SCM
Studies in Communication and Media

RESEARCH-IN-BRIEF

Establishing standards for human-annotated samples applied in supervised machine learning – Evidence from a Monte Carlo simulation

Manuelle Inhaltsanalysen für das maschinelle Lernen – Etablierung von Standards durch eine Monte-Carlo-Simulation

Corinna Oschatz, Marius Sältzer & Sebastian Stier

Corinna Oschatz (Ass.-Prof. Dr.), University of Amsterdam, Amsterdam School of Communication Science (ASCoR), Postbus 15791, 1001 NG Amsterdam, The Netherlands. Contact: c.m.oschatz@uva.nl
Marius Sältzer (Prof. Dr.), University of Oldenburg, Institute for Social Sciences, Department of Digital Social Science, Ammerländer Heerstraße 114-118, 26129 Oldenburg, Germany. Contact: marius.saeltzer@uol.de
Sebastian Stier (Prof. Dr.), GESIS – Leibniz Institute for the Social Sciences, Department Computational Social Science, Unter Sachsenhausen 6-8, 50667 Cologne, Germany / Professor for Computational Social Science, School of Social Sciences, University of Mannheim, Mannheim, Germany. Contact: sebastian.stier@gesis.org

RESEARCH-IN-BRIEF

Establishing standards for human-annotated samples applied in supervised machine learning – Evidence from a Monte Carlo simulation

Manuelle Inhaltsanalysen für das maschinelle Lernen – Etablierung von Standards durch eine Monte-Carlo-Simulation

Corinna Oschatz, Marius Sältzer & Sebastian Stier

Abstract: Automated content analyses have become a popular tool in communication science. While standard procedures for manual content analysis were established decades ago, it remains an open question whether these standards are sufficient for the use of human-annotated data to train supervised machine learning models. Scholars typically follow a two-stage procedure to obtain high prediction accuracy: manual content analysis followed by model training with human-annotated samples. We argue that a loss in prediction accuracy in supervised machine learning builds up over this two-stage procedure. In a Monte Carlo simulation, we tested (1) human coder errors (random, individual systematic, joint systematic) and (2) curation strategies for human-annotated datasets (one coder per document, majority rule, full agreement) as two sequential sources of accuracy loss of automated content analysis. Coder agreement prior to conducting manual content analysis remains an important quality criterion for automated content analyses. A Krippendorff's alpha of at least 0.8 is desirable to achieve satisfying prediction results after machine learning. Systematic errors (individual and joint) must be avoided at all costs. The best training samples were obtained using one coder per document or the majority coding curation strategy. Ultimately, this paper can help researchers produce trustworthy predictions when combining manual coding and machine learning.

Keywords: Supervised machine learning, prediction accuracy, impact of coder errors, impact of curation strategies, Monte Carlo simulation.

Zusammenfassung: Automatisierte Inhaltsanalysen sind ein häufig genutztes Instrument zur Beantwortung kommunikationswissenschaftlicher Forschungsfragen. Während Standards für die manuelle Inhaltsanalyse bereits vor Jahrzehnten etabliert wurden, bleibt zu klären, ob diese Standards für den Einsatz manuell generierter Daten im maschinellen Lernen ausreichen. Wissenschaftler folgen in der Regel einem zweistufigen Verfahren, um mit ihren Modellen qualitativ hochwertige Vorhersagen zu treffen: eine manuelle Inhaltsanalyse, gefolgt von einem Modelltraining mit diesen handcodierten Daten. Bei diesem Vorgehen können allerdings Verzerrungen entstehen, die wir in einer Monte-Carlo-Simulation identifizieren. Simuliert werden (1) Kodierfehler (zufällig, individuell systematisch, gemeinsam systematisch) und (2) Kuratierungsstrategien (ein Kodierer pro Dokument, Mehrheitsregel, vollständige

Übereinstimmung) als zwei aufeinanderfolgende Fehlerquellen. Die Ergebnisse zeigen, dass die Übereinstimmung der Codierer vor der manuellen Inhaltsanalyse ein wichtiges Qualitätskriterium für automatisierte Inhaltsanalysen bleibt. Koeffizienten von mindestens Krippendorff's Alpha = .8 sind wünschenswert, um zufriedenstellende Vorhersageergebnisse durch maschinelles Lernen zu erzielen. Systematische Fehler der Codierer (individuelle und gemeinsame) müssen unbedingt vermieden werden. Die besten Ergebnisse erzielen die Kurationsstrategien „ein Kodierer pro Dokument" oder „Mehrheitscodierung". Die Studie dient Forschern dazu, zuverlässige Vorhersagen beim Einsatz manueller Inhaltsanalysen im maschinellen Lernen zu erzielen.

Schlagwörter: Supervised machine learning, Genauigkeit der Vorhersagen, Einfluss von Kodierfehlern, Einfluss von Kuratierungsstrategien, Monte Carlo Simulation.

