of applying ML to qualitative analysis, proposed alternative uses of ML to support qualitative coding, and demonstrated how ML can be used to address coding ambiguity. In our MTurk study, our findings suggest ambiguity by one coder under the same coding scheme can be predicted to other coders and disagreement between master coders can be predicted to other coders. Based on these findings, we argue for investing greater research efforts in bridging the gaps between ML and social science in both quantitative and qualitative methods.

Our reasons are three-fold. Firstly, as so-called "big data" is becoming more and more pervasive in our daily life and changing the ways people interact, we need better tools to facilitate social science and help understand human relationships and behaviors in such evolving environments. Secondly, the complexity of social science data provides significant research opportunities for the ML community, yet research developments with social emphases have largely been limited to private companies and government agencies [38]. Therefore, we need to encourage academia to move toward richer collaborations between the disciplines, with improved transparency and rigor. Last but not least, more studies in the development of ML methods to support social scientists' deep analyses may address how to generate and communicate results that can stand under careful scrutiny, as well as helping to identify potential pitfalls of the models in a real world context. Similar to how making an end-user friendly operating system (Ubuntu) could contribute back to the development of a comprehensive system (Debian) [70], making ML more applicable to social science in both quantitative and qualitative ways could bring direct benefits to ML.

By pursuing these objectives and prioritizing features and methods that can fulfill these goals, ML can be more human-centered overall. With this goal in mind, we discuss future research directions to engage ML and social science. We also review some of the works that have begun to define some of these directions.

4.1 Increase Machine Learning Transparency and Interpretability for Social Science Analysis

Since social science claims must be grounded in empirical evidence and connected or contrasted with existing theories, social scientists do not confine their analysis to statistical overviews (e.g., frequency of keyword usage during a particular period of time), but rather focus on how and why phenomena develop [70]. Thus, it is critical to make ML methods and results transparent and interpretable. As in our case of highlighting ambiguity, disagreement can be a useful and interpretable proxy for metrics of interest.

Nevertheless, ML models are not always interpretable. Complex models such as neural networks in deep learning are usually difficult to understand. Some recent work (e.g., [28, 54, 76]) has attempted to explain machine learning model behaviors via visualization. Other researchers have introduced techniques for interpretation of models that do not require understanding of the underlying algorithm [31, 49]. Although it is helpful to see how particular inputs are processed through a model and how varying inputs affects output, many other aspects of model development remain unclear.

For instance, it is usually an ad-hoc process to choose hyper-parameters of deep learning models (e.g., learning rate, optimization metrics, and batch size). When it is impossible to exhaustively experiment with millions of possibilities, it can be challenging to justify the meaning of these hyper-parameters in the context of the targeted analysis. As social scientists usually have some theoretical foundations for their analysis, it is still an open research challenge to ensure the model development process is interpretable and transparent enough to help social scientists to connect the design back to these theoretical foundations, which helps ensure the validity of derived scientific understanding [19].

Other previous research has also attempted to address the issue of transparency and interpretability of ML. For instance, Kim et al. proposed a generative approach to select and extract human interpretable features [36]. Brooks et al. suggested that using interpretable algorithms is important [10]. Others have introduced techniques for interpreting ML models that are algorithm-independent [37, 41, 48, 49]. Visualization is another approach that is considered to be helpful for interpretability [52, 65, 67]. One example of a visualization designed for interpretability is a self-organizing map to depict the high dimensional feature space of SVM models [71]. Several visualization methods have also been proposed to help better understand the learning processes and behaviors of artificial neural networks (ANNs). Darrah surveys techniques to help clarify ANN behavior, including Hinton diagrams, Bond diagrams, hyperplane diagrams and animators, Self-organizing Maps (SOMs), and Voronoi diagrams [17]. Smilkov et al. created an interactive visualization for understanding word-embedding [58]. However, research in the area of visualizing ML algorithms is still relatively scarce, and further exploration is needed to develop models and tools that make ML more human-understandable.

4.2 Explore Ways to Make Model Building A Meaningful Task in Social Science Practices

In addition to making ML more transparent and interpretable, exploring new ways to make model building a meaningful task in social science practices is an important direction for bridging the gap between ML and social sciences. In our case study of applying ML to qualitative coding, we found that identifying ambiguity in coding can be helpful in addition to existing attempts of automatic or semi-automatic coding methods [15, 16, 25, 40, 64, 77]. A key reason that this direction is promising is the alignment of the human effort required in ML (i.e., labeling data for training) with tasks that are meaningful in social science practices (i.e., qualitative coding). Even though the trained model may not be very accurate since the amount of the coded data is limited, social scientists can still gain better understanding of their data.

