Personal Context Recognition via Skeptical Learning

Wanyi Zhang
DISI, University of Trento, Italy
wanyi.zhang@unitn.it

Abstract

In personal context recognition many solutions rely on supervised learning that uses sensor data collected from the users' mobile devices. However, the recognition performance is significantly affected by the annotations' quality. The problem lies in the fact that the annotator in such scenarios is usually the user herself which is not an expert and thus provides a significant amount of incorrect labels, while existing solutions can only tolerate a small fraction of mislabels. Our solution is *skeptical learning*, a framework for interactive machine learning where the machine uses all its available knowledge to check the correctness of its own and the user labeling. This allows us to have a uniform confidence measure to be used when a contradiction arises that applies to both the annotator and the machine. The criteria of success is an improvement of the final recognition accuracy with respect to traditional supervised approaches.

1 Introduction

Machine learning is becoming viral and applied in many areas, such as pervasive and ubiquitous computing. Some pervasive computing applications (e.g., decision system and health monitoring system), usually use supervised learning approach to recognize user's context before providing an appropriate level of services. For example, weather prediction systems could learn from the previous weather information, and based on user's present environmental context including location, temperature and humidity, system would return a weather prediction to user. However, the Fact is, the performance of supervised machine learning relies on the quality of labels of the data which they are training with.

In this research we focus on the applications that recognize user's personal context using sensor data collected from user's mobile devices and labels given by user herself. Many solutions of context recognition would utilize supervised learning techniques, such as decision tree, random forest, SVM. Although a perfectly labelled training set is rarely to meet in real-world, most modern classification algorithms can naturally tolerate a small fraction of noise labels. The examples of ensemble methods are bagging and boosting, decision tree, bayesian approaches[Frénay and Verleysen, 2014].

Note that these labels in above research are generally given by domain experts, and mislabeled samples have extremely limited scale. However, in many situations as we are focusing on in this research (e.g., pervasive, ubiquitous, life long), the main source for the annotations is not experts but normal users. The quality of those non-expert annotations is hard to control. According to the research in Social Science, people are unreliable and would give incorrect labels when they are asked to fill questionnaires [West and Sinibaldi, 2013]. Moreover, this is related to user's *response biases* (e.g., memory bias, unwillingness to report) and *cognitive bias* (e.g., careless). This non-expert label noise is a common issue especially in pervasive and ubiquitous computing, and lifelong learning.

Moreover, the growing size of label noise will significantly decrease the prediction performance and increase the number of training samples required for learning and the complexity of the learning model. These label noise issues can not be solved by traditional machine-learning methods.

In order to solve this problem, we propose *Skeptical Learning* (SKEL) as a framework for interactive machine learning. The key idea is that the machine uses all the available knowledge to check the correctness of machine's prediction and the labels provided by the user. The novelty of our work is that we involve the user directly in an interactive process with consistent checks and labeling revisions.

2 Context Recognition

The research in [Giunchiglia *et al.*, 2017b] provides the definition of the real world user's personal Context, aggregating different elements surrounding the user. Context can be formulated as:

$$\text{Cxt } = WA \cup WE \cup WO$$

where the *WA* is the *temporal context*. It is short for "What are you doing?", consisting in the user's activities. The *WE* is the *spacial context*. It is short for "Where are you?", meaning the user's location. While the *WO* is the *social context*. It is short for "Who are you with?", presenting the surrounding users.

In order to recognize user's context, the machine needs to learn not only user's sensor data from smartphone, but also

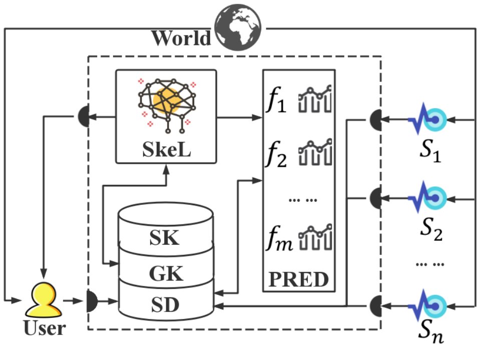

Figure 1: The SKEL Architecture.

the context annotations from user. However, unlike the annotations from domain experts, labels in our research are from common users which are non-experts in general.

