Unit 5 Case Studies

- ➢ IBM deep Blue
- ➢ AI AlphaGo
- ➢ Parkinson’s Disease Prediction
- ➢ Tesla Autopilot
- ➢ Robinhood’s AI-Driven Fraud Detection
- ➢ IBM Watson
- ➢ Alexa
- ➢ Siri
- ➢ Eliza
- ➢ ChatGPT

IBM Deep Blue

- **IBM Deep Blue** was a chess-playing computer developed by IBM. It became famous for defeating world chess champion **Garry Kasparov** in 1997 — the first time a computer had beaten a reigning world champion under standard tournament conditions.

- **Key Features:**

- **Type:** Specialized supercomputer for chess.

- **Search Algorithm:** Used a brute-force search combined with heuristic evaluation.

- **Processing Power:** Could evaluate about **200 million positions per second**.

- **Programming:** Incorporated extensive chess knowledge from grandmasters.

- **Significance:** Marked a milestone in **Artificial Intelligence (AI)** and **computational power**, showcasing how machines could perform complex reasoning in a structured domain.

System Architecture

- **Hardware:**
  - Contained **30 IBM RS/6000 SP processors**.
  - Equipped with **480 custom VLSI chess chips** for parallel computation.
  - Could evaluate **200 million chess positions per second**.
- **Software:**
  - Used **minimax search algorithm** with **alpha-beta pruning**.
  - Integrated **opening book** and **endgame databases**.
  - Employed **heuristic evaluation functions** to assess chessboard positions.

Working Principle

1. **Input:** Current chessboard configuration.

2. **Move Generation:** Lists all possible legal moves.

3. **Search:** Explores millions of move sequences using a **tree search algorithm**.

4. **Evaluation:** Scores positions using heuristics (material, king safety, mobility, pawn structure, etc.).

5. **Selection:** Chooses the move leading to the highest evaluated score.

6. **Output:** Best move recommendation.

AI Concepts Involved

- **Game Tree Search**
- **Heuristic Evaluation**
- **Alpha-Beta Pruning**
- **Parallel Processing**
- **Domain-Specific Knowledge Representation**
- **Expert System Principles**

AI AlphaGo

- **Introduction**

- **AlphaGo** is an artificial intelligence program developed by **DeepMind Technologies** (a subsidiary of Google).

- It was designed to play the ancient board game **Go**, which is significantly more complex than chess.

- In **2016**, AlphaGo made history by **defeating world Go champion Lee Sedol (9-dan professional)** — a landmark event in AI research.

Why Go Was a Major Challenge for AI

- The game of Go has an extremely large **state space** — about **10^{170} possible board configurations** (compared to chess's ~10^{47}).
- Traditional brute-force methods (like those used by IBM Deep Blue) were **impractical**.
- Success required **learning**, **intuition**, and **pattern recognition**, mimicking human cognitive strategies.

Development Background

- Developed by DeepMind Technologies, founded by Demis Hassabis, Shane Legg, and Mustafa Suleyman.
- Built upon deep learning and reinforcement learning architectures.
- Published in Nature (2016) under the paper titled “Mastering the game of Go with deep neural networks and tree search.”
- AlphaGo defeated:
- Fan Hui (European Champion, 2015)
- Lee Sedol (2016) — winning 4–1
- Ke Jie (2017) — the world’s #1 Go player at that time

System Architecture

- AlphaGo combines deep neural networks and tree search algorithms:
- 1. Neural Networks
  - Policy Network: Suggests possible moves (probability distribution over actions).
  - Value Network: Predicts the outcome (win/loss probability) from a given board position.
- Both trained using supervised learning from expert games and reinforcement learning through self-play.
- 2. Monte Carlo Tree Search (MCTS)
  - Simulates many possible future games (playouts).
  - Combines the policy network to focus the search and the value network to evaluate positions efficiently.
  - Balances exploration and exploitation using probabilistic methods.

Learning Process

- 1. Supervised Learning Phase:
- Trained on 30 million positions from expert human games.
- Learned to imitate professional moves (~57% accuracy).
- 2. Reinforcement Learning Phase:
- Played millions of games against itself (self-play).
- Used reinforcement signals (win/loss) to improve its policy and value estimation.
- Gradually surpassed human-level performance.

Key Algorithms Used

1. **Deep Neural Networks (DNNs)** for move prediction and board evaluation.

2. **Reinforcement Learning (RL)** for improvement through experience.

3. **Monte Carlo Tree Search (MCTS)** for efficient exploration of move sequences.

