Federated Learning: Collaboration with no compromise

IFoA Federated Learning Working Party
Malgorzata Śmietanka, Dylan Liew, Claudio Giancaterino

10 October 2021

Background

1. The need and problem of data

- data comprises distributed and isolated data sets;
- analytics requires models to be trained across these independent data sets;
- data sovereignty/privacy legislation is making collecting, sharing and analysing data increasingly difficult.

2. Privacy of data

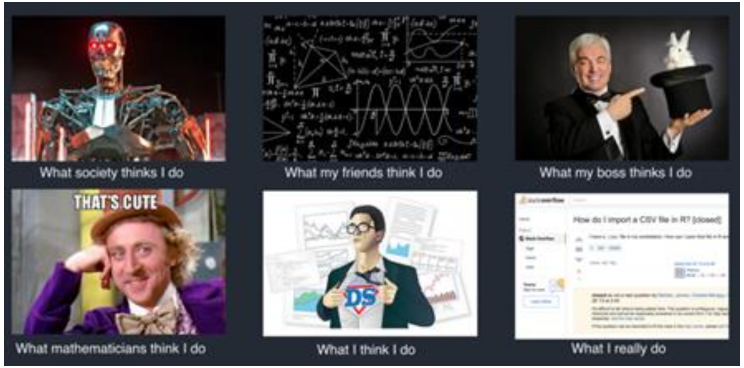
Privacy of data poses many challenges:

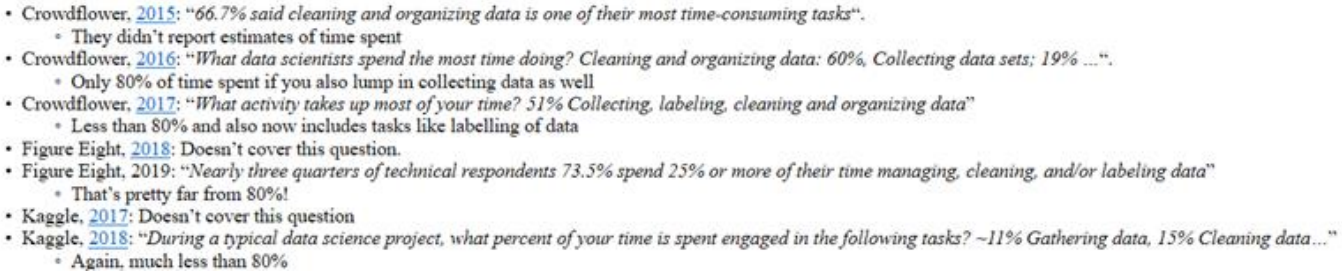
- **Internal data management-** need to control data-access and be able to store data in a way that will enable to do analytics.
- **Collaboration with third parties-** companies recognizing value of collaboration need a way to do that without compromising sensitive data access.
- **Monetising the data**

Motivation

What I think I do

Or?


THAT'S CUTE
DS
How do I import a CSV file in R? [closed]
What society thinks I do
What my friends think I do
What my boss thinks I do
What mathematicians think I do
What I think I do
What I really do

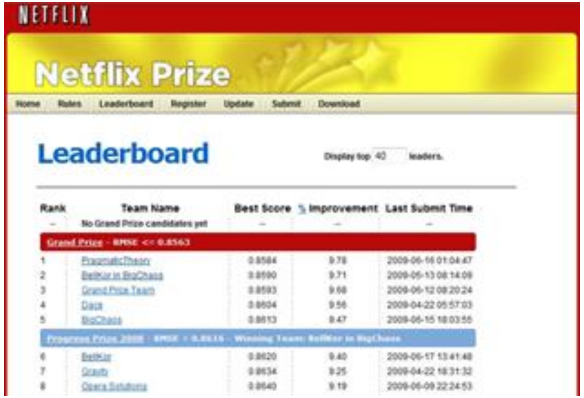


- Crowdflower, 2015: "*66.7% said cleaning and organizing data is one of their most time-consuming tasks*".
  - They didn't report estimates of time spent
- Crowdflower, 2016: "*What data scientists spend the most time doing? Cleaning and organizing data: 60%, Collecting data sets; 19% …*".
  - Only 80% of time spent if you also lump in collecting data as well
- Crowdflower, 2017: "*What activity takes up most of your time? 51% Collecting, labeling, cleaning and organizing data*"
  - Less than 80% and also now includes tasks like labelling of data
- Figure Eight, 2018: Doesn't cover this question.
- Figure Eight, 2019: "*Nearly three quarters of technical respondents 73.5% spend 25% or more of their time managing, cleaning, and/or labeling data*"
  - That's pretty far from 80%!
- Kaggle, 2017: Doesn't cover this question
- Kaggle, 2018: "*During a typical data science project, what percent of your time is spent engaged in the following tasks? ~11% Gathering data, 15% Cleaning data…*"
  - Again, much less than 80%

