

# Distribuirani aktorski sistem za predikciju potrošnje električne energije u domaćinstvima koristeći federativno ulenje, CRDT i klasterovanje

Veljko Vulin RA-69/2021

## Tehnologije:

Python, Docker

Biblioteke:

- Numpy
- Pandas
- Scikit-learn
- Crdt
- Matplotlib
- Thespian

## Aktori:

### HouseholdActor

- Aktor kuća koji uči lokalno na svojim podacima, šalje parametre modela i trenutno potrošnju CentralActor-u i prima poruke o smanjenju potrošnje po sobama.

Prima:

Reduce\_Consumption(target\_reduction, reason)

Shutdown\_Room (room\_name, reason)

Update\_Model(weights)

Šalje:

Model\_Update (household\_id, weights, evaluation\_metrics)

Consumption\_Data (household\_id, consumption, room\_details, crdt\_state)

### CentralActor

- Prima parametre modela i trenutnu potrošnju od kuća, kombinuje ih u globalni model i prati ukupnu potrošnju pomoću CRDT-a, reaktivno šalje poruke kućama (za smanjenje potrošnje). Više CentralActor'a komuniciraju peer-to-peer pomoću TCP soketa.

Prima:

Model\_Update (household\_id, weights, evaluation\_metrics)

Consumption\_Data (household\_id, consumption, room\_details, crdt\_state)

Šalje:

Reduce\_Consumption(target\_reduction, reason)

Shutdown\_Room (room\_name, reason)

Update\_Model(weights)

Menjanje ponašanja:

HouseholdActor i CentralActor menjaju ponašanje koristeći state machine.

### **Federativno učenje:**

Specifikacija algoritma:

Svaki HouseholdActor trenira lokalni model predikcije potrošnje. Nakon lokalnog treninga, HouseholdActor šalje parametre modela CentralActor-u. CentralActor kombinuje parametre svih kuća koristeći federated averaging i dobija globalni model. CentralActor zatim šalje globalni model nazad ka HouseholdActor-ima koji ažuriraju svoje lokalne modele.

Skup podataka:

Podaci se generišu u runtime-u pomoću ConsumptionSimulator-a koji simulira realističnu potrošnju električne energije po sobama (living room, bedroom, bathroom, kitchen) sa varijacijama između kuća.

Način distribucije treniranja:

Svaka kuća trenira lokalni model nezavisno, težine modela se šalju CentralActor-u bez deljenja sirovih podataka.

Metod evaluacije:

Predikcija potrošnje električne energije iz lokalnih i globalnih modela će se evaluirati korišćenjem Mean Squared Error (MSE), Mean Absolute Error (MAE) i Root Mean Squared Error (RMSE).

Klasterovanje:

Sistem koristi Docker Compose sa 6 kontejnera:

2 CentralActor kontejnera (central\_1, central\_2) sa P2P komunikacijom

4 HouseholdActor kontejnera (household\_1-4), distribuirana 2 po centrali

P2P Komunikacija:

CentralActor-i razmenjuju podatke preko raw TCP soketa:

Razmena consumption podataka,

Sinhronizacija globalnih modela

Supervizija

CentralActor supervizira HouseholdActor-e

Prati child actore,

detektuje njihov pad,

graceful shutdown svih children pri exit-u

Vizuelizacija:

Sistem generiše matplotlib grafike: