Detection of Falsified Interviews in Surveys of Households in Crisis Regions

Date: November 30, 2020
Authors: Siyuan Luo, Romina Jafaryanyazdi, Julie Keisler, Barbara Capl

In collaboration with IMPACT Initiatives
Developed in the context of Hack4Good, 3rd Edition

Abstract

An efficient and effective distribution of humanitarian aid calls for an accurate assessment of the help needed by affected people in crisis regions. This assessment is heavily reliant on data collected via household surveys. Due to limited geographical accessibility of crisis regions and cultural barriers, IMPACT Initiatives often needs to rely on third parties to conduct the surveys. This circumstance gives rise to the problem of possible data falsification by the enumerator, which leads to time-consuming cleaning processes in order to filter out potentially falsified interviews. This work uses a supervised algorithm from the family of ensemble decision trees in order to learn the patterns of potentially falsified interviews.

1 Introduction

Many people and families are affected by humanitarian crises such as displacement and poverty due to war conflicts. As the world becomes more digitized, NPO's can benefit from its possibilities to reach more people in need and distribute the resources of the donors with more efficiency and effectiveness. In order to reap these possibilities, NPO's are heavily dependent on accurate data about the social, financial and health-related conditions of people affected by humanitarian crises.

For this, IMPACT Initiatives conduct household surveys in affected crisis regions in multiple countries in order to assess the needs of humanitarian aid. In many target regions, surveys can only be conducted by engaged third parties and collaborators, a circumstance which results in the problem of data falsification from the surveyor's side. Past research has found that intentional data falsification can occur for various reasons, such as a payment based on the number of surveys completed, hard to reach households or because sensitive questions may be uncomfortable to ask.[1] The consequence of the presence of possible data falsification in a survey is a time-consuming data cleaning process in order to find potentially falsified interviews.

So far, labeling interviews as clean or falsified is done by human intervention and is often done by analyzing a vast set of possible response combinations within the respective interview. Apart from being time-consuming, automation based on such heuristics is not scalable. The goal of this project is therefore to develop a tool for detection of falsified interviews in surveys in order to speed up and simplify the data cleaning process. As the number and frequency of conducted surveys is planned to be increased in coming years, this project could be beneficial for the NPO to save time and resources. We approach this goal with *XGBoost*[2], a supervised machine learning algorithm from the family of decision tree classifiers.

2 Expected Impact

Our trained classification model can be integrated into the existing cleaning pipeline of the IMPACT Initiatives and used to find potentially falsified interviews in a

[1] Birnbaum, B. (2013). Algorithmic approaches to detecting interviewer fabrication in surveys (Doctoral dissertation).
[2] Chen, T., Guestrin, C. (2016, August). Xgboost: A scalable tree boosting system. In Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining (pp. 785-794).

new set of interview instances. The model can be integrated such that it returns the input survey with labels “clean” or “falsified” for each interview, which can then be double-checked by the data experts at the NPO.

With this, there is no need to examine the whole survey manually anymore and the human data expert can focus on the subset identified by the model as critical. The model creates labels based on a certain probability threshold, which is 0.5 by default. This means that if the probability of an interview to be falsified is larger than 50%, the interview is classified as “falsified”. In our case, the NPO favours to keep the number of false negatives[3] as low as possible, even if at the expense of having more false positives[4]. In order to get such a more conservative result, the threshold can be set to a much lower number, while keeping in mind that this will on the other end increase the number of false positives and hence the number of interviews that will have to be double-checked by a human data expert. Hence there is a performance/cost trade-off. The best found threshold results in around overall 40% less cleaning time and less than 2% missed falsified interviews.

3 Approach

Our data set, kindly provided by IMPACT, is a survey from Afghanistan consisting of around 40’000 household interviews, each with around 590 questions. There are four additional smaller surveys with fewer questions each and only filled out if applicable to the household. The whole survey had already been cleaned and labeled by the IMPACT team, which enabled us to create a labeled data set for training and testing purposes. We approached the problem in two steps, data engineering and modeling.

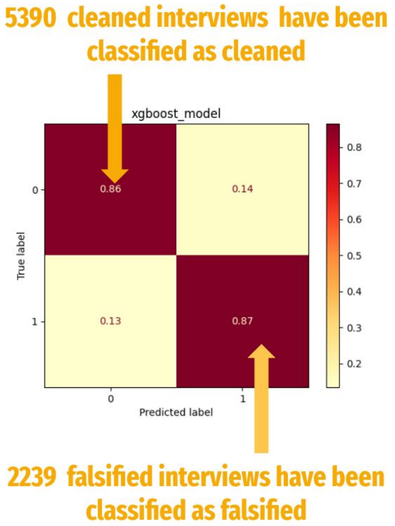
In the data engineering part we converted categorical variables to dummy variables, removed redundant columns and created additional, more meaningful features out of certain questions. Finally, dealing with the large number of missing values was a challenge and a necessity, since many machine learning models can not process them. We observed two possible main reasons of non-randomly missing values in our data set.

Missing values, especially a larger number of them within the same interview, could be an indicator of a falsified interview where the enumerator did not manage to come up with a fitting value. On the other hand, values could also be missing if a question was not applicable. In both cases, values are not missing at random, therefore the standard procedure of only filling them with the mean or median would result in loss of potentially relevant information. Therefore, we created a separate column for each of the columns that had missing values. In these separate columns, we indicated for each row if the original column contained a missing value or not. Subsequently, we filled the missing values in the initial columns with standard procedure according to the meaning of the variable. Finally, we split our clean data set in a train and a test set.