GDPR

https://blog.ldodds.com/2020/01/31/do-data-scientists-spend-80-of-their-time-cleaning-data-turns-out-no/

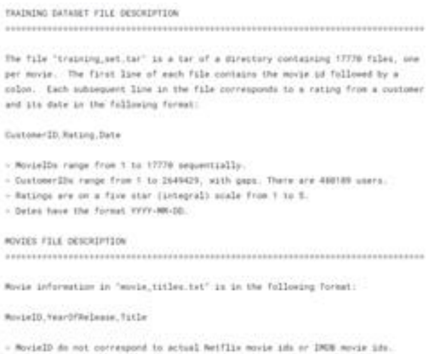
Tiger King



Leaderboard

Display top 40 leaders.

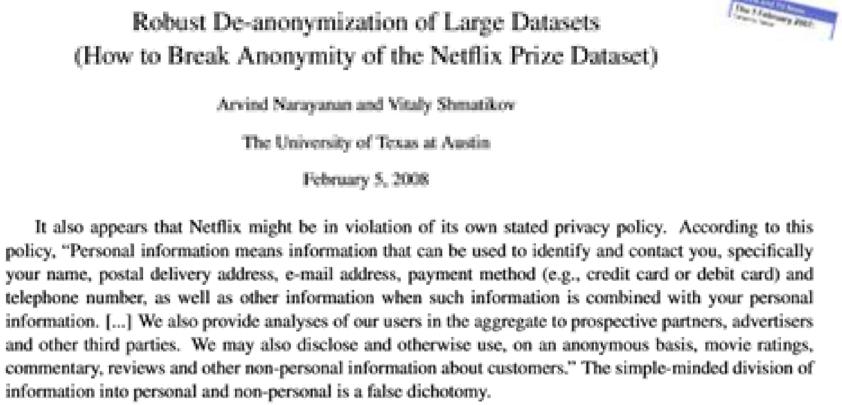
Rank	Team Name	Best Score	% Improvement	Last Submit Time
-	No Grand Prize candidates yet	-	-	-
Grand Prize - RMSE <= 0.8563				
1	PragmaticTheory	0.8584	9.78	2009-06-16 01:04:47
2	BellKor in BigChaos	0.8590	9.71	2009-05-13 08:14:09
3	Grand Prize Team	0.8593	9.68	2009-06-12 08:20:24
4	Dace	0.8604	9.56	2009-04-22 05:57:03
5	BigChaos	0.8613	9.47	2009-06-15 18:03:55
Progress Prize 2008 - RMSE = 0.8616 - Winning Team: BellKor in BigChaos				
6	BellKor	0.8620	9.40	2009-06-17 13:41:48
7	Gravity	0.8634	9.25	2009-04-22 18:31:32
8	Opera Solutions	0.8640	9.19	2009-06-09 22:24:53



```
TRAINING DATASET FILE DESCRIPTION
================================================================================

The file "training_set.tar" is a tar of a directory containing 17770 files, one
per movie.  The first line of each file contains the movie id followed by a
colon.  Each subsequent line in the file corresponds to a rating from a customer
and its date in the following format:

CustomerID,Rating,Date

- MovieIDs range from 1 to 17770 sequentially.
- CustomerIDs range from 1 to 2649429, with gaps. There are 480189 users.
- Ratings are on a five star (integral) scale from 1 to 5.
- Dates have the format YYYY-MM-DD.

MOVIES FILE DESCRIPTION
================================================================================

Movie information in "movie_titles.txt" is in the following format:

MovieID,YearOfRelease,Title

- MovieID do not correspond to actual Netflix movie ids or IMDB movie ids.
```

```
1:
1488844,3,2005-09-06
822109,5,2005-05-13
885013,4,2005-10-19
30878,4,2005-12-26
823519,3,2004-05-03
893988,3,2005-11-17
124105,4,2004-08-05
1248029,3,2004-04-22
1842128,4,2004-05-09
2238063,3,2005-05-11
1503895,4,2005-05-19
2207774,5,2005-06-06
2590061,3,2004-08-12
2442,3,2004-04-14
543865,4,2004-05-28
1209119,4,2004-03-23
804919,4,2004-06-10
1086807,3,2004-12-28
1711859,4,2005-05-08
372233,5,2005-11-23
1080361,3,2005-03-28
1245640,3,2005-12-19
558634,4,2004-12-14
```

IMDb
The Internet Movie Database
Resume Services Presents Hale Appleman



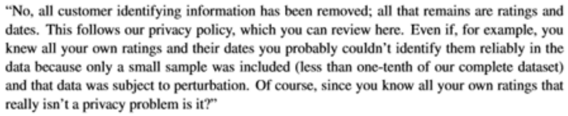
Robust De-anonymization of Large Datasets
(How to Break Anonymity of the Netflix Prize Dataset)

