Creative Intelligence: Applications of Large Language Models in Data Generation and Reasoning

Zhongsheng Wang

School of Computer Science
The University of Auckland

zwan516@aucklanduni.ac.nz

December 19, 2024

School of
Computer Science

Profile

Zhongsheng WANG

王钟声

Education:

- 2025 Spring -: PhD in CS, University of Auckland, Supervisor: Prof. Jiamou Liu & Dr. Qian Liu
- 2022 - 2024: MSc in DS with First-Class Honours, University of Auckland, Supervisor: Prof. Jiamou Liu
- 2019 - 2023: BSc in CS, Southwest University, Supervisor: Prof. Qian Li

Working Experience:

- 2024.03 - 2024.07: AI Engineer, Atom-Intelligence Group
- 2023.11 - 2024.01: AI Scientist, HouGarden Ltd.
- 2023.07 - 2024.11: Graduate Teaching Assistant, UoA

Research Interest:

- Large Language Model Application
- LLM-Based AI Agent
- Natural Language Processing & Reasoning

Personal Website:

- `https://wzs010429.github.io`

Intro of Natural Language Processing (NLP)

Introduce your self.

Hello! I'm ChatGPT, your AI assistant powered by OpenAI. I'm here to help with anything from answering questions and brainstorming ideas to solving problems and learning new concepts. I can assist you with technical topics, creative projects, research, and more.

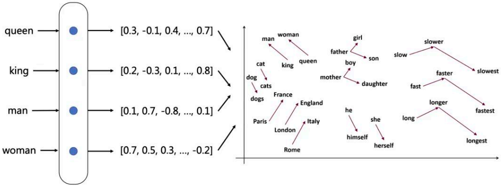
Hello! I'm ChatGPT, your AI assistant powered by (DeepMind ✗ / Meta ✗ / OpenAI ✓).

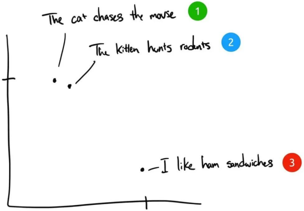
Based on the existing sentence to **predict** the following words.

(Input + Prompt Engineering) $\rightarrow$ *LLM* $\rightarrow$ Ouput

Word Embedding

Represent text semantics as a high-dimensional vector.





- The cat chases the mouse
- The kitten(小猫) hunts rodents(啮齿动物)
- I like ham sandwiches

Evaluating an embedding model: semantically similar texts are closer in vector space (Cosine Similarity, Euclidean Distance, Manhattan Distance, etc.)

Fuzzy Definition of Natural Language Processing (NLP)

- NLP: Model language as a mathematical object and implement language tasks through algorithms
- Given a sequence $x = (x_1, x_2, \ldots, x_n)$, the goal of language modeling is to find its probability distribution

$$
\begin{aligned}
P(x) &= P(x_1, x_2, \ldots, x_n) \\
&= \prod_{i=1}^{n} P(x_i \mid x_1, x_2, \ldots, x_{i-1})
\end{aligned}
$$

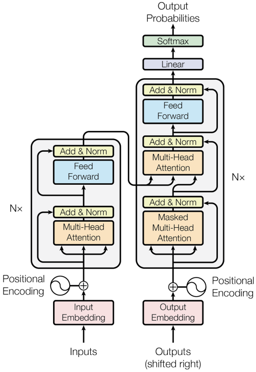
- Conditional probability at each step $P(x_i \mid x_{<i})$ represents the probability of occurrence of x_i in the previous context $x_{<i}$
- NLP Tasks: **Prediction** tasks in different type of outputs $\hat{y}$.
  - Generation: Predict subsequent seq $\hat{y}$ based on given content x
  - Classification: Predict a discrete label $\hat{y}$ (binary, 0 or 1) based on given content x (Sentiment Prediction)

Large Language Model (LLM)

Transformer-based LMs with billions or even hundreds of billions of parameters ($N > 10^9$)