1. Introduction

Establishing intercoder reliability is "near the heart of content analysis; if the coding is not reliable, the analysis cannot be trusted" (Singletary, 1994, p. 294). This often-quoted statement illustrates the relation of the two essential quality criteria of content analysis – reliability and validity. *Reliability*[1] refers to the reproducibility of results (e.g., Krippendorff, 2004; Lombard et al., 2002). It is measured as agreement among coders and achieved when they reach consistent judgements on identical (media) messages. *Validity* refers to the "empirical truth" (Krippendorff, 1980, p. 71). It is the agreement of an empirical measurement with a measurement concept. However, validity cannot be measured directly. Instead, it is inferred from consistently reproduced data (high reliability) that are assumed to accurately describe the population of messages (high validity). Based on this theoretical linearity, intercoder reliability is used as an empirical proxy for valid results that can – in Singletary's (1994) words – be trusted.

1 The glossary for all italic terms can be found on OSF (https://osf.io/rkuj5/).

In past decades, it has been fiercely debated how reliability can be measured appropriately to ensure validity (e.g., Feng, 2014; Krippendorff, 2004; Lombard et al., 2002; Zhao et al., 2013). Despite the central importance of intercoder reliability as a quality criterion in manual content analysis, scholars rely on relatively vague coefficient benchmarks ranging from ≥.60 to ≥.90, depending on the research context (Geiß, 2021; Lombard et al., 2002; Neuendorf, 2017; Zhao et al., 2013). While such conventions have become widely established when conducting manual content analysis, the discussion of appropriate reliability criteria has gained new attention with the increasing popularity of automated content analysis (e.g., Grimmer & Stewart, 2013; Krippendorff, 2021; Song et al., 2020). However, social scientists will not be able to abandon manually coded data because (semi-)supervised machine learning depends on high-quality human annotations as training data to learn the meaning of texts (Grimmer et al., 2021; Grimmer & Stewart, 2013; Nelson et al., 2021; Sebők et al., 2022; van Atteveldt et al., 2021; Wu et al., 2022).

Scholars typically follow a two-stage procedure to obtain accurate predictions in machine learning. The first step is a manual content analysis based on inten-

sive coder training to establish sufficient agreement among carefully selected (e.g., Stoll et al., 2020) or crowdsourced coders (e.g., Budak et al., 2021). In the second step, the human-annotated training dataset is constructed, and the algorithm is trained and validated (Grimmer et al., 2021, p. 398). Coder agreement prior to the manual content analysis has been identified as the most important factor to secure valid predictions of a machine learning algorithm that aims to classify text into "positive" and "negative" examples (Baden et al., 2022, p. 14; Song et al., 2020, p. 558). The accuracy of such predictions is often evaluated via the *F1 score,* which is the harmonic mean of precision (the share of correctly classified positive examples) and recall (the share of correctly classified text among the total number of positive examples = sensitivity). However, recent work indicates that the relation of high coder agreement prior to data collection (high reliability) and the accuracy of computational predictions (F1 score) is not linear (Saeltzer et al., 2022; Viehmann et al., 2022).

We argue that the loss in prediction accuracy builds over the two-stage procedure of automated content analysis. We consider two sequential sources of accuracy loss that affect the quality of the training data: (1) Coder errors (random, individual systematic, joint systematic errors) measured as small disagreements that are overall considered sufficiently reliable for manual content analysis but scale up when used on big data. (2) The curation of the human-annotated dataset, i.e., the allocation of documents to human coders (one coder per document, majority rule, full agreement), impacts the range of positive cases included in the sample from which an algorithm can learn. Both can lead to flawed training data with reduced quality that does not represent the "empirical truth" (Krippendorff, 1980, p. 71). The results of machine learning are then evaluated against this empirical truth, resulting in a seemingly direct test metric of validity. Since training and test sets are typically derived from the same sample, a flawed human-annotated sample can result in high F1 scores, without necessarily being valid.

The goal of this paper is to test whether benchmarks and decisions currently applied in supervised machine learning produce valid predictions at scale for binary classifiers. We build on and connect previous works that examine the quality requirements of manual (e.g., Geiß, 2021) and automated (e.g., Song et al., 2020) content analyses. We conduct a Monte Carlo simulation to test our assumptions. Our contribution can help researchers produce trustworthy predictions when combining manual coding and machine learning.

2. Current standards affecting the quality of automated content analysis

2.1 Quality of the coding

Geiß (2021, p. 65) differentiates three types of coder errors: random individual errors, systematic individual errors, and joint systematic errors. While all error types are addressed in coder training and by randomly assigning materials to respective coders (Maurer & Reinemann, 2006; Rössler, 2017), they can affect the F1 score of predictions at scale.

2.1.1 Random individual errors

Random individual errors emerge when coders are occasionally inattentive. Generally, they have well understood the

phenomenon examined and follow the instructions documented in the codebook. However, coding a large amount of data can lead to sloppiness ("coder fatigue" Potter & Levine-Donnerstein, 1999, p. 271). This means that human coders incorrectly but randomly label text as a false-positive or false-negative example. False-positive labels (= incorrectly identified as a member of the class) lead to reduced precision of the predictions. False-negative labels (= incorrectly identified as *not* being a member of the class) lead to reduced recall. Such errors do not affect inferences about reality but lead to noisy predictions when applied to machine learning.