Arvind Narayanan and Vitaly Shmatikov

The University of Texas at Austin

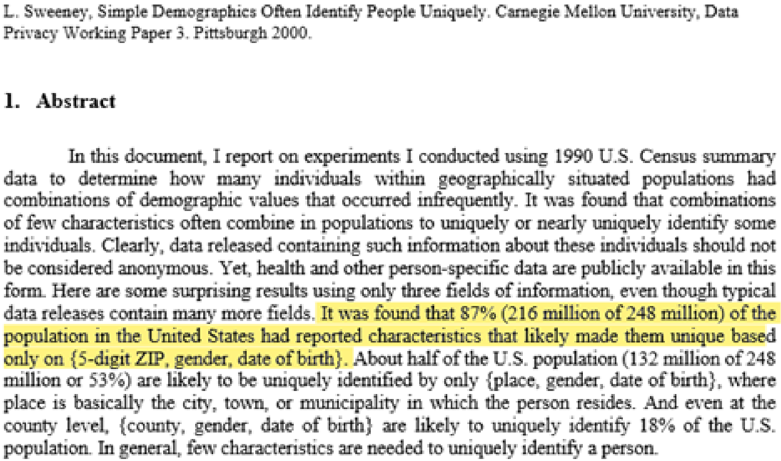
February 5, 2008

It also appears that Netflix might be in violation of its own stated privacy policy. According to this policy, "Personal information means information that can be used to identify and contact you, specifically your name, postal delivery address, e-mail address, payment method (e.g., credit card or debit card) and telephone number, as well as other information when such information is combined with your personal information. [...] We also provide analyses of our users in the aggregate to prospective partners, advertisers and other third parties. We may also disclose and otherwise use, on an anonymous basis, movie ratings, commentary, reviews and other non-personal information about customers." The simple-minded division of information into personal and non-personal is a false dichotomy.



"No, all customer identifying information has been removed; all that remains are ratings and dates. This follows our privacy policy, which you can review here. Even if, for example, you knew all your own ratings and their dates you probably couldn't identify them reliably in the data because only a small sample was included (less than one-tenth of our complete dataset) and that data was subject to perturbation. Of course, since you know all your own ratings that really isn't a privacy problem is it?"

Even more sensitive



L. Sweeney, Simple Demographics Often Identify People Uniquely. Carnegie Mellon University, Data Privacy Working Paper 3. Pittsburgh 2000.

1. Abstract

In this document, I report on experiments I conducted using 1990 U.S. Census summary data to determine how many individuals within geographically situated populations had combinations of demographic values that occurred infrequently. It was found that combinations of few characteristics often combine in populations to uniquely or nearly uniquely identify some individuals. Clearly, data released containing such information about these individuals should not be considered anonymous. Yet, health and other person-specific data are publicly available in this form. Here are some surprising results using only three fields of information, even though typical data releases contain many more fields. It was found that 87% (216 million of 248 million) of the population in the United States had reported characteristics that likely made them unique based only on {5-digit ZIP, gender, date of birth}. About half of the U.S. population (132 million of 248 million or 53%) are likely to be uniquely identified by only {place, gender, date of birth}, where place is basically the city, town, or municipality in which the person resides. And even at the county level, {county, gender, date of birth} are likely to uniquely identify 18% of the U.S. population. In general, few characteristics are needed to uniquely identify a person.

L. Sweeney, Simple Demographics Often Identify People Uniquely. Carnegie Mellon University, Data Privacy Working Paper 3. Pittsburgh 2000.