Such an approach is similar to the "machine teaching" paradigm, in which machines act as students and humans play the role of teachers in a form of interactive learning. This view has been suggested in work by Amershi et al. [1]. While we agree that interactive ML can be powerful, we want to extend the idea to the users' perspective: the power of interactive learning is not only in increasing the accuracy of the ML models, but also in providing a way for the users to reflect on their definitions of concepts, and in suggesting new perspectives from which to examine their data. This view is indeed similar to teaching experiences, where not only are the students learning, but the teachers may also learn from the questions students ask. Furthermore, as students usually have different issues in understanding a concept, teachers must consider different contexts and constantly refine their ideas and explanations. Although it is possible that a machine teaching approach may not result in the most accurate model for automated coding, as too little data might be labeled, the effort the teachers (the social scientists) spend is still be valuable for improving teachers' own understanding of the data. In this case, using ML can be seen as part of the coding process. Specifically, it can serve as a way of pinpointing potential issues in current codes, and we believe more work should be done to find new ways to think about the use of ML in the coding practice.

In order to identify other ways in which ML tasks are not merely meaningful for model-building but also beneficial for social science analysis, we need more studies on the work practice of social scientists to identify needs and opportunities. Works such as that of Brooks [9] that emphasize a human-centered design approach in understanding the needs of social scientists or interviews such as those conducted by Kandel et al. with data analysts [34] are critical for exploring this direction.

4.3 Developing a Common Ground for Both Social Scientists and The ML community

Building connections between ML and social sciences is not merely to facilitate the analytical process, but more importantly, to develop a common ground for both social scientists and members of the ML community. This direction should focus on encouraging publications built upon literature from both sides (e.g., [6]), as well as on further studying the current status and perspectives on the use of ML in social sciences. It should also explore how the ML community views ML applications in social science domains. In addition, efforts should be made to encourage collaborations between social scientists and the ML community. In this paper, we provided initial explorations of the use and challenges of ML in qualitative coding for social scientists, and future work should further investigate the factors that influence the usage and attitudes around ML for social scientists. Questions such as how ML methods are viewed, used, and applied in publications require a larger scale study. Additionally, the ways that publication venues evaluate ML work may also impact whether ML is used in social science. Without linking back to existing theories, methods that are merely novel may not be acceptable for traditional social science journals. All these factors may need more attention in order to build a common ground between both domains and establish communication mechanisms.

Some communities of researchers have begun attempting to create venues for addressing this issue. For example, in 2016, a number of major ML and HCI conferences held workshops aimed at making ML more human-centered (e.g., the Human Centered Data Science workshop in CSCW'16, Human Centered Machine Learning workshop in CHI'16, Human Interpretability in Machine Learning workshop in ICML'16, and Interpretable Machine Learning for Complex Systems workshop in NIPS'16). However, these workshops were held in conjunction with either an HCI or ML conference, and the interactions between the two communities are still limited. Thus, we envision greater efforts toward engaging the two communities, such as inviting submissions across communities in these workshops. In addition, similar efforts should be devoted toward collaboration with other social science disciplines, rather than focusing exclusively on HCI or CSCW communities. We encourage future research and discussion focused on resolving the existing tensions and connecting disciplines.

5 CONCLUSION

As the number of intelligent applications and the amount of collected data increase over time, managing data and making sense of human behaviors have become much more complex and require deep, sustained human consideration. Social scientists have considerable training and insight in the study of human behavior and social interactions, but ML applications for human data have yet to fully leverage their expertise. To help bridge the gap between these two fields, we highlighted challenges in applying ML to qualitative coding in social science. We also outlined our efforts to address these challenges by shifting the focus from building an ML model for automatic qualitative coding to highlighting ambiguity between qualitative coders. We characterized some of the dimensions of ambiguity and explored the use of disagreement as a proxy for ambiguity. Based on the findings from these studies, we proposed three future research directions toward better connecting ML with social science, especially within qualitative methods. We believe these research directions have the potential to make ML more tractable for social scientists as well as to provide a stronger theoretical foundation for the application of ML in social science research. Given the extensive body of knowledge about humans and society in social science, working to establish a stronger connection between these two fields will drive more human-centered ML research overall, supporting more powerful, usable tools that are applicable to a broad range of disciplines and problems.