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V$$

Q, K, V are query, key and value vectors, $\sqrt{d_k}$: scaling factor

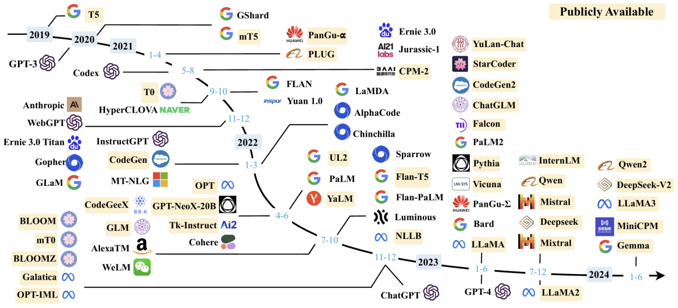


Advantages

- Strong generalization ability: Good results on a variety of tasks without additional fine-tuning
- Contextual understanding: Capture contextual relationships in long texts and generate coherent content
- Versatility: One model can simultaneously complete tasks such as translation, QA, and writing

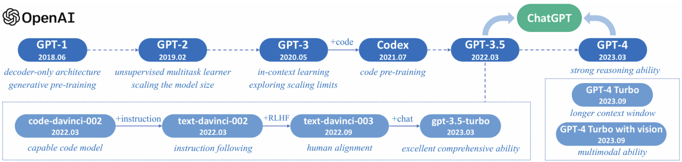
"Attention Is All You Need", `https://arxiv.org/abs/1706.03762`

History of LLM Development



A timeline of existing LLMs (from 2019 to 2024.02).

Yellow color: Open-sourced LLMs.



A brief illustration of the technical evolution of GPT-series models (from 2018 to 2023.09).

"A Survey of Large Language Models", `https://arxiv.org/abs/2303.18223`

LLM Pre-training & Fine-tuning

Pre-training

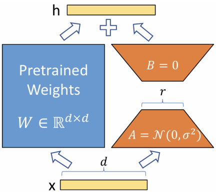
- LLMs learn general language representation and knowledge on massive data through unsupervised learning (High-quality data from any source)

Fine-tuning

- **Based on the pre-trained model**, model is further trained using task-specific labeled data to optimize the model's performance on the task.

Aspect	**Pre-Training**	**Fine-Tuning**
Basic Model	Mathematical architectures	Pre-trained models
Data	Unlabeled, massive datasets	Labeled, task-specific datasets
Objective	General language patterns	Task-specific patterns
Parameter Update	All parameters	Targeted adjustments (some may freeze)
Application Scope	General performance ability	Specific task performance

LLM Fine-tuning Optimization



Difficulty:

- Model parameters are getting larger and larger (13B, 70B, 80B), requiring more computing resources (GPU A100*8, rental: ¥4,000/Week, buy: ¥1,000,000)

Solution: **P**arameter-**E**fficient **F**ine-**T**uning (PEFT)

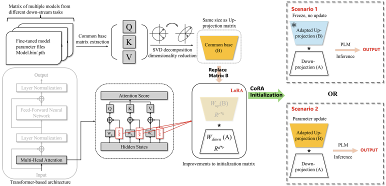
- Fine-tune only a small subset of the model's parameters, rather than the entire model
- **LoRA:** Add a low-rank matrix next to the original weight matrix to reduce the number of trainable parameters (Reduce costs, slightly increase time)

$$\Delta W = BA$$

$$h = (W_0 + BA)x$$

- ΔW: original parm. of LLM
- A: $r \times k$, dim. reduction matrix
- B: $d \times r$, dim. up matrix

CoRA

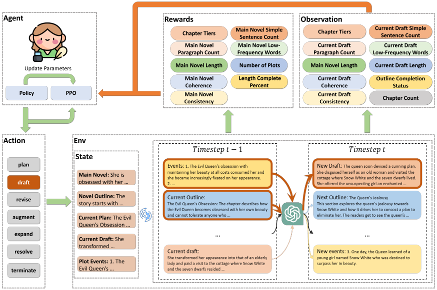


- Extracting common matrix space from the same model fine-tuned on different tasks
- Replace matrix B with the extracted values and freeze it

"LoRA: Low-Rank Adaptation of Large Language Models", https://arxiv.org/abs/2106.09685

