

NPRG035 Nástroj pro prohlížení modelů v systému Mozaik

Kateřina Čížková

23. září 2020

1 Uživatelská dokumentace

Cílem projektu bylo implementovat nástroj který umožní vizualizaci neurálních modelů specifikovaných v systému Mozaik.

Program zobrazuje neurony ve vrstvách a spojení mezi nimi. Data jsou načtena z DataStore z knihovny Mozaik. Spustí se pomocí příkazu

```
bokeh serve --show visualize --args path_to_mozaiik_datastore
```

ve složce mff_nprg035/. Vizualizaci je pak možné zobrazit ve webovém prohlížeči na adrese <http://localhost:5006/visualize>.

V DataStore jsou neurony organizovány do vrstev. Pro každou vrstvu je vykreslen graf s příslušnými neurony a s názvem dané vrstvy. Tento graf je možné zoomovat a posunovat.

Spojení lze pak zobrazit vybráním konkrétních neuronů. Výběr je možný buď pomocí „lasa“ (označení obvodu vybrané oblasti) nebo kliknutím přímo na daný bod. Pokud chceme vybrat více bodů nebo nesouvislých oblastí, stačí při výběru držet klávesu Shift.

Neurony, které jsou přímo vybrané, se zobrazují žlutě, červeně se zobrazují ty, které jsou s nimi propojené, a modře všechny ostatní. V rámci vrstvy jsou spojení znázorněna také čarami. Pokud je vybrán pouze jeden neuron, zobrazí se navíc v tooltipu u jeho sousedů, jaká je váha a zpoždění konkrétního spojení (jinak je u těchto dvou hodnot vždy zobrazena 0) a zároveň se informace o spojení zobrazí vpravo od grafů ve formě textu.

Pomocí tlačítek v horní části se dá zvolit, zda se budou zobrazovat spojení, která přichází do vybraných neuronů, nebo ta která z nich odchází. Také je zde tlačítko reset, kterým lze všechny neurony „odvybrat“.

2 Programátorská dokumentace

2.1 Technologie

Celý program je napsán v programovacím jazyce Python. Pro reprezentaci sítě neuronů a spojení mezi nimi jsem použila knihovnu NetworkX ¹. Pro samotnou vizualizaci pak knihovnu Bokeh ².

2.2 Princip zobrazování obecně

Nejprve jsou načtena data z Mozaik DataStore. Z nich je vytvořena síť, ve které má každý z neuronů atributy s informacemi o svých souřadnicích a vrstvě a každé spojení má atributy s informacemi o váze a zpoždění. Z této sítě jsou pak postupně vytvořeny jednotlivé grafy pomocí funkce `from_networkx` ³ z knihovny Bokeh. Pokud uživatel vybere nějaké neurony a chce tedy zobrazit spojení vedoucí do/z nich, využívá se již vytvořená síť k vyhledání jejich sousedů.

Zdrojový kód je rozdělen do tří souborů: `main.py`, `graph.py` a `interactivity.py`. Část `graph.py` slouží k vytvoření sítě a grafů jednotlivých vrstev, `main.py` slouží k vlastnímu zobrazení vizualizace a `interactivity.py` obsahuje funkce pro obsluhu akcí uživatele (například stisknutí tlačítka nebo výběr některých neuronů).

2.3 `graph.py`

Síť je uložena jako NetworkX DiGraph, tedy orientovaný graf. Do grafu jsou nejprve vloženy všechny neurony postupně po vrstvách. Jejich pozice jsou získány z DataStore pomocí `datastore.get_neuron_positions()[sheet][0]` pro souřadnice x a `datastore.get_neuron_positions()[sheet][1]` pro souřadnice y. Poté jsou přidána spojení z

```
datastore.get_analysis_result(identifier='Connections')
```

Při vytváření GraphRenderer ⁴ pro jednotlivé vrstvy je ze vrcholů sítě vždy vytvořena nová síť obsahující pouze vrcholy z dané vrstvy. Pro vytvoření grafu je použita již zmíněná funkce `from_networkx`, ovšem do jejího

¹<https://networkx.github.io>

²<https://bokeh.org/>

³https://docs.bokeh.org/en/latest/docs/user_guide/graph.html#networkx-integration

⁴<https://docs.bokeh.org/en/latest/docs/reference/models/renderers.html#highlight=graphrenderer#bokeh.models.renderers.GraphRenderer>

ColumnDataSource pro vrchly je třeba přidat sloupec **selected** se samými nulami (značí, že neuron není vybrán, zobrazí se tedy modře – vybraný je pak označen 1, soused vybraného 2), který v původní síti není.

