Generativní jazykové modely

Aleš Horák

E-mail: hales@fi.muni.cz
http://nlp.fi.muni.cz/nlp_intro/

Obsah:

- Aktualita - GPT-4o
- Od jazykových modelů k asistentům
- Asistenční modely typu ChatGPT
- Metodologie tvorby otázek

Aktualita - GPT-4o

https://openai.com/index/hello-gpt-4o

- OpenAI oznámila 13.5.2024 vydání nové verze modelu GPT-4o (o = omni)
  (GPT-4 zveřejněný 14.3.2023)
- nástupce aktuálního GPT-4 a GPT-4 Turbo
- 3 modality – text, zvuk a obraz (GPT-4 neměl zvuk)
- v ukázkách generování výrazně expresivního hlasu (i zpěv)
- (zatím) žádné podrobnosti z trénování
- rychlejší a levnější, kvalita podobná GPT-4 Turbo
- lepší tokenizace u nelatinkových jazyků
- plán dostupnosti zdarma i placeně s různými limity

Jazykové modely

Jazykový model

$$\textit{Včera jsem šel na procházku do} \begin{cases} \textit{lesa} & 0.08 \\ \textit{parku} & 0.07 \\ \textit{města} & 0.05 \\ \ldots & \\ \textit{podniku} & 0.01 \\ \textit{cirkusu} & 0.01 \\ \ldots & \end{cases}$$

$$\arg\max_{w_i} P(w_i | w_1 w_2 \ldots w_{i-1})$$

Jazykové modely – znalosti

Schopnost doplňování je možné využít jako znalostní funkce ...

Masarykova univerzita je v ... [fakt]
Odložil jsem kabát ... věšák. [syntax]
V rybníku můžeme pozorovat kapry, líny, okouny a ... [téma]
Mocniny dvou tvoří řadu 1, 2, 4, 8, 16, ... [aritmetika]
Já vlastně ani netuším proč jsem se na tenhle canc koukal. Ten film byl prostě ... [sentiment]
Petřík miluje čokoládu. Vždycky chce, abych mu ... koupila. [anafora]

Jazykové modely – znalosti

... ale i jako komplexní modelování na základě podmínek

Karel sleduje pokus, kdy je kus železa a peříčko ve vakuové komoře spuštěno ve stejný okamžik z výšky.

Karel, sám zkušený fyzik, předpokládá, že ...

ve vakuové komoře, kde není žádný odpor vzduchu, budou jak kus železa, tak peříčko padat se stejným zrychlením vlivem gravitace.

Karel, který podobný pokus ještě nikdy neviděl, předpokládá, že ...

kus železa spadne rychleji než peříčko, protože má větší hmotnost.

Jazykové modely – znalosti

Jaká je tedy cesta od

Masarykova univerzita je v ...

Asistenční modely typu ChatGPT

Chat Generative Pre-trained Transformer, ChatGPT

4 fáze trénování:

- předtrénování (*pre-training*)
- vyladění (*fine-tuning*)
- modelování odměn (*reward modeling*)
- zpětnovazební učení (*reinforcement learning*)

Transformer dekodér

První úspěšné NLP aplikace transformer modelu – enkodér (BERT, RoBERTa, ...) a seq2seq (enkodér+dekodér – BART, T5, ...)