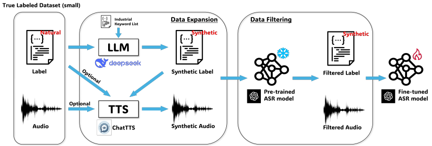
X. Xiao, Z. Wang et al., "CoRA: Optimizing Low-Rank Adaptation with Common Subspace of Large Language Models", Under review NAACL 2024

LLM for Data Generation



- Combination of LLM and RL
- LLM (gpt-3.5-turbo) as planner to generate epic-length novel text (beyond context length)
- Guarantee of novel coherence and logic
- Create a simulation training method for LLM and database interaction to avoid large API overhead

- Generate high-quality, diverse text using prompt engineering
- Use TTS(Text to Speech) model to generate audio data with different accents
- Use this data to fine-tune the ASR model and deploy it in industrial applications

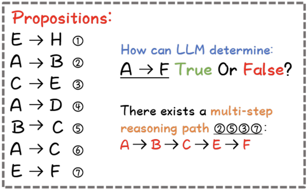


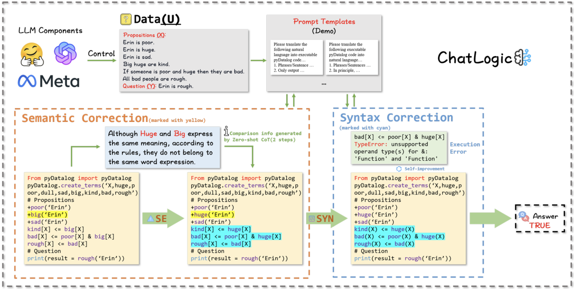
Q. Qi, L. Ni, Z. Wang et al., "Epic-Level Text Generation with LLM through Auto-prompted Reinforcement Learning", IJCNN 2024
Z. Wang et al., "Weak Supervision Techniques towards Enhanced ASR Models in Industry-level CRM Systems", ICONIP 2024

LLM for Reasoning

Multi-step Reasoning

- Difficult to retrieve multi-hop context with LLMs
- LLMs will miss details if too much context

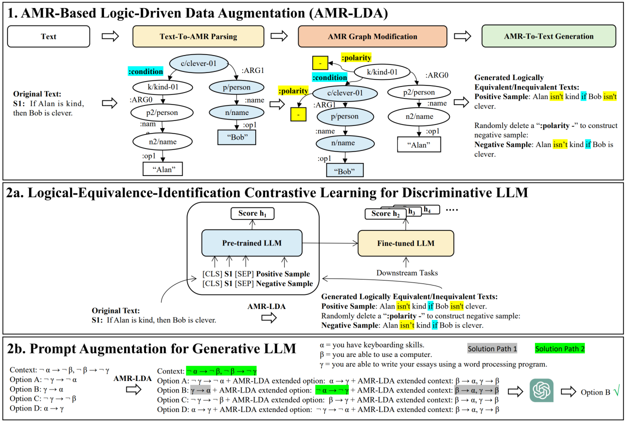


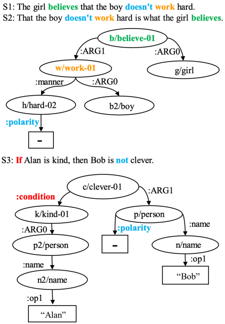


- LLMs generate Python code in a specified format based on the reasoning question
- Code semantic/syntax verification
- Code execution results as reasoning results

Z. Wang et al., "ChatLogic: Integrating Logic Programming with Large Language Models for Multi-Step Reasoning", IJCNN 2024

LLM for Reasoning (Data Augmentation)





- Propose AMR-LDA method to transform and augment logical reasoning datasets using Abstract Meaning Representation (AMR) graphs
- Through contrastive learning for logical equivalence recognition and cue augmentation, AMR-LDA improves the performance of LLMs on logical reasoning tasks

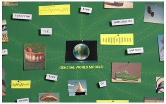
Q. Bao et al., "Abstract Meaning Representation Based Logic-Driven Data Augmentation for Logical Reasoning", ACL 2024

World Model

- From LeCun, A WM must have:
  - Observation
  - Agent environment perception
  - Action proposal
  - latent variable proposal (basis of generalization)