Též jsou přidány nulové sloupce **weight** a **delay**, to proto, aby bylo možné v případě, že je vybrán pouze jeden vrchol, zobrazit u jeho sousedů informace o váze a zpoždění spojení.

2.4 main.py

main.py obsluhuje vytvoření grafů pro každou vrstvu pomocí knihovny Bokeh. Pro každou vrstvu vytvoří figure ⁵ s potřebnými tools. Také jsou vždy počítány rozsahy souřadnic, aby byl výsledný graf správně přiblížený, to ovšem již nebude potřeba ve verzi Bokeh 2.3 ⁶.

Také jsou vytvořena tlačítka pro změnu směru zobrazovaných spojení a tlačítko reset a vše je na závěr přidáno do aktuálně zobrazovaného dokumentu.

2.5 interactivity.py

Pokud uživatel vybere nějaký neuron nebo neurony, je třeba změnit v GraphRendereru všech vrstev informace o tom, které neurony jsou zrovna vybrány, a do vrstvy, ve které výběr proběhl, vykreslit hrany.

Změna probíhá až poté, co je výběr dokončen (**event.final**). Při dynamickém vykreslování při každém přidání nového bodu do výběru to bylo velmi pomalé.

Bokeh bohužel vrací selection pouze jako indexy vybraných vrcholů v ColumnDataSource, to znamená, že je potřeba nejprve zjistit jejich indexy v NetworkX síti. K tomu slouží funkce

```
selected_nx_index(graph_renderer,new_selected_nodes).
```

Ta bere jako první parametr graph_renderer, ve kterém výběr probíhá. Z něj můžeme získat indexy vybraných vrcholů pomocí

```
source = graph_renderer.node_renderer.data_source
```

```
indices_in_renderer = source.selected.indices
```

Jako druhý parametr dostane list nových hodnot „selected“ pro tento graph_renderer a rovnou jej upraví.

⁵<https://docs.bokeh.org/en/latest/docs/reference/plotting.html?highlight=figure#bokeh.plotting.figure>

⁶<https://github.com/bokeh/bokeh/issues/10472>

Pak ze sítě získáme všechny sousedy vybraných vrcholů. Podle toho, zda je proměnná `edges_in` nastavena na `True` nebo `False` jsou jako sousedé vybráni buď předkové nebo následníci daného vrcholu v síti. Pokud je soused ve stejné vrstvě, přidáme příslušnou hranu a jinak jej pouze přebarvíme.

2.6 Možné pokračování v práci

Možná by bylo lepší zkusit najít pro zobrazování jinou knihovnu, než Bokeh, nebo začít úplně od začátku. V průběhu práce jsem totiž zjistila, že se knihovna bokeh na zobrazování sítí příliš nehodí, není na to primárně určena. Například již zmíněné automatické nastavení rozsahu os, které by mělo pro síť fungovat ve verzi Bokeh 2.3, nyní již pro všechny ostatní renderery funguje a ani dokumentace se o tom, že pro grafy nejde použít, nezmiňovala.

Také jsem narazila na problém, že aktuálně nelze přidat tooltip zvlášť pro hrany a vrcholy, proto se nyní váha a zpoždění hrany zobrazuje v tooltipu sousedního vrcholu.

Aktuálně je bug v Bokeh 2.2.1, takže se vizualizace sice zobrazí, ovšem v terminálu hlásí `KeyError: 'id'`. Viz můj dotaz v Bokeh diskuzi <https://discourse.bokeh.org/t/keyerror-id-maybe-issue/6365>.

Též jsem narazila na problém, když jsem se snažila pro aktualizaci vizualizace použít patch ⁷, aby se nemusel pokaždé aktualizovat celý sloupec. I na tento problém jsem se ptala v diskuzi <https://discourse.bokeh.org/t/weird-problem-with-selection-on-change-callback-patching/6289>. Opět jde o bug, který se v bokeh objevuje při vizualizaci sítí.

⁷https://docs.bokeh.org/en/latest/docs/user_guide/data.html?highlight=streaming%20data#patching