

# Softverski agenti - specifikacija projekta

## Federativno učenje

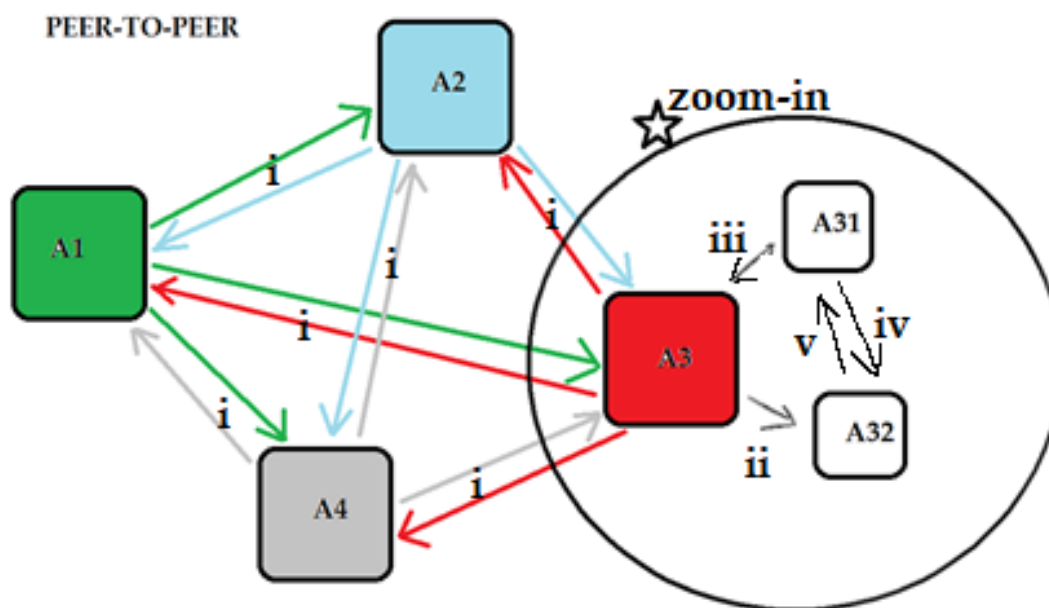
*Boško Kulušić RA122-2020*

*Matija Maksimović RA132-2020*

- Specifikacija algoritma koji će biti implementiran:  
Biće implementirana klasična neuronska mreža (*feedforward neural network*) koja ima ulazni sloj sa 8 neurona (zato što toliko parametara imamo u skupu podataka), dva skrivena sloja (sa 8 i 4 neurona) i izlazni sloj sa 1 neuronom (zato što predviđamo da li osoba ima ili nema dijabetes). Funkcije aktivacije koje se koriste su ReLU za skrivene slojeve i sigmoid za izlazni sloj.
- Skup podataka:  
U pitanju je skup na linku: [Diabetes - Intermediate ➡ Dataset \(kaggle.com\)](#). Skup podataka sadrži karakteristike koje se odnose na dijagnostičke mere za utvrđivanje dijabetesa kod 770 žena i podatak da li imaju ili nemaju dijabetes.
- Tehnologije za rad:  
Neuronska mreža biće implementirana pomoću jezika Golang kao i sami aktori za koje ćemo koristiti protoactor-go radni okvir. Komunikacija između aktera na odvojenim mašinama će se omogućiti uz upotrebu gRPC.

- Način distribucije treniranja algoritma:

Sistem je peer-to-peer, imaćemo 4 aktora u sistemu.



Plan je da jedan aktor zapravo bude tri manja aktora gde to izgleda ovako: imaće jednog aktora koji je zadužen da trenira mrežu(A31), jednog koji je zadužen da usrednjava težine(A32) i jednog koji će da služi kao interfejs za komunikaciju sa ostalima(A3).

Poruke koje se šalju u sistemu primarno će sadržati:

- težine koje se razmenjuju između aktora (A1,A2,A3,A4),
- pristigle težine koje interfejs aktor šalje akтору za usrednjavanje (A3,A32),
- težine između aktora koji obučava mrežu i interfejs aktora (A31,A3)
- težine od aktora koji obučava mrežu ka akтору za usrednjavanje (A31,A32)
- usrednjene težine koje aktor za usrednjavanje šalje akтору za obuku mreže (A32,A31)

Svaki aktor u peer-to-peer sistemu će imati svoj lokalni skup podataka na kojem će vršiti obučavanje mreže (koji predstavlja jedan deo originalnog skupa i biće mu unapred dat).

Nakon što istrenira 10 iteracija na svojim podacima i ažurira sve težine, aktor šalje svoje težine akoru za usrednjavanje i interfejs akтору putem poruka. Aktor za usrednjavanje radi uprosečavanje pristiglih težina sa svojim trenutnima (ovaj aktor može dobiti težine ili od strane drugih aktora iz sistema posredstvom interfejs aktora ili od strane svog aktora za treniranje mreže) i šalje ih akтору za obučavanje mreže. Zatim se ponovo vrši obuka mreže 10 iteracija i proces se ponavlja sve dok se ne prekorači neki unapred zadat broj iteracija(biće naknadno utvrđen) ili dođe do konvergencije loss funkcije(Binary Cross-Entropy Loss).

Algoritam za uprosečavanje: Aktor čuva niz težina za svaki aktor koji postoji u sistemu (u početku su svima postavljene na iste vrednosti), zatim dok trenira ako pristižu on upisuje sve vrednosti koje su stizale na odgovarajuće mesto i zatim kada dođe vreme da uproseči uzima te najaktuelnije i ažurira svoju težinu novim prosekom.

- Metod evaluacije rezultata:

Pošto je ovaj naš sistem federativnog učenja realizovan kao peer-to-peer, imaćemo verovatno različite težine mreže kod svakog od aktora. Nakon što svi aktori prestanu da ažuriraju svoje težine konačni rezultati će se evaluirati na svakom od aktora na validacionom skupu pomoću metrike Recall koja je čest izbor kada su medicinske predikcije u pitanju jer njena formula izgleda ovako:

$$\frac{\text{Tačna pozitivna predviđanja}}{\text{Tačna pozitivna predviđanja} + \text{Tačna negativna predviđanja}}$$