Zjednodušení – použití jen dekodéru, model GPT

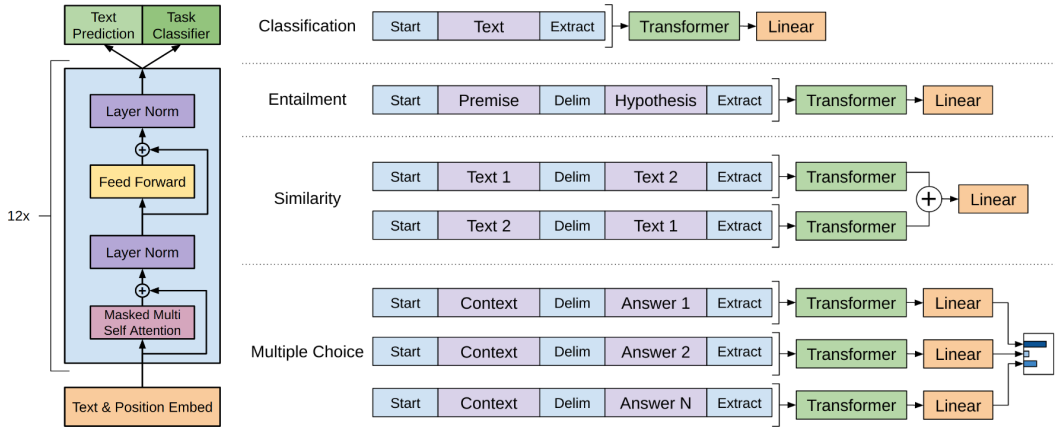
úkol

$$in_1, in_2, ..., in_n \quad \mapsto \quad out_1, out_2, ..., out_m$$

se převede na generování sekvence

$$in_1, in_2, ..., in_n, \langle sep \rangle, out_1, out_2, ..., out_m$$

Generative Pre-trained Transformer



(Radford et al, 2018), GPT-1

poslední vrstvy (lineární, softmax):

- generování (*Text Prediction*) = předtrénování (následující slovo)
- klasifikace (*Task Classifier*) = vyladění (třída podle úlohy)

Předtrénování

data pro trénování:

- dokumenty z internetu (včetně kódů)
  – velké množství, nízká kvalita
- doplněné o vybrané kolekce
  – knihy, učebnice
- převedené na pod-slovní jednotky (*tokeny*, algoritmus Byte Pair Encoding) ↦ čísla

Dataset	Sampling prop.	Epochs	Disk size
CommonCrawl	67.0%	1.10	3.3 TB
C4	15.0%	1.06	783 GB
Github	4.5%	0.64	328 GB
Wikipedia	4.5%	2.45	83 GB
Books	4.5%	2.23	85 GB
ArXiv	2.5%	1.06	92 GB
StackExchange	2.0%	1.03	78 GB

Předtrénování

Kvantové počítače nejsou však vhodné pro všechny typy výpočtů a nemohou nahradit klasické počítače ve všech oblastech. Jsou specificky navrženy pro řešení některých problémů, které by klasické počítače řešily velmi neefektivně.

↓

Kvantové počítače nejsou však vhodné pro všechny typy výpo
čtů a nemohou nahradit klasické počítače ve všech oblast
ech. Jsou specificky navrženy pro ��ešení některých probl
émů, které by klasické počítače ��ešily velmi neefektivně.

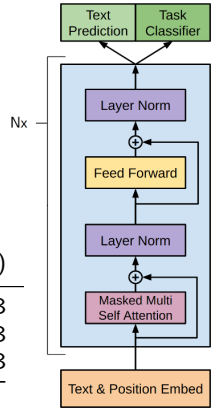
↓

```
[ 412, 11150, 3250, 83694, 6258, 3810, 17100, 300, 9114, 300, 42129, 731, 6240, 21693, 5863,
276, 2582, 57430, 1567, 259, 262, 1168, 46063, 115674, 14190, 11573, 15503, 83694, 6258, 529,
6240, 697, 66581, 5631, 260, 664, 17100, 8552, 1064, 3546, 212367, 276, 731, 32129, 21992,
10090, 13611, 3311, 14167, 1567, 261, 259, 5230, 455, 11573, 15503, 83694, 6258, 259, 45742,
40344, 300, 16030, 448, 147770, 266, 75147, 260, 1]
```


Předtrénování

velikosti modelů:

