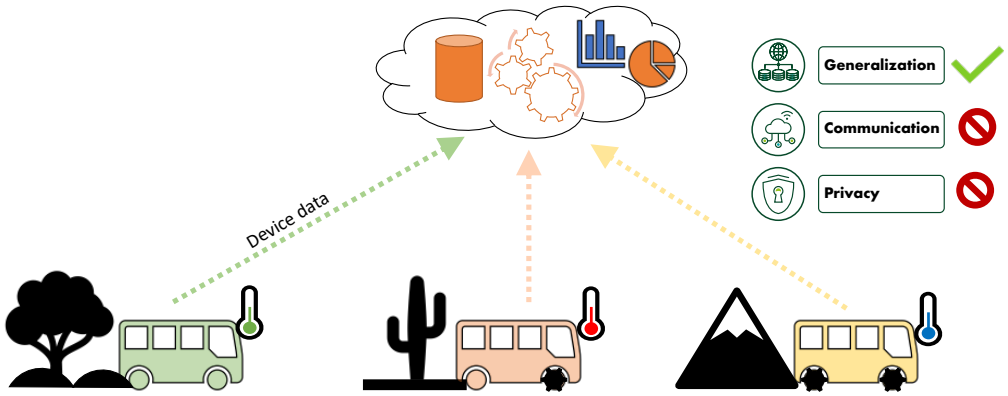
May 2020
Ekkono Solutions AB

Federated Learning

1. Centralized Machine Learning

Machine Learning (ML) for Internet of Things (IoT) has traditionally been done by uploading all data from each connected device to the cloud to train a generic model that can be distributed and applied to all devices. This model answers a specific problem related to the data, for example the battery consumption of the equipment. The advantage of this type of centralized learning is that the model can generalize based on data from a group of devices and thus instantly work with other compatible devices. Centralized learning also entails that data can explain all variations in the devices and their environment. This becomes extremely hard to ensure if there is a large variety of installations or operating environments. Lately, it's has also become apparent that the centralized approach comes with even harder challenges when applied to IoT.



Traditional centralized learning – ML runs in the cloud, gathering info from all connected devices and sending back a model.

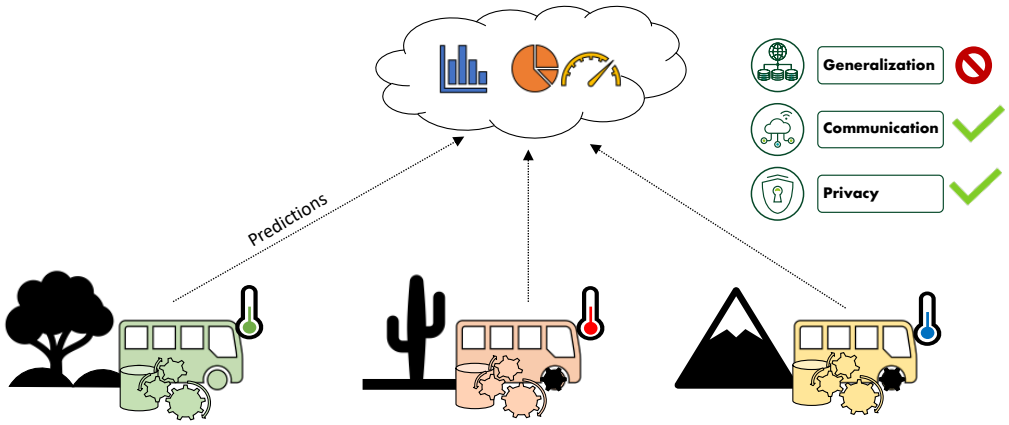
First the bandwidth is often very limited, and the sheer amount of data may exceed what's reasonable to transfer to the cloud e.g., a new ABB electrical substation could generate 5 GB/s of

data. Furthermore, IoT devices are often mobile or distributed in very varying environments making just upholding a stable internet connection a challenge. Finally, latency due to the round trip to the cloud a back is often an issue for application operating in real-time. To summarize the main challenges for traditional centralized learning are:

- **Connectivity** - data must be transmitted over a stable connection
- **Bandwidth** – e.g. a new ABB electrical substation could generate 5 GB/s
- **Latency** - real-time applications, e.g. automation, requires very low latency
- **Privacy** - sensitive operational data must remain on site

2. Decentralized Incremental Learning

Decentralized ML i.e. avoids these problems using Edge Machine Learning*, i.e. ML that runs on site, onboard each connected device. By continuously training the ML model on streaming data the devices learn an individual model for their environment. In practice, this simplifies the task since each model only needs to be able to explain what is normal for itself and not how it varies compared to all other devices. Other benefits of decentralized incremental learning are that the models adapt to changes over time, learning is not constrained by the internet connection and that no confidential information needs to be transferred to the cloud.