ACKNOWLEDGMENTS

We acknowledge all the reviewers for their valuable feedback and the participants for their time.

A CODE DEFINITIONS

This appendix contains the code definitions for five mutually exclusive codes. These were the definitions provided to participants in our MTurk study of ambiguity.

A.1 Category Labels

- **Support** - Explicit or implicit support for Hillary Clinton (regardless of opinions about other candidates)
- **Rejection/Criticism** - Explicit or implicit rejection, criticism, or skepticism about Hillary Clinton, regardless of opinions about others.
- **Neutral** - Statement of fact or quotation from a candidate without explicit or implicit expression of opinion.
- **Unrelated** - Statements that do not reference Hillary Clinton or do not pertain to the 2016 U.S. election
- **Not Understandable** - Statements in which meaning cannot be deciphered (e.g., in a language other than English, gibberish, etc.)

A.2 Ambiguity

- **Unambiguous** - Tweet falls clearly into one and only category for label.
- **Ambiguous** - Tweet may fall into more than one category or its meaning is otherwise unclear.

A.3 Codes for Potential Sources of Ambiguity on Open-ended Feedback

- Limited context
- Interpretation may depend on individuals' perspective/point of view/experiences
- Could be reasonably interpreted in multiple categories
- Content was unfamiliar or not understandable so it was ambiguous
- No clear message or opinion about Clinton
- Potential sarcasm is ambiguous
- Particular phrases are ambiguous
- Url links may have contained more context
- Hashtags seem contradictory or confusing

REFERENCES

[1] Saleema Amershi, Maya Cakmak, W Bradley Knox, and Todd Kulesza. 2014. Power to the people: The role of humans in interactive machine learning. (2014).

[2] Susan Athey and Guido W Imbens. 2015. Machine learning methods for estimating heterogeneous causal effects. *stat* 1050, 5 (2015).

[3] Solon Barocas. 2014. Data Mining and the Discourse on Discrimination.

[4] Gabriela Beirão and JA Sarsfield Cabral. 2007. Understanding attitudes towards public transport and private car: A qualitative study. *Transport policy* 14, 6 (2007), 478–489.

[5] David M Blei, Andrew Y Ng, and Michael I Jordan. 2003. Latent dirichlet allocation. *Journal of machine Learning research* 3, Jan (2003), 993–1022.

[6] Tolga Bolukbasi, Kai-Wei Chang, James Y Zou, Venkatesh Saligrama, and Adam T Kalai. 2016. Man is to Computer Programmer as Woman is to Homemaker? Debiasing Word Embeddings. In *Advances in Neural Information Processing Systems*. 4349–4357.

[7] Natasha K Bowen and Shenyang Guo. 2011. *Structural equation modeling*. Oxford University Press.

[8] Danah Boyd and Kate Crawford. 2012. Critical questions for big data: Provocations for a cultural, technological, and scholarly phenomenon. *Information, communication & society* 15, 5 (2012), 662–679.

[9] Michael Brooks. 2015. *Human Centered Tools for Analyzing Online Social Data*. Ph.D. Dissertation. University of Washington.

[10] Michael Brooks, Katie Kuksenok, Megan K. Torkildson, Daniel Perry, John J. Robinson, Taylor J. Scott, Ona Anicello, Ariana Zukowski, Paul Harris, and Cecilia R. Aragon. 2013. Statistical affect detection in collaborative chat. In *Proceedings of the 2013 conference on Computer supported cooperative work*. ACM, 317–328. http://dl.acm.org/citation.cfm?id=2441813

[11] Claire Cain Miller. 2015. Algorithms and Bias: Q. and A. With Cynthia Dwork. New York Times. (2015). Retrieved from: http://www.nytimes.com/2015/08/11/upshot/algorithms-and-bias-q-and-a-with-cynthia-dwork.html.

[12] Kathy Charmaz. 2014. *Constructing grounded theory*. Sage.

[13] Peter Cihon and Taha Yasseri. 2016. A Biased Review of Biases in Twitter Studies on Political Collective Action. *Frontiers in Physics* 4 (2016). DOI: http://dx.doi.org/10.3389/fphy.2016.00034

[14] Jacob Cohen. 1960. A Coefficient of Agreement for Nominal Scales. *Educational and Psychological Measurement* 20, 1 (1960).

[15] Kevin Crowston, Eileen E. Allen, and Robert Heckman. 2012. Using natural language processing technology for qualitative data analysis. *International Journal of Social Research Methodology* 15, 6 (2012), 523–543.