4. **Gradient Descent Optimization** for model training.

Technical Overview

Component	Description
Game	Go (19×19 grid)
Learning Type	Supervised + Reinforcement Learning
Algorithm	Deep Neural Networks + Monte Carlo Tree Search
Training Data	Human expert games + Self-play
Computing Power	1920 CPUs + 280 GPUs (for AlphaGo Zero, fewer but more efficient)
Programming Language	Primarily Python & C++
Framework	TensorFlow (DeepMind's in-house libraries)

Comparison: AlphaGo vs IBM Deep Blue

Feature	**IBM Deep Blue**	**AlphaGo**
Game	Chess	Go
Approach	Brute-force + heuristics	Deep learning + reinforcement learning
Learning	None (pre-programmed)	Learned via self-play
Algorithm	Minimax + Alpha-Beta Pruning	MCTS + Deep Neural Networks
Victory	Garry Kasparov (1997)	Lee Sedol (2016)
Significance	Power of computation	Power of learning

Parkinson’s Disease Prediction

- **Introduction**
- **Parkinson’s Disease (PD)** is a **neurodegenerative disorder** affecting movement, speech, and balance.
- Early detection is challenging since symptoms appear gradually.
- **AI and Machine Learning** help in **early diagnosis and progression prediction** using medical data (voice, gait, handwriting, and brain scans).

Why AI is Needed

- ➢ Traditional diagnosis relies on clinical observation, which can be subjective.
- ➢ AI can identify subtle patterns in data not visible to human experts.
- ➢ Enables non-invasive, faster, and more accurate screening.

Data Sources Used

- AI systems use multimodal data such as:

- **Voice recordings:** PD affects speech — AI detects tremors or tone changes.
- **Handwriting samples:** Analyzes micrographia (small handwriting).
- **Gait data:** Sensors track walking patterns and tremors.
- **MRI or DaTscan images:** Deep learning identifies brain abnormalities.
- **Clinical and genetic data:** Patient history and biomarkers.

AI Techniques Used

Technique	Description
Machine Learning (ML)	Uses classifiers like SVM, Random Forest, Decision Tree for prediction.
Deep Learning (DL)	CNNs and RNNs analyze MRI scans and temporal gait data.
Speech Analysis Models	Extract Mel-Frequency Cepstral Coefficients (MFCCs) for voice pattern recognition.
Feature Engineering	Selects important biomarkers using PCA or statistical ranking.
Ensemble Learning	Combines multiple models for improved accuracy.

Typical Workflow

1. **Data Collection** – Acquire patient voice, image, or motion data.
2. **Preprocessing** – Noise removal, normalization, segmentation.
3. **Feature Extraction** – Identify relevant features (MFCC, gait speed, handwriting pressure).
4. **Model Training** – Train ML/DL models (SVM, CNN, etc.) using labeled data.
5. **Prediction** – Model classifies subjects as PD or healthy.
6. **Evaluation** – Accuracy, sensitivity, specificity, and ROC curve.

Common Algorithms

- **Support Vector Machine (SVM)** – popular for voice-based PD detection.
- **Random Forest (RF)** – handles diverse medical datasets effectively.
- **Convolutional Neural Networks (CNNs)** – used for MRI image-based diagnosis.
- **Recurrent Neural Networks (RNNs)** – model sequential gait or speech data.
- **XGBoost / Gradient Boosting** – improves classification accuracy.

Example Datasets

- ➢ **UCI Parkinson's Dataset:** Voice recordings from 31 patients.
- ➢ **mPower Dataset (Sage Bionetworks):** Smartphone-based movement and voice data.
- ➢ **PPMI (Parkinson's Progression Markers Initiative):** Clinical and imaging biomarkers.

Performance Metrics

Metric	Description
Accuracy	Overall correctness of the model
Precision	Proportion of true PD predictions
Recall (Sensitivity)	Ability to detect actual PD cases
F1-Score	Balance of precision and recall
AUC-ROC	Measures classification quality

Tesla Autopilot

- **Tesla Autopilot** is an advanced driver-assistance system (ADAS) developed by **Tesla, Inc.** It uses **artificial intelligence (AI)**, **deep learning**, and **sensor fusion** technologies to provide semi-autonomous driving capabilities.

- **Overview**

- Tesla Autopilot is designed to assist drivers by automating various aspects of driving such as steering, acceleration, and braking within its lane. While not fully autonomous, it significantly reduces the driver's workload.

