

அத்தியாயம் 6: மாபெரும் மொழி மாதிரிகள் (LLMs) – ஒரு புதிய பிரக்ஞையின் உதயம்

டிரான்ஸ்ஃபார்மர் என்ற புரட்சிகரமான இயந்திரம் பிறந்துவிட்டது. ஆனால், ஒரு தனி இயந்திரம், ஒரு சகாப்தத்தை உருவாக்குவதில்லை. அந்த இயந்திரத்தின் பில்லியன் கணக்கான பிரதிகளை ஒன்றிணைத்து, ஒரு மாபெரும் நரம்பியல் வலையமைப்பை உருவாக்கும்போதுதான், அதன் உண்மையான சக்தி, அதன் பிரம்மாண்டம் வெளிப்பட்டது.

அந்தப் பிரம்மாண்டத்தின் பெயர்தான் மாபெரும் மொழி மாதிரிகள் (Large Language Models - LLMs).

இது, ஒரு தனி நியூரான்-ஐ உருவாக்குவதிலிருந்து, ஒரு முழுமையான மூளையை உருவாக்குவதைப் போன்ற ஒரு பாய்ச்சல். டிரான்ஸ்ஃபார்மரின் இதயமாக இருந்த "கவனம்" (Attention) என்ற தத்துவத்தை அடிப்படையாகக் கொண்டு, GPT மற்றும் BERT போன்ற டைட்டன்ஸ் பிறந்தன.

அவை, மொழியை வெறும் விதிகளின் தொகுப்பாகப் பார்க்கவில்லை. மாறாக, மனிதனைப் போலவே, பில்லியன் கணக்கான பக்கங்களைப் படித்து, அனுபவத்தின் மூலம் மொழியின் ஆன்மாவைக் கற்றுக்கொண்டன. இதன் விளைவாக, அவை மொழியைப் புரிந்துகொள்வது மட்டுமல்லாமல், நுணுக்கமாகப் பகுப்பாய்வு செய்யவும், மனிதனைப் போலவே புதிய உரைகளை உருவாக்கவும் தொடங்கின.

வாருங்கள், இந்த புதிய டிஜிட்டல் மனங்களின் உருவாக்கத்தை, அவை எப்படிப் பயிற்சி பெறுகின்றன, எவ்வாறு தங்களுக்குள் ஒரு பிரக்ஞையை வளர்த்துக் கொள்கின்றன, மற்றும் அவற்றின் பயன்பாடுகள் என்னென்ன என்பதை இந்த அத்தியாயத்தில் விரிவாக ஆய்வு செய்வோம்.

6.1. கடைக்கால் மாதிரிகள் (Foundation Models)

Foundation Models என்பது, பல்வேறு செயற்கை நுண்ணறிவு (AI) பயன்பாடுகளுக்கு அடித்தளமாக விளங்கும் சக்திவாய்ந்த பொதுவான (general-purpose) மாதிரிகள். இவை ஒரு குறிப்பிட்ட பணிக்காக உருவாக்கப்படாதிருந்தாலும், அதற்கு fine-tune செய்யக்கூடியவை. பொதுவாக, இவை மிகப்பெரிய தரவுத்தொகுப்புகளில் முன்னிலைப் பயிற்சி (Pretraining) பெறும், பின்னர் domain-specific fine-tuning மூலம் தேவைக்கேற்ப தனிப்பயனாக்கப்படுகின்றன.

Foundation Models-ஐ உருவாக்கும் செயல்முறை:

1. பெரிய அளவிலான தரவுகள் (Large-scale Data): இணையம், புத்தகங்கள், கட்டுரைகள், விகிதப்பீடியா போன்றவை.
2. மிகுந்த கணிப்பொறி வளம் (Compute Power): பெரும் அளவிலான GPU/TPU கிளஸ்டர்கள் பயன்படுத்தப்படுகின்றன.
3. அளவுருக்கள் (Parameters): மாடலின் அளவு பில்லியன் அளவுருக்களாக இருக்கலாம் (e.g., GPT-3 – 175B).
4. விரிவான கட்டமைப்புகள் (Architectures): Transformers மற்றும் attention-based deep neural networks.

Foundation Models உருவாக்கம் ஏன் முக்கியம்?

ஒவ்வொரு தனித்தனிப் பணிக்கும் (மொழிபெயர்ப்பு, உரை சுருக்கம்) ஒரு பிரத்யேக AI-ஐ உருவாக்குவதற்குப் பதிலாக, அனைத்தையும் செய்யக்கூடிய ஒரு பொதுவான, சக்திவாய்ந்த மூளையை உருவாக்கினால் என்ன?

இந்த யோசனையிலிருந்து பிறந்தவைதான் அடித்தள மாதிரிகள் (Foundation Models).

இவை பலதரப்பட்ட பணிகளை ஒரே மாதிரி செய்யக்கூடியவை. உரை மொழிபெயர்ப்பு, உரை சுருக்கம், கேள்வி-பதில் அமைப்பு, chatbots, sentiment analysis போன்ற பல NLP பணிகளில் பயன்படுத்தப்படுகின்றன. ஒரே மாடலை, தேவைப்படும் துறைக்கேற்ப fine-tune செய்வதன் மூலம் விரைவில் பயிற்சி செய்து பயன்படுத்த முடிகிறது.

உதாரணம்: GPT-3 போன்ற Foundation Model-ஐ எடுத்துக் கொள்ளுங்கள். இது ஒரே மாதிரி, கேள்விகளுக்கு பதில் அளிக்கும் bot ஆகவும், கட்டுரைகள் எழுதும் உதவியாளராகவும், Python நிரல்களை எழுதும் கோடிங் அசிஸ்டன்டாகவும் செயல்பட முடிகிறது.

இந்த அடித்தள மாதிரிகளை, உலகின் அனைத்துப் பாடங்களையும் கற்றுத் தேர்ந்த ஒரு பல்துறை மேதையாக (Polymath) கற்பனை செய்யுங்கள். அந்த மேதையிடம், "நீங்கள் ஒரு மருத்துவராக வேண்டும்" என்று கூறி, மருத்துவ நுணுக்கங்களை மட்டும் கற்றுக்கொடுத்தால் போதும். குறுகிய காலத்திலேயே, அவர் அந்தத் துறையின் தலைசிறந்த நிபுணராக உருவெடுப்பார்.

இதுவே, ஒரு அடித்தள மாதிரியை நமது தேவைக்கேற்பச் செதுக்கிக்கொள்ளும் (Fine-tuning) கலையாகும்.

Foundation Model உருவாக்கம் பொதுவாக மூன்று முக்கிய அம்சங்களைக் கொண்டு மதிப்பீடு செய்யப்படுகிறது:

1. பயிற்சித் தரவுகள் (Training Data): தரவின் தரமும் பரப்பும்.
2. மாதிரி கட்டமைப்பு (Architecture): Transformer layers, attention heads, hidden sizes போன்றவைகள்.
3. மனிதம் சார்ந்த ஒழுங்குபடுத்தல் (Post-training / Human Feedback): RLHF, human annotation, filter-based scoring போன்றவைகள்.

6.2. முன்னிலைப் பயிற்சி (Pretraining)

மாபெரும் மொழி மாதிரிகளை (LLMs) உருவாக்குவதில் முதற்கட்டமாகச் நடைபெறும் செயல்முறைதான் முன்னிலைப் பயிற்சி அல்லது Pretraining. இது எந்த ஒரு குறிப்பிட்ட பணிக்காக அல்ல, பொதுவான மொழி அறிவைப் பெறுவதற்காக நடைபெறும்.