[16] Kevin Crowston, Xiaozhong Liu, and Eileen E. Allen. 2010. Machine learning and rule-based automated coding of qualitative data. *Proceedings of the American Society for Information Science and Technology* 47, 1 (2010), 1–2.

[17] Marjorie Darrah. 2006. Neural Network Visualization Techniques. In *Methods and Procedures for the Verification and Validation of Artificial Neural Networks*. Springer US, 163–197. DOI: http://dx.doi.org/10.1007/0-387-29485-6_7

[18] N.K. Denzin and Y.S. Lincoln. 2011. *The SAGE Handbook of Qualitative Research*. SAGE Publications. https://books.google.com/books?id=AIRpMHgBYqIC

[19] Finale Doshi-Velez and Been Kim. 2017. Towards A Rigorous Science of Interpretable Machine Learning. In *eprint*

arXiv:1702.08608.

[20] Margaret Drouhard, Nan-Chen Chen, Jina Suh, Rafal Kocielnik, Vanessa Pena-Araya, Keting Cen, Xiangyi Zheng, and Cecilia R Aragon. 2017. Aeonium: Visual analytics to support collaborative qualitative coding. In *Pacific Visualization Symposium (PacificVis), 2017 IEEE*. IEEE, 220–229.

[21] Jeanine C Evers, Christina Silver, Katja Mruck, and Bart Peeters. 2011. Introduction to the KWALON experiment: Discussions on qualitative data analysis software by developers and users. In *Forum Qualitative Sozialforschung/Forum: Qualitative Social Research*, Vol. 12.

[22] Tiffany Derville Gallicano. 2013. An example of how to perform open coding, axial coding and selective coding. (2013). https://prpost.wordpress.com/2013/07/22/an-example-of-how-to-perform-open-coding-axial-coding-and-selective-coding/

[23] Barney G Glaser and Judith Holton. 2004. Remodeling grounded theory. In *Forum Qualitative Sozialforschung/Forum: Qualitative Social Research*, Vol. 5.

[24] Justin Grimmer. 2015. We Are All Social Scientists Now: How Big Data, Machine Learning, and Causal Inference Work Together. *PS: Political Science &amp; Politics* (2015).

[25] Justin Grimmer and Brandon M. Stewart. 2013. Text as Data: The Promise and Pitfalls of Automatic Content Analysis Methods for Political Texts. *Political Analysis* 21, 3 (2013), 267–297.

[26] David J Hand, Heikki Mannila, and Padhraic Smyth. 2001. *Principles of data mining*. MIT press.

[27] Moritz Hardt. 2014. How big data is unfair – Moritz Hardt. (Sept. 2014). https://medium.com/@mrtz/how-big-data-is-unfair-9aa544d739de

[28] Lisa Anne Hendricks, Zeynep Akata, Marcus Rohrbach, Jeff Donahue, Bernt Schiele, and Trevor Darrell. 2016. Generating visual explanations. In *European Conference on Computer Vision*. Springer, 3–19.

[29] Cheri Ann Hernandez. 2009. Theoretical Coding in Grounded Theory Methodology. *Grounded Theory Review* 8, 3 (2009).

[30] Judith A Holton. 2007. The coding process and its challenges. *The Sage handbook of grounded theory* Part III (2007), 265–89.

[31] Giles Hooker. 2004. Discovering additive structure in black box functions. In *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 575–580.

[32] H. V. Jagadish. 2015. Moving Past the Wild West Era for Big Data. IEEE Conference on Big Data Keynote Speech. (2015). Retrieved from: http://static1.squarespace.com/static/55da03c0e4b06261f858e037/t/56383353e4b0c0c519842550/1446523731270/ethics-BD.pdf.

[33] Anil K Jain. 2010. Data clustering: 50 years beyond K-means. *Pattern recognition letters* 31, 8 (2010), 651–666.

[34] Sean Kandel, Andreas Paepcke, Joseph M Hellerstein, and Jeffrey Heer. 2012. Enterprise data analysis and visualization: An interview study. *IEEE Transactions on Visualization and Computer Graphics* 18, 12 (2012), 2917–2926.

[35] Matthew Kay, Cynthia Matuszek, and Sean A Munson. 2015. Unequal representation and gender stereotypes in image search results for occupations. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*. ACM, 3819–3828.

[36] Been Kim, Julie A. Shah, and Finale Doshi-Velez. 2015. Mind the Gap: A Generative Approach to Interpretable Feature Selection and Extraction. In *Advances in Neural Information Processing Systems 28 (NIPS 2015)*.

