Slovenská technická univerzita
Fakulta informatiky a informačných technológií
Ilkovičova 3, 842 16 Bratislava 4

Speed Dating - Binary classification

Autori: Patrik Malina, Martin Bopko
Študijný odbor: Inteligentné softvérové systémy
Predmet: Neurónové siete
Ak. rok: 2023/2024 LS

1 Úvod

Náš projekt bol zameraný na binárnu klasifikáciu. Pracovali sme s datasetom Speed Dating, ktorý je voľne dostupný na: OpenML SpeedDating.
Tento dataset obsahoval zaznamenané dáta párov ktoré sa zúčastnili podujatia „Speed Dating“. Zaznamenane dáta sú napr. vek, pohlavie, koníčky, práca, preferencie partnerov a podobne. Kľúčovým atribútom bola hodnota „match“ ktorý vyjadruje, či sa daný pár znovu stretol alebo nie. Našou úlohou bolo predikovať tuto hodnotu podlá daných parametrov z datasetu.

2 Riešenie

Pred vytvorením nášho modelu sme sa museli najprv oboznámiť so datasetom a predspracovať ho. Dataset bolo potrebné upraviť tak, aby všetky hodnoty boli rovnakého typu a zároveň aby v ňom nechýbali žiadne hodnoty. Na spracovanie sme použili enkodovanie pomocou „one-hot“ enkodera a tiež ordinálneho enkodera. Na doplnenie chýbajúcich hodnôt sme najprv normalizovali pomocou normalizácie MinMax a chýbajúce hodnoty sme doplnili pomocou KNN (k-nearest neighbors).

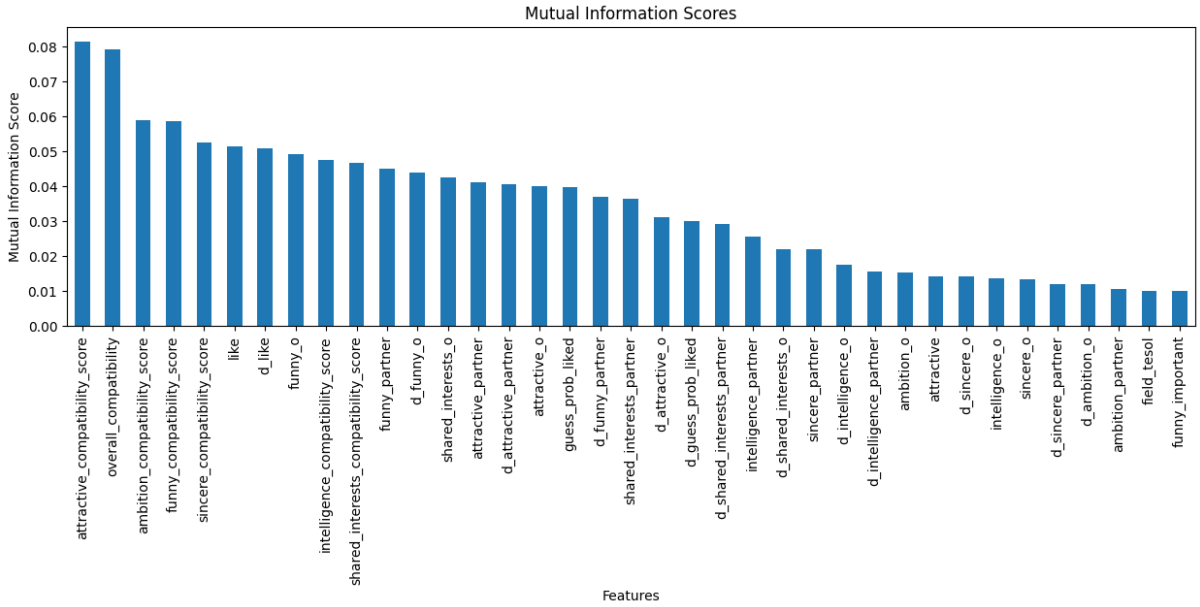
Jednou zo zaujímavých veci ktoré sme spravili bolo vytvorenie nových stĺpcov ktoré vyhodnocovali kompatibilitu ako čo je atraktívnosť alebo inteligenciu a podobne. Na konci sme pridali aj celkovú kompatibilitu medzi pármi. Čo bolo hodnota bližšia k číslu 1 to znamenalo, že bola väčšia kompatibilita.

```
# Calculate the absolute difference between preferences and ratings
compatibility_columns = ['attractive', 'sincere', 'intelligence', 'funny', 'ambition', 'shared_interests']

for column in compatibility_columns:
  clean_data[f"{column}_compatibility_score"] = (
      (1 - abs(clean_data[f"pref_o_{column}"] - clean_data[f"{column}_partner"])) +
      (1 - abs(clean_data[f"{column}_important"] - clean_data[f"{column}_o"]))
  ) / 2
```