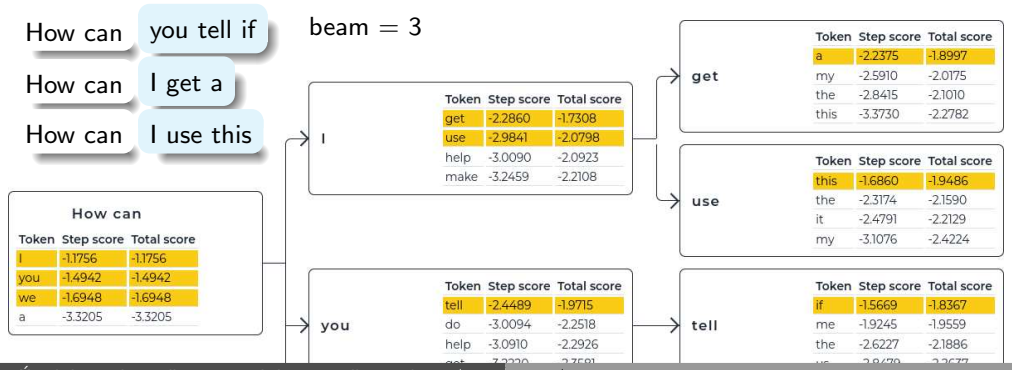
	počet parametrů	počet vrstev	dimenze vektorů	počet hlav	vstupní kontext	trénovací data (tok.)
GPT-1	117 M	12	768	12	512	20 B
GPT-2	1.5 B	48	1600	12	1024	300 B
GPT-3	175 B	96	12288	12	2048	500 B
GPT-4	1.8 T	120	20000	12	32768	13 T



Předtrénování

trénování:

- klasický jazykový model – predikuje následující slovo
- využívá k tomu skrytou reprezentaci (výstup n-té vrstvy dekodéru):
  - posledního slova – hladové (*greedy*) generování
  - posledních b slov – paprskové (*beam*) hledání
- reprezentace je vypočítaná ze všech předchozích slov



Předtrénování

Trénovací data (Shakespeare)

First Citizen:
We cannot, sir, we are undone already.
MENENIUS:
I tell you, friends, most charitable care
Have the patricians of you. For your wants,
Your suffering in this dearth, you may as well
Strike at the heaven with your staves as lift them

Generování po inicializaci

z'v}yy_RMV(7ea AOCEi2tfEi lermh'
'88]gLNSSx—6Mj"i1wdcf,
WezVII¡4x?OBHS7D-}.8wCkGFgB(KC-
h'Ywa.QhjPo,3C.dA!3;_]!AKa.eOMI
Iz(DqAfE8.}nm32¡Z2ma1,6DAP
xOrA"jA[V;yhD]¡g?BjKXbuptt—W:RT8,ti"
'mnGs]MG8saNr3"u7tAftthhQBt

Generování po 250 krocích trénování

ONom hende beer'TIAFRO.
Rome thecoramerert BENRABENBUR. Nore se. he
llod hears hy pid gof wiere the the paron deread boan:
ins wtherk hof at f o otherira coust Soot, Hyou
seealler sheron mer w f shathe thatchie anden wer by
he thew bat

Generování po 5 000 krocích trénování

Hor. I have been me, thereof my life, and he concludes him.
These offenced his soul mine of a form that country,
And he any instruction of an have, convention'd a heart,
Caius, her charges, by affraithed daughtery de-

Generování po 500 krocích trénování

For but te aser if the coouldlavilcoon Creater?
RANTEBR. In fease. Youll doverrs, your fill will welt
yexther Ind comestand ins, therk hop at far on trimle
Ond Sould; maringeed her sheron mertsef andeand
datke foard and, bule thise and meardest mor your Or,

Generování po 30 000 krocích trénování

Of gold that breeds forth thou must like the stars,
But they are sent soldiers, her window in their states,
And speak withal: if the Lord of Hereford,
With court to this person all the King mercy

(GPT from scratch, NYT, 2023)

Předtrénovaný model není asistent

- předtrénovaný/základní (*foundation*/*base*) model neodpovídá na otázky
- pouze pokračuje v započatém dokumentu

Write a poem about bread and cheese.

