A Feature Space Focus in Machine Teaching

Lars Holmberg
Supervised by Paul Davidsson and Per Linde
Department of Computer Science and Media Technology Malmö University Malmö, Sweden
lars.holmberg@mau.se

***Abstract*—Contemporary Machine Learning (ML) often focuses on large existing and labeled datasets and metrics around accuracy and performance. In pervasive online systems, conditions change constantly and there is a need for systems that can adapt. In Machine Teaching (MT) a human domain expert is responsible for the knowledge transfer and can thus address this. In my work, I focus on domain experts and the importance of, for the ML system, available features and the space they span. This space confines the, to the ML systems, observable fragment of the physical world. My investigation of the feature space is grounded in a conducted study and related theories. The result of this work is applicable when designing systems where domain experts have a key role as teachers.**

***Index Terms*—Machine learning, Machine Teaching, Human in the loop**

I. Introduction

Contemporary Machine Learning (ML) is often data-hungry and/or compute hungry and focused on finding correlations in datasets. These systems give impressive results in areas like self-driving cars and image recognition. This currently dominating connectionist approach has its limitations since it lacks logic reasoning and the possibility to identify causal relations.

There is increasing interest in identifying causal relations [1], interpreting predictions [2] as part of a quest for trustworthy AI [3]. The approach I take is to compensate for ML shortcomings by increasing a human domain expert's agency during the total lifetime of the system.

Development of ML systems traditionally starts with data gathering and labeling followed by analysis, algorithm selection, etc. until the trained model can be deployed [4]. Models developed as outlined above risk to become static, hard to evaluate and risks degrade due to changes in the context they are deployed into. When unacceptable degrading is identified the model has to be updated, which can be challenging since ML experts and domain experts could work on other projects and many parts of training and feature engineering is hard to document using traditional tools and processes [5].

An alternative path is some approaches that invite domain experts in a short training loop and consequently can retrain the model continuously so the model can adapt to changes in demands or in the deployment context. Approaches in line with this are, for example, Active Learning (AL), Interactive Machine Learning (IML) and Machine Teaching (MT) [4]. The endpoints in this continuum of human agency are AL and on the other end MT. In AL the ML system remains in control of the learning process and treats the human as an oracle as opposed to MT where a human domain experts have control of the learning process by delimiting the knowledge that they intend to transfer to the ML model.

This work was partially financed by the Knowledge Foundation through the Internet of Things and People research profile.

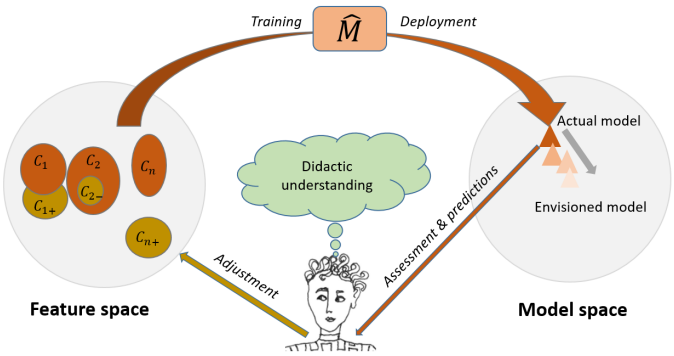

Fig. 1. Overview of machine teaching in this work.

In my work, I focus on MT as an approach and consequently envision an ML system that allows domain experts to control the model building process. A domain expert has, in addition to an ability to label examples, knowledge about the domain as such. This knowledge can, for example, be used to identify unreachable parts of the feature space, missing features that are needed in order to separate classes or identify features not needed for the task at hand. This moves focus in the ML system towards the domain expert's ability to transfer domain knowledge to an ML-model. The purpose of my work is to answer the following question:

- How does an MT approach change the domain expert's role?

An MT system targets a domain by offering a domain-specific language including features to select from, a user interface (UI) and possibilities to interpret predictions made [4]. Initially, a system like this faces a cold start problem a situation where no training data exists. This can then be handled using one or a combination of approaches:

- As a transfer learning situation where a pre-trained model is imported.
- By importing an existing labeled/unlabeled training set from a related domain and label, add, modify or tweak this data.
- By collecting and label new data from the intended deployment context or generate this data.

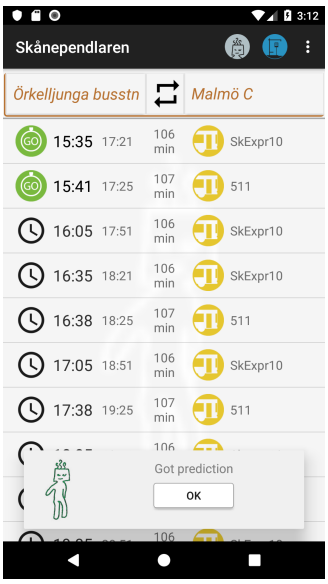

Fig. 2. Prototype where upcoming journeys are predicted using contextual features.