[37] Lingpeng Kong, Nathan Schneider, Swabha Swayamdipta, Archna Bhatia, Chris Dyer, and Noah A Smith. 2014. A dependency parser for tweets. (2014).

[38] David Lazer, Alex (Sandy) Pentland, Lada Adamic, Sinan Aral, Albert Laszlo Barabasi, Devon Brewer, Nicholas Christakis, Noshir Contractor, James Fowler, Myron Gutmann, Tony Jebara, Gary King, Michael Macy, Deb Roy, and Marshall Van Alstyne. 2009. Life in the network: the coming age of computational social science. *Science (New York, N.Y.)* 323, 5915 (Feb. 2009), 721–723. DOI: http://dx.doi.org/10.1126/science.1167742 PMID: 19197046.

[39] Margaret D LeCompte. 2000. Analyzing qualitative data. *Theory into practice* 39, 3 (2000), 146–154.

[40] Seth C. Lewis, Rodrigo Zamith, and Alfred Hermida. 2013. Content Analysis in an Era of Big Data: A Hybrid Approach to Computational and Manual Methods. *Journal of Broadcasting & Electronic Media* 57, 1 (2013), 34–52.

[41] Scott Lundberg and Su-In Lee. 2017. A unified approach to interpreting model predictions. *arXiv preprint arXiv:1705.07874* (2017).

[42] Matthew B Miles and A Michael Huberman. 1985. *Qualitative data analysis*. Sage Newbury Park, CA.

[43] Michael Muller, Shion Guha, Eric P.S. Baumer, David Mimno, and N. Sadat Shami. 2016. Machine Learning and Grounded Theory Method: Convergence, Divergence, and Combination. In *Proceedings of the 19th International Conference on Supporting Group Work (GROUP '16)*. ACM, New York, NY, USA, 3–8. DOI: http://dx.doi.org/10.1145/2957276.2957280

[44] William Lawrence Neuman. 2005. *Social research methods: Quantitative and qualitative approaches*. Vol. 13. Allyn and Bacon Boston.

[45] Cathy O'Neil. 2016. *Weapons of math destruction: How big data increases inequality and threatens democracy*. Crown Books.

[46] Theodore M. Porter. 1994. *From Quetelet to Maxwell: Social Statistics and the Origins of Statistical Physics*. Springer Netherlands, Dordrecht, 345–362. DOI: http://dx.doi.org/10.1007/978-94-017-3391-5_11

[47] Nicholas Ralph, Melanie Birks, and Ysanne Chapman. 2015. The Methodological Dynamism of Grounded Theory. *International Journal of Qualitative Methods* 14, 4 (2015), 1609406915611576.

[48] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. 2016a. Model-agnostic interpretability of machine learning. *arXiv preprint arXiv:1606.05386* (2016).

[49] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. 2016b. Why should i trust you?: Explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, 1135–1144.

[50] A.P. Rovai, J.D. Baker, and M.K. Ponton. 2013. *Social Science Research Design and Statistics: A Practitioner's Guide to Research Methods and IBM SPSS*. Watertree Press. https://books.google.com/books?id=QId2AgAAQBAJ

[51] Cynthia Rudin. 2015. Can Machine Learning Be Useful for Social Science? (Sep 2015). http://citiespapers.ssrc.org/can-machine-learning-be-useful-for-social-science/

[52] D Sacha, M Sedlmair, L Zhang, JA Lee, D Weiskopf, SC North, and DA Keim. 2016. Human-centered machine learning through interactive visualization: Review and open challenges. In *Proceedings of the 24th European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning*.

[53] Johnny Saldana. 2015. An introduction to codes and coding. In *The coding manual for qualitative researchers*. 1–31.

[54] Wojciech Samek, Thomas Wiegand, and Klaus-Robert Müller. 2017. Explainable Artificial Intelligence: Understanding, Visualizing and Interpreting Deep Learning Models. *arXiv preprint arXiv:1708.08296* (2017).

[55] Cyrus Samii, Laura Paler, and Sarah Zukerman Daly. 2016. Retrospective Causal Inference with Machine Learning Ensembles: An Application to Anti-recidivism Policies in Colombia. *Political Analysis* 24, 4 (2016), 434–456.

[56] Burr Settles. 2010. Active learning literature survey. *University of Wisconsin, Madison* 52, 55-66 (2010), 11.