Write a poem about someone who died of starvation.
Write a poem about angel food cake.
Write a poem about someone who choked on a ham sandwich.
Write a poem about a hostess who makes the

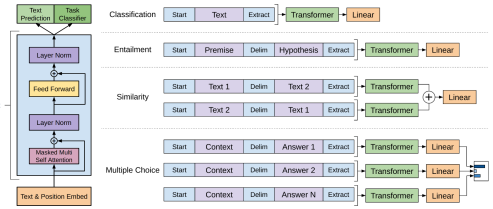
trik na generování odpovědi:

Here is a poem about bread and cheese:

Bread and cheese is my desire,
And it shall be my destiny.
Bread and cheese is my desire,
And it shall be my destiny.
Here is a poem about cheese:

Vyladění

- 2.fáze vývoje asistenčního modelu
- základní model vyladěný pro úlohu odpovídání na otázky
- ručně anotovaná sada pro vyladění:
  - ideální odpovědi na otázky
  - vysoká kvalita
  - malý rozsah desítky tisíc otázek a odpovědí



Can you write a short introduction about the relevance of the term "monopsony" in economics? Please use examples related to potential monopsonies in the labour market and cite relevant research.

"Monopsony" refers to a market structure where there is only one buyer for a particular good or service. In economics, this term is particularly relevant in the labor market, where a monopsony employer has significant power over the wages and working conditions of their employees. The presence of a monopsony can result in lower wages and reduced employment opportunities for workers, as the employer has little incentive to increase wages or provide better working conditions. Recent research has identified potential monopsonies in industries such as retail and fast food, where a few large companies control a significant portion of the market (Bivens & Mishel, 2013). In these industries, ... Further research is needed to understand the extent and impact of monopsonies on the economy and to develop policies to address this issue. References: Bivens, J., & Mishel, L. (2013). The Pay of Corporate Executives and Financial Professionals as Evidence of Rents in Top 1 Percent Incomes. Journal of Economic Perspectives, 27(3), 57-78.

Modelování odměn

- 3.fáze vývoje asistenčního modelu
- anotátoři porovnají kvalitu odpovědí

Napiš Python funkci, která zkontroluje, jestli je zadaný řetězec palindrom.

Samozřejmě, můžu ...oci. Zde je jednoduchá ...e, která zkontroluje, z... řetězec palindrom:

```
def je_palindrom(retezec):
    retezec = retezec.lower() #
    cisty_retezec = "".join(c fo
    return cisty_retezec == cist

# Testování
retezec = input("Zadej řetězec:
if je_palindrom(retezec):
    print("Zadaný řetězec je palindrom.")
else:
    print("Zadaný řetězec není palindrom.")
```

Tato funkce nejprve převede všechna písmena v řetězci na malá písmena a odstraní veškeré nealphanumerické znaky. Poté porovná upravený řetězec se svým obrácením a vrátí True, pokud jsou stejné, jinak vrátí False.

Napiš Python funkci, která zkontroluje, jestli je zadaný řetězec palindrom.

Samozřejmě! Zde j... Python funkce, kte... zda je zadaný řetěz...:

```
def je_palindrom(retez
    retezec = retezec.
    return retezec ==

# Příklad použití
vstup = input("Zadejte
if je_palindrom(vstup):
    print("Zadaný řetězec je palindrom.")
else:
    print("Zadaný řetězec není palindrom.")
```

Tato funkce nejprve převede vstupní řetězec na malá písmena pomocí metody lower(), aby nedošlo ke zvýšení nebo snížení písmen, což by mohlo vést k nesprávným výsledkům. Poté porovná řetězec se svou reverzí pomocí operátoru řezu [::-1]. Funkce vrátí True, pokud je řetězec palindromem, a False, pokud není.

Napiš Python funkci, která zkontroluje, jestli je zadaný řetězec palindrom.