Edge/decentralized learning: ML continuously onboard each device at the edge of the network.

* Edge Machine Learning means running machine learning (ML) at the edge of the network – onboard the connected device. Ekkono develops an Edge Machine Learning software. In Ekkono's case, it is possible to do incremental learning at the edge, which means that the ML model continuously gets better but also that it gets personalized as it is fed with sensor data while in production.

Having access to models for each device does not only facilitate more accurate and adaptive models, but also makes it possible to compare the devices through their models. Since the models are created from observed data, they are, per definition, a representation of the same data and, thereby, of the devices themselves. To achieve this, only the models need to be transmitted to the cloud to perform a comparison, which is a fraction in size compared to the sensor data and contains less sensitive information.

The ability to compare and correlate these ML models in the cloud enables new features:

- **Identification of outliers relative to its peers**. These indicate anomalies (e.g. an engine runs hotter than it should for current conditions, or it is consuming more power than should be needed).
- **Analysis of common learnings from the entire fleet**, in combination and relation to individual learnings for a specific asset.
- **Preloading a device with a model** from a similar unit and then fine-tune for the actual device (i.e. transfer learning).

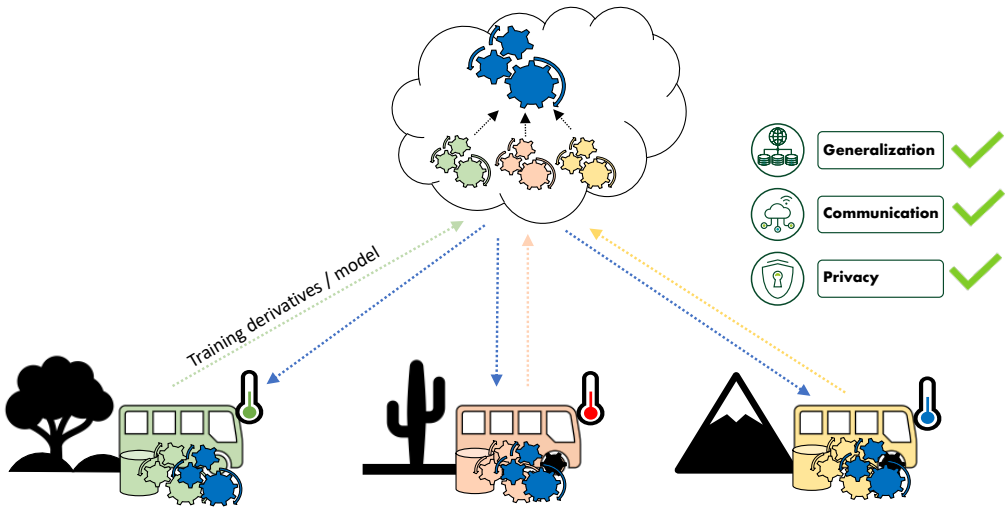
3. Federated Learning

The downside of individualized decentralized learning is that the learning cannot generalize from results derived on other devices as in centralized learning. Combining centralized and individualized decentralized learning is a complex and intricate subject, requiring different solutions for different ML techniques. A new research field, Federated Learning (FL), is addressing this issue. In essence FL is an ML technique to train algorithms across decentralized edge devices while holding data samples locally.

FL is today, arguably the hottest research field within ML with Google as the main player. The main focus of current research has been on in what is termed cross-device FL. In essence this address the problem of training ML models on billions of mobile phones while respecting the privacy of the users. A key aspect is maintaining privacy by only sending fractions of training results, i.e. training derivatives, to the cloud and to never store anything on the device. When collected in the cloud the partial training results from millions of devices can be assembled to a new supermodel that in the next step can be send back to the devices. This is what Google put

all their attention to and what is made available in their open source framework TensorFlow Federated.

Ekkono does instead exploit silo-based FL which is a more versatile technique but much less explored. Silo-based FL deals with the task of training a set a ML models on smaller scale, i.e. 2-10^5 devices, where it's also possible to address specific devices and store information on the device. This is a natural fit for industrial IoT and consequently why Ekkono have put a large research effort to develop its algorithms in this direction.



Federated Learning – learning one from each other while keeping data on the device – creation of Super Models.