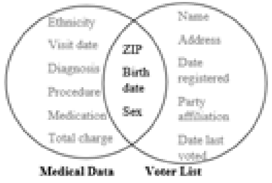
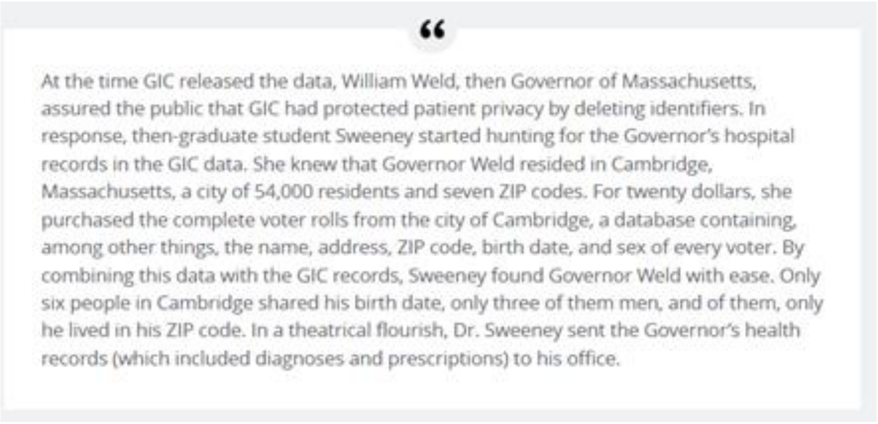
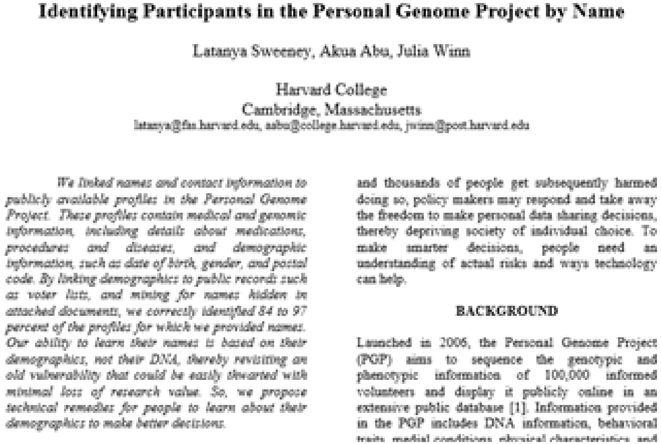


Figure 1 Linking to re-identify data



> At the time GIC released the data, William Weld, then Governor of Massachusetts, assured the public that GIC had protected patient privacy by deleting identifiers. In response, then-graduate student Sweeney started hunting for the Governor's hospital records in the GIC data. She knew that Governor Weld resided in Cambridge, Massachusetts, a city of 54,000 residents and seven ZIP codes. For twenty dollars, she purchased the complete voter rolls from the city of Cambridge, a database containing, among other things, the name, address, ZIP code, birth date, and sex of every voter. By combining this data with the GIC records, Sweeney found Governor Weld with ease. Only six people in Cambridge shared his birth date, only three of them men, and of them, only he lived in his ZIP code. In a theatrical flourish, Dr. Sweeney sent the Governor's health records (which included diagnoses and prescriptions) to his office.



Identifying Participants in the Personal Genome Project by Name

Latanya Sweeney, Akua Abu, Julia Winn

Harvard College
Cambridge, Massachusetts
latanya@fas.harvard.edu, aabu@college.harvard.edu, jwinn@post.harvard.edu

We linked names and contact information to publicly available profiles in the Personal Genome Project. These profiles contain medical and genomic information, including details about medications, procedures and diseases, and demographic information, such as date of birth, gender, and postal code. By linking demographics to public records such as voter lists, and mining for names hidden in attached documents, we correctly identified 84 to 97 percent of the profiles for which we provided names. Our ability to learn their names is based on their demographics, not their DNA, thereby revisiting an old vulnerability that could be easily thwarted with minimal loss of research value. So, we propose technical remedies for people to learn about their demographics to make better decisions.

and thousands of people get subsequently harmed doing so, policy makers may respond and take away the freedom to make personal data sharing decisions, thereby depriving society of individual choice. To make smarter decisions, people need an understanding of actual risks and ways technology can help.

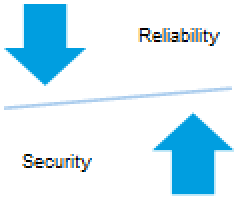
BACKGROUND

Launched in 2006, the Personal Genome Project (PGP) aims to sequence the genotypic and phenotypic information of 100,000 informed volunteers and display it publicly online in an extensive public database [1]. Information provided in the PGP includes DNA information, behavioral traits, medical conditions, physical characteristics, and

Secure Multiparty Computation ("SMPC")

- Alice, Bob, and Carol would like to know their average salary, say:

Name	Salary
Alice	100
Bob	200
Carol	300
Average	200



- Useful but personal information
- They don't want to tell each other
- Could give to a trusted 3rd party, Dylan to compute?
- But can they all trust him? Would he collude?
- Consider randomly adding numbers in and splitting

Name	Share 1	Share 2	Share 3	Total
Alice	50	30	20	100
Bob	-80	100	180	200
Carol	0	350	-50	300

- What if Alice randomly gives Carol her 2nd secret share, and Bob her 3rd etc. and we jumble the shares?

