

Meta-prompty pro optimalizaci promptů velkého jazykového modelu

Vedoucí práce: Ing. Jan Drchal Ph.D.

Vojtěch Klouda

Obhajoba bakalářské práce
18.6.2025



Motivace optimalizace promptů

- ▶ Komplexní úlohy \iff komplexní prompty
- ▶ Manuální ladění promptů vyžaduje čas a expertní znalosti
- ▶ Podobné prompty \neq podobné účinky
- ▶ Data jsou potřeba i k manuálnímu ladění
- ▶ Méně náročné než fine-tuning



Diagram použité metody

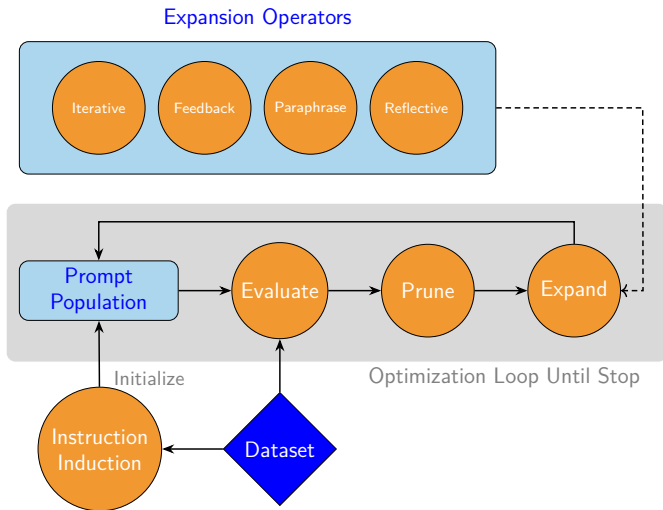


Figure: Diagram použité optimalizační metody



Operátory pro generaci promptů

Instruction Induction

Data: Příklady úloh z datasetu

Meta-prompt: Najdi vhodnou instrukci pro řešení úlohy

Iterative

Data: Prompty+skóre, vzestupně seřazené

Meta-prompt: Vymysli prompt, který pokračuje v trendu

Reflective

Data: Špatný prompt a nevydařený pokus

Meta-prompt: Odhal chybu a oprav prompt

Paraphrase

Data: Prompt

Meta-prompt: Parafrázuj prompt



Self-supervised optimalizace

Data: Prompt a historie jeho porovnání s ostatními prompty pomocí LLM

Feedback

Meta-prompt: Najdi rozdíly mezi dobrými a špatnými prompty a udělej nový

Pro každý pár promptů:

Párová porovnávací metrika

- ▶ Najdi úkoly, pro které oba prompty mají výsledky
- ▶ LLM porovná výsledky a prompty - vynese odůvodněný verdikt
- ▶ Verdikt se uloží do seznamu porovnání, využije ho Feedback operátor



Testovací úlohy

Connections

- ▶ Kategorizace slov na základě netriviálních charakteristik
- ▶ Input: puff, domino, curl, placebo, ball, butterfly, halo, doodle
- ▶ Gold: ball, curl, doodle, puff; butterfly, domino, halo, placebo

CodeContests

- ▶ Řešení úloh kompetitivního programování

Posloupnosti

- ▶ Vlastní úloha - hledání dalšího čísla v posloupnosti
- ▶ Input: 0 3 22 9 44 15 66 21 88 27 110 33 132
- ▶ Output: 39



Hlavní experiment

- ▶ úlohy s objektivním řešením
- ▶ porovnání 4 operátorů
- ▶ 3 operátory používají metriky úloh
- ▶ 1 pracuje s porovnáváním výstupů pomocí LLM
- ▶ referenční řešení - rekonstrukce instrukcí z příkladů
- ▶ každý experiment zopakován třikrát



STEP	REFLECTIVE	ITERATIVE	FEEDBACK	PARAPHRASE
HB	0.70	0.70	0.70	0.70
SB	0.40	0.40	0.40	0.40
1	0.47 (0.23)	0.63 (0.36)	0.43 (0.29)	0.73 (0.38)
2	0.53 (0.39)	0.53 (0.32)	0.47 (0.32)	0.60 (0.33)
3	0.27 (0.15)	0.67 (0.43)	0.47 (0.33)	0.63 (0.37)
4	0.47 (0.21)	0.63 (0.42)	0.47 (0.32)	0.60 (0.42)
5	0.57 (0.33)	0.63 (0.40)	0.40 (0.28)	0.60 (0.38)
6	0.53 (0.35)	0.57 (0.34)	0.27 (0.19)	0.80 (0.45)
7	0.57 (0.44)	0.60 (0.31)	0.37 (0.23)	0.73 (0.37)
8	0.63 (0.46)	0.70 (0.50)	0.37 (0.25)	0.57 (0.29)
9	0.50 (0.40)	0.80 (0.37)	0.43 (0.27)	0.63 (0.44)
10	0.57 (0.41)	0.57 (0.39)	0.33 (0.21)	0.60 (0.35)

Figure: Výsledky jednotlivých operátorů na úloze hledání dalšího čísla v celočíselné posloupnosti. Hard Baseline (HB) - nejlepší z 50 promptů z Instruction Induction operátoru, Soft Baseline (SB) - nejlepší z promptů inicializační populace.