We propose SKEL and implement it as the multi-layer architecture depicted in Figure 1. The key intuition is that the human annotator(s) and the machine learning algorithm(s) are considered as *interpretation channels* that provide their own *fallible* perspective on what the case is in the real world. In Figure 1, the *first* layer, inside the dashed box, is the HW/SW *Machine* (e.g., a smartphone with its software), implementing the all SKEL functionalities. The *third* and last layer is the world, which is only indirectly accessible to the Machine. In the *second* layer, Machine and World are connected, on one side, via a set of n sensors $S_1, ..., S_n$ (e.g., in a smart phone or a smart watch as they exist) that produce a set of streams $\{x_t\}_{S_i}$, with x_t collected at time t by the sensor S_i. On the other side, the machine *reference user*, namely the person who is the direct beneficiary of its services, provides a label y_t which is interpreted as the value at time t of a certain property P_j on request (outgoing arrow on the left of the dashed box).

As shown in figure 1, the Machine is composed by three components:

1. A *Predictor* PRED consisting of an *ensemble* of m learning algorithms $f_1, .., f_m$, and each algorithm takes input with an array of data $\mathbf{x}_t$ (the concatenation of all sensor readings at a certain time) and produces a score $f_k(\mathbf{x}_t, y)$ for all possible labels $y \in \mathcal{Y}_j$ of a given property P_j. These scores are then aggregated by PRED into a single label y_t.
2. The main algorithm of SKEL is designed by taking a continuous stream of sensor data stored in SD as input. There are three modalities in the algorithm, namely: *Train* mode performed in usual supervised learning, *Refine* mode that checks the quality of the user answers and challenges these answers under certain conditions, and *Regime* mode that starts being autonomous and only queries the user for particularly ambiguous instances.
3. A *Knowledge component* stores any prior knowledge that the machine has accumulated in time. It has three sub-components: The *Stream Data* (SD) stores the data streams from sensors and the labels from the predictor and the user. The *Ground Knowledge* (GK) contains factual knowledge about the world, and in the actual implementation it is stored as a knowledge graph. GK is the place where the machine accumulates the knowledge learned in time. One example of GK is knowledge about specific locations, for instance, the fact that Prof.Fausto's office is part of the Department building, which in turn is a part of the University premises. The *Schematic Knowledge* (SK) contains general knowledge, for instance, in the form of a hierarchy of concepts stated in a certain language [Giunchiglia *et al.*, 2017a].

The key observation is that the GK and the SK, providing a model-driven view of the world, are unavoidable components whenever there is an interest in making the machine capable of fully understanding the user input, in its intended semantics, in enabling the user in providing semantics to (i.e. in fully understanding) the output of the machine internal data-driven machine learning algorithms.

We validate SKEL on the data collected in one of our experiments with students. Data analysis provides the evidence that users make mistakes. The results indicate that our system achieves 30.7% relative improvement in performance (from f_1=0.26 to f_1=0.34). We also analyze the performance graphs of the different users and identify four patterns as highly common, namely inattentive user, predictable user, reliable user and tricksy user. [Zeni *et al.*, 2019] reports the details of our experiments.

Our research provides a first attempt at integrating the model-driven input from the user with the data-driven input provided by machine learning. The main goal was to exploit this integration in order to minimize the impact of labeling mistakes both on the machine and on the user side. The experimental results are promising and the general idea seems generalizable in many dimensions.

References

[Frénay and Verleysen, 2014] Benoît Frénay and Michel Verleysen. Classification in the presence of label noise: a survey. *IEEE transactions on neural networks and learning systems*, 25(5):845–869, 2014.

[Giunchiglia *et al.*, 2017a] Fausto Giunchiglia, Khuyagbaatar Batsuren, and Gabor Bella. Understanding and exploiting language diversity. In *IJCAI*, pages 4009–4017, 2017.

[Giunchiglia *et al.*, 2017b] Fausto Giunchiglia, Enrico Bignotti, and Mattia Zeni. Personal context modelling and annotation. In *Pervasive Computing and Communications Workshops (PerCom Workshops), 2017 IEEE International Conference on*, pages 117–122. IEEE, 2017.

[West and Sinibaldi, 2013] Brady T West and Jennifer Sinibaldi. The quality of paradata: A literature review. *Improving Surveys with Paradata*, pages 339–359, 2013.

[Zeni *et al.*, 2019] Mattia Zeni, Wanyi Zhang, Enrico Bignotti, Andrea Passerini, and Fausto Giunchiglia. Fixing mislabeling by human annotators leveraging conflict resolution and prior knowledge. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 3(1):32, 2019.