Name	Share 1	Share 2	Share 3
Alice	50	180	350
Bob	-80	-50	20
Carol	0	30	100

- Carol got a figure of 30 from Alice, Alice got a figure of 350 from Carol, can they infer anything from about their salaries?
- Everyone can sum their "secret shares" and glean nothing, but observe:

Name	Share 1	Share 2	Share 3	Total
Alice	50	180	350	580
Bob	-80	-50	20	-110
Carol	0	30	100	130
Average				200

Secure Multiparty Computation (“SMPC”)

Insurer 3's secret	25
Insurer 1 receives	15
Insurer 2 receives	7
Insurer 3 keeps	3

1. Insurers 1,2, and 3 all decide to build their claims model on their own data, and calculate their gradients (=> parameters)

Insurer	Gradient	Out-of-sample error
1	11	1.3
2	15	2.7
3	-2	10.3
Sum	24	17.2
Average	8	5.7

2. Some prime number **p**, say 27 is picked, and all gradients are taken to **mod p**

Insurer	Gradient	Mod27
1	11	11
2	15	15
3	-2	25

3. Scrambling and sharing occurs as before

Insurer	Mod27	Share 1	Share 2	Share 3	Total
1	11	5	4	2	11
2	15	3	8	4	15
3	25	15	7	3	25

Insurer	Share 1	Share 2	Share 3	Total
1	5	3	15	23
2	4	8	7	19
3	4	2	3	9
Total				51

X (Possible secrets)	x mod27 (You receive)
15	15
-12	15
42	15
690	15
…	…

4. Total is taken to mod p and then divided by the number of insurers:

51 mod 27=24

24/3=8

5. Everyone updates their gradients (=>parameters) to get (hopefully) improved mode

Insurer	Gradient	Out-of-sample error
1	8	0.9
2	8	4.2
3	8	5.6
Sum	24	10.7
Average	8	3.6

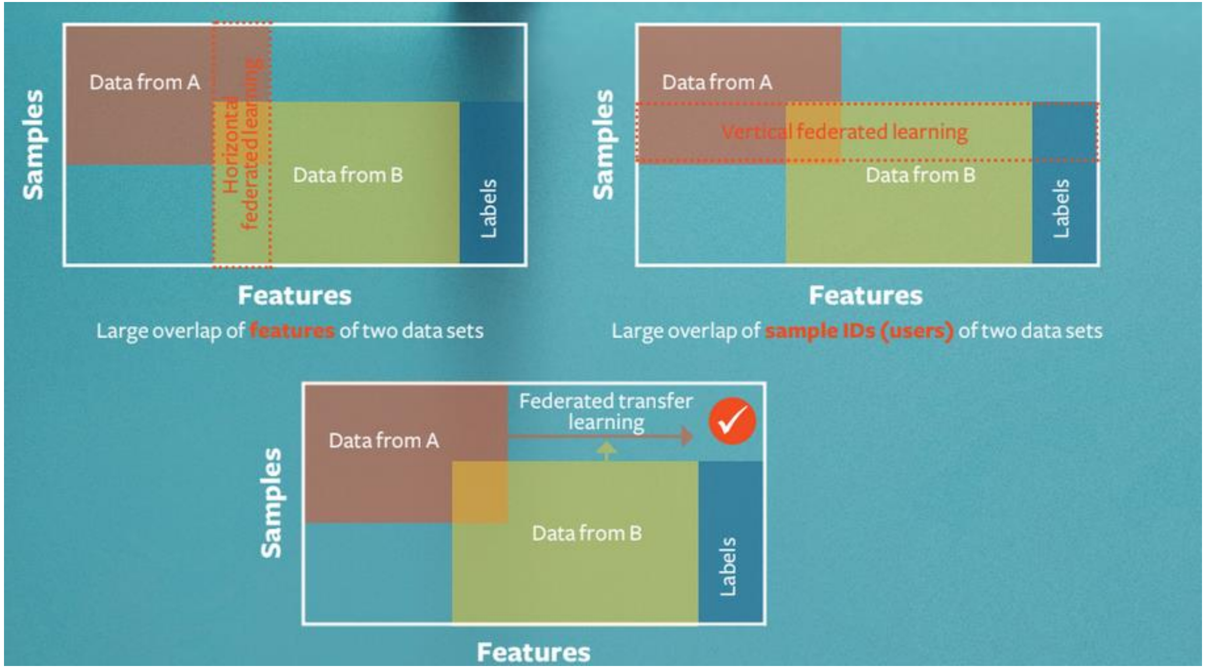
Federated Learning Ecosystem

- **Federated data infrastructure** – privacy-preserving data infrastructure; a framework for collaboration, allowing secure communication with collaborating parties, such that ‘raw’ data does not leave the owner.

- **Federated machine learning** – decentralized training of a machine learning model which enables collaborative learning while keeping data sources in their original location. For example, Google’s mobile phone users benefit from obtaining a well-trained model without sending their personal data to the Cloud.