2.1.2 Systematic individual errors

Predominantly for latent content, however, it is more likely that coders produce systematic individual errors due to personal routines, attitudes, and heuristics (Geiß, 2021, p. 65). Potter and Levine-Donnerstein (1999, pp. 259–261) differentiate two types of latent content: pattern content and projective content. Pattern content assumes that there is an objective pattern that all coders can uncover based on symbols and cues. Such coding decisions are then based on the coders' experience and prior knowledge and might lead to judgment bias. For example, partisanship has been found to substantially affect an individual's judgment of political contexts and situations (Bakker et al., 2020; Kim, 2018). Thus, messages from political actors close to a coder's position might be coded more positively than messages from an actor of the opposing political spectrum. Projective content "shifts the focus more onto coders' interpretations of the meaning of the content" (Potter & Levine-Donnerstein, 1999, p. 259). Coders use schemas to interpret messages. For example, concepts such as incivility are judged against one's own standards (e.g., Nai & Maier, 2021). A comment might be judged as uncivil by one coder and as merely impolite by another. Coders might thus also vary in their sensitivity to detect latent variables. As it is impossible to discuss all facets and semantic appearances of a target concept during coder training, the likelihood of sensitivity bias increases with the size of the dataset and difficulty of the examined concept.

2.1.3 Joint systematic errors

Joint systematic errors are most problematic for machine learning, as they are not identifiable through reliability measures. Such errors might occur due to a lack of clarity in the codebook (e.g., missing definitions/examples), insufficient coder training, or "interpretative congruence" (Potter & Levine-Donnerstein, 1999, p. 271) when coders share interpretative schemas for coding projective latent content. Joint systematic errors can also occur over time (stability of reliability; Krippendorff, 1980, p. 71) during the coding process if coders do not code independently or simply become more/less sensitive toward the concepts examined due to greater familiarity with the task and the material. In this case, reliability tests will document high coder agreement, but the human annotations deviate from the empirical truth. While previous literature has acknowledged the presence of these three error types (Geiß, 2021), it has not yet considered to what extent they affect the relation between coder reliability and model performance.

RQ1: What is the relation between reliability and validity (F1 scores), given different types of coder errors?

2.2 Curation of the training dataset

The decision on curation strategies is essential, as it determines what data are used for training and validation. There are several options (Barberá et al., 2021) We focus on the consequences of three curation strategies: one coder per document, majority rule, or full agreement. These strategies differ in their consequences for the sample size and representativeness of the cases included in the human-annotated sample. Researchers typically aim to maximize their human coders' tasks according to the budget, i.e., most projects have a predefined sample size when coding begins. After sufficient agreement has been established among coders, a share of documents (e.g., social media posts) is randomly assigned to the human coders. When the one coder per document strategy is used, each coder receives a unique subsample, thereby maximizing the size of the human-annotated sample. In contrast, when applying the majority rule or full agreement, all coders receive the same subsample. At least three coders are required to apply the majority rule, according to which the code assigned by most coders is included in the training dataset. Even more strictly, complete correspondence between coder decisions is required when the full agreement rule is applied. Thus, the full agreement rule results in the smallest human-annotated sample given a predefined sample size.

Moreover, while improving the internal consistency and substantive quality of the sample compared to a one coder per document strategy, the majority and full agreement rules reduce its representativeness. This is particularly pronounced for full agreement, as only the 'easiest' and 'indisputable' cases are used for training. Consequently, the classifier will not see the more difficult cases as they are absent from the training dataset (Krippendorff, 2018, p. 285). The consequences of curation decisions have not yet been properly reflected, especially in conjunction with the different types of coder errors.

> *RQ2: To what extent do different curation strategies lead to over/underestimation of F1 scores, given different types of coder errors?*

3. The simulation study

To test our assumptions, we conducted a Monte Carlo simulation, which is a useful approach to evaluate the implications of diverse options in complex models (e.g., Geiß, 2021; Scharkow & Bachl, 2017). We proceeded in the following stepwise workflow:

I. Simulating a dataset with realistic characteristics (Dataset generation).
II. Simulating human annotators, inducing three error types (Quality of the coding).
III. Comparing Krippendorff's alpha as a broadly used reliability coefficient in communication science with F1 scores (RQ1).
IV. Testing how different curation strategies induce bias in our ability to correctly observe the performance of the machine learning classification (RQ2).

3.1 Dataset generation (I)

We simulate a binary dependent variable (concept absent = 0, present = 1), i.e., the empirical truth that researchers in reality cannot observe, and assign three parameters to the simulation:

- n is the number of data points to be "annotated",
- p is the prevalence of the target category, which we keep constant at 0.2,
- *Covars* are the characteristics of the document that relate to the true coding and must be interpreted by coders. We approximate this relationship by giving the covariates random (between -1 and 1) influences on the Y label (= dependent variable/true coding).