இது மனிதனுக்கு பள்ளிக்குச் செல்லும் முன் கற்றுக் கொள்ளும் அடிப்படை அறிவைப் போன்றது. எழுத்து, சொற்கள், வாக்கிய அமைப்புகள், புணர்ச்சிப் பொருள் போன்றவற்றைச் சராசரியாகப் புரிந்து கொள்வது போல, LLM-களும் மொழியின் இலக்கணம் மற்றும் சொற்கள் இடையிலான உறவுகளை இந்த கட்டத்தில் கற்றுக்கொள்கின்றன.

6.2.1 பயிற்சி தரவுகளின் தன்மை

Pretraining நடக்கும் தரவுகள் பொதுவாக மிகப்பெரிய அளவிலான, தன்னிச்சையான மற்றும் வெவ்வேறு துறைகளிலிருந்து பெறப்பட்ட உள்ளடக்கங்களை கொண்டிருக்கும். இதில் பொதுவாக பின்வரும் தரவுத்தொகுப்புகள் அடங்கும்:

- Wikipedia: தகவல் மற்றும் பொதுவான அறிவைப் பெற உதவும்.
- BooksCorpus: இலக்கியம் மற்றும் கதைகளை உள்ளடக்கியது.
- Common Crawl: இணையத்தில் உள்ள பல்லாயிரக்கணக்கான இணையதளங்களில் இருந்து சேகரிக்கப்பட்ட திறந்த தரவுகள்.
- OpenWebText: Reddit போன்ற சமூக வலைத்தளங்களில் பரிந்துரைக்கப்படும் தரமான இணையப்பக்கங்கள்.
- Project Gutenberg: புத்தகங்கள், சிறுகதைகள் போன்ற மரபு இலக்கியத் தொகுப்புகள்.

இந்த தரவுகள் அனைத்தும், மாடலுக்கு பன்மொழி, பல்துறை மற்றும் பல்லாயிரக்கணக்கான மொழி நடைகளைப் பற்றிய புரிதலை வழங்குகின்றன. சில தரவுகள் பிழைகள் கொண்டிருந்தாலும், அவை பெரிய அளவில் படிக்கப்படுவதால், மாடலுக்கு ஒரு பரந்த மொழி அறிவைப் கட்டியெழுப்ப உதவுகின்றன.

6.2.2 பயிற்சியின் விதிகள் – Masking மற்றும் Autoregression

Pretraining பல்வேறு பயிற்சி நுட்பங்களைக் கொண்டுள்ளது. அதில் முக்கியமானவை இரண்டு:

1. Masked Language Modeling (MLM)

இது BERT போன்ற Encoder-based மாதிரிகளால் பயன்படுத்தப்படும் முறை.

- ஒரு வாக்கியத்தில் சில சொற்கள் மறைக்கப்பட்டு (masked) விடப்படும்.
- மாடல், அந்த இடத்தில் எந்த சொல் வரும் என கணிக்கக் கற்றுக்கொள்ளும்.
- இது மாடலுக்கு வாக்கியத்தின் முழு சூழ்நிலையை (context) புரிந்துகொள்ள உதவுகிறது.

உதாரணம்:

“தமிழ் ஒரு மிகப் பழமையான [MASK] ஆகும்.”

மாடல், அதன் முன்னும் பின்னும் உள்ள சொற்களை பார்த்து, ‘மொழி’ என்பதை எடுக்கும்.

2. Causal Language Modeling (Autoregressive)

இது GPT மாதிரிகளின் பயிற்சி முறை. இதில், ஒரு சொல் வருவதற்கு முன் வந்த சொற்கள் மட்டுமே பார்க்க அனுமதிக்கப்படுகிறது.

- மாடல், வாக்கியத்தை இடதிலிருந்து வலது நோக்கி படிப்பதுபோல் பயிற்சி பெறுகிறது.
- ஒவ்வொரு கட்டத்திலும், பிந்தைய சொல் என்னவாக இருக்கலாம் எனக் கணிக்கின்றது.

உதாரணம்:

“தமிழ் ஒரு மிகப் பழமையான ...” → அடுத்த சொல்லை ‘மொழி’ என கணிக்க முயற்சி செய்கிறது.

இந்த இரண்டு பயிற்சி முறைகளும் மாடலுக்கு:

- சொற்களின் இடமாற்றங்களை (word associations)
- வாக்கிய கட்டமைப்பை (sentence structure)
- சூழ்நிலைப் பொருத்தத்தை (semantic context)

புரிந்து கொள்ளும் திறனை அளிக்கின்றன.

6.2.3 Pretraining எதற்காக?

Pretraining என்பது, ஒரு LLM-ஐ மனித மொழியில் பொதுவான அறிவு மற்றும் நுண்ணறிவு தொடர்புகள் கொண்டது போல் உருவாக்குவதற்காகவே.

- இது domain-specific அறிவு இல்லாமல் இருந்தாலும், மொழியின் இயல்பை (linguistic intuition) கற்றுக்கொள்கிறது.
- பரந்த அளவிலான உரை தரவுகளை வாசிப்பதன் மூலம், synonyms, grammar rules, idiomatic expressions போன்றவை பற்றி புரிந்து கொள்கிறது.
- இது zero-shot அல்லது few-shot learning-க்கு வழிவகுக்கும் – அதாவது, ஒரு புதிய வேலைக்காக மிகக் குறைந்த data-விலேயே பயன்படுத்தக்கூடிய அளவுக்கு general knowledge உருவாகிறது.

6.2.4 திறன்கள் மற்றும் வரம்புகள்

Pretraining மாடலுக்கு பல திறன்களைத் தருகின்றது, ஆனால் சில வரம்புகளும் உள்ளன.

திறன்கள்:

- பொதுவான உரை புரிதல் மற்றும் பதிலளித்தல்
- வினா-விலக்குகள், சுருக்கம், மொழிபெயர்ப்பு, creative writing
- பன்மொழி புரிதல் (Multilingual understanding), குறிப்பாக multilingual datasets-ஐ பயன்படுத்தினால்

வரம்புகள்:

- மாடலுக்கு task-specific விளக்கம் அல்லது domain-specific knowledge இல்லை

- Data Bias – பயிற்சி தரவுகள் இருந்த இடைச்செருக்குகள், சமூக முன்னேற்பாடுகள் அல்லது தவறான தகவல்களால் மாடல் பயிற்சி பெறலாம்
- hallucination – உண்மையில்லாத, ஆனால் நம்பத்தக்கதாகத் தோன்றும் தகவல்களை உருவாக்கும் சிக்கல்

6.2.5 பயிற்சி செலவுகள் மற்றும் கணிப்பொறி ஆதாரம்

Pretraining என்பது மிகப் பெரிய கணினி வளங்களை (compute resources) தேவைப்படுத்தும். இது பொதுவாக Google TPU, NVIDIA A100 GPU போன்ற மென்மையான hardware-ஐ நாடும்.

- GPT-3 போன்ற மாடலை பயிற்சி செய்ய நூற்றுக்கணக்கான GPU-கள், வெகுநாட்கள் தேவைப்படும்.
- Pretraining செலவு மில்லியன் டாலர் வரையில் இருக்கலாம் (பண்புத்தன்மை, தரவளவு, iteration எண்ணிக்கை போன்றவற்றைப் பொருத்து).

இந்த Pretraining கட்டத்தில், ஒரு LLM-க்கு ஒரு பொதுவான மொழி அறிவுத் தளமுறை உருவாக்கப்படுகிறது. இது, இத்துடன் Fine-tuning மற்றும் Post-training ஆகிய செயல்முறைகள் வந்தால், அந்த மாடல் ஒரு சிறந்த தொழில்நுட்ப உதவியாளராக, அல்லது நிபுணத் துறையில் பேசக்கூடிய language model-ஆக மாறும்.