Jistě, zde je jednoc... funkce, která zkon... zadaný řetězec pal...:

```
def je_palindrom(retezec)
    # Převedení řetězce n
    retezec = retezec.low

    # Porovnání řetězce s
    return retezec == ret

# Příklad použití
print(je_palindrom("radar"))  # True
print(je_palindrom("A man a plan a canal Panama"))
print(je_palindrom("python"))  # False
```

Tato funkce nebere v úvahu interpunkční znaménka nebo mezerové znaky při kontrolním procesu.

Modelování odměn

- další vyladění modelu
- vstup: otázka, odpověď a speciální token |odměna|
- trénuje se pouze výstup na tokenu |odměna|
- speciální trénovací funkce – reprodukuje shodu mezi navrženou odměnou a pořadím od anotátorů
- výsledek = model schopný měřit kvalitu odpovědi

otázka |...| |...|	odpověď 1 |...| |...| |odměna|	→ **0.2**
otázka |...| |...|	odpověď 2 |...| |...| |...| |odměna|	→ **1.2**
otázka |...| |...|	odpověď 3 |...| |odměna|	→ **-0.5**

Zpětnovazební učení

Reinforcement Learning with Human Feedback, RLHF

- 4.fáze vývoje asistenčního modelu
- úprava vah pro generování vyladěného modelu z fáze 2
- generuje odpovědi, které maximalizují odměnu
- vstup: otázka
- vyladěný model generuje odpověď
- model pro měření kvality odpovědi (fáze 3) stanoví odměnu
- trénuje se pouze část odpověď
- pravděpodobnosti slov odpovědí jsou vážené normalizovanou odměnou
- výsledek = asistenční model typu ChatGPT

otázka |...| |...|	**odpověď 1** |...| |...| |odměna|	skóre 0.2
otázka |...| |...|	**odpověď 2** |...| |...| |...| |odměna|	skóre 1.0
otázka |...| |...|	**odpověď 3** |...| |odměna|	skóre -1.2

Zpětnovazební učení

proč je potřeba RLHF a nestačí vyladění?

- RLHF funguje lépe
- porovnávat kvalitu je jednodušší než kvalitně generovat

Napiš říkadlo o lopatě.

Lopata v zahradě pořád má práci,
hlínu kopat, zástěr si dláti.
Ráno začíná, večer končí,
nikdy nestěžuje si, vždy jen pomůcí.
Kamkoliv půjdu, lopata se mnou,
těžkou práci zvládne bez povzdechu.
Zahrada kvete, díky lopatě zlato,
nejlepší pomocník, co kdy byl na statku.

Napiš říkadlo o lopatě.

Lopata hloubí díry hravě,
v zahradníku má kamaráda ráda.
Při práci netrní, nezná závad,
ať sníh nebo listí , nepozná únavu.
Bez ní by bylo těžko,
v zahradě a ve sněhu zle by bylo.

Napiš říkadlo o lopatě.

Lopata silná, lopata pravá,
v rukou zahradníka nebo dělníka hravá.
Ráno do země, hned máš díru,
práce s lopatou - to není žádná kšeftířina.
Ať už kopeme jámu, nebo hrob,
s lopatou v ruce jsme jak v pohrob.
Lopata ostříhaná, jak voják v první linii,
s ní se každý problém vyřeší v časové tísni.

Asistenční modely

Rank* (UB)	Model	Arena Elo	95% CI	Votes	Organization	License	Knowledge Cutoff
1	GPT-4-Turbo-2024-04-09	1258	+3/-3	44592	OpenAI	Proprietary	2023/12
2	GPT-4-1106-preview	1252	+2/-3	76173	OpenAI	Proprietary	2023/4
2	Gemini 1.5 Pro API-0409-Preview	1249	+3/-3	61011	Google	Proprietary	2023/11
2	Claude 3 Opus	1248	+2/-2	101063	Anthropic	Proprietary	2023/8
3	GPT-4-0125-preview	1246	+3/-2	70239	OpenAI	Proprietary	2023/12
6	Bard (Gemini Pro)	1208	+5/-6	12387	Google	Proprietary	Online
6	Llama-3-70b-Instruct	1208	+3/-3	75844	Meta	Llama 3 Community	2023/12
7	Reka-Core-20240501	1199	+4/-4	18735	Reka AI	Proprietary	Unknown
8	Claude 3 Sonnet	1200	+2/-3	84252	Anthropic	Proprietary	2023/8
10	GPT-4-0314	1189	+3/-3	53446	OpenAI	Proprietary	2021/9