Federated Learning

… by data partition


Samples
Data from A
Horizontal federated learning
Data from B
Labels
Features
Large overlap of features of two data sets
Samples
Data from A
Vertical federated learning
Data from B
Labels
Features
Large overlap of sample IDs (users) of two data sets
Samples
Data from A
Federated transfer learning
Data from B
Labels
Features

Federated Learning Pipeline

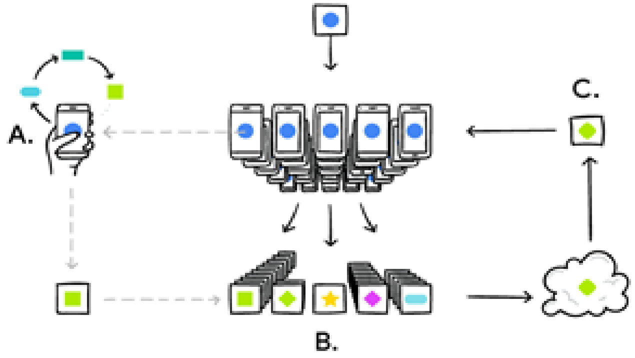
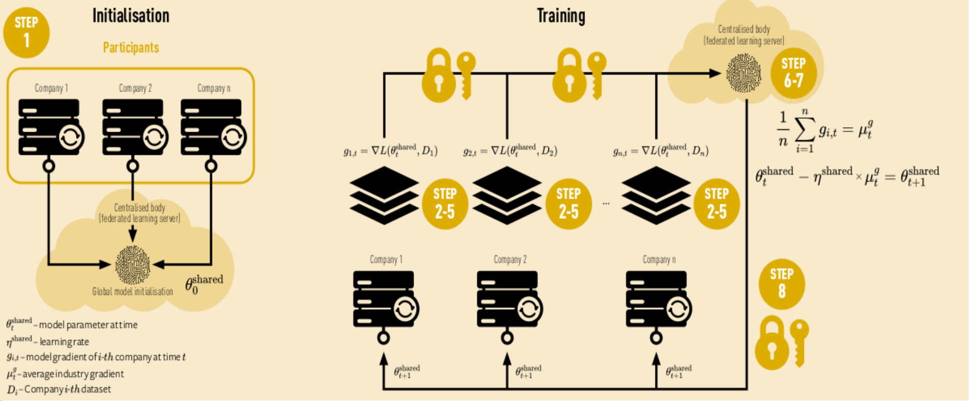


Figure 2: Google Federated Learning

FIGURE 3: Federated learning pipeline.


STEP 1
Initialisation
Participants
Company 1
Company 2
Company n
Centralised body (federated learning server)
Global model initialisation
θ_0^{shared}
Training
$g_{1,t} = \nabla L(\theta_t^{\text{shared}}, D_1)$
$g_{2,t} = \nabla L(\theta_t^{\text{shared}}, D_2)$
$g_{n,t} = \nabla L(\theta_t^{\text{shared}}, D_n)$
STEP 2-5
STEP 2-5
...
STEP 2-5
Company 1
Company 2
Company n
$\theta_{t+1}^{\text{shared}}$
$\theta_{t+1}^{\text{shared}}$
$\theta_{t+1}^{\text{shared}}$
Centralised body (federated learning server)
STEP 6-7
$\frac{1}{n}\sum_{i=1}^{n} g_{i,t} = \mu_t^g$
$\theta_t^{\text{shared}} - \eta^{\text{shared}} \times \mu_t^g = \theta_{t+1}^{\text{shared}}$
STEP 8

θ_t^{shared} – model parameter at time
η^{shared} – learning rate
$g_{i,t}$ – model gradient of i-th company at time t
μ_t^g – average industry gradient
D_i – Company i-th dataset

A) your phone personalises the model locally depending on your usage;
B) many users' updates are aggregated;
C) the aggregated updates form a consensus change to the shared model; and
D) the shared models are updated.

Connecting to Deep Learning

- Gradient Descent is an optimization algorithm used in the back-propagation step of Neural Networks with the goal to minimize the loss function updating parameters by a loop process

$$\theta_t - \eta * g_{i,t} = \theta_{t+1}$$

- Averaging the gradient in the Federated Learning process

$$\theta_t^{Shared} - \eta^{Shared} * \mu_t^g = \theta_{t+1}^{Shared}$$

Where:

μ_t^g represents the average of local gradients -> $\frac{1}{n}\sum_{i=1}^{n} g_{i,t} = \mu_t^g$

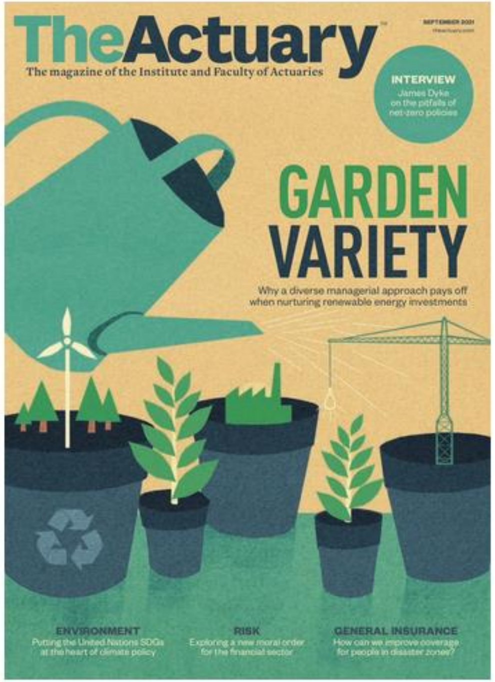
Open-Source Tools

- **Google TensorFlow Federated (TFF) -** Google TFF focuses on horizontal federated learning with a large population of client devices with heterogeneous computing capabilities.