6.3. நுண்-பயிற்சி (Fine-Tuning) – விரிவான விளக்கம்

மிகப்பெரிய தரவுத்தொகுப்புகளை வைத்து Pretraining செய்யப்பட்ட ஒரு LLM, மனித மொழியை ஒரு அளவிற்கு பொதுவாகப் புரிந்துகொள்ளும். ஆனால், நாம் அதை ஒரு குறிப்பிட்ட செயற்கை நுண்ணறிவு பணிக்கோ, ஒரு தொழில்நுட்பம் சார்ந்த சூழலுக்கோ, அல்லது ஒரு துறைமுகமாக செயல்படுத்த விரும்பினால், நாம் அதை Fine-Tune செய்யவேண்டும்.

இந்த Fine-Tuning என்பது, மாடலை ஒரு வழக்கமான பொதுவான அறிவு உள்ள நபர் என எடுத்துக்கொள்வதுபோல். அந்த நபரை நீங்கள் மருத்துவராக்க விரும்பினால், அவருக்கு மருத்துவப் பாடங்கள் சொல்லித் தர வேண்டும். அதேபோல், LLM-ஐ மருத்துவ நோக்கத்துக்குப் பயன்படுத்த, அதை medical datasets மூலம் fine-tune செய்ய வேண்டும்.

வாழ்க்கைசார்ந்த உதாரணம்:

ஒரு சிறந்த சமையல்கலை நிபுணரை நினைத்துப் பாருங்கள். அவர் இந்தியா முழுக்க பயணம் செய்து பல்வேறு வகையான உணவுகளை சுவைத்திருக்கிறார் (Pretraining). ஆனால், நீங்கள் அவரை ஒரே நிகர் மட்டமான ஹைதராபாத் பிரியாணி சமையல்காரராகக் உருவாக்க விரும்பினால், அவர் ஹைதராபாத் சமையல் வழிமுறைகளை மட்டும் பயில வேண்டும் (Fine-

Tuning).

உதாரணம் 1: மருத்துவம்

Pretrained LLM, “புற்றுநோய் என்றால் என்ன?” என்ற கேள்விக்கு பொதுவான பதிலை வழங்கும்.

பொதுவான பதில்:

“புற்றுநோய் என்பது ஒரு உயிரணுக்களின் கட்டுப்பாடற்ற பெருக்கம், இது உடலின் பல பகுதிகளில் பாதிப்புகளை ஏற்படுத்தும்.”

ஆனால் நீங்கள் அதே LLM-ஐ, PubMed கட்டுரைகள், நோயாளி விவரங்கள், மருந்து தொடர்பான குறிப்புகள், மருத்துவ வழிகாட்டிகள் போன்றவை மூலம் Fine-Tune செய்தால், அது மேலும் விபரமாகவும், நிலைத்த பதில்களையும் வழங்கும்:

Fine-Tuned பதில்:

“புற்றுநோய் என்பது ஒரு neoplastic condition ஆகும், இதில் abnormal cells G1/S phase-ல் cell cycle checkpoints-ஐ மீறி பெருகுகின்றன. அதனாலேயே metastasis ஏற்படுகிறது.”

இது ஒரு நோயாளிக்கு மருந்து பரிந்துரை செய்யும் Clinical Decision Support System (CDSS)-இல் நேரடியாக பயன்படுத்தலாம்.

உதாரணம் 2: சட்டத் துறை (Legal AI)

நீங்கள் ஒரு LLM-ஐ சட்டம் தொடர்பான வேலைக்குப் பயன்படுத்த விரும்பினால், அதை Indian Penal Code (IPC), Supreme Court judgments, contracts, legal memos போன்றவற்றுடன் fine-tune செய்யலாம்.

- Fine-Tuning இல்லாமல்: “ஒரு ஒப்பந்தம் என்ன?” என்றால், பொதுவான பதில்தான் வரும்.
- Fine-Tuned LLM: Section 10, Indian Contract Act 1872-ஐ மேற்கோள்களுடன் விவரிக்கும்.

“ஒரு ஒப்பந்தம் சட்டரீதியாகச் செல்லத்தக்கதா என்பதை Article 299(1) of the Constitution மற்றும் Section 23, 24 of the Indian Contract Act இன் அடிப்படையில் மதிப்பீடு செய்ய வேண்டும்” என்று பதிலளிக்க முடியும்.

உதாரணம் 3: வணிகம் மற்றும் வாடிக்கையாளர் சேவை

நீங்கள் ஒரு SaaS நிறுவனத்திற்கு FAQ bot உருவாக்க விரும்புகிறீர்கள் என்றால், அதற்காக அந்த நிறுவனம் வைத்திருக்கும்:

- வாடிக்கையாளர் support chat logs
- Troubleshooting articles
- Onboarding guides

போன்றவற்றை fine-tune செய்யலாம்.

Generic LLM பதில்: “இந்த மென்பொருளைப் பயன்படுத்த நீங்கள் login செய்ய வேண்டும்.”

Fine-Tuned LLM பதில்: “Click on ‘My Account’ in the top-right corner, then choose ‘Manage Subscription’ to view your active usage limits and billing plan.”

இதுபோன்ற deep product awareness பதில்கள், only fine-tuning மூலமே பெறக்கூடியவை.

Fine-Tuning வகைகள் மற்றும் modern techniques

FINE-TUNING முறை	விளக்கம்	நன்மை
Full Fine-Tuning	முழு மாடலையும் மீண்டும் train செய்வது	பயிற்சிக் கட்டுப்பாடு அதிகம்
Adapter Layers	மாடலுக்கு இடையே சிறிய layers சேர்த்து, அவை மட்டுமே update செய்யும்	மெமரி இடம் குறைவு
LoRA (Low Rank Adaptation)	Matrix decomposition மூலம் fine-tune	resource-efficient
Prompt Tuning	Prompt vectors-ஐ மட்டுமே update செய்வது	மெகா LLM-களுக்கேற்ப சிறந்தது

Bias handling during Fine-Tuning

Fine-Tuning தரவுகள் தவறாக curate செய்யப்பட்டிருந்தால், அது bias-ஐ உருவாக்கலாம்.

உதாரணம்:

ஒரு customer support LLM-ஐ fine-tune செய்யும் போது, training dataset-இல் அனைத்தும் *angry customers* ஆக இருந்தால், அதுவே மாடலின் சுயநிலையை உருவாக்கிவிடும் – பயனர் எதையாவது கேட்டால், அது “மன்னிக்கவும், இது உங்கள் தவறே!” என்று பதிலளிக்கும் அபாயம் உள்ளது.

அதனால், fine-tuning dataset-ஐ தொடர்ந்து validate செய்வது முக்கியம்.

Fine-Tuning with Human-in-the-Loop (HITL)

இன்று அதிகமான AI நிறுவனங்கள் Fine-Tuning-ஐ மனித பார்வையுடன் இணைத்துப் பணியாற்றுகின்றன:

- மனித annotators, model-generated response-ஐ score செய்கின்றனர் (better/worse)
- இவை reward function ஆகும்
- Reinforcement Learning with Human Feedback (RLHF) மூலமாக policy gradients update செய்யப்படும்

GPT-4, ChatGPT போன்றவை இதைத்தான் பயன்படுத்துகின்றன.

இவ்வறு, Fine-Tuning என்பது ஒரு LLM-ஐ முழுமையான பொதுத்திறமையிலிருந்து, நுட்பமான பணிக்குச் சீரமைக்கும் முக்கிய அடிக்கோலாகும். அது இல்லாமல், LLM பொதுவான, ஆனால் மேற்பரப்பான பதில்களை மட்டுமே வழங்கும்.

