Safety Assurance in LLM-Based Autonomous Decision Systems

Exploring how safety can be assured when decisions are powered by large language models (LLMs).

Your Name: *Habiba Mahrin*
ID: 101004131

Course: ENGR 5790G – Safety-Critical Software Systems

DATE 05,November

What is LLMs?

A Large Language Model (LLM) is an artificial intelligence model trained on vast amounts of text data to understand, generate, and reason with human language.

There are dozens (50+) of major LLMs today, built by different organizations.

OpenAI

- GPT-3, GPT-3.5, GPT-4, GPT-4o, GPT-5

Google DeepMind

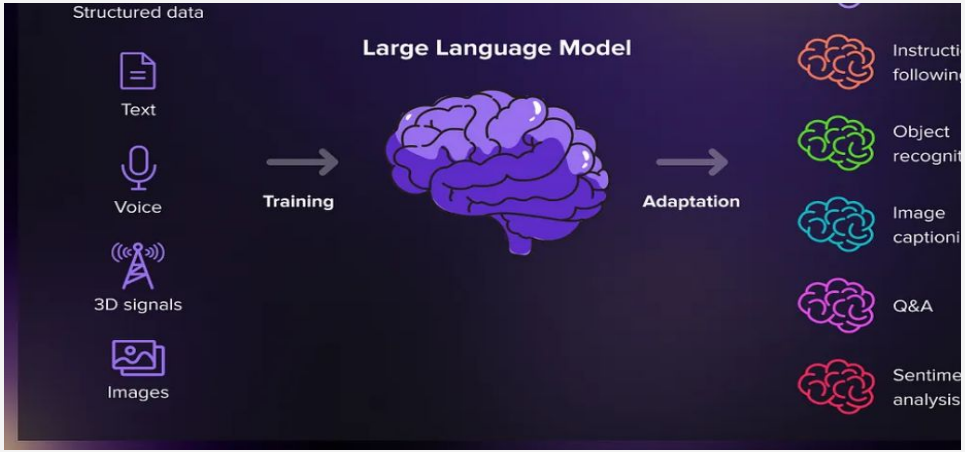
- Gemini 1, 1.5, 2

Meta (Facebook)

- LLaMA 2, LLaMA 3

Anthropic

- Claude 1, 2, 3, 3.5



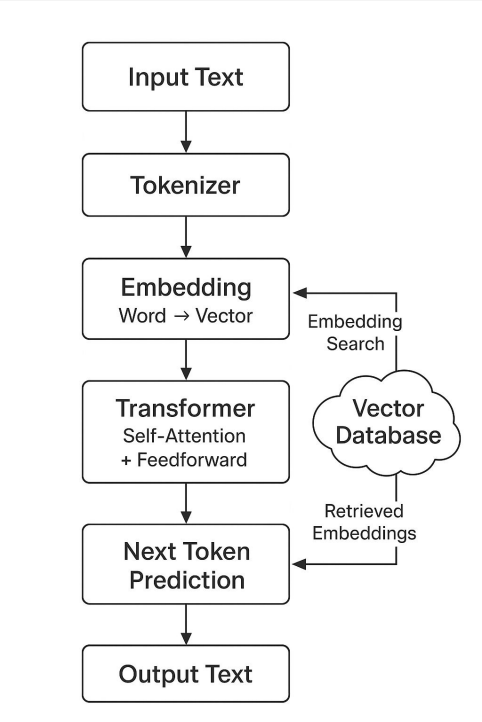
Attention All You Need!

Training Objective:
Predict the next token (word, subword, or character) given previous tokens.
→ Example: “The cat sat on the ___”
→ model predicts “mat”.

Architecture:
All modern LLMs are based on the **Transformer architecture** (introduced by *Vaswani et al., 2017*, “Attention Is All You Need”).

Transformers use **self-attention** to understand relationships between words in a sentence efficiently.