When combined with incremental learning, silo-based FL starts with training an individual model on each device. At a given frequency all or a set of models are sent to the cloud and combined into a supermodel (blue cogs in the figure above) while preserving the general learning from each model. Finally, the super model can be sent down to the devices to replace or be merged with the local model and then adapted to the current environments by again applying incremental learning. Silo-based and cross device FL both benefits from the advantages of applying ML at the edge and in the cloud. Furthermore, silo-based approach adds important functionality that cross device FL cannot provide:

- **Model inspection** – evaluation of device behavior through its model
- **Model comparison** – comparing models in the cloud to find outliers, super models

- **Robust learning** - learning can continue even if connection to the cloud is lost
- **Tailored initialization** – new devices can start with a model from a similar device, instead of a general super model

4. Use Cases

There are three main application areas for FL:

Smartphones – As mentioned above cellphones is use case that normally come first to mind when speaking of FL. There is million or even billions of devices, there data is very sensitive, they vary in both hardware and the data they produce and due to battery constraints, they are only available for FL while charging. A real-world application in this area is googles next word prediction in Gmail. Naturally, the data you type in your emails is very sensitive but by processing on the phone and only sending an encrypted partial training derivative, you may still learn among devices while preserving privacy.

Orgnizations – Some organizations, e.g. hospitals and airports, may have need of sharing data but are restricted due to regulation. Here, it is rarley a problem of bandwith, power or the number of devices, but privacy and hetergoenicty of the systems.

Internet of Things – Billions of devices are being connected and naturally it is not feasible to send all the data they generated to the cloud due both bandwidth and privacy constraints. In addition, typical IoT devices have very limited hardware which require even more efficient solutions. Nonetheless, applications within industrial IoT is especially promising since they typically concern a smaller set of devices, i.e. < 10^5, often have a steady power supply and may exploit persistent models. Most applications hence typically fit within silo-based FL. Typical use cases relates to predictive maintenance and

5. Challenges

There are of course challenges of applying federate learning and they vary from application to application. The main technical challenges are:

- **Communication** – Even if FL drastically reduce the amount of data that needs to be sent to the cloud communication is still needed to transmit the training result for each round. Hence, a lot of research is focusing on reducing the number of update rounds and the update message itself, i.e. the training derivatives.
- **System heterogeneity** – A FL systems is almost by definition heterogenic in that the device may vary in storage, computational and communication capabilities. Hence a FL training scheme must be dynamic or conform to the lowest denominator of the devices.
- **Statistical heterogeneity** – Due to varying operation, environments or configuration devices in a FL system frequently generate data that is statistically different from each other, i.e. in a non I.I.D. manner. Since ML typically rely on the I.I.D. assumption special techniques must be developed to handle the statistical heterogeneity if present.
- **Privacy** – FL is designed to preserve privacy but care still need to be given to ensure that sensitive information is not reviled for specific users or devices. Typically, it is deviating units that are most likely to be compromised since their usage pattern stand out and may influence the model I a unique way.

6. Conclusions

There are of course challenges of applying FL and they vary from application to application. The main technical challenges are:

- **Communication** – Even if FL drastically reduce the amount of data that needs to be sent to the cloud communication is still needed to transmit the training result for each round. Hence, a lot of research is focusing on reducing the number of update rounds and the update message itself, i.e. the training derivatives.
- **System heterogeneity** – A FL systems is almost by definition heterogenic in that the device may vary in storage, computational and communication capabilities. Hence a FL training scheme must be dynamic or conform to the lowest denominator of the devices.
- **Statistical heterogeneity** – Due to varying operation, environments or configuration devices in a FL system frequently generate data that is statistically different from each

other, i.e. in a non I.I.D. manner. Since ML typically rely on the I.I.D. assumption special techniques must be developed to handle the statistical heterogeneity if present.

- **Privacy** – FL is designed to preserve privacy but care still need to be given to ensure that sensitive information is not reviled for specific users or devices. Typically, it is deviating units that are most likely to be compromised since their usage pattern stand out and may influence the model I a unique way.

FL techniques should be computationally cheap, communication-efficient, and tolerant to dropped devices – all without overly compromising accuracy or privacy. Naturally this is not easy to accomplish but FL is designed with these challenges in mind and luckily, all challenges does rarely apply to a specific application.

***

Ekkono #openfika is a short open, online fika[†] session, hosted by Ekkono, on hot, contemporary and relevant topics, where a 15 minutes presentation is followed by discussion and Q&A. Keep an eye on www.ekkono.ai and LinkedIn for the next #openfika session.

Ekkono Solutions AB is a software company that develops Edge Machine Learning. Our product is the result of seven years of research at the University of Borås, Sweden, and assists product OEMs in different industries to rapidly develop smart features onboard their products, using machine learning to make them self-learning and predictive. For more information, visit www.ekkono.ai.

[†] fika (wikipedia.org); Swedes have fika (pronounced [ˈ fiːka]), meaning "coffee break". The tradition has spread throughout Swedish businesses around the world. Fika is a social institution in Sweden and a common practice at workplaces in Sweden. Fika may also function partially as an informal meeting between co-workers and management people, and it may even be considered impolite not to join in.