Three Challenges in Utilizing Machine Learning to Predict Human Behavior from Observational Data

Thomas Manzini
tmanzini@tamu.edu
Texas A&M University
College Station, Texas, United States

Priyankari Perali
perali@tamu.edu
Texas A&M University
College Station, Texas, United States

Robin R. Murphy
robin.r.murphy@tamu.edu
Texas A&M University
College Station, Texas, United States

ABSTRACT

This paper outlines the three principal challenges encountered during the machine learning efforts of the Real-Time Adaptive Systems (R-TAPS) project to learn the behavior of chemical plant workers and provides recommendations for future HRI projects that face similar problems. This paper specifically focuses on data labeling, annotation processes, and model evaluation. The R-TAPS machine learning efforts aimed to predict worker behavior during task execution in real-time. It employed a step-level label system, which caused difficulties in predicting worker behavior on a timestamp level. The annotation process that was carried out lacked uniformity, leading to inconsistencies in the data entries. The model performance presentation caused confusion due to multiple performance values and a lack of understanding of what metric to evaluate. In response, this paper offers recommendations that address each challenge for future efforts.

CCS CONCEPTS

• **Computing methodologies** → **Machine learning**; • **Human-centered computing** → **Interaction design**.

KEYWORDS

Human Robot Collaboration, Human Robot Interaction, Machine Learning

ACM Reference Format:
Thomas Manzini, Priyankari Perali, and Robin R. Murphy. 2024. Three Challenges in Utilizing Machine Learning to Predict Human Behavior from Observational Data. In *Companion of the 2024 ACM/IEEE International Conference on Human-Robot Interaction (HRI '24 Companion), March 11–14, 2024, Boulder, CO, USA.* ACM, New York, NY, USA, 3 pages. https://doi.org/10.1145/3610978.3640729

1 INTRODUCTION

With the increasing demand for human-robot collaboration (HRC) in various industries, predicting human behavior is essential. In industrial applications, given the potential occurrence of unforeseen events or human error, robots adapting based on predicted worker behavior can avoid collisions and injuries, allowing for safer human-robot interaction (HRI) [3]. Previous work in HRI that involves human behavior modeling is spread across various applications, and many utilize machine learning techniques. Tsitos et al.[9] present the potential for real-time predictions of human behavior within the application of competitive tasks through various machine learning classifiers. Liu et al.[4] utilize human behavior modeling for dealing with varying team members' expertise in tasks to adapt the structure in an effort to improve human-robot teaming. Schirmer et al.[8] focus on predicting anomalies or unexpected human behavior with a LSTM model and their possible effect in an industrial assembly use case. Al-Saadi et al.[1] propose using human behavior predictions for any necessary conflict resolutions in collaborative tasks with a random forest classifier.

To enable these future scenarios, the authors embarked upon the Real-Time Adaptive Procedure System project (R-TAPS). The R-TAPS project's objective is real-time worker behavior prediction during task execution. The motivation is that these predictions will allow for adaptions and interventions in high-risk environments in an attempt to minimize risk and worker errors. Labeled observational data collected from workers performing three different procedural tasks in a chemical plant was used to train a machine learning model that would predict worker behavior during task execution in real time. This led to a complex data management and model training pipeline, the development of which resulted in three clear lessons learned.

The paper's primary contribution is a description of the three challenges and recommendations to consider when utilizing machine learning in similar HRC and HRI applications. The rest of the paper details these three challenges and provides recommendations for each (Section 2) and ends with a summary and concluding remarks (Section 3).

Table 1: "Work As Done" (WAD) Labels

WAD Label	Description
A	Step skipped
B1	Step done out of order but right action
B2	Step done out of order but wrong action
C	Step done in order but wrong action
D	Step done as prescribed

2 CHALLENGES

The primary challenges faced were with respect to the data labels, annotation process, and model evaluation; these are discussed separately, and recommendations are provided for each.

