**Dokumentacja projektu na zajęcia z Podstaw sztucznej inteligencji 2016/17 – Marcin Lichota**

**Cel projektu :** Stworzenie programu który będzie w stanie rozpoznawać 3 kategorie artykułów (związane z : pogodą , sportem oraz szeroko rozumianą informatyką) . Program powinien mieć możliwość stworzenia oraz nauczenia sieci neuronowej jak również ostatecznego działania na nauczonej sieci.

**Cel dodatkowy:** Wizualizacja procesu uczenia.

**Realizacja:**

1. Pierwsza część projektu obejmowała gromadzenie oraz konwersje danych. W moim programie rola konwersji danych przypada modułowi formatter (formatter.cpp, formatter.hpp). Celem konwersji było utworzenie „wygodnych” i zminimalizowanych danych służących dalej jako wejście sieci neuronowej. Proces gromadzenia i konwersji danych przebiegał następująco:

* Zgromadzenie artykułów z tematycznych stron internetowych.
* Usunięcie wszystkich znaków końca linii z pojedynczego artykułu (dla prostoty implementacji). Dane te znajdują się w pliku „artykuly.txt” w folderze „src”.
* Podzielenie słów występujących we wszystkich zgromadzonych artykułach na klasy odmian. (zrealizowane dzięki bazie odmian wszystkich słów języka polskiego ok 4 mln słów).
* Zgromadzenie wszystkich odmian znajdujących się w zgromadzonych artykułach. (To stanowi wejście sieci neuronowej)
* Oznaczenie występowania słowa danej klasy jako 1 oraz jego braku jako 0.

2. Druga część projektu obejmowała implementacje generatora sieci neuronowej wraz z wszystkimi funkcjami potrzebnymi do realizacji zadania. Jako tym sieci zdecydowałem się na feedforward ze względu na prostotę implementacji oraz możliwość względnie wygodnego sposobu wyświetlania procesu uczenia. Jako technikę trenowania sieci wybrałem propagacje wsteczną. (Moduł generatora sieci neuronowej znajduje się w plikach Net.cpp oraz Net.hpp).

3. Ostatnią częścią projektu była implementacja modułu rysującego proces uczenia. Do tego celu wykorzystałem bibliotekę SFML. Wynikiem tej części projektu jest dynamiczna wizualizacja warstwy wejścia, neuronów warstw ukrytych oraz warstwy wyjścia, połączonych ze sobą liniami o grubości odpowiadającej wartości wag i kolorze determinującym znak danej wagi (niebieski – dodatnia, czerwony – ujemna, szara – bliska 0 lub równa 0).

**Wyniki:**

Zrealizowałem wszystkie założone cele oraz otrzymałem wyniki satysfakcjonujące. Dla nauczonej sieci przeprowadziłem test i otrzymałem 90% skuteczności. W celu przetestowania sieci wystarczy uruchomić program, wczytać nauczoną sieć (lub nauczyć nową), wkleić do pliku „ask\_neuron\_net.txt” dowolny arytkuł dot. pogody, sportu lub IT i w menu programu wybrać opcje nr 4 zatwierdzając klawiszem ENTER.

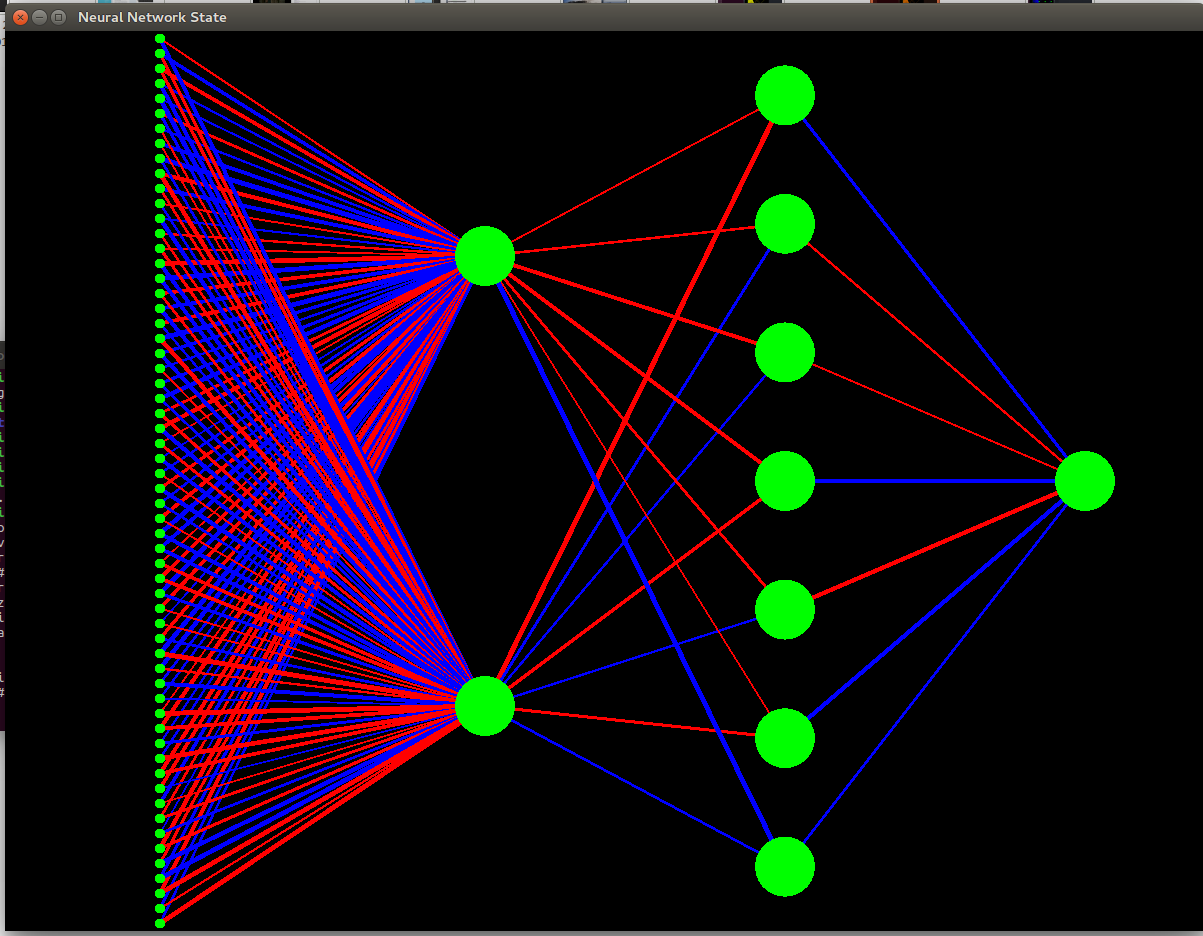
**Wnioski:**

Ten projekt wskazał mi główne problemy jak i samą idee działania sieci neuronowej. Dzięki podzieleniu programu na moduły mam możliwość wykorzystania generatora sieci do innych programów. Sukces działania mojego programu wskazuje iż sieci neuronowe jest to zagadnienie przyszłościowe i można za ich pomocą rozwiązywać problemy na różnych poziomach abstrakcji.

**Uwagi:**

1. Poniżej zamieszczam zdjęcie z procesu uczenia sieci.

2. Należy zauważyć iż program jest napisany dla systemów Linux i nie daje gwarancji na uruchomienie go na innych systemach operacyjnych. Aby uruchomić program należy poprać zawartość mojego repozytorium i w konsoli wykonać polecenie ./main .

****