Key Technologies Used

1. **Computer Vision** – Tesla vehicles use **8 cameras**, **ultrasonic sensors**, and **radar (in older versions)** to perceive the environment.

2. **Neural Networks** – Deep learning models process video data to detect:
   - Vehicles
   - Pedestrians
   - Road signs and lane markings
   - Traffic lights and obstacles

3. **Sensor Fusion** – Combines data from multiple sensors to create a 360° real-time map of the vehicle's surroundings.

4. **AI Chip (Tesla FSD Chip)** – Custom-designed hardware that enables on-board neural network processing.

5. **Over-the-Air (OTA) Updates** – Tesla continuously improves Autopilot capabilities using OTA software updates.

Main Features

1. **Traffic-Aware Cruise Control (TACC):** Adjusts speed according to surrounding traffic.

2. **Autosteer:** Keeps the car centered in its lane.

3. **Auto Lane Change:** Allows the vehicle to switch lanes automatically with driver confirmation.

4. **Navigate on Autopilot:** Suggests and performs lane changes, highway exits, and merges.

5. **Autopark:** Assists in parallel and perpendicular parking.

6. **Summon:** Moves the car in and out of parking spaces remotely.

7. **Full Self-Driving (FSD) Beta:** Currently in testing—aims to enable full urban and highway autonomy.

- **Advantages**

➢ Reduced driver fatigue on highways
➢ Improved safety through collision avoidance
➢ Constant software improvements
➢ Data-driven learning from Tesla’s global fleet

- **Challenges & Ethical Concerns**

➢ **Driver complacency** due to overreliance
➢ **Accidents and liability** during Autopilot use
➢ **Regulatory and legal issues**
➢ **Ethical decision-making** in unavoidable crash scenarios
➢ **Data privacy** and constant video capture

AI Algorithms Involved

- ➢ **Convolutional Neural Networks (CNNs)** for visual perception
- ➢ **Reinforcement Learning** for decision-making and planning
- ➢ **Sensor fusion and path planning algorithms** for navigation

Robinhood’s AI-Driven Fraud Detection

- **Robinhood**, a popular U.S.-based stock trading and investment platform, uses **Artificial Intelligence (AI)** and **Machine Learning (ML)** to **detect and prevent fraudulent activities** in real time. Its AI-driven system safeguards millions of users from scams, identity theft, and financial fraud.

- **Overview**

- Robinhood processes millions of transactions daily. To maintain security, it employs **AI-powered fraud detection models** that automatically identify suspicious user behavior and prevent potential misuse—such as unauthorized account access, money laundering, and fake identity creation.

Key Technologies Used

- **Machine Learning Algorithms**
  - Detect unusual trading patterns, login behavior, and transaction anomalies.
  - Models are trained on historical fraud data and user activity patterns.

- **Natural Language Processing (NLP)**
  - Analyzes communication patterns (like customer support chats or emails) to detect phishing or social engineering attempts.

- **Graph Neural Networks (GNNs)** *(recently emerging)*
  - Identify relationships between accounts (shared devices, IPs, or payment methods).
  - Detect **fraud rings** or **coordinated attacks**.

- **Behavioral Biometrics**
  - Tracks typing speed, mouse movement, and mobile gestures to verify legitimate users.

- **Anomaly Detection Models**
  - Unsupervised learning algorithms (e.g., Isolation Forest, Autoencoders) identify patterns that deviate from normal activity.

AI Workflow

- **1. Data Collection**
- User account data, device fingerprints, IP address, login history, transaction logs, and behavioral data are collected.

- **2. Feature Engineering**
- Convert user actions (e.g., trade frequency, transaction size, login time) into measurable numerical features.

- **3. Model Training**
- ML algorithms like **Random Forest**, **Gradient Boosted Trees**, or **Neural Networks** are trained using labeled datasets of legitimate and fraudulent activity.

- **4. Real-Time Detection**
- Each new action (login, trade, withdrawal) is scored by the AI model for fraud probability.

- **5. Risk Scoring and Action**
- Low-risk: Proceed
- Medium-risk: Request additional verification (OTP, ID check)
- High-risk: Temporarily block or report the transaction

- **Advantages**

➢ Real-time fraud detection and prevention
➢ Reduced manual review workload
➢ Continuous learning from new fraud patterns
➢ Protects users and improves trust

-

- **Challenges**

➢ **Evolving fraud tactics** require frequent model retraining
➢ **Data privacy and ethical AI** concerns
➢ **False positives** may block legitimate users
➢ **Scalability** as transaction volume grows

IBM Watson

- **IBM Watson** is a powerful **AI platform** developed by **IBM** that uses **Natural Language Processing (NLP)**, **Machine Learning (ML)**, and **Data Analytics** to understand, reason, and learn from large volumes of structured and unstructured data.

