**Projekat iz predmeta Neuralne Mreže**

**Studenti :**

**Boris Deletić 2022/0255**

**Mihajlo Đurović 2022/0256**

**Projektovanje potpuno povezane neuralne mreže**

**A graph with blue squares

AI-generated content may be incorrect.**

**Slika:** Broj primeraka po klasama

Problem koji se rešava sadrži 9752 uzoraka koje je potrebno razvrstati u 2 okupacije.

Svaki uzorak sadrži sledeće atribute:

* Temperature
* Humidity
* Light
* CO2
* HumidityRatio
* Occupancy

Kriterijumska funkcija koja se koristi je binary\_crossentropy koja služi za binarnu klasifikaciju uzoraka u odgovorajuće klase.

Za izbor funckije aktivacije izabraće se relu ili tanh funckija aktivacije, u zavisnosti od najboljih hiperparametara koje model bude odredio.

Za minimizaciju funckije greške koriste se Droput, L2 regularizator i tehnika rano zaustavljanja.

Dropout se koristi da bi svaki ulaz prošao kroz različitu mrežnu arhitekturu. Na taj način mreža ne može lako da se preobuči. U toku treniranja nasumični neuroni se isključuju sa određenom verovatnoćom, dok tokom testiranja se svi neuroni koriste.

L2 regularizator dodaje penal na funkciju gubitka kako bi se sprečilo preobučavanje. Penal kojise dodaje jednak je λR(θ). Lambda predstavlja koeficijent regularizacije a R(θ) predstavlja sumu kvadrata svih težina u modelu.

Tehnika ranog zaustavljanje se koristi tako što zadamo maksimalan broj epoha i neku metriku koja se prati, recimo validation\_accuracy. Ako posle zadatog broja epoha nema napretka u pogledu metrike treniranje se zaustavlja.

Hiperparametri koje će model pronaći unakrsnom validacijom su:

* no\_units – Broj neurona u skrivem sloju
* act – Aktivaciona funckija
* reg- Koeficijent regularizacije
* drop – Verovatnoća za Dropout sloj.
* lr – Konstanta obučavanja

A screen shot of a computer program

AI-generated content may be incorrect.

**Slika**: Vrednosti hiperparametara koje će biti testirane

Iz priloženog koda vide se vrednosti hiperparametara koje će biti testirane tokom validacije.

A graph of accuracy and loss

AI-generated content may be incorrect.

**Slika**: Grafik promene performansi tokom epoha treniranja

Plava linija predstavlja trening skup a narandžasta validacioni skup.

A yellow and purple squares with numbers

AI-generated content may be incorrect.

**Slika**: Matrica konfuzije na trening skupu

A yellow and purple squares with numbers

AI-generated content may be incorrect.

**Slika**: Matrica konfuzije na test skupu

A screen shot of a computer

AI-generated content may be incorrect.

**Slika**: Vrednosti performansi klasifikacije

**Projektovanje potpuno konvolucione neuralne mreže**

Problem koji se rešava: klasifikacija zečeva i mačaka.

A blue rectangles with black text

AI-generated content may be incorrect.

**Slika**: Broj primeraka po klasama

A cat and rabbit in a grass field

AI-generated content may be incorrect.

**Slika**: Po jedan primerak svake klase

**Predprocesiranje**

Potrebno je izvršiti augmentaciju podataka kako bi se povećao broj primeraka kojim se trenira mreža. Primenjuje se augmentacija za rotaciju, uvećanje, flipovanje, kontrast i osvetljenost.

**Formiranje CNN**

**A screenshot of a computer

AI-generated content may be incorrect.**

Model se sastoji od sloja za augmentaciju, 3 konvoluciona sloja, 3 max pooling sloja, jednog potpuno povezanog sloja sa 128 neurona i jednog izlaznog sloja. Prvi konvolucioni sloj ima 448, drugi 4640 I treći ima 18496 parametara. Koristimo dropout tehniku kako se mreža ne bi preobučila. Na početku je izvršeno skaliranje pomoću Rescalling sloja kojim se ulazne vrednosti skaliraju u novi opseg.

A black background with white text

AI-generated content may be incorrect.

Ukupan broj parametara je 2 121 122.

Prilikom treniranja mreže izabrana je kriterijumska funkcija sparse categorical crossentropy koja je kodirana celim brojevima. Na primer za tri klase labele će biti 0, 1, 2.

Aktivaciona funckija koja se koristi prilikom treniranja je ReLu. Ako je vrednost ulaza manja ili jednaka nuli rezultat će biti 0 inače ta vrednost.

A graph of a function

AI-generated content may be incorrect.

Aktivaciona funkcija koja se korsiti u poslednjem sloju je softmax funkcija koja daje niz verovatnoća koje predstavljaju pripadnost klasama.

Tehnike koje se koriste za minimizaciju funkcije greške su Dropout i L2 regularizator.

Dropout se koristi da bi svaki ulaz prošao kroz različitu mrežnu arhitekturu. Na taj način mreža ne može lako da se preobuči. U toku treniranja nasumični neuroni se isključuju sa određenom verovatnoćom, dok tokom testiranja se svi neuroni koriste.