- In simple, the core of WM is to build a simplified but sufficiently accurate **environment simulator** through learning.
  - State Representation: $s_t = f(o_t)$
  - Dynamic Model: $s_{t+1} = g(s_t, a_t)$
  - Reward Function: $R(s_t, a_t) = \mathbb{E}[r_t | s_t, a_t]$

Dogs like to eat snacks; Snack shops; Chicken scraps dropped by passers-by



WM Reification

Yann LeCun @ylecun

Lots of confusion about what a world model is. Here is my definition:

Given:
- an observation x(t)
- a previous estimate of the state of the world s(t)
- an action proposal a(t)
- a latent variable proposal z(t)

A world model computes:
- representation: h(t) = Enc(x(t))
- prediction: s(t+1) = Pred(h(t), s(t), z(t), a(t))
Where
- Enc() is an encoder (a trainable deterministic function, e.g. a neural net)
- Pred() is a hidden state predictor (also a trainable deterministic function).
- the latent variable z(t) represents the unknown information that would allow us to predict exactly what happens. It must be sampled from a distribution or or varied over a set. It parameterizes the set (or distribution) of plausible predictions.

The trick is to train the entire thing from observation triplets (x(t),a(t),x(t+1)) while preventing the Encoder from collapsing to a trivial solution on which it ignores the input.

Auto-regressive generative models (such as LLMs) are a simplified special case in which
1. the Encoder is the identity function: h(t) = x(t),
2. the state is a window of past inputs
3. there is no action variable a(t)
4. x(t) is discrete
5. the Predictor computes a distribution over outcomes for x(t+1) and uses the latent z(t) to select one value from that distribution.
The equations reduce to:
s(t) = [x(t),x(t-1),...x(t-k)]
x(t+1) = Pred(s(t), z(t), a(t))
There is no collapse issue in that case.

Idea: Construction of World Model and Model Checking

Below is an example of a natural language multi-step reasoning question answering (QA) dataset. (Not a formal language)

Example Question

Context: In jurisdictions where use of headlights is optional when visibility is good, drivers who use headlights at all times are less likely to be involved in a collision than are drivers who use headlights only when visibility is poor. Yet Highway Safety Department records show that making use of headlights mandatory at all times does nothing to reduce the overall number of collisions.

Question: Which one of the following, if true, most helps to resolve the apparent discrepancy in the information above?

Options:
A. In jurisdictions where use of headlights is optional when visibility is good, one driver in four uses headlights for daytime driving in good weather.
B. Only very careful drivers use headlights when their use is not legally required.
C. The jurisdictions where use of headlights is mandatory at all times are those where daytime visibility is frequently poor.
D. A law making use of headlights mandatory at all times is not especially difficult to enforce.

Answer: B.

An example of the ReClor dataset.

W. Yu et al., "ReClor: A Reading Comprehension Dataset Requiring Logical Reasoning", ICLR 2020

Inference Analysis in Natural Language

Para. and Option B

Context: In jurisdictions where use of headlights is optional when visibility is good, drivers who use headlights at all times are less likely to be involved in a collision than are drivers who use headlights only when visibility is poor. Yet Highway Safety Department records show that making use of headlights mandatory at all times does nothing to reduce the overall number of collisions. (在能见度好时可选择是否使用前灯的管辖区，始终使用前灯的司机发生碰撞的可能性比仅在能见度差时使用前灯的司机要小。然而，公路安全部门的记录显示，强制始终使用前灯并不能减少总体碰撞次数)

Question: Which one of the following, if true, most helps to resolve the apparent discrepancy in the information above? (以下哪一项如果为真，最有助于解决上述信息中明显的差异？)

Option B: Only very careful drivers use headlights when their use is not legally required. (只有非常小心的司机才会在法律不要求使用前灯时使用前灯)

- Option B can explain the Context highlighted in yellow
- Introducing a "mandatory use of headlights" scenario: Many less cautious drivers will also start using their headlights
- The reason for the conclusion that "the number of accidents overall has not decreased" is that careless drivers will not become more careful simply because they are forced to use headlights
- Inference result: Even if all drivers were forced to use headlights, it would not reduce the overall number of accidents