HRI '24 Companion, March 11–14, 2024, Boulder, CO, USA

ACM ISBN 979-8-4007-0323-2/24/03.
https://doi.org/10.1145/3610978.3640729

2.1 Data Labels

There were five different "Work as Done" (WAD) labels [5] considered, shown in Table 1, that can describe the worker's behavior in completing a task with a given procedure. This label taxonomy has been used in the literature to compare the performance of workers as they complete steps in procedures [5]. However, the nature of these labels caused substantial friction during the machine learning process.

At first glance, this label taxonomy appears reasonable in this setting. Nevertheless, because the R-TAPS project was focused on predicting worker behavior in real time, at the timestamp level, this taxonomy became inadequate. This is because this taxonomy is designed to be applied at the step level instead of at the timestamp level. In other words, the labels were associated with steps within the procedures rather than timestamps within the task execution.

At the project's inception, the expectation was that a model could predict the WAD label of the worker's *current step*. However, this was not possible because the current step cannot be known definitively. This is because workers do not necessarily complete steps in order or discretely, and they may return to steps that they have previously started.

As a result, it is impossible to disambiguate, in real-time, the step that the worker is performing. Since the data was annotated at the step level, inference must be done at the step level. It quickly became unclear what step should be the target of that inference.

The output of the model would be a distribution over WAD labels. This approach is beneficial because it is easy to train: it is a simple classification problem, and it is connected to the procedure in a way that can be leveraged in a real-world setting. However, as discussed, these WAD labels are temporally aligned, and thus models trained on this data cannot predict when an intervention should be made as the worker progresses through the procedure. In hopes of addressing this problem, predictions for all steps in the worker's task needed to be performed in parallel, and thus, models were trained with this prediction target in mind.

This prediction target created a substantial amount of noise in the target function. This is because the same real-time features could correspond to different WAD labels when conditioned on different steps.

In short, if predictions must be at the timestamp level, labels must be at the timestamp level. This incongruence between the operational use case and the available data hampered the machine learning process. In the future, should timestamp level prediction be required, it is highly recommended that the data be annotated at the timestamp level.

2.2 Annotation Process

The annotation process involved paid annotators who were assigned videos of workers completing tasks to watch and annotate. Annotators were instructed to annotate all of the worker's actions in executing the given task, consisting of each procedure step, the worker's behavior, and other relevant data points. Each annotator was provided with a template spreadsheet which they then filled out for the various data fields that were required.

This process became difficult to manage because there were no guardrails in place to ensure the uniformity of data entered across the different annotators. Certain annotators would label complete spans of time for the videos, while others would annotate only the transition points. This would result in certain spans of data within a video being unannotated, annotated with spurious data, or annotated with data that is incorrectly formatted.

Additionally, different annotators developed their own shorthand for certain fields, and each annotator would have different ways of spelling colloquial terms (such as "walkytalkie," "walkietalky," "radio," etc.). Some annotators utilized different timestamp formats, which required different parsers during data cleaning. This effect is well known in language technologies and has been utilized to generate diverse training data when desirable [7]. Finally, the Google Sheets UI would occasionally reformat certain numerical fields resulting in malformed data that had to be recovered manually. This type of inconsistency between the different annotators, tasks, and fields became an additional source of noise in an already noisy dataset.

The lessons learned offer two suggestions for the future focused on a theme: guardrails. In other words, additional infrastructure is required to manage label noise generated by annotators.

(1) Utilize an additional layer of processing to ensure uniformity among the labels generated by the annotators. This would ideally be a layer of software (e.g., a data entry tool) that verifies the consistency of the entered data between the annotators. Alternatively, inter-annotator agreement or averaging annotations could be leveraged as another means to manage label noise [2, 6].
(2) Prior to the annotation process, develop an ontology/taxonomy of devices, subtasks, and actions for each specific procedure. Then during the annotation process, instruct annotators to select from this ontology/taxonomy while entering data. This will ensure uniformity among the entered data and serve to decrease noise.

The utilization of these two suggestions would have mitigated the vast majority of noise present in the R-TAPS project. This guidance will translate to future related machine learning projects.

2.3 Evaluation

At its core, the R-TAPS project was a multiclass classification effort focused on the A, B1, B2, C, and D WAD labels discussed above. Early on in this project, it was discussed that certain WAD labels may be more valuable than others (e.g., it may be more important to predict steps that were completed incorrectly rather than correctly). As a result, model performance metrics were presented for each individual class rather than collectively using an aggregation function. This decision ultimately generated confusion as it became difficult to determine when one trained model was performing better than another.