Obrázok 1 Kód na výpočet kompatibility

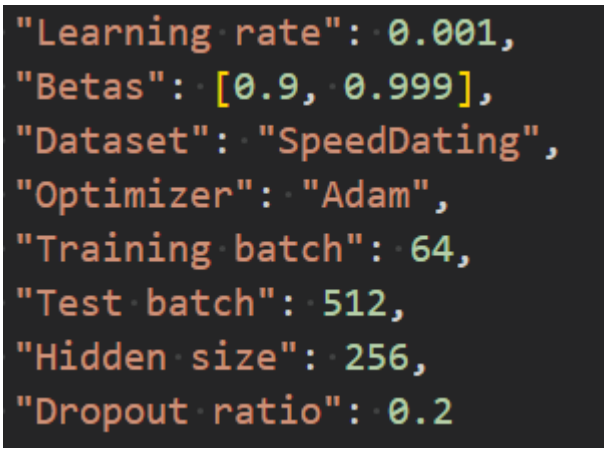
Taktiež sme vytvoril graf ktorý nám znázorňuje ako hodnoty medzi sebou korelujú a to sme urobili pomocou korelácie „Mutual Information“. Na grafe sú znázornene iba hodnoty s koreláciou väčšia ako 0.01.



Obrázok 2 Mutual Information korelácia

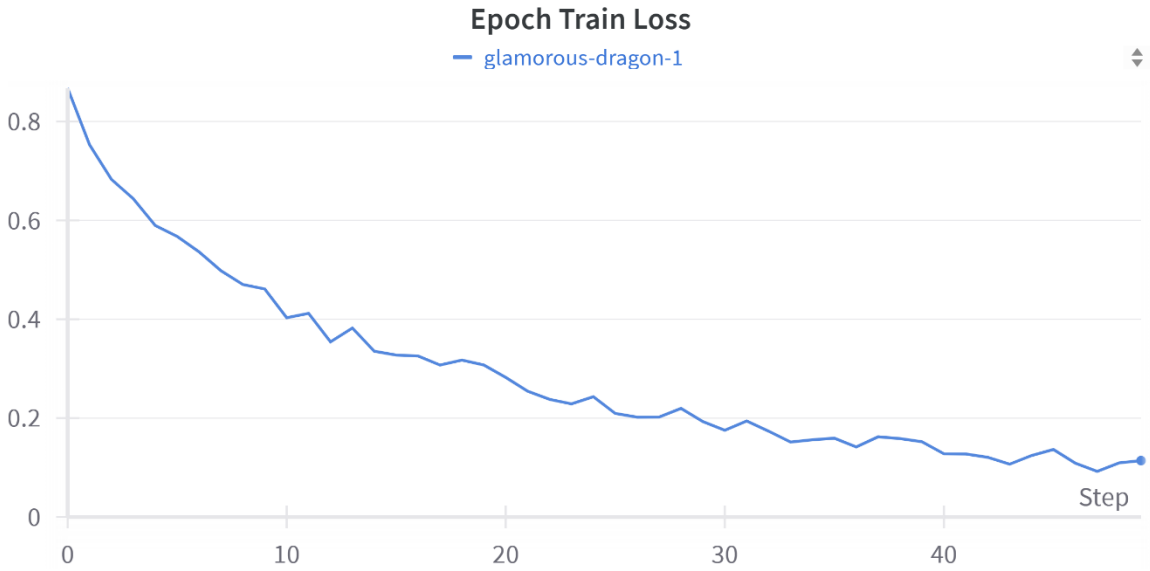
Údaje sme ďalej rozdelili na tréningové údaje, ktoré obsahovali 80 % celkových údajov, a testovacie údaje, ktoré tvorili 20 %. Naša trieda "SimpleMLP" obsahuje 2 skryté vrstvy, zároveň sme pridali „Batch normalisation“ a tiež „Dropout“ vrstvy. Na optimizer sme použili Adam a na stratovú funkciu BCE. Do stratovej funkcie sme tiež vložili parameter ktorý dáva väčší význam menšinovej triede z dôvodu, že náš dataset je nevyvážený (84% un-matched, 16% matched).