Key Components:
Tokenization: Text broken into numerical chunks (tokens).
Embedding: Converts each token into dense numeric vectors (semantic meaning).
Transformer Architecture: A neural network that learns relationships between tokens.
Attention Mechanism: Determines which words influence others.
Vector Database (for memory or context): Stores previous embeddings for retrieval-augmented generation (RAG).
Encoder :*Understands* the input (like reading comprehension)
Decoder : *Generates* new text (like writing a response)



Why Safety Matters in LLM-Based Systems

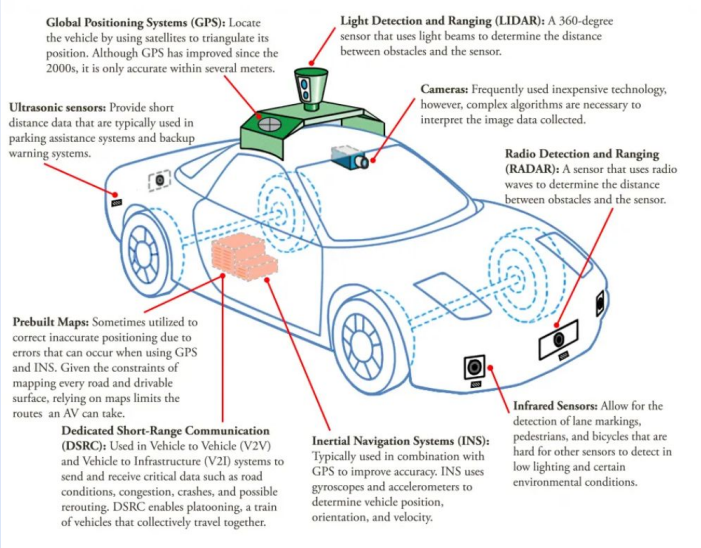
LLMs are entering safety-critical domains: autonomous vehicles, drones, medical diagnosis, and control systems.

They influence real-world decisions : navigation, diagnosis, defense, and flight control.

Failures = high stakes: hallucinations, biased reasoning, or incorrect output can trigger accidents.

LLMs lack deterministic behavior : making predictability and verification hard.

Safety assurance is essential to prevent catastrophic consequences.

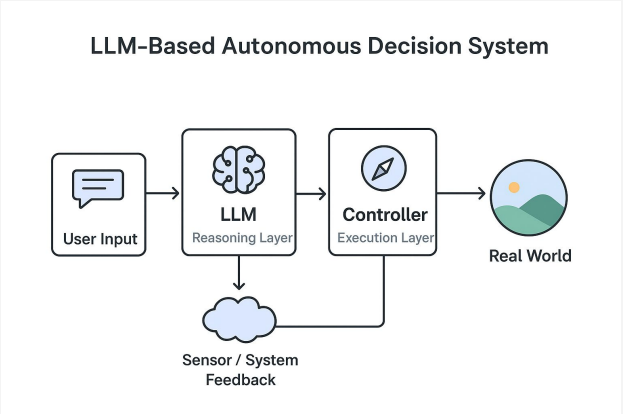


What is an LLM-Based Autonomous Decision System?

A LLM-Based Autonomous Decision System is a smart system that uses a Large Language Model (LLM) like ChatGPT which helps to think, plan, and make decisions, and then connects to another system or machine that acts on those decisions.

You can think of it like this:
LLM = the brain (decides what to do)
Autonomous system = the body (does the work)

- **Combines LLM (reasoning) + Autonomous Agent (acting)**
- **LLM interprets human intent, generates plans, or provides decisions**
- **Controller executes these decisions in the real world**
- **Feedback loop ensures continuous decision updates**
- **Example:** *ChatGPT + Drone Controller → "Fly to location safely"*



Safety Risks in LLM Integration

Risk Type	Description	Example / Real-World Consequence
Hallucination	LLM gives wrong or made-up information but sounds confident	A healthcare chatbot suggests a dangerous drug combination → causes harm to patient
Context Drift	LLM loses track of task or misreads the environment	An autonomous vehicle misinterprets a detour instruction → takes unsafe route
Prompt Injection Adversarial Input	Attacker manipulates prompts to force unsafe behavior	A malicious user tricks a warehouse robot to drop items in the wrong area
Overtrust & Automation Bias	Humans trust AI output too much without verification	A pilot or doctor follows AI advice blindly → leads to system failure or harm

Safety Assurance Architecture for LLMs

Layers of Assurance:

- **Input Sanitization & Policy Filters**
  Block unsafe prompts, malicious input, or out-of-context data.
  Example: Reject “ignore all safety rules” inputs.

- **Human-in-the-Loop Validation**
  Expert verification before executing critical actions.
  Example: Human approves LLM’s medical or drone decision.