- It became widely known after defeating human champions on the quiz show **Jeopardy!** in 2011 — showcasing its advanced language understanding and reasoning capabilities.

- **Overview**

- IBM Watson is designed to **analyze data, extract insights, answer questions**, and **assist in decision-making** across various domains such as **healthcare, finance, education, and customer service**.

Core Technologies Used

- **Natural Language Processing (NLP)**
  - Understands and interprets human language — questions, documents, and conversations.
  - Performs entity recognition, sentiment analysis, and question answering.
- **Machine Learning (ML)**
  - Learns from data patterns to make predictions or recommendations.
  - Continuously improves through feedback.
- **Deep Learning**
  - Uses neural networks for advanced text, speech, and image analysis.
- **Cognitive Computing**
  - Mimics human thought processes — understanding context, ambiguity, and intent.
- **Big Data Analytics**
  - Processes massive data volumes to extract meaningful insights.

Major Components / Services

Component	Description
Watson Assistant	AI-powered chatbot for conversational interfaces.
Watson Discovery	Intelligent document search and knowledge mining.
Watson Studio	Platform for building, training, and deploying ML models.
Watson Natural Language Understanding (NLU)	Text analysis — sentiment, emotion, and entity detection.
Watson Speech to Text / Text to Speech	Converts voice to text and vice versa.
Watson Machine Learning (WML)	Model management and deployment in the cloud.
Watson Knowledge Catalog	Data governance and metadata management.

Applications

1. **Healthcare** – Diagnosing diseases, recommending treatments, analyzing medical literature.
   Example: Watson for Oncology suggests cancer treatment options.
2. **Finance** – Fraud detection, risk analysis, and customer engagement.
3. **Customer Support** – AI chatbots and virtual agents for 24/7 assistance.
4. **Education** – Personalized learning experiences.
5. **Business Intelligence** – Data-driven insights and automated reporting.

Example: Watson in Healthcare

- **Watson for Oncology (in partnership with Memorial Sloan Kettering Cancer Center):**
- Analyzes patient health records and medical literature.
- Suggests personalized cancer treatment options.
- Helps doctors make evidence-based decisions faster.

- **Advantages**

➢ Understands natural human language
➢ Provides explainable AI insights
➢ Processes vast amounts of unstructured data
➢ Integrates easily with cloud-based services (IBM Cloud)
➢ Supports multiple industries and use cases

-

- **Limitations**

➢ High implementation and training costs
➢ Requires domain-specific data for accuracy
➢ Some models underperformed in real-world clinical applications
➢ Complex setup for small organizations

Alexa

- **Amazon Alexa** is an **AI-powered virtual assistant** developed by **Amazon** that uses **Natural Language Processing (NLP)**, **Machine Learning (ML)**, and **Speech Recognition** to understand and respond to user voice commands.

- It powers devices like **Amazon Echo**, **Echo Dot**, **Fire TV**, and many **smart home systems**, making everyday tasks easier through voice interaction.

- **Overview**

- Alexa acts as a **voice-controlled intelligent assistant** that can answer questions, play music, control smart devices, make calls, set reminders, and more. It continuously learns from user interactions to improve responses and personalize experiences.

Core Technologies Used

- **Automatic Speech Recognition (ASR)**
  - Converts the user’s **spoken words** into text.
- **Natural Language Understanding (NLU)**
  - Interprets the meaning, **intent**, and **context** behind user commands. *(e.g., “Turn on the lights” → intent = smart home control)*
- **Text-to-Speech (TTS)**
  - Converts Alexa’s textual responses into **spoken voice output**.
- **Machine Learning (ML)**
  - Continuously improves Alexa’s understanding of language, accents, and user preferences.
- **Cloud Computing (AWS)**
  - Alexa’s brain resides in the **Amazon Web Services (AWS)** cloud, which processes requests and returns results.
- **Skill Kit (ASK)**
  - Developers can create custom “**skills**” (apps) that add new capabilities to Alexa.