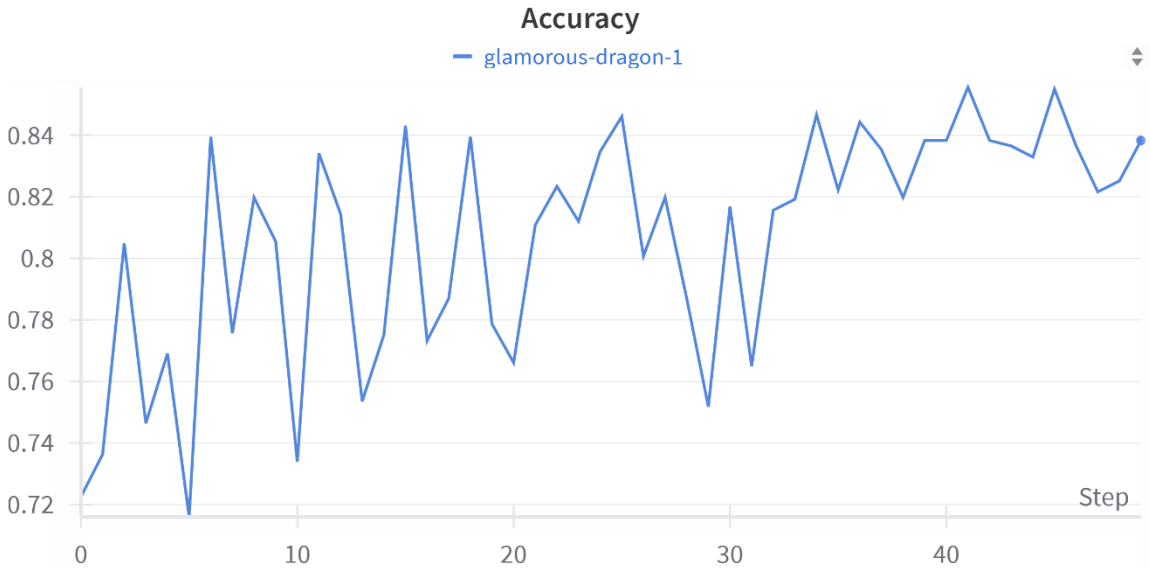
Pri trénovaní modelu sme nastavili na 50 epoch s danými parametrami:

