

KNN: Rozpoznání matematických rovníc

Jakub Málek
Tomáš Milostný
Marek Večeřa

17. května 2025

`{xmalek17,xmilos02,xvec31}@vutbr.fit.cz`

Akademický rok 2024/2025



Obsah

1	Úvod	3
2	Popis úlohy	3
3	Přehled existujících řešení	3
4	Popis vlastního řešení	4
4.1	Experimenty	4
4.2	Výsledky ~ Evaluace (Kvantitativní vs. kvalitativní)	4
5	Závěr ~ Takeaway messages	4
6	Závěr	5
7	Kuba CHATGPT	5
	Zdroje	6

1 Úvod

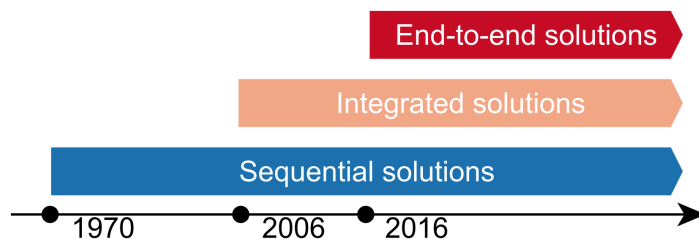
Tento dokument popisuje problematiku a řešení rozpoznání matematických rovnic. Jedná se o velmi aktuální téma, které se v posledních letech stalo předmětem mnoha výzkumů a aplikací. Rozpoznávání matematických vzorců je důležité pro automatizaci různých úloh, jako je převod tištěných nebo ručně psaných vzorců do digitálního formátu, což usnadňuje jejich zpracování a analýzu.

2 Popis úlohy

Úkolem projektu je vytvoření a natrénování modelu pomocí strojového učení, který bude schopný rozpoznat obrázky matematických výrazů na vstupu a převést je do strojového zápisu. Formát vstupních obrázků jsme si zvolili jako obrázky ručně psaných rovnic, které budou následně převedeny do formátu LaTeX.

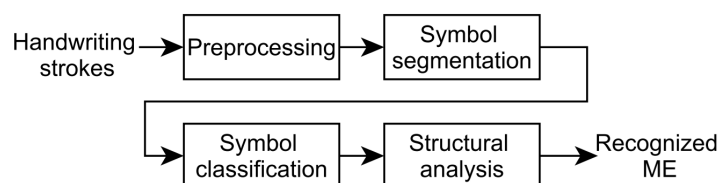
3 Přehled existujících řešení

V celé historii problematiky rozpoznávání ručně psaných matematických rovnic se objevily tři odlišné přístupy 1. Na počátku se jednalo o jednodušší více přímočaré metody označované jako sekvenční řešení [1].



Obrázek 1: Časový vývoj různých metod pro rozpoznávání matematických vzorců

Sekvenční řešení použije princip dekompozice a tuto metodu lze rozložit do několika částí (obrázek 2). Modely pracující na tomto principu nejprve (1) rozpoznají jednotlivé symboly a následně provedou (2) strukturní analýzu, kde jednotlivé symboly poskládají do celého výrazu. Rozpoznávání symbolů lze pak dále rozložit na segmentaci jednotlivých symbolů a jejich klasifikaci. Tato metoda o několika krocích má ale nevýhodu v akumulaci chyb během výpočtu jednotlivých kroků.



Obrázek 2: Diagram postupu sekvenčního řešení

Integrovaná řešení: generují množinu symbolických hypotéz a následně pomocí strukturní analýzy vyberou tu nejlepší na základě znalosti odpovídající gramatiky a sémantiky. Tento přístup je složitější a vyžaduje více výpočetních prostředků, ale může poskytnout lepší výsledky v případě složitějších vzorců.

Zatímco předchozí dvě metody řešení problematiky rozpoznávání matematických rovnic byly v posledních dekáдах studovány důkladně, **end-to-end řešení** na principu **encoder-decoder** se začaly objevovat relativně nedávno. Tento přístup se snaží vyřešit problém rozpoznávání vzorců jako celek, bez rozdělení na jednotlivé části. Používá hluboké učení a neuronové sítě bez jakékoliv znalosti konkrétní domény pro tvorbu modelu – **data-driven** přístup. Enkodér převede vstup na latentní reprezentaci a dekodér extrahuje skrytou reprezentaci za pomoci attention mechanismu a generuje výstupní sekvenci.

4 Popis vlastního řešení

4.1 Experimenty

4.2 Výsledky ~ Evaluace (Kvantitativní vs. kvalitativní)

5 Závěr ~ Takeaway messages

6 Závěr

Tato dokumentace popisuje strukturu a implementaci projektu KNN-Math. Projekt je navržen modulárně, což umožňuje snadnou údržbu a rozšiřování. Implementovaný systém používá algoritmus k-nejbližších sousedů pro rozpoznávání matematických symbolů a převod do LaTeX formátu.

7 Kuba CHATGPT

Tak víc než toto nemám :/

Jistě, zde je shrnutí průběhu experimentů a výsledků:

První pokusy (malý batch, limit dat, základní augmentace), , Batch size: 16, Epochs: 5, Limit: 1000 vzorků, Výsledek: Train accuracy: až 1.0 (100%), Train loss: 0.4592, Val accuracy: 0.0355 (3.55%), Val loss: 5.1749, , Závěr: Model se naučil trénovací data nazpaměť, ale na validaci selhával (overfitting).,

Zvětšení batch, více dat, zvýšený dropout, regularizace, , Batch size: 32–128, Dropout: 0.7 → 0.5, Weight decay: 0.01–0.02, Label smoothing: 0.1–0.3, Learning rate: 1e-4 → 1e-5 (příliš nízký nefungoval), Výsledek: Train accuracy: 0.3141 (31.41%) po 4 epochách (batch 128), Val accuracy: 0.0178 (1.78%), Train loss: 1.1588, Val loss: 7.5405, , Závěr: Overfitting se snížil, ale validace stále velmi slabá.,

Oprava resize a paddingu, fixní výška 80, šířka 256, , Vstupní obrázky: [3, 80, 256], Batch size: 128, Epochs: 5 (test), plánováno 67–100 (celonoční běh), Výsledek: Epoch 1: Train loss: 1.0333, Train accuracy: 0.3446, Val loss: 6.2991, Val accuracy: 0.0000, , Epoch 2: Train loss: 0.8438, Train accuracy: 0.4296, Val loss: 6.7513, Val accuracy: 0.0178, , , Závěr: Model se učí, trénovací přesnost roste, validace je stále velmi nízká, ale už není nulová.,

Technické úpravy Opraven resize s poměrem stran, Padding na fixní šířku, Opraveno stackování batchů, Opraven import a použití F.pad

Zdroje

- [1] ZHELEZNIAKOV, Dmytro; ZAYTSEV, Viktor; RADYVONENKO, Olga. Online Handwritten Mathematical Expression Recognition and Applications: A Survey. *IEEE Access*. 2021, roč. 9, s. 38352–38373. Dostupné z DOI: [10.1109/ACCESS.2021.3063413](https://doi.org/10.1109/ACCESS.2021.3063413).
- [2] LI, Zhe; JIN, Lianwen; LAI, Songxuan; ZHU, Yecheng. Improving Attention-Based Handwritten Mathematical Expression Recognition with Scale Augmentation and Drop Attention. In: *2020 17th International Conference on Frontiers in Handwriting Recognition (ICFHR)*. 2020, s. 175–180. Dostupné z DOI: [10.1109/ICFHR2020.2020.00041](https://doi.org/10.1109/ICFHR2020.2020.00041).