Applications

Category	Examples
Smart Home	Control lights, fans, thermostats, and security cameras
Entertainment	Play songs, audiobooks, podcasts
Communication	Make voice calls, send messages, drop-in feature
Productivity	Set alarms, reminders, manage calendar
Shopping	Order products directly from Amazon
Education	Answer factual questions, define words, assist in learning

- **Advantages**

➢ Hands-free operation and convenience
➢ Expands functionality with thousands of third-party “skills”
➢ Continuous learning and personalization
➢ Integrates with IoT and smart home ecosystems
➢ Multilingual support

-

- **Challenges & Concerns**

➢ **Privacy issues** (voice data stored on cloud)
➢ **Accidental activations** (“false wake words”)
➢ **Dependence on internet connectivity**
➢ **Security risks** from third-party skill misuse

Siri

- **Siri** is Apple's **AI-powered virtual assistant**, integrated into **iPhones, iPads, Macs, Apple Watches, and HomePods**.
-
  It uses **speech recognition**, **natural language processing (NLP)**, and **machine learning (ML)** to understand spoken commands, answer questions, and perform tasks for users — all with a focus on **privacy and personalization**.

- **Overview**

- Launched in **2011**, Siri was one of the first mainstream voice assistants.
  It helps users perform hands-free actions like sending messages, setting reminders, playing music, navigating routes, or answering queries using voice commands such as:

- "Hey Siri, what's the weather today?"

Core Technologies Behind Siri

- **Automatic Speech Recognition (ASR)**
    - Converts user speech into text.
    - Handles different accents, pronunciations, and noise environments.

- **Natural Language Understanding (NLU)**
    - Interprets user intent and extracts key information (like contact names, places, or actions).
    - Example: “Call Mom” → Action = Call, Target = Mom.

- **Machine Learning (ML)**
    - Learns from user behavior to personalize responses.
    - Improves accuracy with usage patterns.

- **Knowledge Graph + Search Integration**
    - Uses Apple’s and third-party knowledge bases (like WolframAlpha, Maps, Wikipedia) to answer factual questions.

- **Text-to-Speech (TTS)**
    - Converts Siri’s text responses into natural-sounding speech.

- **On-Device and Cloud AI**
    - Simple tasks (like opening apps) are processed **on-device** for speed and privacy.
    - Complex queries (like web searches) are sent securely to Apple’s servers.

Applications of Siri

Category	Examples
Communication	Make calls, send messages, read notifications
Productivity	Set reminders, schedule meetings, manage calendar
Entertainment	Play music, podcasts, or control Apple TV
Navigation	Provide directions via Apple Maps
Smart Home (HomeKit)	Control lights, locks, and thermostats
Information Retrieval	Weather, sports scores, general knowledge

- **Advantages**

➢ Hands-free convenience
➢ Deep integration with Apple ecosystem
➢ High privacy — minimal data stored in the cloud
➢ Personalized responses based on user habits
➢ Continuous learning and updates

- **Limitations**

➢ Limited integration with non-Apple apps and devices
➢ Context understanding still basic compared to newer LLMs (like ChatGPT)
➢ Requires internet for most functions
➢ Regional language support still growing

Eliza

- **ELIZA** is one of the earliest examples of **Artificial Intelligence in natural language processing**, developed in **1966** by **Joseph Weizenbaum** at the **MIT Artificial Intelligence Laboratory**.
- It is considered the **first chatbot** and an early demonstration of **human-computer conversation** through text.
-
- **Overview**
- ELIZA was designed to simulate conversation by using **pattern matching** and **substitution rules** to respond to user input.
- One of its most famous scripts was **“DOCTOR”**, which mimicked a **Rogerian psychotherapist** — responding with open-ended questions or reflective statements.

- Example conversation:
- User: I'm feeling sad today.
- ELIZA: Why do you think you are feeling sad?
- Even though ELIZA had **no real understanding** of language, many users felt like they were genuinely conversing with an empathetic listener — a powerful early example of AI’s psychological impact.

Key Working Principles

- **Pattern Matching**
  - ELIZA scans the input for keywords (e.g., *mother, father, happy, sad*).
  - It matches these patterns against a set of predefined rules.
- **Decomposition Rules**
  - Breaks down user input into components.
  - Example: “I am X” → “Why are you X?”
- **Reassembly Rules**
  - Constructs a reply by rearranging the input text and adding canned phrases.
- **Script-Based Design**
  - Each behavior or personality (like the “DOCTOR”) was defined by a **script** containing the rules.