CTL-like Representation

All the CTL-like representations below are generated by GPT-4o

- In jurisdictions where use of headlights is optional when visibility is good, drivers who use headlights at all times are less likely to be involved in a collision than are drivers who use headlights only when visibility is poor.
  - $\mathcal{AG}(Visibility_{good} \land Headlight \land \neg Mandatory \rightarrow CollisionRateLow)$
  - $\mathcal{AG}(Visibility_{bad} \land Headlight \land \neg Mandatory \rightarrow \neg CollisionRateLow)$
  - intuition: $\mathcal{AG} Headlight \rightarrow CollisionRateLow$
- Yet Highway Safety Department records show that making use of headlights mandatory at all times does nothing to reduce the overall number of collisions.
  - $\mathcal{AG}(Headlight \rightarrow Collision) \land \mathcal{AG}(\neg Headlight \rightarrow Collision)$
  - intuition: $\mathcal{AG}(\neg CarefulDriver \land Headlight \land Mandatory \rightarrow \neg CollisionRateLow)$
- **B.** Only very careful drivers use headlights when their use is not legally required.
  - $\mathcal{AG}(CarefulDriver \land Headlight \land \neg Mandatory \rightarrow CollisionRateLow)$

Idea: Construction of World Model and Model Checking

Extract necessary elements from natural language content, following the building blocks of the world model. (Ignore potential perceptions, action proposals)

Expanding propositional logic to first-order logic & temporal logic.

- **Option B**: Only very careful drivers use headlights when their use is not legally required.

- $D(x)$: x is a very careful driver
- $U(x)$: x used headlights.
- R: Using headlights is required by law.

FOL: $\Rightarrow$ $\quad \forall x(R \to U(x)) \to D(x)$
LTL: $\Rightarrow$ $\quad \Box(\neg R \to (U \to D))$

Task Details:

- Accurately extract **all available information** $I = \{E, A, R\}$ from natural-language text $T = \{w_1, w_2, \ldots, w_n\}$
- Generate formalized logical expressions using semantic extension with $I = \{E, A, R\}$.
- Expand the knowledge space using logical reasoning tools or LLMs (Under Model Checking)
- Convert the results of reasoning back into natural language expressions to achieve a closed loop from knowledge extraction to reasoning

Demo

Knowledge Extraction

Given text T, we extract:

- Entities: $E = \{\text{Alice}, \text{CompanyX}\}$
- Relationships: $R = \{\text{works\_for(Alice, CompanyX)}\}$

Mathematical Representation:

$$E, R = f_{\text{entity/relation}}(T)$$

Semantic Expansion and Logical Expression Construction

Expand knowledge semantically and construct logical expressions.

- Expanded Relationship:

$$(\text{CompanyX}, \text{employs}, \text{Alice})$$

- Logical Expression Construction:

$$\phi_1 = \text{works\_for}(\text{Alice}, \text{CompanyX})$$

$$\phi_2 = \forall x, \text{works\_for}(x, \text{CompanyX}) \Rightarrow \text{has\_income}(x)$$

Logical Reasoning

Use logical rules to infer new knowledge.

- Given:

$$\phi_1 = \text{works\_for}(\text{Alice}, \text{CompanyX})$$

$$\phi_2 = \forall x, \text{works\_for}(x, \text{CompanyX}) \Rightarrow \text{has\_income}(x)$$

- Inference:

$$\phi_1 \wedge \phi_2 \models \text{has\_income}(\text{Alice})$$

Cont.

Result Generation
Transform logical inference results into natural language.

- Inferred result: "Alice has an income because she works for CompanyX."

Natural language conclusion will be combined with the option information as inference text to improve the accuracy of the answer.

Dificulties:

- LLMs infer the accuracy of entity relationships at the **semantic level** without ground truth (Open-world Assumption).
  Example: Puppies like to go to pet supermarkets. (missing puppies like eating snacks)
- Using LLM to generate some necessary supporting intermediate materials
- Ready-to-use logical reasoning tools/programs/models for correct and stable knowledge extension (model checking)

Thanks for listening!

Zhongsheng Wang
zwan516@aucklanduni.ac.nz

WeChat:

UNIVERSITY OF AUCKLAND
Waipapa Taumata Rau
NEW ZEALAND

School of Computer Science