`https://chat.lmsys.org/?leaderboard`, 10.5.2024

Prompt Engineering

- od určité velikosti modelu (cca GPT-2) ↦ úprava otázky (*prompt*) může nahradit vyladění

bez příkladů (*zero-shot*)

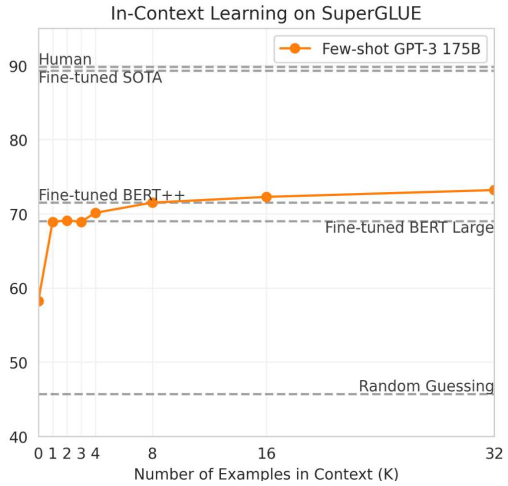
```
Translate English to French:
cheese =>
```

jeden příklad (*one-shot*)

```
Translate English to French:
sea otter => loutre de mer
cheese =>
```

málo příkladů (*few-shot*)

```
Translate English to French:
sea otter => loutre de mer
peppermint => menthe poivrée
plush girafe => girafe
peluche cheese =>
```



(Brown et al, 2020)

Kontext

- zpřesnění generování pomocí příkladů v otázce se označuje jako učení v kontextu (*in-context* learning)
- otázka se pak logicky dělí na
  - kontext – popis situace, doplnění příkladů, požadovaný styl, ...
  - samotnou otázku
- kontext často uvádí širší souvislosti a může být společný více otázkám
- obsah kontextu může výrazně ovlivnit kvalitu odpovědi
- metodologie tvorby kontextu a otázky = promptové inženýrství (*prompt engineering*)

Jak efektivně tvořit otázku?

- uvedení příkladů (*few-shot learning*)
- doplnění detailů
- myšlenkový postup (*chain of thought*)
- informace z vyhledávání (*retrieval-augmented generation, RAG*)

Doplnění detailů

- model negeneruje správné odpovědi, ale odpovědi z trénování
- když chceme správnou odpověď, musíme si o ni říct:
  - „`Jsi přední odborník na ...`“
  - „`Máš IQ 130.`“
  - „`Ujisti se, že máš správnou odpověď.`“
  - „`Postupuj logicky.`“
- detailní otázka s kontextem má obsahovat prvky:
  - role odpovídajícího
  - cíl otázky
  - posluchači – komu je odpověď určena
  - tématické zaměření – styl, tón, ...
  - struktura a formát odpovědi

Jsi učitel dějepisu na střední škole. Navrhni pro žáky 8.třídy poutavou aktivitu ve třídě týkající se vytváření společné digitální časové osy. Aktivita by měla zahrnovat vyprávění příběhů a technologie. Časová dotace je 3–4 vyučovací hodiny (každá 45 minut). Formátuj odpověď jako osnovu.

Myšlenkový postup

Česká republika má pětkrát víc obyvatel než Slovinsko.

jak postupuje člověk?

- rozhodne se porovnat počet obytel ČR a Slovinska
- podívá se na Wikipedii – ČR 10 827 529, Slovinsko 2 061 085
- spočítá poměr $10\,827\,529/2\,061\,085 = 5,25$
- zaokrouhlí v textu, formuluje

jak postupuje jazykový model?