3.2 Quality of the coding (II)

3.2.1 Random individual errors

We simulate 21 error levels, ranging from 0 to 2.[2] In this way, we generate a variation in Krippendorff's alpha simultaneously with a change in F1 scores. In other words, we do not directly manipulate the alpha but model the factors that cause it. In all simulations, we observe how central metrics change by adding random errors in a symmetric manner. In formal terms, coders producing random errors can be described as:

$$\beta_0 + \beta_1 * x_1 + \beta_2 * x_2 + \beta_3 * x_3 + \varepsilon_{random}$$

3.2.2 Individual and joint systematic errors

For each coder, we manipulate how they code different features of the document. In a mathematical sense, coders make their decisions not with a true coefficient, as defined during data generation, but with a biased estimator. They will over- or underestimate the effect of the visible characteristics of a specific category:

$$\beta_0 + (\beta_1 + \varepsilon_1) * x_1 + (\beta_2 + \varepsilon_2) * x_2 + (\beta_3 + \varepsilon_3) * x_3 + \beta_2 * x_2 + \beta_3 * x_3 + \varepsilon_{random}$$

Coders are not homogeneous actors but have idiosyncrasies. If several coders with uncorrelated biases predict the same variable, the means cancel out. If these biases are correlated, this is not the case. For modeling these scenarios, we use 3 covariates and 5 coders, allowing for 3 variable-based biases. The fourth column is the random error. These specifications enable us to generalize coder biases in a succinct form. If all coders have a random error, the matrix looks like this:

$$\begin{pmatrix} 0 & 0 & 0 & 0 & E \\ 0 & 0 & 0 & 0 & E \\ 0 & 0 & 0 & 0 & E \\ 0 & 0 & 0 & 0 & E \\ 0 & 0 & 0 & 0 & E \end{pmatrix}$$

Now, we add variable-based biases that might cancel each other out

$$\begin{pmatrix} -.2 & -.2 & -.2 & 0 & E \\ .2 & .2 & .2 & 0 & E \\ -.2 & -.2 & -.2 & 0 & E \\ .2 & .2 & .2 & 0 & E \\ 0 & 0 & 0 & 0 & E \end{pmatrix}$$

or they might be correlated, i.e., the coders observe reality in a similarly distorted fashion.

$$\begin{pmatrix} .2 & .2 & .2 & 0 & E \\ .3 & .3 & .3 & 0 & E \\ .1 & .1 & .1 & 0 & E \\ .4 & .4 & .4 & 0 & E \\ 0 & 0 & 0 & 0 & E \end{pmatrix}$$

2 The simulation function requires a random error term for the coders. It should vary between 0 and 2, as 1 indicates a random classifier with an F1 of .5. We vary the amount of random error from 0 to 2 to also achieve low F1 scores and alphas. We separate the values from 0 to 2 into 21 bins (0.1, 0.2 etc.).

3.3 Comparing Krippendorff and F1 (III)

To investigate RQ1, we plot different levels of coder agreement against the F1 score, given different types of coder errors (see II). Every simulation is conducted 1,000 times with 1,000 documents. The number of covariates is set to 3, and the number of coders is 5. In total, we ran 63,000 simulations: 21,000 (21 error levels * 1000 iterations) * 3 (error structures) with 2000 codings each (= full simulated dataset). For the presentation of results, we focus on visual inference in graphical form.

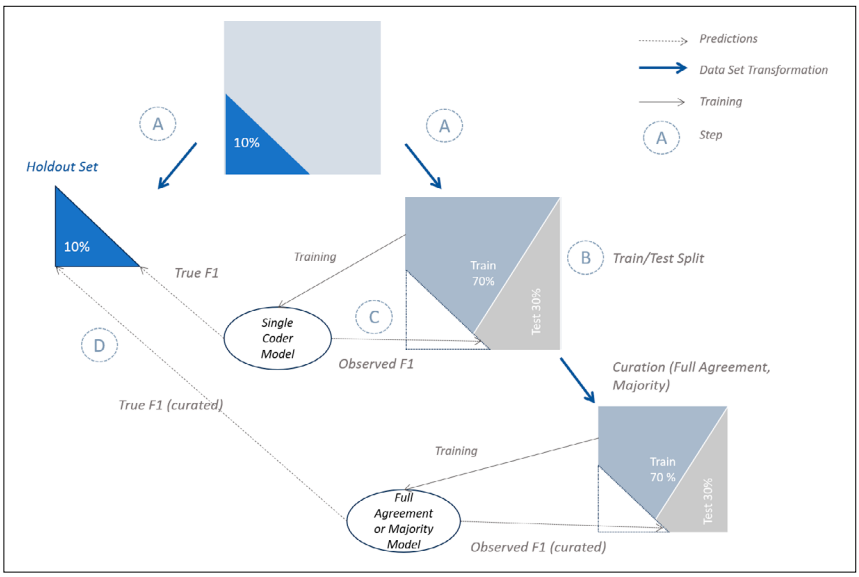
3.4 Simulating curation strategies (IV)

In the final step, we tested the impact of the coding errors when these data were used in machine learning (Figure 1). We simulated how three curation strategies affect our ability to assess the actual performance of a machine learning model, which enables us to evaluate the TRUE F1 score (coder annotations compared against the simulated – in reality unobservable – empirical truth) in contrast to the OBSERVED F1 score (coder annotations compared against the test set during supervised machine learning, which might differ from the empirical truth) after curation.