மிகவும் சரி. இப்போது நீங்கள் கேட்டதுபோல், Post-training மற்றும் Human Feedback பற்றி மேலும் ஆழமாகவும், நூல்வகைத் தரம் கொண்ட தருணச்சிந்தனையுடனும் விரிவாக்குகிறேன். இவை LLM-களின் மனிதர்களுடன் ஏற்கத்தக்க முறையில் செயல்படும் திறனை உருவாக்குவதற்கான அறியமான கட்டமாகும்.

6.4. பிந்தைய பயிற்சி (Post-training) மற்றும் மனிதப் பின்னூட்டம் (Human Feedback)

பெரிய மொழி மாதிரிகள் (LLMs) இயற்கை மொழியைப் புரிந்துகொள்வதிலும், தரவுகளில் இருந்து பொதுவான மற்றும் துறையறிந்த அறிவைப் பெற்றுக்கொள்வதிலும் திறமை பெறுகின்றன. ஆனால், இவை மனிதர்களின் எதிர்பார்ப்பு, நாகரிக கட்டுப்பாடுகள், மதிப்பீடு மற்றும் ஒழுங்குமுறைகள் போன்றவற்றுக்கு ஒத்திசைவாக செயல்பட வேண்டுமானால், இன்னும் ஒரு மேலதிக பயிற்சி கட்டம் தேவைப்படுகிறது. அதுவே Post-training எனப்படும்.

இந்த Post-training கட்டத்தில், LLM-கள் தொடர்ந்து மொழியை நிர்வாகிக்கின்றன. ஆனால் அவை இப்போது மனிதர்களுடன் உரையாடல்களில் ஏற்படக்கூடிய நுண்ணிய சூழ்நிலைகளை புரிந்துகொள்வதற்கும், அதன் அடிப்படையில் பதில்களை உருவாக்குவதற்கும் முயற்சிக்கின்றன.

6.4.1 Post-training: நோக்கம் மற்றும் தேவை

ஒரு pretrained மற்றும் fine-tuned LLM, ஒரு கேள்விக்குத் தரமான பதிலை அளிக்கத் தயாராக இருக்கலாம். இருப்பினும், அவற்றின் பதில்கள்:

- பயனாளியின் உணர்வுகளை புறக்கணிக்கக்கூடியது,
- சமூக நெறிமுறைகளுக்கு எதிரானதாக இருக்கக்கூடியது,
- தவறான/பொதுவாகச் சங்கடமுறையாக்கும் தகவல்களை வழங்கக்கூடியது

எனும் சிக்கல்களை ஏற்படுத்தலாம்.

இதனைத் தவிர்க்க, LLM-களில் மனிதர்களின் மதிப்பீடு மற்றும் விருப்பங்களை அடிப்படையாகக் கொண்டு உருவாக்கப்படும் “alignment” என்பது ஒரு முக்கிய அம்சமாக செயல்படுகிறது. Post-training என்பது alignment-ஐ மேம்படுத்தும் பயிற்சியாகும்.

6.4.2 Reinforcement Learning with Human Feedback (RLHF)

Post-training ல் பயன்படுத்தப்படும் முக்கிய நுட்பம், மனிதப் பின்னூட்டத்துடன் கூடிய பலப்படுத்தும் கற்றல் (Reinforcement Learning with Human Feedback - RLHF) ஆகும். இது மூன்று அடுக்குகளில் செயல்படுகிறது:

1. மேற்பார்வையுடன் கூடிய நுட்ப பயிற்சி (Supervised Fine-Tuning - SFT)

முன்கூறிய human-like response-கள் கொண்ட தரவுகளை கொண்டு, மாடல் ஒரு conversation அல்லது task-க்கு ஏற்ற பதில்கள் அளிக்க கற்றுக்கொள்கிறது.

உதாரணமாக, StackExchange, Reddit போன்ற இணைய சமூகங்களில் இருந்து தொலைக்காட்சி உரையாடல்கள் மற்றும் உத்தரவு-பதில்கள் (instruction-response pairs) எடுக்கப்படும்.

2. விருப்பம் அடிப்படையிலான மதிப்பீட்டு மாதிரி (Reward Model Training)

மாடல் இரண்டு அல்லது மூன்று பதில்களை வழங்கும். பின்னர், மனித மதிப்பீட்டாளர்கள் (human annotators) அவற்றில் எது சிறந்தது என்று தேர்வு செய்கின்றனர்.

இதில் உருவாகும் தரவுகள் ஒரு reward model உருவாக்குவதற்காக பயன் படுத்தப்படும். இந்த reward function என்பது “இந்த பதில் நல்லதா, தீமையா?” என்பதற்கான மதிப்பீடு அளவுகோலாகும்.

3. கொள்கை மேம்பாடு (Policy Optimization)

இந்த reward model-ஐ பயன்படுத்தி, Reinforcement Learning மூலம் LLM-ஐ மாற்றுகின்றனர். Proximal Policy Optimization (PPO) என்ற ஒரு ஆல்கொரிதம் பயன்படும். இதன் மூலம், LLM புதிய பதில்களை உருவாக்கும்போது, அந்த reward மாடலின் விருப்பங்களை பூர்த்தி செய்யும் வண்ணம் செயல்படும்.

6.4.3 RLHF பயிற்சி முறை: செயல்முறை சுழற்சி

1. ஒரு கேள்வி அல்லது உத்தரவு வழங்கப்படுகிறது.
2. LLM இரண்டு அல்லது அதற்கு மேற்பட்ட பதில்களை உருவாக்குகிறது.
3. மனித annotators அவற்றை மதிப்பீடு செய்து ஒரு பதிலை மேலாக மதிப்பீடுகிறார்கள்.
4. இந்த human-preference தரவுகள் மூலம் reward model பயிற்சி பெறுகிறது.
5. அந்த reward-ஐ அதிகபட்சமாகச் செய்யும் (maximize) வகையில், LLM அதன் பதில்களை மாற்றிக்கொள்கிறது.

இந்த செயல்முறை தொடர்ந்து மேற்கொள்ளப்படுவதால், LLM மெதுவாக மனிதர்களின் நுண்ணிய எண்ணங்களுக்கும், ஒழுங்குப்பாடுகளுக்கும் ஒத்திசைவாக செயல்படுகிறது.

6.4.4 RLHF மூலம் ஏற்படும் செயல்மாற்றங்கள்

PRETRAINING + FINE-TUNING வரை	RLHF/ POST-TRAINING பின்
பதில்கள் factually சரியாக இருக்கலாம், ஆனால் தொடர்பற்றது	பாத்திரம், நுண்மை, சூழ்நிலை ஆகியவற்றுடன் பதில்கள்
சமூகமையற்ற, சில சமயம் அசிங்கமாகக் கேட்கும் பதில்கள்	மனித நாகரிக ஒழுக்க விதிகளுக்கு ஏற்ப பதில்கள்
hallucination இருக்கக்கூடும், ஆனால் பகுத்தறிவு குறைவாக இருக்கும்	உருவாக்கப்படும் தகவல்களில் நிலைத்த நுண்மை ஏற்படும்
Prompt sensitivity குறைவாக இருக்கும்	Prompt-இன் பாணி, எழுத்தியல் சிற்றுணர்வுகள் மற்றும் பயன்பாட்டு வழிமுறைகளை அனுசரிக்கும்

6.4.5 Bias Alignment மற்றும் Value Steering

முன்னிலைப் பயிற்சியில் bias உள்ள இடங்களை RLHF மூலமாக சீராக்க முடியும். உதாரணமாக, ஒரு pretrained LLM ‘engineer’ என்ற சொல்லை வலுவாக ‘he/his’ என்ற pronoun-களுடன் தொடர்புபடுத்தலாம். ஆனால் human feedback மூலம், அது gender-neutral பதில்களாக மாறும்.