- **Features**

➢ Text-based interaction
➢ No memory of previous conversations
➢ No true understanding — just syntactic manipulation
➢ Modular “scripts” allowed new personalities or topics

-

- **Advantages**

➢ First successful demonstration of natural language conversation
➢ Inspired later chatbot systems (e.g., PARRY, ALICE, Siri, ChatGPT)
➢ Showed human tendency to attribute understanding to machines

-

- **Limitations**

➢ No semantic understanding (didn’t actually “know” anything)
➢ Could not maintain conversation context
➢ Repetitive and predictable after a few exchanges
➢ Dependent on predefined scripts and keywords

Legacy & Influence

- ➢ **ELIZA Effect:** The human tendency to assume intelligence or empathy in machines that simply simulate language.
- ➢ Inspired decades of AI development in natural language understanding and conversational agents.
- ➢ Forms the foundation for modern **AI assistants** like **Alexa**, **Siri**, and **ChatGPT**.

ELIZA’s Technological Evolution

Generation	Example	Description
1st Gen (1960s)	**ELIZA**	Rule-based pattern matching
2nd Gen (1970s–2000s)	**PARRY, ALICE**	More advanced rules and keyword logic
3rd Gen (2010s–Present)	**Siri, Alexa, ChatGPT**	Deep learning, NLP, context awareness

ChatGPT

- **ChatGPT** (short for *Chat Generative Pre-trained Transformer*) is an **AI-powered conversational agent** developed by **OpenAI**.
- It uses advanced **Natural Language Processing (NLP)** and **Deep Learning** techniques to understand, generate, and interact in human-like language.
- It represents one of the most sophisticated **Large Language Models (LLMs)** ever built.
- **Overview**
- ChatGPT is based on the **GPT (Generative Pre-trained Transformer)** architecture — a type of neural network designed to process and generate natural language text.
- It can **answer questions, summarize text, write essays, generate code, translate languages, and carry on conversations** with remarkable fluency.

Core Technologies Behind ChatGPT

- **Transformer Architecture**
  - The foundation of GPT models.
  - Uses **self-attention mechanisms** to understand relationships between words across long contexts.
- **Pre-training (Generative)**
  - Trained on vast amounts of text from the internet to learn grammar, facts, reasoning, and language patterns.
- **Fine-tuning**
  - Refined on curated datasets with human feedback using **Reinforcement Learning from Human Feedback (RLHF)** to make responses helpful and safe.
- **Tokenization**
  - Text is broken into small pieces (*tokens*) before processing and reassembled into coherent output.
- **Natural Language Understanding (NLU) + Generation (NLG)**
  - Understands the intent of the user's input (NLU) and generates context-aware responses (NLG).

- **Training Process**

1. **Data Collection:** Massive datasets of text (books, articles, web pages).
2. **Pre-training:** Learns language structure and general knowledge.
3. **Fine-tuning:** Adjusted for human-like conversation.
4. **Reinforcement Learning (RLHF):** Trained using human feedback to refine tone, accuracy, and safety.

Capabilities

Category	Examples
Conversational AI	Engage in natural conversations
Content Creation	Essays, reports, poetry, stories
Programming Help	Debug, explain, and generate code
Education	Explain complex topics, tutoring
Research Assistance	Summarize papers, suggest literature
Translation	Convert text between multiple languages
Creative Work	Idea generation, design concepts

- **Advantages**

➢ Understands and generates human-like text
➢ Learns context from conversation history
➢ Supports multiple domains and languages
➢ Enhances productivity and creativity
➢ Continuously improved with newer versions (GPT-3 → GPT-4 → GPT-5)

-

- **Limitations**

➢ May produce **incorrect or biased information**
➢ Lacks **real understanding** (patterns, not consciousness)
➢ Cannot access **real-time data** (unless connected to the web)
➢ May generate **plausible but false** or **nonsensical** answers

- **Ethical & Safety Considerations**

➢ Ensures **responsible AI use** through content moderation
➢ Uses **alignment techniques** to avoid harmful outputs
➢ Strives for **transparency, privacy, and fairness**

-

- **Future Directions**

➢ Integration with **real-time web access and databases**
➢ **Multimodal AI** – understanding **text, images, audio, and video**
➢ Improved **reasoning and explainability**
➢ Domain-specific and **personalized AI assistants**

Key Versions

Model	Year	Description
GPT-1	2018	Proof of concept (117M parameters)
GPT-2	2019	Fluent text generation (1.5B parameters)
GPT-3	2020	Massive model (175B parameters)
GPT-4	2023	Multimodal reasoning (text + images)
GPT-5	2025	Advanced reasoning, contextual memory, multimodal intelligence