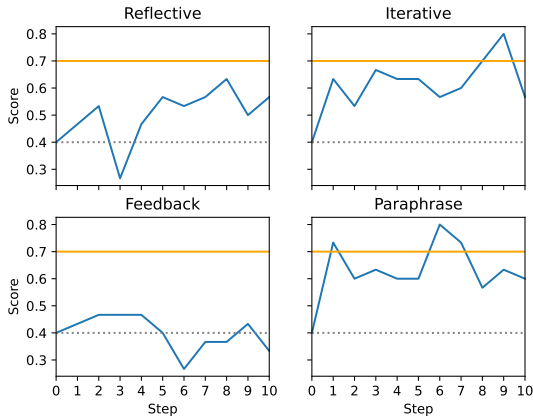


Figure: Výsledky jednotlivých operátorů na úloze hledání dalšího čísla v celočíselné posloupnosti. Hard Baseline (Oranžová) - nejlepší z 50 promptů z Instruction Induction operátoru, Soft Baseline (Šedá, čárkovaná) - nejlepší z promptů inicializační populace.



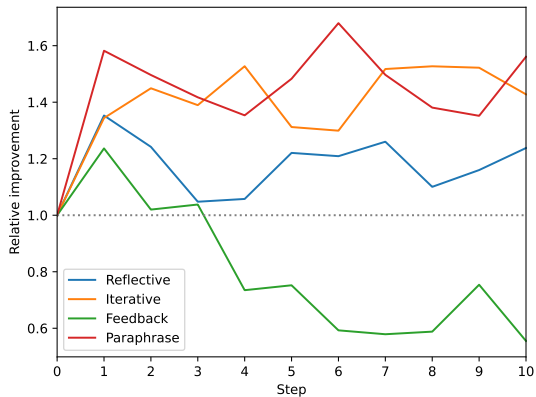


Figure: Relativní zlepšení vzhledem k inicializačním promptům (Soft Baseline).



Kreativní úlohy



(a) Initial



(b) Step 1



(c) Step 2



(d) Step 3

Figure: Využití operátoru Feedback pro kreativní úlohu - generace promptu pro difuzní obrázový model pro daná přídavná jména: “wistful”, “hopeful” and “lonely”.



- ▶ rozsáhlý přehled literatury
- ▶ zasazení do kontextu jiných metod škálování
- ▶ modulární optimalizační metoda promptů
- ▶ 5 operátorů včetně self-supervised operátoru Feedback
- ▶ framework pro strukturovanou generaci s inference-time metodami
- ▶ porovnání operátorů na různých úlohách oproti silnému referenčnímu řešení



Otázka

Do jaké míry lze navržené řešení použít obecně v praxi, např. pro úlohu převodu PDF obsahu na strukturovaný JSON výstup?

Odpověď

- ▶ Prompt lze optimalizovat pro jakoukoliv úlohu, kde je dobře definovaná metrika a jsou dostupná data, případně lze použít LLM v hodnotící roli.
- ▶ Čím komplexnější úkol, tím složitější a delší prompt a tím větší hledací prostor.
- ▶ Vedoucí práce používá optimalizaci promptů v úlohách NLP, jako například rozpoznávání entit v novinových článcích.



Otázka

Jakým způsobem je zajištěno, že párové hodnocení promptů a jejich výsledků nevede ke zhroucení prohledávání pouze na úzkou oblast prostoru? Jsou nějaké zkušenosti s nasazením v obecných praktických úlohách?

Odpověď

- ▶ Problém diverzity byl velkým tématem pro všechny operátory, nejen pro operátor Feedback, který staví na párovém porovnávání.
- ▶ Každé volání operátoru Feedback používá náhodně vybraný prompt, jehož historie porovnáváním je pak využita ke generaci nového promptu.



- ▶ Operátor Feedback měl suveréně nejhorší výsledky a docházelo ke kolapsu vyhledávání.
- ▶ Nejdůležitější mi přijde diverzita u inicializační populace a u Instruction Induction přidávám seed v podobě Persony.
- ▶ Obecně jsem se tento problém snažil řešit prořezáváním duplikátů v populaci na základě editační vzdálenosti a dále v rámci návrhu meta-promptů.
- ▶ Článek, kterým jsem se při návrhu metody inspiroval, ji testuje například na MT-Bench, což je soubor úkolů, testující užitečnost odpovědí LLM při vícekových konverzacích. Často se také k testování používají například reálná řešení z Githubu.



Děkuji za pozornost

