

Sovelluksen toiminnot

Projektissa syntynyt sovellus tunnistaa käsinkirjoitettuja numeroita neuroverkkoon perustuvan tekoälyn avulla. Kyseessä on neuroverkkoa soveltavan järjestelmän prototyyppi, jota on tarkoitus tulevaisuudessa jatkokehittää ratkaisemaan monimutkaisempia ongelmia.

Sovelluksessa on käyttöliittymä, jossa käyttäjä voi piirtää hiirellä numeron. Lähetettyään numeron neuroverkolle käsiteltäväksi, näytölle ilmestyvät eri vaihtoehtojen todennäköisyydet. Suurimman todennäköisyyden saanut numero näkyy näytöllä lihavoituna.

Sovellusta opetetaan valmiin kuva-aineiston avulla. Käyttäjä voi käynnistää opetuksen käyttöliittymästä, jossa hän voi määrittää kuinka paljon kuvia neuroverkolle näytetään 1 ja 60000 kuvan välillä. Käynnistettyään harjoittelun, käyttäjä voi seurata harjoituskuvien lukemisen etenemistä progressbarin avulla. Ennen harjoittelun käynnistämistä käyttäjä voi määrittää neuroverkolle learning raten 0 ja 1 väliltä. Harjoittelun tuloksena muodostuneet synapsien ja neuronien painot voidaan tallentaa tietokantaan käyttöliittymän "save session" -nappia painamalla ja hakea ne seuraavalla kerralla "load session" nappia painamalla.

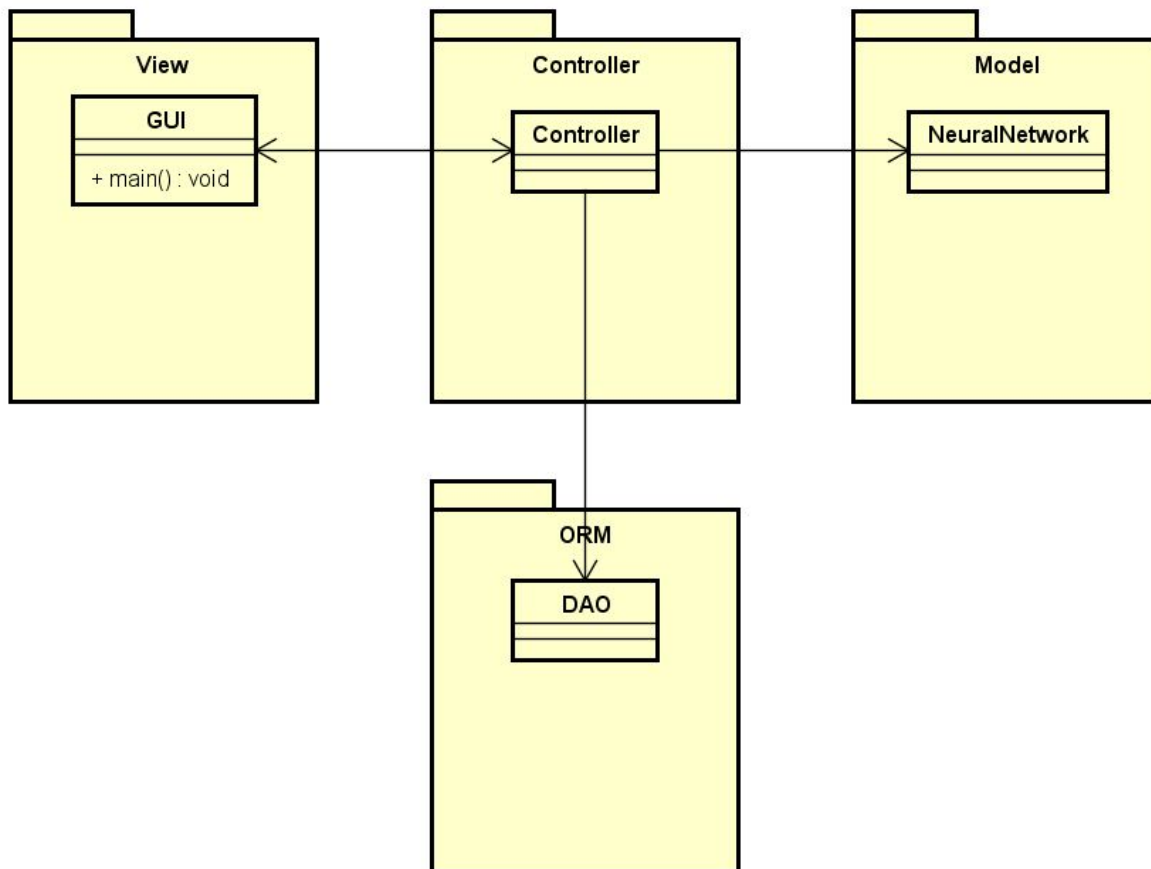
Valmiin kuva-aineiston avulla harjoittelun lisäksi käyttöliittymästä löytyy ominaisuus, jolla käyttäjä voi itse opettaa neuroverkkoa. Käyttäjä voi syöttää neuroverkolle oikean numeron, mikäli neuroverkon ilmoittama tulos on väärä. Tätä ominaisuutta ei kuitenkaan vielä ole toteutettu model-tasolla, koska käyttäjän syöttämien kuvien määrä tulisi olla todella suuri, jotta opetuksella olisi vaikutusta neuroverkon oppimiseen. Tämä ominaisuus toteutetaan projektin jatkokehitysvaiheessa.

Arkkitehtuuri

Java-koodi koostuu neljästä pakkauksesta: Model-, View-, Controller- ja Orm-pakkauksesta. Controller-pakkaus sisältää ohjelmaa hallitsevan Controller-luokan, joka kommunikoi muiden pakkausten kanssa ja ohjaa sovellusta. Controller-pakkaus sisältää myös IDXReader-luokan, jonka tarkoitus on lukea harjoittelu- ja testaus-tiedostoja.

Model-pakkaus sisältää neuroverkon toiminnallisuuden (NeuralNetwork) sekä luokat neuroverkon käyttämille matriiseille(Matrix) ja inputeille(InputData). View sisältää toiminnallisuuden näkymän luomiseksi käyttäjälle. Gui-luokka luo näkymän sekä NeuroCanvas-luokan ilmentymän, jolle käyttäjä voi piirtää numeron neuroverkon tunnistettavaksi.

Orm-pakkaus sisältää työkalut, joilla kommunikoidaan sovelluksen tietokannan kanssa. Sovellus käyttää Hibernatea ja Orm-pakkaus sisältää Hibernaten hyödyntämät Data Access Object(DAO) luokat, joilla luodaan olion ja tietokannan välille yhteys.



Tietokanta

Tietokanta koostuu kolmesta taulusta: Node, NodeSynapse ja Synapse. Node sisältää yhden solmun ID:n, solmun järjestysluvun tasossa, tason numeron sekä solmun biaksen. Synapse kuvaa kahden solmun yhteyttä ja taulu sisältää painon sekä tason numeron. NodeSynapse-taulu on Noden ja Synapsen välinen liitostaulu, jonka tarkoitus on Noden ja Synapsen välisen m-m suhteen muuttaminen kahdeksi 1-m suhteeseen. Direction on vanhentunut kenttä, joka todennäköisesti poistetaan tulevaisuudessa.