Our strategy involves splitting the dataset into three subsamples (holdout dataset, training dataset, and test dataset). We now elaborate on the process in detail. First, we sample (10%) a holdout dataset from the full simulated dataset (A). On the remaining full simulated dataset (90%), we perform a train/test split (70/30) and train a machine learning model, i.e., a logistic regression model on the covariates we simulated for the coding process (B).[3] We use this model to predict the test set. The performance of this model on the test data is the OBSERVED F1 score (Single Coder Model in Figure 1) (C). Then, we predict the holdout set (D). This is the TRUE F1 score. The difference between OBSERVED and TRUE performance is the mean absolute prediction error (MAPE), "to what degree observed F1 scores deviate from true F1 scores – when using the observed F1 score as the best possible "prediction" of the true F1 score" (Song et al., 2020, p. 557) (E). Next, we apply the curation strategies to the data that are split into training/test data (Majority/Full agreement Coder Model in Figure 1). We apply the same procedure as in (C) and (D). The difference between these experiments is the effect of curation on MAPE and the misinterpretation of the OBSERVED model performance against the TRUE model performance.

3 Any other machine model can be used. We chose the simplest model that best fits the binary coding process with a limited number of covariates

Figure 1. Workflow of the MAPE estimation



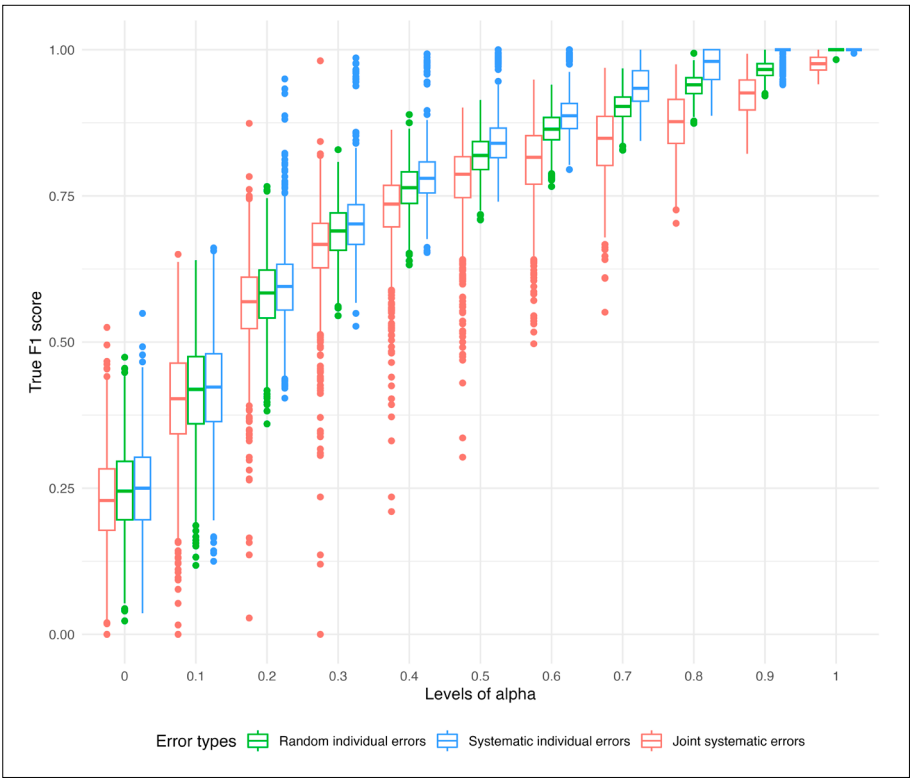
Note. MAPE is the difference between the observed F1 score and true F1 score.

4. Results

4.1 RQ1: The relationship between reliability and validity

Figure 2 shows how the relationship between coder reliability (different levels of Krippendorff's alpha) and the prediction accuracy (true F1 scores) differs given the three types of coder errors. Both Krippendorff's alpha and F1 scores are functions of the coding errors. Random individual errors (green), systematic individual errors (blue), and joint systematic errors (red) can still lead to acceptable Krippendorff's alpha levels, but they differ to varying degrees from the true F1 scores. Joint systematic errors affect the F1 score negatively across all levels of alpha. Interrater tests therefore indicate high reliability in terms of alpha values but miss systematic performance problems. The less intuitive finding is that if errors are systematic but compensating between individuals (as we would expect for uncorrelated errors on average), the F1 score is even better at all alpha levels than random errors.