மேலும், ஒன்று மிக முக்கியம்: RLHF-ஐ human values, ethics, emotional sensitivity மற்றும் legal constraints-க்கு ஒத்திசைக்கும் விதமாக அமைக்க முடியும். இதுவே “value steering” எனப்படும்.

6.4.6 Post-training பயன்பாடுகள்

Post-training இல்லாமல் ஒரு LLM ஒரு தொழில்நுட்ப கையேடாக மட்டுமே இருக்கும். ஆனால் Post-training பின், அது:

- வாடிக்கையாளர் உதவித் திட்டங்களில், நடுத்தரமான, பயனுள்ள பதில்கள் அளிக்கிறது
- கல்வியில், மாணவர் உணர்வுகளைக் கவனித்துப் பதிலளிக்கிறது
- உணர்வுப் பகுப்பாய்வு, mentorship, therapeutic response ஆகியவற்றில் மனித உணர்வுகளை பரிசீலிக்கிறது
- பொதுத்தகவல் பரிமாற்றம், உண்மை தவறான செய்தி மீதான தடுப்பு, சமூக ஒழுக்க மரபுகள் ஆகியவற்றை பின்பற்றுகிறது

6.4.7 தனி மனித பாணி மற்றும் கேள்வி உணர்திறன் (Prompt Sensitivity)

Post-training செய்த LLM-கள் prompts-இன் பாணியும் கற்பனையும் வாசிக்கக்கூடிய நிலைக்கு உயர்கின்றன.

“தலைவலி என்றால் என்ன?” என்பது போன்ற கேள்விக்கு ஒரு பொதுவான பதில் தரும் LLM, Post-training பின், “தலைவலி என்றால் என்ன? அதில் உள்ள மருத்துவ விளக்கங்களை உணர்த்தவும்” என்ற வழிகாட்டல் prompt-ஐக் கொண்டு துல்லியமான, செயல்படக்கூடிய பதில்களை வழங்கும்.

6.4.8 முடிவுகள்

Post-training என்பது ஒரு LLM-ஐ:

- வெறும் தரவுப் பிணைவுகளை அள்ளும் statistical engine-இலிருந்து
- மனித மொழியின் அழகு, நாகரிகம், உணர்வுப் பாங்கு ஆகியவற்றுக்கு ஒத்திசையும் ஒரு மனிதநேயம் வாய்ந்த உரையாடல் செயற்கை நுண்ணறிவு அமைப்பாக மாற்றும் வகை பயிற்சி கட்டம் ஆகும்.

இந்த கட்டத்துடன் முடியும் போது தான், LLM ஒரு நம்பகமான, பாதுகாப்பான, நுண்ணிய மற்றும் விழிப்புணர்வுடைய சிந்தனையாளர் ஆக செயல்படத் துவங்குகிறது.

மிக நன்றாக. இப்போது நாம் 6.5. பதில்கள் உருவாக்கும் நடைமுறைகள் (Sampling Techniques), Prompt Engineering மற்றும் Context Handling ஆகியவற்றை விரிவாகப் பார்ப்போம். இது LLM-களின் செயற்கை நுண்ணறிவு உள்கட்டமைப்பின் செயல்பாடுகளை, மனிதர்களுக்குப் போல் மொழியை உளவியல் மற்றும் சூழ்நிலை அடிப்படையில் உருவாக்கும் திறனை விளக்கும் ஒரு மைய அத்தியாயம்.

6.5. பதில்கள் உருவாக்கும் நுணுக்கங்கள் (Sampling, Prompting, Context)

Pretraining, Fine-Tuning மற்றும் Post-training ஆகிய மூன்று பயிற்சி கட்டங்களின் முடிவில், LLM ஒன்று மனிதர்களைப் போல உரையாடல்களில் கலந்துகொள்ளும், கேள்விகளுக்கு பதிலளிக்கும், மற்றும் ஆழமான புரிதலைக் காட்டும் திறனுடன் வெளிப்படுகிறது. ஆனால், இவ்வாறு ஒரு பதிலை உருவாக்கும் போது LLM எவ்வாறு சொற்களைத் தேர்ந்தெடுக்கிறது என்பது குறித்த சரியான புரிதல் இல்லையெனில், நாம் அதன் உள்ளார்ந்த செயல்பாட்டை சரியாகக் கவனிக்க முடியாது.

இந்த அத்தியாயத்தில், LLM-கள் ஒரு வாக்கியத்தை உருவாக்கும் போது என்னை அடிப்படையாக வைத்து செயல்படுகின்றன, மற்றும் எப்படி prompt-களின் வடிவமைப்பு மற்றும் context-இன் சீர்திருத்தம் முடிவுகளை மாற்றும் என்பதை விவாதிப்போம்.

6.5.1 பதில்கள் உருவாக்கும் திசைகள்: ஒரு நிலைப்பாடு

Transformer அடிப்படையிலான LLM-கள், உரையை வாக்கியங்களாக அல்லாது tokens எனப்படும் சிறு அலகுகளாகப் பிரிக்கின்றன. ஒவ்வொரு நிலைப்பதிலும், மாடல்:

- “அடுத்த token என்னவாக இருக்கக்கூடும்?” என்பதைப் probabilistically கணிக்கின்றது.
- அது softmax செயல்முறையின் மூலம், அனைத்து சாத்தியமான சொற்களுக்கும் நிகழ்தகவு (probability) மதிப்புகளை அளிக்கின்றது.

- அதன் பிறகு ஒரு sampling strategy அல்லது decoding method மூலம், அந்த எண்ணிக்கை வாய்ப்பு பட்டியலில் இருந்து ஒரு token தேர்ந்தெடுக்கப்படுகிறது.

6.5.2 Deterministic vs. Stochastic Generation

பதிலை உருவாக்கும் இரண்டு முக்கிய முறைகள் உள்ளன:

1. Deterministic (உருதியான) – ஒவ்வொரு கட்டத்திலும் மிக அதிக வாய்ப்பு கொண்ட token-ஐ மட்டும் தேர்ந்தெடுக்கின்றது.
 - இதில் greedy decoding அல்லது beam search போன்ற முறைகள் பயன்படும்.
 - Output ஒன்று மட்டுமே, எப்போதும் ஒரே மாதிரியாக இருக்கும்.
2. Stochastic (வாய்ப்பான அடிப்படை) – probability-யை அடிப்படையாகக் கொண்டு சற்று randomness கொண்ட தேர்வைச் செய்யும்.
 - இதில் Top-k, Top-p (nucleus sampling) மற்றும் Temperature-based sampling போன்றவை அடங்கும்.
 - Output ஒவ்வொரு முறையும் மாறக்கூடும்; ஆனால் இதில் படைப்பாற்றல் அதிகம்.

6.5.3 முக்கிய Sampling முறைகள்

A. Temperature Sampling

- Temperature என்பது ஒரு அளவீட்டு அளவுரு; அது randomness-ஐ கட்டுப்படுத்துகிறது.
- Temperature = 0 → மிகக் குறைந்த randomness; எப்போதும் probability-யில் முதலிடம் வகிக்கும் token மட்டுமே தேர்ந்தெடுக்கப்படும்.
- Temperature = 0.1 - 0.8 → probability படி தேர்வு நடைபெறும்.
- Temperature = 1 → மிகவும் படைப்பாற்றலான, ஆனால் அதிக வழக்கல்கள் உள்ள பதில்கள்.

உதாரணம்:

Prompt: “தமிழ் மொழி என்பது...”

- Temperature 0: “தமிழ் மொழி என்பது ஒரு மிகப் பழமையான மொழியாகும்.”