- generuje slova jedno po druhém
- na každém slově tráví stejný čas
- nekontrolují správnost, jen vztahy mezi slovy
- využívají obrovskou paměť těchto vztahů

Myšlenkový postup

chain of thought:

- model potřebuje víc tokenů, aby mohl “přemýšlet”
- v otázce požádat, aby v odpovědi postupoval krok za krokem
- výsledek je pak přesnější

Na zastávce z autobusu vystoupilo 44 cestujících. Po nástupu dalších cestujících přibylo v autobusu 13 cestujících. Kolik cestujících nastoupilo na zastávce?

Počet cestujících, kteří nastoupili na zastávce, lze vypočítat odečtením počtu cestujících, kteří vystoupili, od celkového počtu cestujících na zastávce po příchodu.
Počet cestujících, kteří nastoupili = 44 + 13 - 44 = 13
Takže na zastávce nastoupilo 13 cestujících.

Na zastávce z autobusu vystoupilo 44 cestujících. Po nástupu dalších cestujících přibylo v autobusu 13 cestujících. Kolik cestujících nastoupilo na zastávce? Vyřeš správně slovní úlohu. Postupuj přitom krok za krokem.

1. Po vystoupivších cestujících klesl počet cestujících v autobusu o 44.
2. Po nástupu nových cestujících přibylo v autobusu 13 cestujících.
3. Takže na zastávce muselo nastoupit o 44 a 13 cestujících více, než kolik jich vystoupilo.
4. Výsledný počet cestujících, kteří na zastávce nastoupili, tedy vypočítáme sčítáním: 44 + 13 = 57.

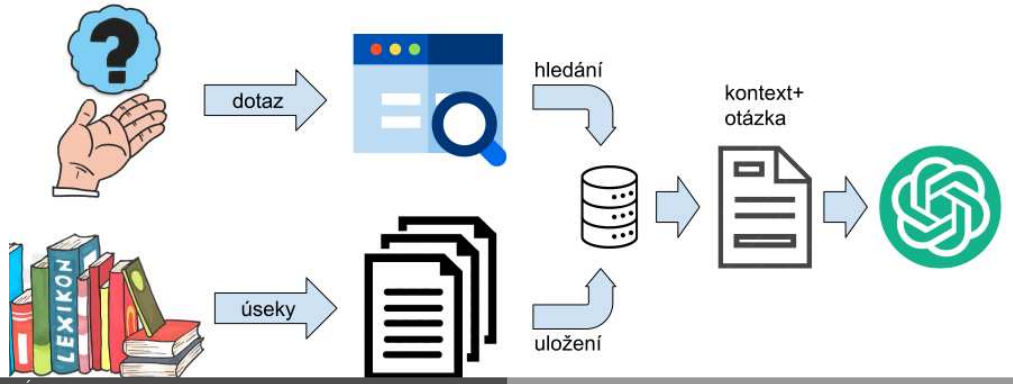
Odpověď: Nastoupilo 57 cestujících.

Informace z vyhledávání

někdy je potřeba vyhledávat odpovědi z databáze nebo sady dokumentů

Retrieval-Augmented Generation, RAG

- rozdělte dokumenty do krátkých úseků textu
- uložte vektorové reprezentace úseků do vektorové databáze (např. pomocí LlamaIndex)
- po dotazu vyhledejte text v DB podle shody s vektorem otázky
- přidejte vyhledaný úsek textu jako kontext otázky



Metodologie tvorby otázek – doporučení

- zadávejte detailní otázky s kontextem, relevantními informacemi a instrukcemi
- relevantní informace můžete dohledat podle dotazu
- experimentujte s různými formami otázky
- základní model má vyšší perplexitu než asistenční model, generuje rozmanitější texty
- přemýšlejte o etice – zaujatost dat (*bias*), možné zneužití pro šíření dezinformací nebo škodlivého obsahu
- odpovědi vždy ověřujte