Figure 2. Relation between reliability (Krippendorff's alpha) and validity (true F1 scores) given different types of coder errors



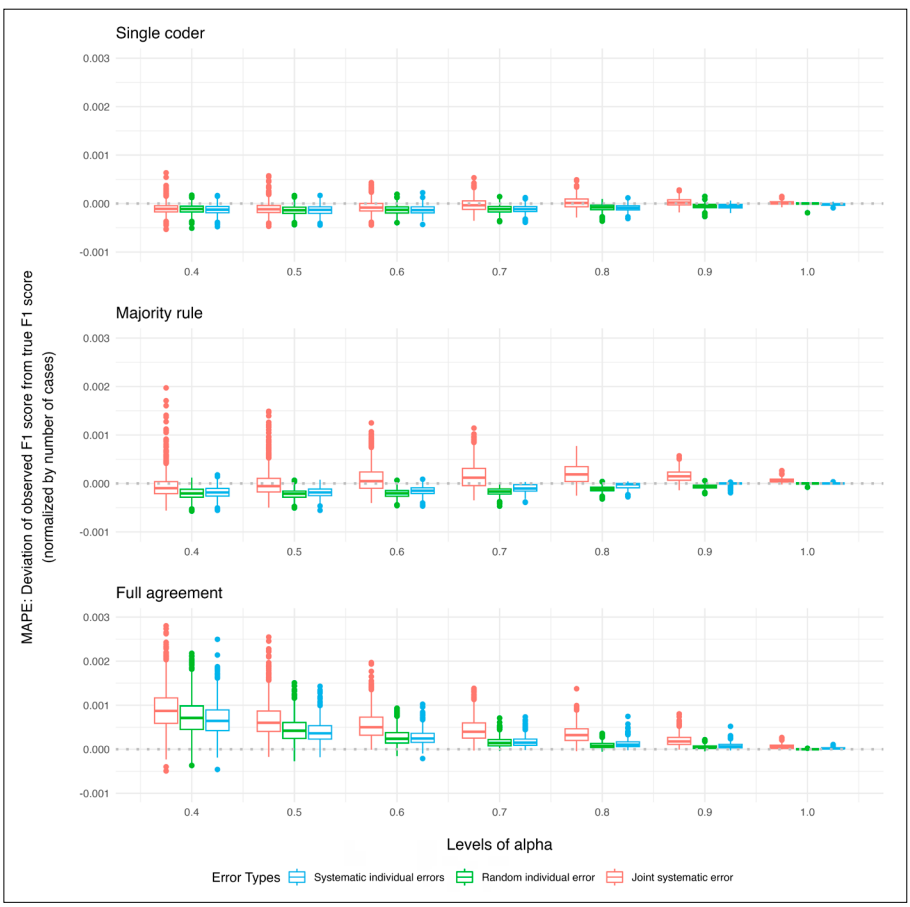
4.2 RQ2: The effects of curation strategies on F1 scores

Next, we focus on the effects of common strategies to curate coded training data for supervised machine learning (Figure 3). On the x-axis, we display the reliability of the coded dataset before curation, produced by introducing random errors. We focus on variations along alpha values of at least 0.4, as realistically, researchers would not train machine learning models based on data with lower reliability values. The y-axis shows the MAPE prediction errors, more precisely, the weighted differences between the true F1 score and the observed F1 score for different researcher strategies. The MAPE is normalized by dividing it by the fluctuating n in each simulation, as full agreement reduces the sample size depending on the amount of coder errors. Therefore, normalization of the y-axis is necessary to make the different samples comparable.

We simulate how three common practices of curating the training dataset (one coder per document, majority rule, full agreement) affect the training and validation upon the occurrence of random and joint systematic errors. As one would expect, MAPE converges to 0 in all three cases when the reliability of the coding (alpha) improves. Both majority rule and one coder per document tend to result in slight underestimation of the actual

model performance (positive MAPE). A lack of reliability in coding intensifies this underestimation. Nonetheless, these false negatives are less consequential in practice than the relatively larger overestimation of model performance (negative MAPE) when only taking the codes with full agreement into account. The errors of full agreement are most consequential, as positive values of MAPE indicate that the observed F1 score is higher than the true F1 score. For better interpretation, the values on the Y-axis can be interpreted as follows when readjusting the MAPE values based on the number of simulated cases for one concrete scenario: given that the reliability is alpha =.4, joint systematic errors (in red) under the full agreement rule could lead to a true F1 score that is up to .5 lower than the observed F1 score reported in a validation, and on average, the observed F1 score is .15 to .2 higher than the true F1 score. Full agreement therefore overestimates the validity of a machine learning model, leading to false-positive results.

Figure 3. MAPE given three different dataset curation strategies, different types of coder errors and levels of reliability (Krippendorff's alpha)



As two further robustness tests, we used the Brennan-Prediger coefficient instead of Krippendorff's alpha, and in another full iteration of the analysis, we repeated all analyses simulating 3 instead of 5 coders. The results of the two robustness tests in the supplementary material on OSF are very similar to the findings reported in the main paper.