In the modeling part of our project we constructed different classifiers such as *Random Forest*, *Naive Bayes* and multiple *Neural Network* architectures. Training and hyper-parameter tuning was conducted on the train set via cross-validation. The best prediction performance on the test set was achieved by ***XGBoost***, a boosting decision tree algorithm.

4 Results

As outlined previously, the best prediction performance was achieved with the *XGBoost* classifier. Since the data set was highly imbalanced, we measured the performance using performance measures such as balanced accuracy[5] and F1 score[6]. We consulted the confusion matrix in order to get an overview over the number of falsified interviews wrongly classified as "clean", a number the NPO wishes to keep as low as possible. The confusion matrix for the base case (probability threshold as explained before at 0.5) is displayed in *Figure 1* and comes with a balanced accuracy of 87%.


5390 cleaned interviews have been classified as cleaned

2239 falsified interviews have been classified as falsified

[3] Falsified interviews classified wrongly as "clean".

[4] Clean interviews classified wrongly as "falsified".

[5] Accuracy measures the number of correctly identified falsified interviews among all interviews. Balanced accuracy is accuracy corrected for the class imbalance in the data set.

[6] Please refer to `https://en.wikipedia.org/wiki/F-score` and `https://scikit-learn.org/stable/modules/generated/sklearn.metrics.f1_score.html` for more information.

Figure 1: *Figure: Confusion matrix based on test set. Model: XGBoost, separating probability threshold 0.5.*

The finally delivered *XGBoost* model was tuned on the train set using cross-validation and uses random undersampling to balance the data set. Additionally, the pipeline contains various feature selection steps based on different feature selection methods. The codes and final results can be found on the GitLab repository[7]. By changing the probability threshold of the model, the number of false positives and false negatives can be regulated as wished.

In addition to *XGBoost*, we examined and tested other classification algorithms such as *Random Forest*, *Support Vector Machines* and *CatBoost*. Among *Neural Network* architectures, different combinations of Convolutional layers, LSTM, and Dense layers were built and tested as well. None of the other classifiers as well as the *Neural Networks* couldn't beat the prediction performance of *XGBoost*.[8] The inferior performance of the *Neural Networks* in our case might be because of the large number of features and the relatively small data set, a problem of fitting noise as well as a possibly suboptimal hyperparameter tuning.

Finally, we deliver the project split in two separate main pipelines, data engineering and modeling. These pipelines are kept apart in order to simplify the usage and extension of the project.

5 Difficulties, Limitations and Risks

One of the difficulties we faced during the project was the large number of survey questions, from which most of them were categorical. Additional data engineering based on domain knowledge could potentially enhance the performance of the supervised algorithm. With such knowledge, one could also try a more sophisticated unsupervised modelling approach.

One of the main limitations of our model lies in the fact that as a supervised learning approach, a labelled data set is needed in the beginning for training the model and evaluating the performance of the test set. In our case, these labels still need to be generated by human annotation, as done up to now. Once the model has been trained on these labels, the trained model can be used for detecting falsified interviews automatically in subsequent surveys. There is reason to believe that the way surveyors falsify interviews and hence the respective pattern of answer combinations will be similar in subsequent years or also across different regions. However, we did not get to test this assumption on any other survey and suggest this to be done as a next step in order to test the applicability of the model.

Finally, it needs to be emphasized that the labels generated by human intervention are not the ground-truth. There is a chance that actually clean interviews have been labeled as falsified or falsified interviews were not detected by human data experts of the NPO. Since the model uses these human-created labels as input, it can only learn the patterns of falsification related to what has been detected by human annotators. Therefore, it would be risky to rely too much on the model based on the current model inputs only and the process of human labeling for creating training data needs to be revised regularly.

6 Conclusion and Recommendations

In conclusion, we find that supervised learning is capable enhancing the process of detecting falsified interviews in surveys. Using our trained *XGBoost* classifier, it is possible to save around 40% of cleaning time while missing less than 2% of falsified interviews. As already discussed, the performance of the model could be enhanced by incorporating more domain knowledge into the data engineering process in order to create even more meaningful variables out of the given interview questions.

We recommend to test the model on future surveys for Afghanistan and extend the testing to surveys from other regions but similar questions and structures, in order to elaborate the generalization performance of the classifier. Performance and robustness can be enhanced by training the trained model on additional labeled data.

Due to the limited project time, it was not possible to investigate all conceivable approaches to solving the problem. In future projects, an extension with unsupervised learning methods could be beneficial and would be interesting to investigate. Especially paired with domain knowledge, an interpretable algorithm could be created in this field. An unsupervised algorithm could possibly detect falsified interviews human annotators did not detect. However, in order to evaluate the model performance, there would be a need for ground-truth labels. We also recommend to investigate a possible approach with active learning, which is particularly interesting if the number of labels available for training the model are low or must be kept low for cost efficiency.

[7] Please refer to: `https://gitlab.com/analytics-club/hack4good/hack4good-fall-2020/impact.git`.

[8] The final best-performing *Neural Network* architecture is included in the final delivered code for future purposes.