L2 regularizator dodaje penal na funkciju gubitka kako bi se sprečilo preobučavanje. Penal kojise dodaje jednak je λR(θ). Lambda predstavlja koeficijent regularizacije a R(θ) predstavlja sumu kvadrata svih težina u modelu.

A graph of loss and accuracy

AI-generated content may be incorrect.

**Slika**: Grafik promene performansi tokom epoha treniranja

Plava linija predstavlja trening skup, a narandžasta validacioni skup.

A screenshot of a computer program

AI-generated content may be incorrect.

**Slika**: Vrednosti performansi klasifikacije na trening i test skupu

A yellow and purple squares with black text

AI-generated content may be incorrect.

**Slika**: Matrica konfuzije na trening skupu

A chart of different colored squares

AI-generated content may be incorrect.

**Slika**: Matrica konfuzije na test skupu

A cat and a rabbit

AI-generated content may be incorrect.

**Slika**: Primer dobro klasifikovane i loše klasifikovane slike iz skupa podataka

**Projektovanje fuzzy regulatora**

**Sistem fuzzy upravljanja za praćenje referentne vrednosti objekta upravljanja**

Opredelili smo se za intuitivno fuzzy upravljanje.

Postupak projektovanja:

1. Podešavamo odgovarajuću funkciju prenosa na osnovu zahteva.
2. Traženje univerzuma za grešku koja će biti ulazna varijabla za fuzzy mašinu. Univerzum određujemo na osnovu ograničenja upravljanja.

A graph on a black background

AI-generated content may be incorrect.

Greška e predstavlja odstupanje od reference pa se računa po formuli:

e = r – y, pa je naš univerzum za grešku [-9 9].

1. Kada dobijemo odgvorajuci univerzum greske pravimo fuzzy masinu sa jednim ulazom: Modelovanje se bazira na podesavanju univerzuma ulaza(e) I izlaza , stepena pripadanja i fuzzy pravila.
2. Pošto nije došlo do očekivane vrednosti zasićenja ubačen je integrator i blok za saturaciju sa ograničenjem na [-1 1], ograničenje upravljanja, kako bi dostigli finalno stanje ali i ograničili da ne pređemo finalno stanje.

A screen shot of a graph

AI-generated content may be incorrect.

1. Pošto nije došlo do zasićenja uvodimo novu ulaznu varijablu, izvod greske. Univerzum izvoda greške odredili smo sa grafika izvoda greske [-0.5,0.5]. Dodajemo nova pravila.
2. Kao rezultat dobijena je funkcija koja ima zasićenje u maksimalnoj referentnoj vrednosti.

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

Žuta linija predstavlja ulaz u fuzzy inference.

Plava linija predstavlja regulisanu varijablu.

Signal upravljanja:

A black screen with white lines

AI-generated content may be incorrect.

A diagram of a machine

AI-generated content may be incorrect.

**Slika**: Celokupan model upravljanog sistema

**Sistem fuzzy upravljanja za potiskivanje nemerljivog poremećaja na ulazu objekta upravljanja**

Opredelili smo se za intuitivno fuzzy upravljanje.

Postupak projektovanja:

1. Podešavamo odgovarajuću funkciju prenosa na osnovu zahteva.
2. Traženje univerzuma za grešku koja će biti ulazna varijabla za fuzzy mašinu. Univerzum određujemo na osnovu ograničenja upravljanja.

A graph on a black background

AI-generated content may be incorrect.

Greška e predstavlja odstupanje od reference pa se računa po formuli:

e = r – y. Pošto je referentna vrednost uvek 0 naš univerzum za je grešku [-7 7].

1. Kada dobijemo odgvorajuci univerzum greske pravimo fuzzy masinu sa jednim ulazom: Modelovanje se bazira na podesavanju univerzuma ulaza(e) I izlaza , stepena pripadanja i fuzzy pravila. Razlika u odnosu na prethodni zadatak je što sada imamo i dodatni poremaćaj čija je maksimalna amplituda jednaka polovini maksimalne amplitude upravljačkog signala.
2. Pošto nije došlo do očekivane vrednosti zasićenja ubačen je integrator i blok za saturaciju sa ograničenjem na [-1 1], ograničenje upravljanja, kako bi dostigli finalno stanje ali i ograničili da ne pređemo finalno stanje.

A screen shot of a graph

AI-generated content may be incorrect.

1. Pošto nije došlo do zasićenja uvodimo novu ulaznu varijablu, izvod greske. Univerzum izvoda greške odredili smo sa grafika izvoda greske [-0.2, 0.2]. Dodajemo nova pravila.
2. Kao rezultat dobijena je funkcija koja ima zasićenje u maksimalnoj referentnoj vrednosti.

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

A black screen with a black grid

AI-generated content may be incorrect.

Žuta linija predstavlja ulaz u fuzzy inference.

Plava linija predstavlja regulisanu varijablu.

Signal upravljanja:

A screen shot of a graph

AI-generated content may be incorrect.

A diagram of a computer system

AI-generated content may be incorrect.

**Slika**: Celokupan model upravljanog Sistema

Razlika u odnosu na prethodni sistem je jedino što imamo dodatni poremećaj pa je potrebno i njega uključiti kao ulaz u objekat upravljanja.