[57] Burr Settles. 2013. Machine Learning and Social Science: Taking The Best of Both Worlds. (2013). https://slackprop.wordpress.com/2013/02/05/machine-learning-and-social-science/

[58] Daniel Smilkov, Nikhil Thorat, Charles Nicholson, Emily Reif, Fernanda B Viégas, and Martin Wattenberg. 2016. Embedding Projector: Interactive Visualization and Interpretation of Embeddings. *arXiv preprint arXiv:1611.05469* (2016).

[59] Kate Starbird, Dharma Dailey, Ann Hayward Walker, Thomas M. Leschine, Robert Pavia, and Ann Bostrom. 2015. Social Media, Public Participation, and the 2010 BP Deepwater Horizon Oil Spill. *Human and Ecological Risk Assessment: An International Journal* 21, 3 (April 2015), 605–630. DOI: http://dx.doi.org/10.1080/10807039.2014.947866 00005.

[60] Anselm L Strauss. 1987. *Qualitative analysis for social scientists*. Cambridge University Press.

[61] Latanya Sweeney. 2013. Discrimination in online ad delivery. *Queue* 11, 3 (2013), 10.

[62] Renata Tesch. 2013. *Qualitative research: Analysis types and software*. Routledge.

[63] 2016. Economists are prone to fads, and the latest is machine learning. *The Economist (US)* (Nov 2016).

[64] Patrick Tierney. 2012. A qualitative analysis framework using natural language processing and graph theory. *The International Review of Research in Open and Distributed Learning* 13, 5 (2012), 173–189.

[65] Vanya Van Belle and Paulo Lisboa. 2013. Research directions in interpretable machine learning models.. In *ESANN*.

[66] Hal R Varian. 2014. Big data: New tricks for econometrics. *The Journal of Economic Perspectives* 28, 2 (2014), 3–27.

[67] Alfredo Vellido, José David Martín-Guerrero, and Paulo JG Lisboa. 2012. Making machine learning models interpretable.. In *ESANN*, Vol. 12. Citeseer, 163–172.

[68] Sarah Vieweg, Amanda L. Hughes, Kate Starbird, and Leysia Palen. 2010. Microblogging During Two Natural Hazards Events: What Twitter May Contribute to Situational Awareness. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '10)*. ACM, New York, NY, USA. DOI: http://dx.doi.org/10.1145/1753326.1753486

[69] Hanna Wallach. 2014. Big Data, Machine Learning, and the Social Sciences: Fairness, Accountability, and Transparency. (2014). https://medium.com/@hannawallach/big-data-machine-learning-and-the-social-sciences-927a8e20460d

[70] H. Wallach. 2016. *Computational Social Science: Toward a Collaborative Future*. Cambridge University Press. https://www.microsoft.com/en-us/research/publication/computational-social-science-toward-a-collaborative-future/

[71] Xiaohong Wang, Sitao Wu, Xiaoru Wang, and Qunzhan Li. 2006. SVMV–a novel algorithm for the visualization of SVM classification results. In *International Symposium on Neural Networks*. Springer, 968–973.

[72] Duncan J. Watts. 2004. The "New" Science of Networks. *Annual Review of Sociology* 30, 1 (2004). DOI: http://dx.doi.org/10.1146/annurev.soc.30.020404.104342

[73] Gregor Wiedemann. 2013. Opening up to big data: Computer-assisted analysis of textual data in social sciences. *Historical Social Research/Historische Sozialforschung* (2013), 332–357.

[74] Gregor Wiedemann and Wiedemann. 2016. *Text Mining for Qualitative Data Analysis in the Social Sciences*. Springer.

[75] Ian H Witten, Eibe Frank, Mark A Hall, and Christopher J Pal. 2016. *Data Mining: Practical machine learning tools and techniques*. Morgan Kaufmann.

[76] Kanit Wongsuphasawat, Daniel Smilkov, James Wexler, Jimbo Wilson, Dandelion Mane, Doug Fritz, Dilip Krishnan, Fernanda B Viégas, and Martin Wattenberg. 2017. Visualizing Dataflow Graphs of Deep Learning Models in TensorFlow. *IEEE transactions on visualization and computer graphics* (2017).

[77] Jasy Liew Suet Yan, Nancy McCracken, Shichun Zhou, and Kevin Crowston. 2014. Optimizing Features in Active Machine Learning for Complex Qualitative Content Analysis. *ACL 2014* (2014), 44.

Received January 2017; revised August 2018; accepted February 2018