- **OpenMined PySyft** – PySyft uses the PyTorch machine learning platform to implement a federated learning model. PySyft is a Python library for secure and private deep learning. PySyft supports PyTorch, Tensorflow, and Keras with varying capabilities for remote execution, federated learning, differential privacy, homomorphic encryption, and secure multi-party computation.

- **WeBank FATE** – WeBank's Federated AI Technology Enabler (FATE) supports horizontal FL, vertical FL and federated transfer learning with a focus on secure protocols based on homomorphic encryption and multi-party computation (MPC).

Use Case



TheActuary

The magazine of the Institute and Faculty of Actuaries

SEPTEMBER 2021

theactuary.com

INTERVIEW
James Dyke on the pitfalls of net-zero policies

GARDEN VARIETY

Why a diverse managerial approach pays off when nurturing renewable energy investments

ENVIRONMENT
Putting the United Nations SDGs at the heart of climate policy

RISK
Exploring a new moral order for the financial sector

GENERAL INSURANCE
How can we improve coverage for people in disaster zones?

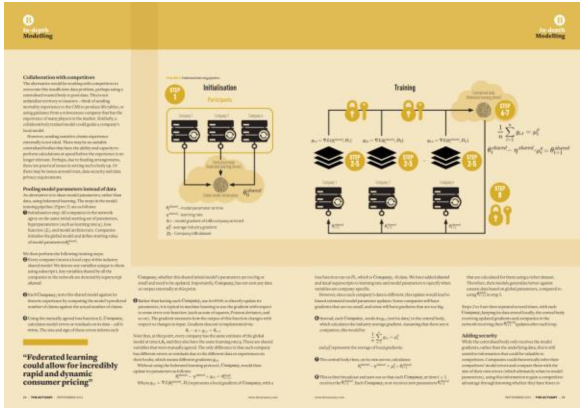
Issue date: September 2021

In-depth Modelling

ALL TOGETHER NOW

Małgorzata Śmietanka, Dylan Liew and Claudio Giorgio Giancaterino demonstrate how to model claims anonymously using federated learning

"Once the model is initialised and its architecture defined, participating companies train the global model locally and send back the gradients"



In-depth Modelling

Initialisation

Training

"Federated learning could allow for incredibly rapid and dynamic consumer pricing"

Use Case: Data

Data frame has 678.013 individual car insurance policies and for each policy there are 12 variables

N	Name	Description
1	IDPol	Policy number
2	ClaimNb	Number of claims on the given policy
3	Exposure	Total exposure in yearly units
4	Area	Area code
5	VehPower	Power of the car
6	VehAge	Age of the car in years
7	DrivAge	Age of the driver in years
8	BonusMalus	Bonus-malus level between 50 and 230 (reference level 100)
9	VehBrand	Car brand
10	VehGas	Diesel or regular fuel car
11	Density	Density of inhabitants per km-square in the city of the living place of the driver
12	Region	Regions in France (prior to 2016)

Use Case: Data

- Data transformation
  - ➢ Data cleaning: some features have been corrected for anomaly values, following Mario V. Wutrich approach in "Case Study: French Motor Third-Party Liability Claims"
  - ➢ Data pre-processing: encoding categorical features and scaling all features

- Constraints about data input for federated learning
  - ➢ Same data transformation and same features are required for all participants involved in the Federated Learning training

Use Case: Results

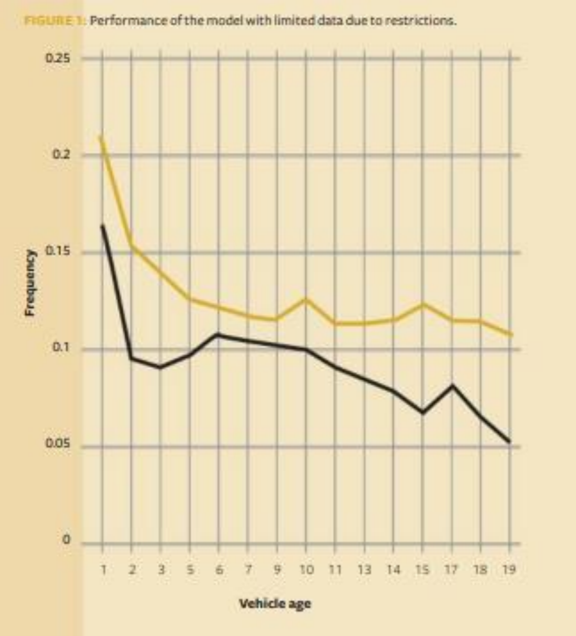


FIGURE 1: Performance of the model with limited data due to restrictions.