- Temperature 1: “தமிழ் மொழி என்பது தமிழர்களின் பண்பாட்டு சிந்தனையின் உயிரெழுத்தாகக் கருதப்படுகிறது.”

B.Top-k Sampling

- Softmax மூலம் வரும் vocabulary-யில் இருந்து k மிக உயர் probability-யை கொண்ட tokens மட்டும் shortlist செய்யப்படும்.
- அதன் பிறகு, அந்த k token-களில் இருந்து ஒரு token சிறிது randomness உடன் தேர்ந்தெடுக்கப்படும்.

உதாரணம்:

k = 50 என்றால், vocabulary-யில் இருந்து மிக உயர்ந்த 50 tokens மட்டும் பயன்படுத்தப்படும்.

C.Top-p Sampling (Nucleus Sampling)

- இதுவும் Top-k போன்றது, ஆனால் cumulative probability $\geq p$ (பொதுவாக $p = 0.9$) என்ற அளவின் கீழ் tokens சேர்க்கப்படும்.
- இதனால் மிக அதிக probability உள்ள tokens மட்டும் பயன்படுத்தப்படுகின்றன.

Top-p என்பது dynamic shortlist உருவாக்குகிறது, ஆனால் Top-k என்பது fixed shortlist.

6.5.4 Prompt Engineering

LLM-கள் மிகவும் prompt-sensitive. அதாவது, ஒரு கேள்வி எப்படி உருவாக்கப்படுகிறது என்பதற்கேற்ப, பதில் கணிசமாக மாறிவிடும்.

உதாரணம் 1:

- Prompt A: “எனக்கு சளி இருக்கு. என்ன மருந்து சாப்பிடலாம்?”
- Prompt B: “ஒரு மருத்துவராக, ஒரு வயதான நபர் சளியால் அவதிப்படுகிறார் என்றால், நீங்கள் என்ன பரிந்துரை செய்வீர்கள்?”

Prompt B-க்கு வரும் பதில் clinical tone உடன் இருக்கும். A-க்கு வரும் பதில் சாதாரண consumer-level ஆவியாக இருக்கும்.

உதாரணம் 2:

- Prompt: “ஒரு 100 வார்த்தைகளுக்குள் ‘இயற்கை’ பற்றி ஒரு கட்டுரை எழுதவும்.”
- Prompt: “இயற்கை என்றால் என்ன?”

முன்னைய prompt எழுத்து அளவு, பாணி ஆகியவற்றை கட்டுப்படுத்தும். LLM-ஐ ஒரு எழுத்தாளனாக அமைக்கும்.

6.5.5 Prompt Design Strategies

Prompt Engineering இப்போது ஒரு தனி துறையாக வளர்ந்து வருகிறது. இதில் பின்வரும் உத்திகள் முக்கியமானவை:

- Instructional Prompting: “Translate this to French”, “Summarize this paragraph” போன்றவை.
- Chain-of-Thought Prompting: ஒரு தீர்வு எவ்வாறு அடையப்படுகிறது என்பதை reasoning வழியாக கூறும்.
- Few-shot Prompting: ஒரு கேள்விக்கு முன் 2-3 உதாரணங்களை கொடுத்து, மாடலை அதை புரிந்து பதிலளிக்கச் செய்வது.
- Zero-shot Prompting: context இல்லாமல், ஒரு instruction மட்டுமே.

6.5.6 Context Window மற்றும் Memory Limitations

LLM-களுக்கு context window எனப்படும் ஒரு வரம்பு உள்ளது. இது பொதுவாக 2,000 tokens முதல் 128,000 tokens வரை (GPT-4, Claude 2.1 போன்ற மாடல்களில்).

- ஒரு conversation அல்லது document-ல் ஒரு LLM context-ஐ வைத்துக்கொண்டு reasoning செய்ய முடியும் என்பது அதன் limitations-க்கு உட்பட்டது.
- Context overflow-ஆனால் பழைய விவரங்கள் “தொலைந்து” போகலாம்.
- Context-aware prompting-ஐ optimize செய்வதன் மூலம், தேவையான history மட்டுமே மாடலுக்கு feed செய்யப்படுகிறது.

6.5.7 Practical Case: தமிழ் புத்தக சுருக்கம்

Prompt:

“தெனாலிராமனின் சிறுகதை தொகுப்பில் உள்ள மூன்றாவது கதையை சுருக்கமாக, 5 வரிகளில் சொல்லவும்.”

இந்த prompt:

- Story retrieval
- Logical condensation
- Cultural tone alignment

என மூன்றையும் பொருத்தப்படுத்த வேண்டிய ஒரு sample task ஆகும்.

LLM, இதற்குள் reasoning செய்து:

- கதை யார் பற்றி
- முக்கிய conflict என்ன
- தீர்வு என்ன

இவற்றை பட்டியலிட்டு, ஒரு அட்டவணையை மனதிற்குள் கட்டமைத்துப் பதிலளிக்கும்.

6.5.8 சாராம்சம்

- LLM ஒரு பதிலை உருவாக்கும் போது, அது வாய்ப்பு அடிப்படையில் சொற்களை தேர்ந்தெடுக்கும்.
- இந்த தேர்வில் randomness-ஐ கட்டுப்படுத்த Sampling Techniques முக்கிய பங்கு வகிக்கின்றன.
- Prompt-ஐ எந்த வார்த்தைகளில், எந்த அமைப்பில் தருகிறோம் என்பதைப் பொருத்து பதில்கள் மாறுகின்றன.
- Context window வரம்புகள் உள்ளன; ஆனால் அவற்றை முறையாக நிர்வகித்தால் தரமான உரையாடல்கள் உருவாக்கலாம்.

மிகச் சிறப்பு. இப்போது நாம் 6.6: LLM-களின் பயன்பாடுகள், சவால்கள் மற்றும் தமிழ் போன்ற மொழிகளில் அமலாக்கம் என்ற பகுதிக்குச் செல்வோம். இது LLM-களின் செயல்திறன் எவ்வாறு நவீன தொழில்நுட்பங்களில் பல்வகையாக பயன்படுகிறது, மற்றும் குறைவான மூலங்களுடன் (low-resource settings) இந்த மாதிரிகளை உருவாக்குவதற்கான நுட்பங்களை ஆராயும்.

6.6. பயன்பாடுகள், சவால்கள் மற்றும் தமிழ் போன்ற மொழிகளில் LLM வளர்ச்சி

பெரிய மொழி மாதிரிகள் (LLMs) இன்று வெறும் ஆராய்ச்சி லாபங்களிலேயே நின்றுவிடவில்லை. அவை பன்முகப் பயன்களுடனும், பல்துறை பயன்பாடுகளுடனும் உலகளாவிய மாற்றங்களை ஏற்படுத்தி வருகின்றன. ஆனால் இவ்வளவு வலிமையுள்ள தொழில்நுட்பத்தை தமிழ் போன்ற குறைவான மூலமுள்ள (low-resource) மொழிகளில் கொண்டு வருவதற்கு பல சவால்களும், அதனை மீறிச் செல்லக்கூடிய புதிய முயற்சிகளும் உள்ளன.