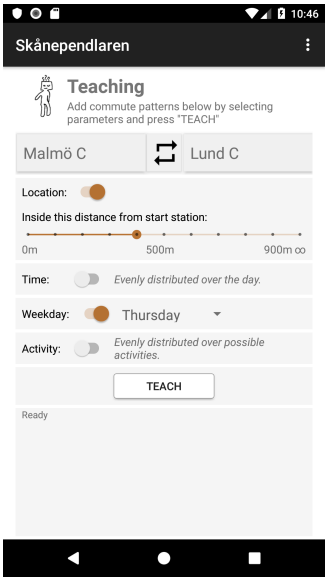

Fig. 3. Teaching interface that shows how labeling a sub-space in the features space is done.

- By labeling sub-spaces in the feature space so examples representing these spaces can be generated and used for training (figure 3).

Teaching in an MT system is as Simard et al. [4] points out an iterative process and if a system is well designed the need for an ML expert can be marginal during teaching and usage. The MT system, as such, has to be updated if it, for example, lacks features available to separate classes or if the chosen ML method(s) cannot learn the intended task.

This approach is outlined in figure 1, inspired by Zhu [6]. The cold start situation is here addressed by allowing a human teacher to select and label sub-spaces in a feature space $C_1, ..., C_n$ and synthetic examples are then generated for this sub-space and used to train a model. That model is then assessed by the human to evaluate if the model's knowledge maps the, by the human teacher envisioned, model knowledge. The teacher moves the actual model in the model space towards the envisioned in an iterative process where labeled sub-spaces are added or removed $(C_{1+}, C_{2-}, ..., C_{n+})$ and the model $\hat{M}$ retrained until it fulfills the teacher's expectations.

II. Case study

My initial case study targeted the domain commuting. As a domain, commuting is well-known whereas commute patterns are individual and the commuters themselves are experts in their own patterns. A commuter can use the prototype to teach journeys from their commute patterns, they do this by labeling sub-spaces in the feature space. By connecting a time-span, day and location to a journey such as commute-to-work C_{ToWork} the system generates labeled examples that represent that sub-space. By using the commuters context the prototype predicts taught upcoming journeys in real-time (figure 2). This MT system was evaluated using qualitative methods in a Participatory Design study that involved eight users over eight weeks.

III. Analysis and result

From our study of MT, we see a focus shift from the labeled examples towards the gap between the domain expert's knowledge and the ML systems knowledge. A domain expert could, given the right tools, bridge this gap by for example include new features such as weather data, calendar bookings or restrict journey predictions at nights. Personalizing the user experience is possible by connecting higher-level concepts with feature sub-spaces using names, so the app can predict "To rugby training" instead of a journey from station A to station B.

IV. Future work

We are interested in further research in pervasive machine teaching settings. Domains like indoor climate and human activity recognition are interesting application areas that build on, as commuting, understandable feature space, and understandable labels. For activity recognition features could be: number of persons, multiple speakers, etc and labels could be: presentation, seminar, etc. This situation can be addressed by building systems where a domain expert bridge the gap in a teaching process by identifying the current status of the system and the real world status in order to make adjustments to the ML-system. This puts demands on the control interface so it can present current status in an interpretable and actionable fashion. The goal is to build a trustworthy ML-systems where the domain experts are in control of adjusting the system so it matches real-world demands. Systems like these can, for example, be implemented to save energy since they can match needs and demands closer.

We are open to cooperation with other researches to target domains in for example the areas of assistive technology, intelligent personal assistants or personal informatics.

References

[1] Yongkai Wu, Lu Zhang, Xintao Wu, and Hanghang Tong. PC-Fairness: A Unified Framework for Measuring Causality-based Fairness. In *Nips*, 2019.

[2] Finale Doshi-Velez and Been Kim. Towards A Rigorous Science of Interpretable Machine Learning. *arXiv preprint arXiv:1702.08608*, 2 2017.

[3] AI HLEG. Ethics Guidelines for Trustworthy AI. Technical report, High-Level Expert Group on Artificial Intelligence, 2019.

[4] Patrice Y. Simard, Saleema Amershi, David M. Chickering, Alicia Edelman Pelton, Soroush Ghorashi, Christopher Meek, Gonzalo Ramos, Jina Suh, Johan Verwey, Mo Wang, and John Wernsing. Machine Teaching: A New Paradigm for Building Machine Learning Systems. Technical report, Microsoft Research, 2017.

[5] D Sculley, Gary Holt, Daniel Golovin, Eugene Davydov, Todd Phillips, Dietmar Ebner, Vinay Chaudhary, and Michael Young. Machine Learning : The High-Interest Credit Card of Technical Debt. *NIPS 2014 Workshop on Software Engineering for Machine Learning (SE4ML)*, pages 1–9, 2014.

[6] Xiaojin Zhu. Machine Teaching: An Inverse Problem to Machine Learning and an Approach Toward Optimal Education. In *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligenc*, pages 4083–4087, 2015.