

Python pro praxi – 2. domácí úkol

One-shot klasifikace pomocí siamské sítě.

Cílem DU je sestavit, natrénovat a otestovat siamskou síť pro potřeby one-shot klasifikace (klasifikace na základě jediného známého příkladu pro některé třídy). One-shot klasifikace je důležitá zejména v situacích, kdy se počet tříd může dynamicky měnit a není možné pro každou změnu přetrénovávat celý model (například vizuální identifikace aktuálních zaměstnanců velké korporace). Pro tyto případy je možné použít nějakou variantu siamských sítí (viz https://en.wikipedia.org/wiki/Siamese_neural_network), které se zaměřují na binární klasifikaci páru objektů (jsou/nejsou reprezentanty stejné třídy).

V DU budete vycházet z Fashion-MNIST datasetu (<https://github.com/zalandoresearch/fashion-mnist>), který pro potřeby testování obohatíme o několik nových tříd. Váš úkol bude postupně:

- vytvořit model siamské sítě (přepokládá se použití `tf.keras`)
- natrénovat model na trénovacích datech Fashion-MNIST datasetu (train set budete muset upravit pro potřebu pairwise binární klasifikace)
- obohatit Fashion-MNIST dataset o několik nových tříd (detaily níže)
- otestovat model jak na původních testovacích datech, tak i na reprezentantech nových kategorií
- reportovat dosažené výsledky (ideálně také alespoň částečně prozkoumat vliv hyperparametrů modelu na výsledky)

Jako základ pro vytvoření modelu siamské sítě můžete použít script z posledního cvičení (https://www.ksi.mff.cuni.cz/~peska/vyuka/nprg065/2019/nprg065_03_task.ipynb). Minimální očekávaná přesnost modelu na validačních datech je alespoň 95%. Obohacení Fashion-MNIST datasetu proveďte dle vlastního uvážení. Zvolte ale alespoň 10 nových tříd, které jsou disjunktní s původními Fashion-MNIST datasetem: třídy buď rozšiřují doménu módy (hodinky, čepice,...), nebo se týkají jiné ucelené domény (například druhy nábytku). Z každé třídy vyberte alespoň 10 reprezentantů a převedte je do Fashion-MNIST formátu (greyscale, 28x28). Využít můžete například podmnožinu ImageNet datasetu. Z každé třídy pak vyberte jednoho reprezentanta (known class member), vůči kterému budete ostatní data testovat.

Při testování porovnejte zvláště přesnost klasifikace na původních testovacích příkladech, tak i nově vytvořených třídách (nicméně, každý testovací objekt porovnáváte s reprezentanty všech tříd, tj. provádíte klasifikaci alespoň do 20 tříd). Zobrazte také nějakou variantu confusion matrix (například jako heatmapu). Očekává se rozumná performance jak na původních, tak rozšířených datech (rozumnost je definována překonáním jednoduchého baselinu implementovaného vyučujícím). Preferovaný způsob odevzdání je (komentovaný) python notebook a zip archiv obsahující rozšiřující objekty datasetu.

Možné rozšíření:

- Jaký vliv na výsledek má počet a výběr reprezentativních class members při testování (náhodný výběr, centroidy,...)?
- Jsou si nějaké třídy výrazně podobné (např. i dle class probability)?
- Má na výsledky nějaký vliv podobnostní metrika implementovaná siamskou sítí?