6.6.1 LLM-களின் முக்கிய பயன்பாடுகள்

LLM-கள் இன்று பல்வேறு துறைகளில் வலுவான செயல்திறனைக் காண்பிக்கின்றன:

1. கல்வி:

- மாணவர்கள் கேட்கும் கேள்விகளுக்கான விளக்கங்களை தருவது
- கட்டுரை எழுத்து, பதிலளிப்பு, மொழிபெயர்ப்பு
- ஆசிரியர்கள் பாடநெறிகளை வடிவமைப்பதில் உதவி

2. மருத்துவம்:

- நோயாளியின் அறிகுறிகளை வைத்து நோய்க்களங்களை சுட்டிக்காட்டுவது
- மருந்து தொடர்பான வினாக்களுக்கு அறிவுப்பூர்வமான பதில்கள்
- மருத்துவ ஆவணங்களை சுருக்கம் செய்தல் (Clinical Summarization)

3. வணிகம் மற்றும் வாடிக்கையாளர் சேவை:

- Chatbot ஆக வாடிக்கையாளர்களுக்கு பதிலளித்தல்
- தயாரிப்பு பற்றிய தகவல் வழங்கல்
- விற்பனை, சந்தைப்படுத்தல் மற்றும் பயனர் ஆதரவு

4. சட்டம் மற்றும் சட்ட ஆலோசனை:

- சட்ட பத்திகளை விளக்குதல்
- நீதிமன்ற தீர்ப்புகளை தெளிவுபடுத்துதல்
- சட்ட வழிகாட்டி உதவிகள்

5. பொருளாதாரம் மற்றும் பங்கு சந்தை:

- பங்கு சந்தை மாற்றங்களை பகுப்பாய்வு செய்தல்
- காலாண்டு அறிக்கைகள், விளக்கப்படுத்தல்
- புள்ளிவிவர தரவுகளை மொழிபெயர்க்கின்ற அறிவு உதவியாளர்

6. உள்ளமைப்பு மேலாண்மை (Infrastructure Management):

- நிரல் உதவி (e.g., code autocomplete)
- டாகுமென்டேஷன் உருவாக்கம்
- API design/validation

6.6.2 தமிழ் போன்ற குறைவான மூலமுள்ள மொழிகளுக்கான சவால்கள்

Low-resource languages எனப்படும் மொழிகள் (தமிழ், மலையாளம், கன்னடம், சிங்களம், ஹவுசா, சோமாலி போன்றவை) LLM-களால் குறைவாக சேவை செய்யப்படுகின்றன. இதற்குப் பின்னே சில முக்கிய காரணங்கள் உள்ளன:

1. தரவின் அளவு குறைவு:

- Wikipedia, Common Crawl போன்ற இடங்களில் தமிழ் போன்ற மொழிகளில் உள்ள உரைத் தொகுப்பு மிகக் குறைவாக உள்ளது.
- நூல்கள், செய்தித்தாள்கள், வலைப்பதிவுகள் போன்ற தரவுகள் மொத்தமாகவும் குறைவாக சேகரிக்கப்பட்டுள்ளன.

2. தரவின் தரம்:

- தமிழில் காணப்படும் பல இணையதள தரவுகள் பிழைகள் நிறைந்தவை, அல்லது conversational tone-ல் இல்லை.
- கிராமிய மொழி நுணுக்கங்கள், பன்மொழி கலப்புகள் உள்ளன.

3. Unicode & Encoding சிக்கல்கள்:

- தமிழ் எழுத்துக்குறிகள் Unicode-ல் சீர்படுத்தப்படாமல், பன்முக சிக்கல்கள் ஏற்படுகின்றன.
- எழுத்துப்பிழைகள், font mismatch, transliteration usage என அனேக encoding சிக்கல்கள் இருக்கின்றன.

4. Annotation பணியின் விலை:

- தமிழ் போன்ற மொழிகளில் பணம் செலவழித்து data curators/annotators வைத்துக்கொள்வது பெரும்பாலான opensource குழுக்களுக்கு சாத்தியமில்லை.

6.6.3 தொழில்நுட்ப நோக்கில் செய்திச் சுருக்கங்கள்

1. Transfer Learning

- English மற்றும் high-resource மொழிகளில் பயிற்சி பெற்ற LLM-ஐ, குறைவான தமிழ் தரவுகளுடன் fine-tune செய்வது.
- multilingual transformers (e.g., mBERT, XLM-R) பயன்படுகின்றன.

2. Backtranslation & Data Augmentation

- உரையை தமிழில் மொழிபெயர்த்து, அதை மாற்றியமைத்த உரையாக சேகரித்து data volume அதிகரிக்கும்.
- GPT + T5 போன்ற பயிற்சி செயற்கைகள் பயன்படுத்தப்படும்.

3. Tokenization மேலாண்மை

- Byte-level tokenizers (e.g., SentencePiece, Byte-Pair Encoding) தமிழ் எழுத்துக்களுக்கு மேம்பட்ட coverage தருகின்றன.
- Custom tokenization: Tamil letters & grammar-based segmentation models.

4. LoRA / Adapter Fine-Tuning

- Resource constraint-உள்ள சூழலில் சிறிய LLM மாதிரிகள் (e.g., Alpaca, TinyLlama) – தமிழ் உரை தரவுகளில் Adapter layers வைத்து fine-tune செய்ய முடியும்.

6.6.4 தமிழில் LLM உருவாக்கத் தகுந்த கட்டமைப்புகள்

கூறு	விளக்கம்
Tokenizer	IndicNLP, Byte-BPE
Datasets	Tamil Wikipedia, Common Crawl-Ta, OpenSubtitles-Ta
Architecture	DistilGPT-2, LLaMA, Falcon, Mistral
Fine-tuning	Instruction Tuning using Tamil dialogues
Evaluation	BLEU, ROUGE, perplexity for Tamil corpora

6.6.5 எதிர்காலம்

தமிழில் LLM உருவாக்கம் என்பது வெறும் மொழிக்காக அல்ல, அது ஒரு மாற்றுத்திறன் வாய்ந்த சமூக நியாயமாக மாறியுள்ளது.

- மருத்துவ chatbot-கள் கிராமப்புற மக்களுக்காக தமிழில் பேசும்
- பள்ளி மாணவர்களுக்கு ஊக்கமளிக்கும் தமிழ் ஆசிரியர் avatars
- தமிழ்ப் பழமொழிகளை பயன்பாடுகளுக்குத் தந்திடும் இலக்கிய-நுண்ணறிவு LLM-கள்

எல்லாம் விரைவில் நம்மை நோக்கி வருகின்றன.

- LLM-கள் பல்துறை பயன்பாடுகளை உருவாக்குகின்றன – கல்வி முதல் மருத்துவம் வரை.
- தமிழ் மொழி உள்ளிட்ட low-resource setting-இல் உள்ள மொழிகள், தரவுக்குறைவால் பின்தங்கியுள்ளன.
- Opensource இயக்கங்கள், transfer learning, tokenization மற்றும் adapter tuning ஆகியவையின் மூலம் LLM-ஐக் கொண்டு வர முடிகிறது.
- இது வெறும் தொழில்நுட்ப நடவடிக்கையாக அல்ல; ஒரு மொழி நாகரிகத்தை தாங்கும் ஒரு புதுவழி.

6.7 எதிர்காலப் பரிமாணங்கள் மற்றும் நாகரிகக் கேள்விகள் (Ethical and Societal Implications)

பெரிய மொழி மாதிரிகள் (LLMs) மனிதர்களின் மொழி, அறிவு, தர்க்கம், மற்றும் சமூக நடைமுறைகளை உருவாக்கி பயன்படுத்தும் ஒரு நுண்ணறிவு சீரமைப்பாக வளர்ந்துள்ளன. ஆனால், இவை ஒரு நன்மைதரும் தொழில்நுட்பமாக மட்டுமல்ல, சமூகப் பொறுப்பு, ஊழியம், மற்றும் மனித அடையாளங்களுக்குள் நுழையும் ஆபத்தான வழிகளையும் கொண்டுள்ளன. இதனால்தான், LLM-களைப் பொறுப்புடன் உருவாக்கும் மற்றும் பயன்படுத்தும் பண்பாட்டியலில்

நம்மை நாம் முன்வைக்க வேண்டும்.