In the future, an effort should be made to converge on a single numerical value to judge model performance. It may not always be possible to converge on an ideal metric. However, even when there is uncertainty about the relevance of that metric. It can be further refined. The complete class-level performance should not be discarded but rather should be presented alongside the single value for situational awareness. Within this work, once the team aligned on a weighted sum of class labels, the ablation studies that were

conducted became far more clear, and understanding which models and features were better than others was an easier conversation to have.

3 CONCLUSION

This paper outlines three challenges encountered during the R-TAPS machine learning efforts and provides recommendations for each.

(1) When making real-time predictions, data should be annotated on a timestamp level.
(2) To reduce noise due to inconsistencies within an annotated dataset, utilize an additional verification layer for annotation consistency between annotators and develop an ontology/taxonomy for annotators to select from.
(3) To increase model performance comprehension, efforts should be made to converge on a single value to represent model performance so comparisons can be easily made.

As machine learning techniques are utilized more in HRC and HRI applications, these recommendations should taken into consideration.

ACKNOWLEDGMENTS

This material is based upon work supported by the National Science Foundation under Grant No. 2106963.

REFERENCES

[1] Zaid Al-Saadi, Yahya M Hamad, Yusuf Aydin, Ayse Kucukyilmaz, and Cagatay Basdogan. 2023. Resolving Conflicts During Human-Robot Co-Manipulation. In *Proceedings of the 2023 ACM/IEEE International Conference on Human-Robot Interaction*. 243–251.
[2] Ron Artstein. 2017. Inter-annotator agreement. *Handbook of linguistic annotation* (2017), 297–313.
[3] Abdelfetah Hentout, Mustapha Aouache, Abderraouf Maoudj, and Isma Akli. 2019. Human–robot interaction in industrial collaborative robotics: a literature review of the decade 2008–2017. *Advanced Robotics* 33, 15-16 (2019), 764–799.
[4] Ruisen Liu, Manisha Natarajan, and Matthew C Gombolay. 2021. Coordinating human-robot teams with dynamic and stochastic task proficiencies. *ACM Transactions on Human-Robot Interaction (THRI)* 11, 1 (2021), 1–42.
[5] Atif Mohammed Ashraf, Changwon Son, S Camille Peres, and Farzan Sasangohar. 2021. Navigating operating procedures in everyday work in a petrochemical facility: A comparative analysis of WAI and WAD. In *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, Vol. 65. SAGE Publications Sage CA: Los Angeles, CA, 623–627.
[6] Stefanie Nowak and Stefan Rüger. 2010. How reliable are annotations via crowdsourcing: a study about inter-annotator agreement for multi-label image annotation. In *Proceedings of the international conference on Multimedia information retrieval*. 557–566.
[7] Abhilasha Ravichander, Thomas Manzini, Matthias Grabmair, Graham Neubig, Jonathan Francis, and Eric Nyberg. 2017. How Would You Say It? Eliciting Lexically Diverse Dialogue for Supervised Semantic Parsing. In *Proceedings of the 18th Annual SIGdial Meeting on Discourse and Dialogue*, Kristiina Jokinen, Manfred Stede, David DeVault, and Annie Louis (Eds.). Association for Computational Linguistics, Saarbrücken, Germany, 374–383. https://doi.org/10.18653/v1/W17-5545
[8] Fabian Schirmer, Philipp Kranz, Jan Schmitt, and Tobias Kaupp. 2023. Anomaly Detection for Dynamic Human-Robot Assembly: Application of an LSTM-based autoencoder to interpret uncertain human behavior in HRC. In *Companion of the 2023 ACM/IEEE International Conference on Human-Robot Interaction*. 881–883.
[9] Athanasios C Tsitos, Maria Dagioglou, and Theodoros Giannakopoulos. 2022. Real-time feasibility of a human intention method evaluated through a competitive human-robot reaching game. In *2022 17th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*. IEEE, 1080–1084.