

# Introduction & Foundation:

## ১. Large Language Model (LLM) কী?

Large Language Model (LLM) হলো একটি **Deep Learning**—ভিত্তিক ভাষা মডেল, যা মানুষের ভাষা বুঝতে, বিশ্লেষণ করতে এবং নতুন ভাষা তৈরি করতে সক্ষম।

LLM মূলত কাজ করে পরবর্তী শব্দ (**next token**) অনুমান করার মাধ্যমে। একটি বাক্যের আগের শব্দগুলো দেখে সম্ভাব্য সবচেয়ে উপযুক্ত পরের শব্দটি কী হতে পারে—এটাই মডেলের মূল কাজ।

LLM ট্রেন করা হয়:

- বই
- ওয়েবসাইট
- আর্টিকেল
- কোড
- ডকুমেন্ট

এই বিশাল ডেটার কারণে মডেল ভাষার গঠন, অর্থ, প্রসঙ্গ (context) এবং কিছু মাত্রায় যুক্তি (reasoning) শিখে নেয়।

## ২. কেন একে “Large” বলা হয়?

LLM-কে “Large” বলা হয় কারণ এতে থাকা প্যারামিটার (**parameters**) সংখ্যা অত্যন্ত বেশি।

### Parameter কী?

Parameter হলো নিউরাল নেটওয়ার্কের ভেতরের শেখার উপাদান—যেগুলো training-এর সময় আপডেট হয়।

আধুনিক LLM-এ:

- বিলিয়ন ( $10^9$ )
- এমনকি ট্রিলিয়ন ( $10^{12}$ ) পর্যন্ত parameter থাকতে পারে

বেশি parameter মানে:

- ভাষার সূক্ষ্ম পার্থক্য বোঝা
- দীর্ঘ context মনে রাখা
- জটিল প্রশ্নের ভালো উত্তর

## ৩. Traditional NLP Model বনাম LLM

**Traditional NLP:**

- Task-specific
- আলাদা কাজের জন্য আলাদা মডেল
- যেমন:
  - Translation model
  - Sentiment model

## LLM:

- General-purpose
- একই মডেল দিয়ে বহু কাজ
- শুধু instruction (prompt) বদলালেই কাজ বদলে যায়

এটিই LLM-এর সবচেয়ে বড় শক্তি।

## ৪. LLM কেন স্ক্র্যাচ থেকে শেখা জরুরি?

শুধু LLM ব্যবহার করলে:

- ভিতরের কাজ বোঝা যায় না
- Debug বা optimize করা কঠিন
- Interview-এ concept clear থাকে না

স্ক্র্যাচ থেকে বানালে:

- প্রতিটা ধাপ পরিষ্কার বোঝা যায়
- Model কেন এমন output দেয় বোঝা যায়
- Deep Learning foundation শক্ত হয়

**Use** করার আগে **Understand** করা—এটাই মূল লক্ষ্য।

## ৫. Transformer Architecture: LLM-এর ভিত্তি

LLM-এর মূল ভিত্তি হলো **Transformer Architecture**।

Transformer-এর প্রধান উপাদান:

- Tokenization
- Embedding
- Positional Encoding
- Self-Attention
- Multi-Head Attention
- Feed Forward Network

এই আর্কিটেকচার মডেলকে সক্ষম করে:

- একই সাথে অনেক শব্দ প্রসেস করতে
- দূরের শব্দের সম্পর্ক বুঝতে
- দ্রুত এবং scalable training করতে

## ৬. Key Concepts in LLM

LLM বুঝতে নিচের ধারণাগুলো অত্যন্ত গুরুত্বপূর্ণ:

- **Token:** শব্দ বা শব্দের অংশ
- **Embedding:** শব্দকে সংখ্যায় রূপান্তর
- **Positional Encoding:** শব্দের অবস্থান বোঝানো
- **Attention:** কোন শব্দ গুরুত্বপূর্ণ তা নির্ধারণ
- **Key, Query, Value:** Attention হিসাবের ভিত্তি

এগুলো না বুঝলে LLM আসলে বোঝা যায় না।

## ৭. Open Architecture কেন গুরুত্বপূর্ণ?

Open architecture মানে:

- Model structure উন্মুক্ত
- Code দেখা যায়
- Experiment করা যায়

এর ফলে:

- শেখা সহজ হয়
- Research করা যায়
- নিজস্ব model তৈরি সম্ভব হয়

শেখার জন্য open architecture সবচেয়ে উপযোগী।

## ৮. Generative AI এবং LLM

LLM হলো Generative AI-এর একটি অংশ।

Generative AI পারে:

- নতুন টেক্সট তৈরি করতে
- ছবি বানাতে
- অডিও / ভিডিও তৈরি করতে

LLM মূলত:

- Text generation
- Question answering
- Summarization
- Translation

এই কাজগুলো করে।

## ৯. শেখার রোডম্যাপ (High-Level)

একটি LLM স্ক্যাচ থেকে বানাতে সাধারণত ধাপগুলো হলো:

1. Text data সংগ্রহ
2. Tokenization
3. Vocabulary তৈরি
4. Model architecture ডিজাইন
5. Training loop লেখা
6. Loss calculation
7. Model evaluation

প্রতিটা ধাপ একে অপরের সাথে গভীরভাবে যুক্ত।

## ১০. কেন **Foundation** বোঝা সবচেয়ে গুরুত্বপূর্ণ?

Foundation শক্ত হলে:

- নতুন model বুঝতে সময় লাগে না
- Research paper পড়া সহজ হয়
- Industry-ready হওয়া যায়

শুধু application বানানো সাময়িক সুবিধা দেয়,  
কিন্তু **foundation** দীর্ঘমেয়াদি দক্ষতা তৈরি করে।