Engineer

Велика мовна модель

💬 нейронна мережа, яка передбачає наступний токен у тексті

Що це таке

Велика мовна модель — це нейронна мережа, навчена на дуже великому обсязі тексту робити одну річ: за вже написаним текстом призначити ймовірність кожному токену, який міг би йти далі. Токен — це короткий шматок тексту: часто ціле слово, іноді частина слова або окремий розділовий знак. Відповідь пишеться тим самим кроком, повтореним у циклі. Програма вибирає токен відповідно до цих імовірностей, дописує його до тексту й питає знову, доки модель не видасть токен, що означає «стоп», або доки не буде досягнуто обмеження довжини. Профіль NIST для генеративного ШІ, опублікований у липні 2024 року, вміщує це в один рядок: великі мовні моделі передбачають наступний токен або слово в реченні. Абревіатура LLM, від англійського large language model, означає те саме, і саме цю форму вживає англійський текст профілю.

Архітектура, на якій побудовано моделі останніх років, називається Transformer; її описала стаття про машинний переклад 2017 року. Вона відмовилася від рекурентних і згорткових шарів, з яких складалися попередні моделі, і спиралася лише на увагу (attention) — механізм, що дає кожній позиції в тексті дивитися безпосередньо на кожну іншу. Через це навчання стало набагато легше розподіляти між багатьма процесорами, і модель зі статті навчалася значно швидше за попередниць.

«Велика» стосується кількості параметрів — ваг, яких мережа набуває під час навчання. Стаття 2020 року описала модель зі 175 мільярдами параметрів, удесятеро більше, ніж мала будь-яка попередня нерозріджена мовна модель, і показала, що за такого розміру модель може братися за нове завдання за кількома розв’язаними прикладами, вставленими у вхідний текст, без жодного додаткового навчання.

Модель, яку навчили лише продовжувати текст, саме це й робить: продовжує текст, а це не те саме, що виконати прохання. Стаття 2022 року описала крок, який перетворює таку модель на асистента. Люди написали відповіді, які хотіли б отримати на набір прохань, і модель донавчили на них; потім люди впорядкували власні відповіді моделі від кращої до гіршої, і її навчили далі на цих рейтингах. Цей метод називають навчанням із підкріпленням на основі відгуків людей. В оцінках зі статті люди віддавали перевагу відповідям моделі з 1,3 мільярда параметрів, навченої в такий спосіб, перед відповідями початкової моделі, що мала в сто разів більше параметрів.

Застосуйте поняття до рішення

Застосуйте «Що таке велика мовна модель?» до конкретного рішення. Ми можемо пов'язати поняття із системою, яку ви створюєте або купуєте. Агент, під'єднаний до систем, якими ви вже користуєтеся, перевірений на прикладах, які складають ваші експерти, під наглядом ваших людей і не прив'язаний до однієї моделі.

Переглянути наші послуги Зв'язатися з нами →

Перевірте припущення за терміном

Використайте пояснення та пов'язані роботи, щоб сформулювати запитання для проєкту. Ми можемо перевірити припущення разом до вибору інструмента. Побудувала механізм із 268 перевірок репозиторію — 268 перевірок коміту і 146 тестів самих перевірок

Побудувала механізм із 268 перевірок репозиторію Зв'язатися з нами →

Чому це важливо

Мовна модель береться за роботу, яку важко охопити наперед заданими правилами: розкласти довільний лист у правильну папку, витягти поля з відсканованої накладної, написати чернетку відповіді мовою клієнта, підсумувати довге листування. Роботу, для якої раніше людина мусила прочитати кожен документ, тепер може підготувати програма, а людина — перевірити.