5. Discussion and recommendation

This simulation study contributes to our understanding of the requirements of manually coded training data applied in supervised machine learning. We examined how coder errors (random, individual systematic, joint systematic) and researcher decisions on sample curation (one coder per document, majority rule, full agreement) affect the accuracy of predictions at scale. The simulation yields two main results: (1) Error types differentially affect Krippendorff's alpha and the true F1 score. In line with previous research (Song et al., 2020), we show that coder agreement prior to conducting the manual content analysis remains an important quality criterion for automated content analyses. A Krippendorff's alpha of at least .8 is desirable to achieve satisfying prediction results after machine learning. Systematic errors (individual and joint) must be avoided at all costs. To our knowledge, the simulation reveals for the first time the effect of a systematic error that researchers are usually not aware of. One practical way to reduce judgment biases (e.g., due to a coders' political orientation or gender) is to provide the coding material in the most anonymous way possible. We have had good experience using a shiny app database that allocates the text of social media posts to coders without disclosing the source of the post (Saeltzer et al., 2022). If coders are suspected of sharing political leanings, it might also make sense to find additional coders that cancel out these biases. Moreover, the database approach ensures random allocation of posts, as coders can only see one post at a time and cannot self-select certain posts (e.g., coding short posts first, coding all posts including the same picture at once). (2) Using the full agreement rule to curate a sample is the least preferable curation strategy. This leads to overconfidence in predictions (false positives), presumably because only simple and indisputable cases are detected (and validated). We thus recommend using either one coder per document (see also the recommendation by Barberá et al., 2021, p. 30) or the majority rule to include borderline cases that will be highly insightful for the learning algorithm (also see Card & Smith, 2018, p. 1644). A researcher must have a clear idea of the required sample size of the human-annotated sample. In the case of insufficient agreement among the coders, additional subsamples must be coded to fulfill the sample size requirements. Communicating such requirements to human coders might even increase their attentiveness and reduce random errors.

While the advent of large language models (LLMs) appears to have aided researchers in scaling up automated content analysis (Törnberg, 2023) reliability and bias of the Large Language Model (LLM, this by no means invalidates these findings. LLMs depend less on training data, but using them for scientific research requires transparency about potential errors that only manual annotation and the respective reliability measures can provide. If these annotations follow the same problems as discussed in this paper, understanding their effect

on validity remains central in the age of LLMs as well (Reiss, 2023).

We would also like to address limitations that have potential for future (simulation) studies. The findings are based on specific parameters we deemed realistic but might have to be adjusted in other research contexts. We share replication materials that enable researchers to probe the effects of specific configurations that might better represent their data (https://osf.io/rkuj5/). Finally, the patterns we identify usually become of substantive relevance in subsequent stages of analysis, namely, statistical hypothesis testing. In future research, the effect of systematic errors on downstream analyses should be considered.

6. Conclusion

To conclude, agreement among coders is central for valid predictions at scale. While this is common knowledge to scholars conducting content analysis, our standards for achieving agreement must be revisited with the increasing popularity of automated content analyses. We evaluated common researcher decisions to generate a human-annotated sample for machine learning. The best training samples were obtained using one coder per document or majority coding. From such samples, trusted conclusions that most accurately describe the population of documents can be obtained.

Funding

This manuscript emerged from a broad research program funded by the German Research Foundation (Deutsche Forschungsgemeinschaft, DFG) to study "Negative Campaigning in German Election Campaigns" (grant no: 441574527). We would like to thank the DFG for supporting our research.

References

Baden, C., Pipal, C., Schoonvelde, M., & van der Velden, M. A. C. G. (2022). Three gaps in computational text analysis methods for social sciences: A research agenda. *Communication Methods and Measures*, *16*(1), 1–18. https://doi.org/10.1080/19312458.2021.2015574

Bakker, B. N., Lelkes, Y., & Malka, A. (2020). Understanding partisan cue receptivity: Tests of predictions from the bounded rationality and expressive utility perspectives. *The Journal of Politics*, *82*(3), 1061–1077. https://doi.org/10.1086/707616

Barberá, P., Boydstun, A. E., Linn, S., McMahon, R., & Nagler, J. (2021). Automated text classification of news articles: A practical guide. *Political Analysis*, *29*(1), 19–42. https://doi.org/10.1017/pan.2020.8

Budak, C., Garrett, R. K., & Sude, D. (2021). Better crowdcoding: Strategies for promoting accuracy in crowdsourced content analysis. *Communication Methods and Measures*, *15*(2), 141–155. https://doi.org/10.1080/19312458.2021.1895977

Card, D., & Smith, N. A. (2018). The importance of calibration for estimating proportions from annotations. *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, 1636–1646. https://doi.org/10.18653/v1/N18-1148

Feng, G. C. (2014). Intercoder reliability indices: Disuse, misuse, and abuse. *Quality & Quantity*, *48*(3), 1803–1815. https://doi.org/10.1007/s11135-013-9956-8

Geiß, S. (2021). Statistical power in content analysis designs: How effect size, sample size and coding accuracy jointly affect hypothesis testing – A Monte Carlo sim-

ulation approach. *Computational Communication Research*, *3*(1), 61–89. https://doi.org/10.5117/CCR2021.1.003.GEIS

Grimmer, J., Roberts, M. E., & Stewart, B. M. (2021). Machine learning for social science: An agnostic approach. *Annual Review of Political Science*, *24*(1), 395–419. https://doi.org/10.1146/annurev-polisci-053119-015921

Grimmer, J., & Stewart, B. M. (2013). Text as data: The promise and pitfalls of automatic content analysis methods for political texts. *Political Analysis*, *21*(3), 267–297. https://doi.org/10.1093/pan/mps028

Kim, J. W. (2018). Online incivility in comment boards: Partisanship matters – But what I think matters more. *Computers in Human Behavior*, *85*, 405–412. https://doi.org/10.1016/j.chb.2018.04.015

Krippendorff, K. (1980). Validity in content analysis. In E. Mochmann (Ed.), *Computerstrategien für die Kommunikationsanalyse* (pp. 69–112). Campus. http://repository.upenn.edu/asc_papers/291

Krippendorff, K. (2004). Reliability in content analysis: Some common misconceptions and recommendations. *Human Communication Research*, *30*(3), 411–433. https://doi.org/10.1111/j.1468-2958.2004.tb00738.x

Krippendorff, K. (2018). *Content analysis: An introduction to its methodology* (4th edition). SAGE.