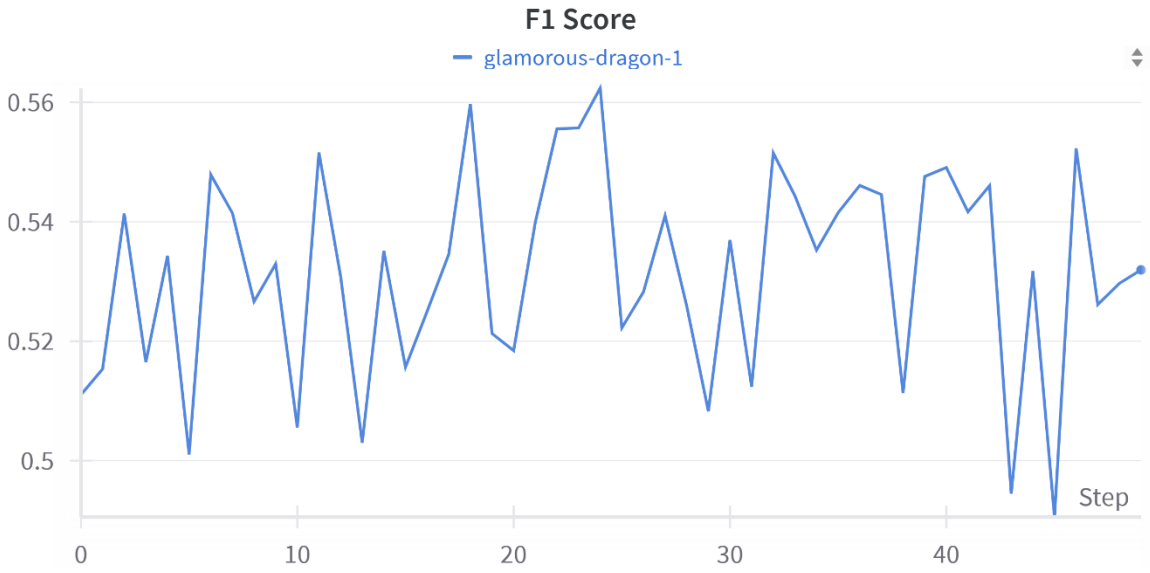


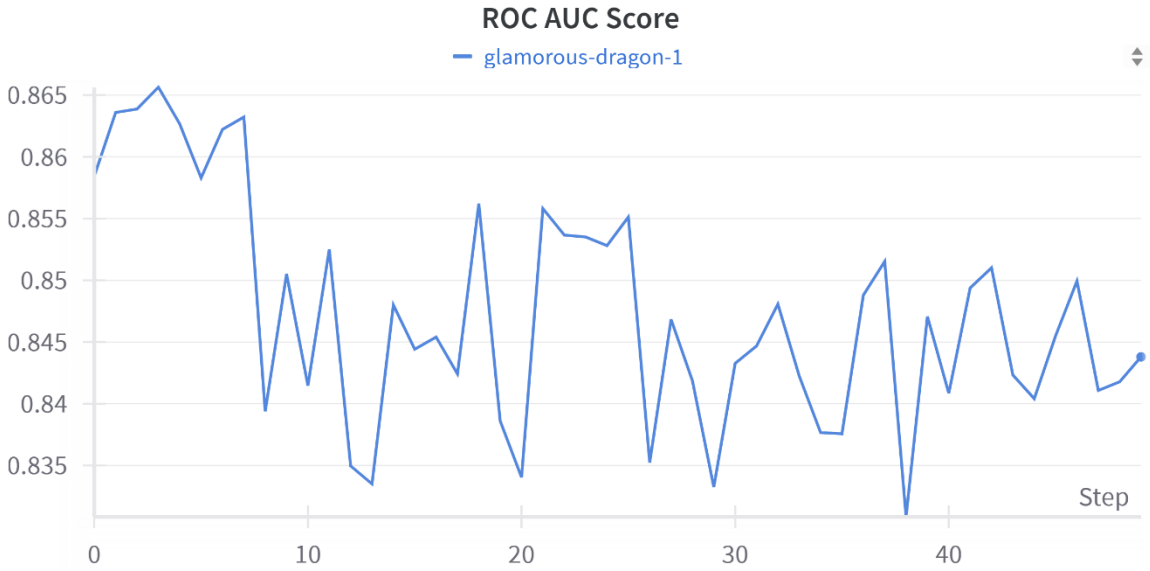
Obrázok 3 Parametre modelu

Výsledky ktoré sme získali:









Taktiež jednou z hlavných veci bolo použiť experimentálnu metódu „weighted averages“ pri používaní Adam optimajzera. To znamená, že počas trénovania sa uchovávajú všetky váhy a na konci trénovania sa vytvorí priemer tých váh a nasadí sa na novy model. Taktiež sme aj skúsili použiť nie všetky váhy ale iba váhy pomocou ktorých model dostal výsledok F1 viac než 0.51.

Metoda	Accuracy	ROC AUC	F1 score
Žiadna metóda	0.83831	0.8438	0.53195
Weighted averages	0.83233	0.8341	0.51801
Weighted averages > 0.51	0.83353	0.8404	0.52951

3 Záver

V niekoľkých pokusoch sme získali podobné hodnoty a zistili sme, že nie vždy je vhodné použiť metódu „weighted averages“. Je vhodné ju použiť len v prípadoch, keď je model nestabilný a keď hodnoty metrík počas trénovania dosť skáču.