Та сама властивість, що робить модель корисною, ускладнює побудову на ній. Її вихід зазвичай вибирається з розподілу ймовірностей навмання, тож той самий вхід може дати різні відповіді за різних запусків, і модель пише гладко незалежно від того, чи має рацію. NIST називає впевнено сформульований хибний результат конфабуляцією, а частіше кажуть «галюцинація», і описує його як природний наслідок того, як ці моделі побудовано: вони створюють текст, що відповідає статистиці тексту, на якому їх навчали, і такий текст часто точний, а іноді ні. NIST також зазначає, що модель може вигадати й міркування чи посилання, які нібито обґрунтовують

відповідь. Хибна відповідь виглядає точнісінько як правильна, тож єдиний спосіб дізнатися, як часто модель має рацію у вашій роботі, — виміряти це на вашій роботі.

Немає й однієї моделі, на якій можна зупинитися. Нові моделі виходять регулярно, дві з них можуть зовсім по-різному поводитися на тому самому вході, а коли зміниться модель, яку викликає ваша програма, вирішує її постачальник. Команда, що будує на мовній моделі, будує на частині, якої не контролює, і довговічна робота — це все навколо моделі: кейси, що її вимірюють, інструменти, які їй дозволено викликати, і правила того, що вона може робити сама.

Профіль NIST називає дванадцять сфер ризику для генеративного ШІ, серед них конфабуляцію, приватність даних, інформаційну безпеку та інтелектуальну власність. Його написано для організацій, що керують ризиками, а не для розробників, але це придатний контрольний список для команди, яка вирішує, чого може торкатися функція на основі моделі.

Як це виглядає на практиці

Функція на основі мовної моделі починається з написаних кейсів, а не з промпту. Кожен кейс — це справжній вхід і відповідь, яку прийняла б людина, що знає цю роботу; кейси зберігають у файлі під контролем версій поруч із кодом:

```
# cases/delivery-notes.yaml
- id: note-0142
  input: scans/note-0142.txt
  expect:
    supplier: "Example Freight"
    reference: "DN-58213"
    pallets: 6
- id: note-0157
  input: scans/note-0157.txt
  expect:
    supplier: "Sample Haulage"
    reference: "DN-59002"
    pallets: 14
```

Яка модель виконує який тип завдання — це налаштування, тож мала й швидка модель може вилучати дані, а більша — писати чернетки відповідей, що потребують судження:

```
# routing.yaml
jobs:
  extract-delivery-note:
    model: small-fast
    cases: cases/delivery-notes.yaml
  draft-claim-reply:
    model: large
    cases: cases/claim-replies.yaml
```

Кожна зміна, чи то новий промпт, новий інструмент або нова модель, проходить усі кейси, перш ніж потрапити в роботу, а результат — це оцінка для кожного кейсу, яку зберігають у часі, щоб виправлення одного кейсу не могло непомітно зламати інший. Коли виходить нова модель, спробувати її — це зміна одного рядка в `routing.yaml` і прогін кейсів, а рішення перемкнутися спирається на оцінки, а не на анонс.

Саме так цей сайт описує розробку ШІ-агентів для вашої галузі. Вибір моделі задається в конфігурації, а не в коді: мала й швидка модель — для сортування та вилучення даних, велика — для завдань, що потребують судження. Співпраця починається з того, що разом із людьми, які виконують роботу, записують від тридцяти до п’ятдесяти справжніх прикладів із правильними відповідями, і кожна зміна — промпту, інструмента чи моделі — проходить їх усі, перш ніж потрапити в роботу. Як сказано на сторінці, модель завжди можна замінити, а код, кейси й дані належать клієнтові, тож агент лишається його власністю навіть тоді, коли змінюється модель, на якій він працює.

Де виникають проблеми

Впевнені відповіді, які виявляються хибними. Модель, яку питають про пункт договору, якого перед нею немає, може все одно його описати тим самим упевненим тоном, яким дає правильні відповіді, і послатися на джерело, якого не існує. Дайте моделі документи, на основі яких вона має відповідати, попросіть посилатися на них і перевіряйте в коді, що кожне посилання вказує на справжній документ і фрагмент. Вимірюйте частку помилок на написаних кейсах, а відповіді з наслідками передавайте людині, перш ніж на їхній основі буде вчинено будь-яку дію.