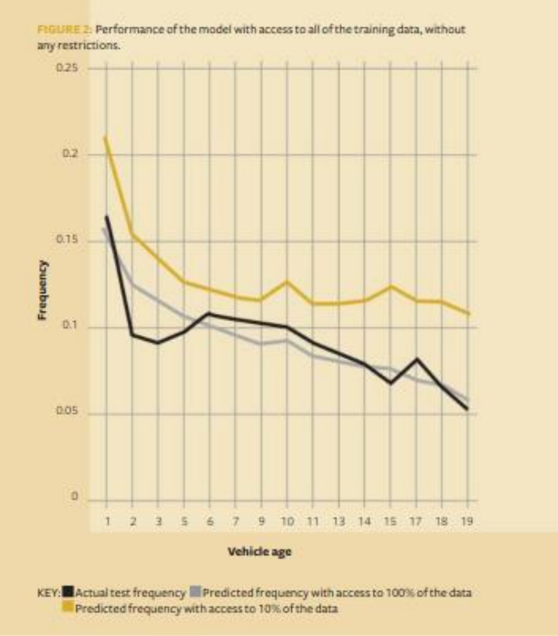


FIGURE 2: Performance of the model with access to all of the training data, without any restrictions.

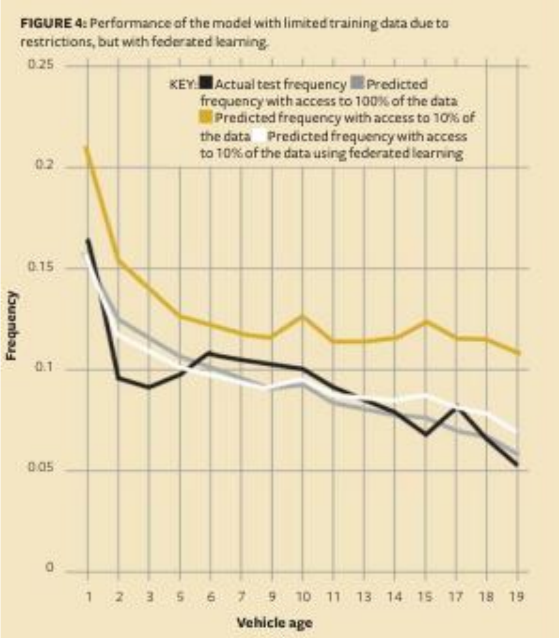


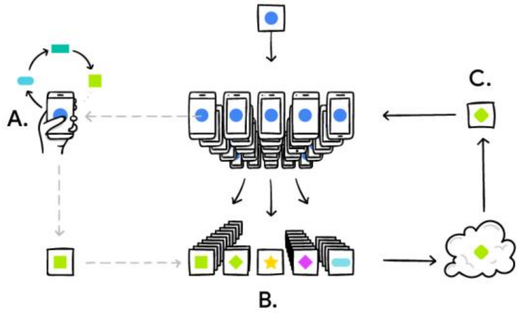
FIGURE 4: Performance of the model with limited training data due to restrictions, but with federated learning.

Use Case Challenges: Federated Learning Engineering Issues

- **Data characteristics**
- **Model characteristics** - choosing the *hyperparameters* (values used to control the learning process) and optimisers
- **Performance efficiency** - performance is impacted firstly by node communications requirements and secondly by privacy-preserving cryptographic techniques.
- **Availability of nodes** - distributed and collaborative training is impacted by availability of communications and nodes. Systems need to be reliable or tolerant to failure, otherwise it may interrupt the entire training process that may render all the work done by the other nodes as void.

Applications of Federated Learning

- Federated Learning has been introduced by Google in 2017 for the mobile keyboard prediction



- Federated Learning applications in healthcare are used to improve real-world AI models for COVID-19 diagnosis on chest X-rays, pancreas segmentation AI model, prostate cancer AI model...

- Federated Learning applications in finance are used to build a Fraud Detection System to minimize loss for banks and cardholders, for credit scoring...

- Federated Learning applications are used in smart cities to estimate air quality, to detect abandoned and suspicious objects, to reduce the traffic congestion...

- Federated Learning applications are used in insurance for claims fraud detection, for risk modelling and pricing...

Questions

Comments