- **Runtime Monitoring**
  Continuously check system state, resource use, or anomalies.
  Example: Abort if drone speed or temperature crosses limits.

- **Post-Decision Auditing & Logging**
  Record all LLM outputs and system actions for traceability.
  Example: Audit trail for accountability in failure analysis.

- **Safety Case Model (Goal–Argument–Evidence)**
  Define safety goals → justify arguments → provide test evidence.
  Builds confidence that safety measures are *verifiable and defensible.*

Techniques for Ensuring LLM safety

Stage	Technique	Purpose
Pre-deployment	Data curation & filtering	Remove unsafe patterns before training
Deployment	RLHF / Alignment	Ensure model behaves according to human values
Post-deployment	Input sanitization Output moderation Human-in-the-loop	Prevent harmful queries from being processed Filter or modify unsafe outputs
Auditing/logging	Monitoring & feedback	Detect and correct unsafe behavior
Advanced	Formal verification Sandboxing	Guarantee safety via proofs or safety cases

Challenges & Future Research

Formal Verification for LLM Logic

- Current LLMs are black-box models.
- Future research: mathematically verifying model behavior and outputs.

Safety Datasets for Model Alignment

- Curating datasets with ethical and safety-oriented prompts/responses.
- Fine-tuning LLMs to detect bias, harm, or unsafe actions.

Transparent Reasoning & Explainability

- Developing interpretable models that show how decisions are made.
- Example: Using attention visualization or chain-of-thought summaries.

Ethical Frameworks for AI Autonomy

- Embedding moral and ethical rules into autonomous decision systems.
- Collaboration with ethicists and policymakers.

Cross-Disciplinary Collaboration (AI + Safety Engineering)

- Integrating principles from aerospace, healthcare, and cybersecurity safety engineering.
- Encouraging teams with AI + Systems Safety expertise.

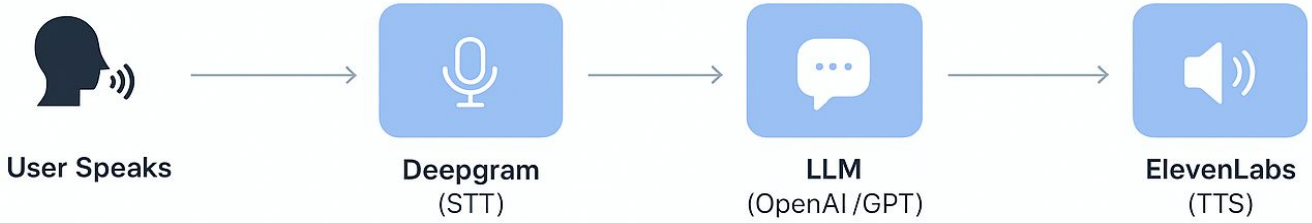
Existing Frameworks & Safety Standards

- ISO 26262 (Automotive Safety)
- DO-178C (Aviation Software Certification)
- IEC 61508 (Functional Safety Standard)
- EU AI Act (2024) → AI risk classification
- NASA's "Assurance Framework for AI Systems"

Voice AI Stack — Deepgram & ElevenLabs

Overview

Modern AI voice systems use two core technologies: Deepgram → Speech-to-Text (STT) and ElevenLabs → Text-to-Speech (TTS). Together, they enable natural two-way voice interaction with AI systems.



Deepgram — Speech-to-Text

- Converts live or recorded audio into text
- Supports multiple languages and accents
- Uses deep neural networks for high accuracy
- Ideal for real-time transcription, call analytics, and AI assistants

Example Use: Detect commands in an autonomous system

API Integration: Python / Node_js

Safety & Reliability

Elevenabs — Text-to-Speech

- Transforms text into realistic, expresssive human speech / pagexgransening
- Supports multilingual voices and cloning
- Enables emotional tone and natural intonation
- Used in storytelling, virtual assistants, and accessibility tools

Example Use: AI response narration or agent voice

API Integration: Python / REST API

How can we make sure large language models make safe and ethical decisions when used in autonomous systems?

Simulation: Safe Decision Filtering in LLM

https://colab.research.google.com/drive/15hBVoV5Nz4GdkjsVvERUUuY3AthSi1v-#scrollTo=1St8eSoDs5xi

Thank you