Інструкції, сховані в даних. Програма, яка дає моделі читати електронні листи, вебсторінки чи завантажені файли, дає авторам цього тексту говорити з моделлю. Стаття 2023 року показала це на справжніх застосунках: інструкції, підкинуті у вміст, який модель, найімовірніше, прочитає, змушували її розкривати дані й змінювати поведінку застосунку, а профіль NIST зараховує непряму ін'єкцію промптів до ризиків безпеки генеративного ШІ. Вважайте все, що читає модель, недовіреними вхідними даними, перевіряйте дозволи в інструментах, а не в промпті, давайте інструментам лише права людини, від імені якої діє модель, і вимагайте схвалення людини для кожної дії, яку не можна скасувати.

Модель, що змінюється без вашого відома. Нова версія чи інший постачальник можуть відповісти на той самий вхід інакше, і відповідь, яка проходила минулого місяця, сьогодні не проходить, хоча ваш власний код не змінився. Вказуйте точну версію моделі скрізь, де постачальник це дозволяє, і проганяйте весь файл кейсів за кожної зміни, зокрема зміни моделі, перш ніж вона дійде до користувачів.

Вихід, який код не може розібрати. Модель, яку просять повернути JSON, повертає JSON і дружнє речення після нього, пропускає поле або записує число як рядок. Опишіть очікувану структуру в JSON Schema, перевіряйте кожну відповідь за нею і вважайте відповідь, що не пройшла перевірку, помилкою, яку треба повторити або передати людині, а не даними.

Персональні дані в промптах і журналах. Усе, що надсилають моделі на серверах постачальника, залишає ваші власні системи, а журнал промптів і відповідей зберігає персональні дані ще довго після того, як запит виконано. NIST додає, що моделі можуть повторювати чутливі дані зі своїх навчальних даних, і називає це запам'ятовуванням даних. Надсилайте лише ті поля, яких потребує завдання, зберігайте журнали промптів за тими самими правилами зберігання й доступу, що й дані в них, і прочитайте умови постачальника щодо зберігання та навчання, перш ніж до нього потраплять будь-які персональні дані.

Що почитати далі

- NIST AI 600-1: Generative Artificial Intelligence Profile — супровідний документ NIST до його рамки управління ризиками ШІ, липень 2024 року: дванадцять ризиків генеративного ШІ, серед них конфабуляція, і запропоновані дії щодо кожного.

- Attention Is All You Need — стаття 2017 року, що представила Transformer, основу сучасних мовних моделей.
- Language Models are Few-Shot Learners — стаття 2020 року про модель зі 175 мільярдами параметрів, яка береться за нові завдання за прикладами у вхідному тексті.
- Training language models to follow instructions with human feedback — донавчання на написаних відповідях і впорядкованих результатах: крок від передбачувача тексту до асистента.
- Not what you've signed up for: Compromising Real-World LLM-Integrated Applications with Indirect Prompt Injection — інструкції, підкинуті в дані, які читає модель, продемонстровані на справжніх застосунках.

Словник

Обирайте за фактичними умовами

Опишіть робочий процес, наявні інтерфейси та користувачів, яким потрібна зміна. Ми можемо визначити перевірюваний результат і спосіб передачі, який дасть вашій команді змогу підтримувати його.

Дивитися, скільки це коштує Зв'язатися з нами →

Зробіть термінологію практичною

Запитайте, як «Що таке велика мовна модель?» стосується вашої організації. Конкретного прикладу з поточної системи достатньо для початку розмови.

Переглянути наші послуги

welcome@engineer.company +45 50 10 61 84 Копенгаген, Данія

Візьміть цю сторінку із собою

Візитівка

Дбайливо зроблено з любов'ю — D.

CVR 45969703 · Tagensvej 105C, 3. th, 2200 København N

Авторське право © 2025 – дотепер · Engineer ApS · Усі права захищено.

Надруковано з https://engineer.company/uk/glossary/large-language-model/