6.7.1 விஸ்தாரமான தாக்கங்கள்

LLM-கள் இன்று மக்கள் வாழ்க்கையின் அனைத்துத் துறைகளிலும் இடம்பிடிக்கின்றன: கல்வி, சட்டம், ஊடகம், அரசியல், மருத்துவம், வேலைவாய்ப்பு. ஆனால் இந்த இடம்பிடிப்பு பழைய மனித-மனித இடைமுகங்களை இடிக்கக்கூடிய, அல்லது மீண்டும் பிணைக்கும் இருவழி பாதையாக இருக்க முடியும்.

தாக்கங்கள்:

- கல்வித் துறையில் மனித ஆசிரியர்களின் பணிபலனை மாறும்.
- வேலைவாய்ப்பு நிலைகளில் பலவகை ஆட்கள், குறிப்பாக எழுத்தாளர், மொழிபெயர்ப்பாளர், data analyst போன்ற பணிகள் மாறும்.
- பொது அறிவுத் தளங்களில் misinformation மற்றும் hallucination அதிகரிக்கக்கூடியது.
- மொழி சமத்துவம் சீரழியக்கூடியது — குறைந்த ஆதாரமுள்ள மொழிகள் புறக்கணிக்கப்படலாம்.

6.7.2 நாகரிகப் பொறுப்புகள்

LLM-கள் மனிதர்களைப் போல பதிலளிக்கக்கூடியவை என்பதால், அவை சுயமாகத் தவறான அல்லது கேலிகூத்தான தகவல்களையும் வழங்கக்கூடும். இதனால், அவை பின்வரும் நாகரிகக் கேள்விகளை எழுப்புகின்றன:

1. பொறுப்புமிக்க பதில்:

- தவறான தகவலால் பாதிப்பு ஏற்படும்போது, பதிலுக்குப் பொறுப்பானவர் யார்? மென்பொருள் உருவாக்குநரா, பயனாளரா, அல்லது மாடலா?

2. முன்னெச்சரிக்கை வடிவமைப்பு (Safety by Design):

- ஒரு LLM கற்றுக்கொள்ளும் தரவுகளில் இருந்து தீவிரவாதம், பாகுபாடு, அடக்குமுறை சார்ந்த சொற்பிரயோகங்களை எவ்வாறு தவிர்க்க முடியும்?

3. பட்டயமற்ற சிகிச்சை (Unlicensed Advice):

- மருத்துவம், சட்டம் போன்ற துறைகளில், ஒரு LLM தவறாக வழங்கும் தீர்வு ஒரு மனிதரின் வாழ்வை மாற்றக்கூடியதாய் இருக்கலாம். இவ்வாறான நிலைகளில் LLM-ஐ முறையாக சார்பு கூறுவதற்கான சட்டக்கட்டுப்பாடுகள் தேவைப்படுகிறது.

6.7.3 பாகுபாடும் (Bias), புறக்கணிப்பும் (Exclusion)

LLM-கள் கற்றுக்கொள்ளும் அனைத்து தரவுகளும் ஒரு சமூகத்தைக் பிரதிபலிக்கின்றன. ஆனால் அவை அதே சமயம்:

- குறிப்பிட்ட மொழிகள், சமூகங்கள், பாலினங்கள், அல்லது கலாசாரங்களை புறக்கணிக்கக்கூடும்.
- தோற்றம், மதம், இனம், பாலின அடையாளம் போன்ற பிரிவுகளில் bias-ஐ அதிகரிக்கும் அபாயம் உள்ளது.

உதாரணம்:

ஒரு LLM “CEO” என்ற சொல்லுக்கு 90% male context-இல் பதிலளிக்கலாம், ஏனெனில் கற்ற தரவுகளில் அதுவே அதிகம்.

6.7.4 தனியுரிமை மற்றும் தரவு உரிமைகள்

- Open datasets இல் இருக்கும் தரவுகள் பெரும்பாலும் copyright அல்லது தனிப்பட்ட தகவல்களைக் கொண்டிருக்கக்கூடும்.
- ஒரு LLM, பயிற்சி தரவிலிருந்தே பழைய email, சொந்த பெயர்கள், தொடர்புகள் போன்றவற்றை நினைவில் வைத்திருக்கக்கூடும் — இது ஒரு தனியுரிமை மீறல் ஆகும்.
- தொழில்துறையின் தரவுகள் (confidential documents, chat logs) கூட கேட்கப்படும் query-க்கேற்ப ஒரு LLM-ஐ influence செய்யக்கூடியது.

இந்தநிலையில், தரவு உரிமைகள் மற்றும் GDPR போன்ற சட்டங்கள் LLM-களுக்கு அளவிடப்பட்ட learning boundary-ஐ கட்டுப்படுத்த வேண்டும்.

6.7.5 Regulating LLMs — ஒரு நாகரிக வாதம்

உலகளாவிய அளவில், LLM-களுக்கு உரிய சட்டங்களை அமைக்கிறது என்பது சுருக்கப்பட்ட நாகரிக ஆணையம் உருவாக்குவது போல.

- European Union: AI Act மூலமாக LLM-களை regulate செய்ய முயற்சிக்கிறது.
- UNESCO மற்றும் OECD: AI-இல் Ethics Guidelines வெளியிட்டுள்ளன.
- India மற்றும் Tamil Nadu போன்ற மாநில அளவிலான governing body-கள், LLM-களின் பயிற்சி தரவுகள், பயன்பாட்டு எல்லைகள் குறித்த தங்களது கருத்துகளை உருவாக்க வேண்டிய அவசியம் உள்ளது.

6.7.6 தமிழ் மொழியில் நாகரிக ஒழுங்குகள்

- தமிழில் LLM-கள் உருவாக்கும் போது, இலக்கிய மரபுகளையும், நுண்ணிய கலாசாரப் பிம்பங்களையும் சரியாகக் கையாள வேண்டியது மிக முக்கியம்.
- ‘பெரியார்’, ‘திருவள்ளுவர்’, ‘சங்க இலக்கியம்’ போன்ற வரலாற்று சம்பந்தப்பட்ட பிரச்சனைகளில் தவறான பதில்கள் பொது நலத்துக்கு ஆபத்தாக இருக்கக்கூடும்.
- அதனால், தமிழ்நாடு அரசும், தமிழ் பல்கலைக்கழகங்களும் சேர்ந்து மொழி, கலாசார செம்மை என்பதை அடிப்படையாகக் கொண்டு LLM alignment-ஐ வடிவமைக்க வேண்டியது அவசியம்.

6.7.7 முடிவுரை

பெரிய மொழி மாதிரிகள் (LLMs) ஒரு தொழில்நுட்ப புரட்சியைத் தூண்டியிருக்கலாம். ஆனால், அவை மனித சமூகத்தின் தத்துவச் சூழலுக்குள் நுழைந்துவிட்டன. இதனால்:

- இது ஒரு algorithmic achievement மட்டுமல்ல, ஒரு நாகரிகப் பொறுப்பு.
- இது ஒரு செயற்கை நுண்ணறிவு சாதனை மட்டுமல்ல, ஒரு மனித பன்முகம் தரும் ஒப்புதல்.
- இது மொழி உருவாக்கம் மட்டுமல்ல, அருவணக்கத்தின் ஒரு தளமாக மாறுகிறது.

எனவே, LLM-களைப் புரிந்து கொள்வது என்பது மொழியையும், சமூகத்தையும், நீதியையும், எதிர்காலத்தையும் ஒரே நேரத்தில் நோக்கிப் பார்க்கும் ஒரு முயற்சியாகும்.