

Specifikacija projekta – Softverski agenti

Projekat će biti rađen samostalno.

Ime, prezime i broj indeksa člana tima: Davor Homa RA 47/2021

1. Zadatak

Medicinske ustanove poseduju podatke o pacijentima, ali ti podaci su osetljivi i ne mogu se deliti direktno zbog pravnih i etičkih razloga.

Cilj projekta je da se omogući federativno učenje prediktivnog modela za medicinsku dijagnozu, gde svaka bolnica trenira lokalni model na sopstvenim podacima i deli samo parametre modela.

Za sinhronizaciju koristi se akterski sistem, koji se sastoji od tri vrste aktera: trening, agregatora i aktera za perzistenciju.

2. Federativno učenje

Algoritam

Koristiće se:

- Algoritam logističke regresije za treniranje lokalnih modela,
- *FedAvg* algoritam za potrebe federativnog učenja.

Skup podataka

[Logistic regression To predict heart disease](#)

Način distribucije treniranja algoritma

Treniranje modela logističke regresije će se vršiti na više klijenata koji će imati približno jednaku količinu podataka za treniranje modela i evaluaciju rezultata.

Metod evaluacije rezultata

Za evaluaciju rezultata će se koristiti *Accuracy* metrika:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

3. Implementacija akterskog radnog okruženja

Elementi koji će biti implementirani:

- Obavezni elementi: Aktori, Asinhrono slanje i primanje poruka, Sanduče, Menjanje ponašanja (stanja) aktera, Reagovanje na lifecycle događaje stvaranja i otkazivanja aktera, Udaljena komunikacija aktera
- Proizvoljni elementi: Supervizija aktera i perzistencija stanja aktera

Aktori

Training aktor:

- Trenira lokalni model i čuva ažurirane parametre modela
- Šalje ažurirane parametre agregatorima svih bolnica.
- Šalje ažurirane parametre Persistence akтору iste bolnice.
- Prima globalni model od agregatora i ažurira lokalni model.
- Evaluira lokalni i globalni model.

Agregator aktor:

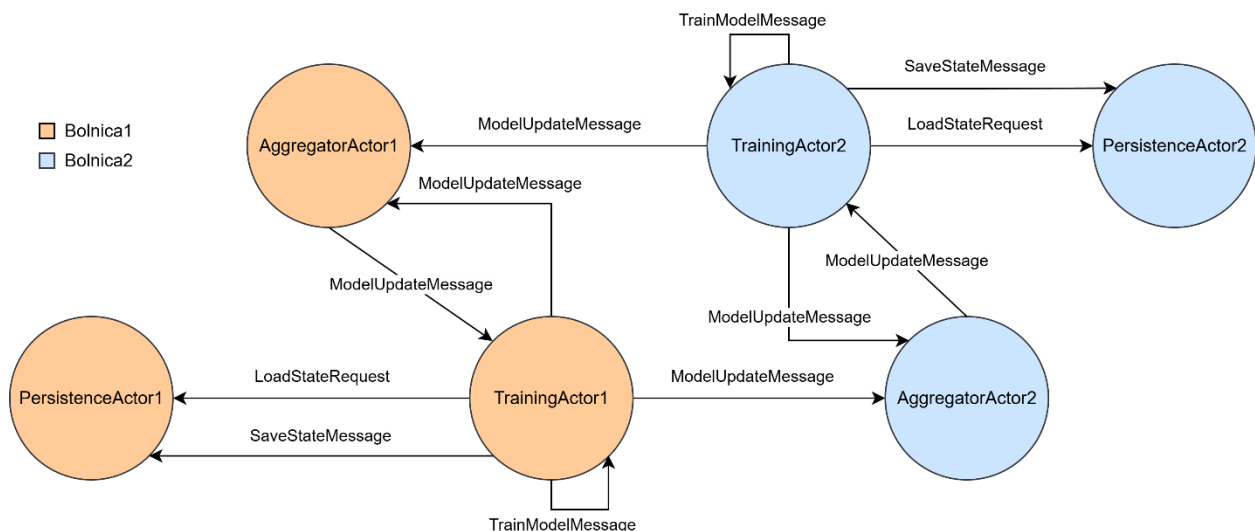
- Agregira parametre svih lokalnih modela i kreira globalni model.
- Šalje globalni model Training akтору iste bolnice.

Persistence aktor:

- Serijalizuje i skladišti parametre modela u datoteke.
- Učitava i deserijalizuje podatke iz datoteka i šalje Training akтору.

Poruke

- TrainModelMessage – poruka za pokretanje treniranja Training aktora
- ModelUpdateMessage – poruka sa ažuriranim parametrima modela
- LoadStateRequest – zahtev za učitavanje parametara modela iz datoteka
- SaveStateMessage – poruka za čuvanje stanja (parametara modela) Training aktora



Detalji implementacije

Koristiće se programski jezik *Go* za implementaciju svih delova projektnog zadatka.

Za udaljenu komunikaciju će se koristiti *gRPC* radni okvir.

Za perzistenciju stanja aktora će se koristiti biblioteka *BoltDB* (<https://github.com/etcd-io/bbolt>). Podaci će biti čuvani lokalno u datotekama u binarnom formatu kao ključ-vrednost.

Za serijalizaciju i deserijalizaciju podataka će se koristiti standardna *Go* biblioteka *encoding/json*.

Za obučavanje modela logističke regresije će se koristiti biblioteka *goml* (<https://github.com/cdipaolo/goml>).