Krippendorff, K. (2021). A quadrilogy for (big) data reliabilities. *Communication Methods and Measures*, *15*(3), 165–189. https://doi.org/10.1080/19312458.2020.1861592

Lombard, M., Snyder-Duch, J., & Bracken, C. C. (2002). Content analysis in mass communication: Assessment and reporting of intercoder reliability. *Human Communication Research*, *28*(4), 587–604. https://doi.org/10.1111/j.1468-2958.2002.tb00826.x

Maurer, M., & Reinemann, C. (2006). *Medieninhalte: Eine Einführung* [Media contents: An introduction]. VS Verlag für Sozialwissenschaften.

Nai, A., & Maier, J. (2021). Is negative campaigning a matter of taste? Political attacks, incivility, and the moderating role of individual differences. *American Politics Research*, *49*(3), 269–281. https://doi.org/10.1177/1532673X2096554

Nelson, L. K., Burk, D., Knudsen, M., & McCall, L. (2021). The future of coding: A comparison of hand-coding and three types of computer-assisted text analysis methods. *Sociological Methods & Research*, *50*(1), 202–237. https://doi.org/10.1177/0049124118769114

Neuendorf, K. A. (2017). *The content analysis guidebook* (2nd edition). SAGE.

Potter, W. J., & Levine-Donnerstein, D. (1999). Rethinking validity and reliability in content analysis. *Journal of Applied Communication Research*, *27*(3), 258–284. https://doi.org/10.1080/0090988 9909365539

Reiss, M. V. (2023). *Testing the reliability of ChatGPT for text annotation and classification: A cautionary remark*. https://doi.org/10.48550/ARXIV.2304.11085

Rössler, P. (2017). *Inhaltsanalyse* [Content analysis] (3rd, fully revised edition). UVK.

Saeltzer, M., Oschatz, C., & Stier, S. (2022, March 26). *Classifying negative campaigning at scale: A study of candidates' social media communication across eight German elections*. Spring conference of the section "methods of empirical social research" of the German Association for Sociology, Virtual conference.

Scharkow, M., & Bachl, M. (2017). How measurement error in content analysis and self-reported media use leads to minimal media effect findings in linkage analyses: A simulation study. *Political Communication*, *34*(3), 323–343. https://doi.org/10.1080/10584609.2016.1235640

Sebők, M., Kacsuk, Z., & Máté, Á. (2022). The (real) need for a human touch: Testing a human-machine hybrid topic classification workflow on a New York Times corpus. *Quality & Quantity*, *56*(5), 3621–3643. https://doi.org/10.1007/s11135-021-01287-4

Singletary, M. W. (1994). *Mass communication research: Contemporary methods and applications*. Longman.

Song, H., Tolochko, P., Eberl, J.-M., Eisele, O., Greussing, E., Heidenreich, T., Lind, F., Galyga, S., & Boomgaarden, H. G. (2020). In validations we trust? The impact of imperfect human annotations as a gold standard on the quality of validation of automated content analysis. *Political Communication*, *37*(4), 550–572. https://doi.org/10.1080/10584609.2020.1723752

Stoll, A., Ziegele, M., & Quiring, O. (2020). Detecting impoliteness and incivility in online discussions: Classification approaches for German user comments. *Computational Communication Research*, *2*(1), 109–134. https://doi.org/10.5117/CCR2020.1.005.KATH

Törnberg, P. (2023). *ChatGPT-4 outperforms experts and crowd workers in annotating political Twitter messages with zero-shot learning*. https://doi.org/10.48550/ARXIV.2304.06588

van Atteveldt, W., van der Velden, M. A. C. G., & Boukes, M. (2021). The validity of sentiment analysis: Comparing manual annotation, crowd-coding, dictionary approaches, and machine learning algorithms. *Communication Methods and Measures*, *15*(2), 121–140. https://doi.org/10.1080/19312458.2020.1869198

Viehmann, C., Beck, T., Maurer, M., Quiring, O., & Gurevych, I. (2022). Investigating opinions on public policies in digital media: Setting up a supervised machine learning tool for stance classification. *Communication Methods and Measures*, 1–35. https://doi.org/10.1080/19312458.2022.2151579

Wu, X., Xiao, L., Sun, Y., Zhang, J., Ma, T., & He, L. (2022). A survey of human-in-the-loop for machine learning. *Future Generation Computer Systems*, *135*, 364–381. https://doi.org/10.1016/j.future.2022.05.014

Zhao, X., Liu, J. S., & Deng, K. (2013). Assumptions behind intercoder reliability indices. *Annals of the International Communication Association*, *36*(1), 419–480. https://doi.org/10.1080/23808985.2013.11679142