AI CASE STUDIES

CHATBOTS

"computer programs designed to simulate human conversation through text or voice."

Use cases:

1. customer services 2. lead generations 3. PA
4. ↑user experience 5. ↑efficiency

Core Components:

1> user input: text typed by user

2> natural language processing (NLP): gives computer ability to understand, translate, generate human language

3> natural language understanding (NLU)

- Intent Recognition: proc of figuring out what user wants (user's intent)
- Entity Extraction: proc of extracting specific pieces of info relevant to that intent, called entities

4> Dialouge Management

- State tracking: keeps track of convo history & context.

5> Response Selection / Action Determination

based on intent, entity and dialogue state, this

module decides what the chatbot should do next - ask a follow-up question, retrieve info or an API call etc.

6> Knowledge Base / Database Integeration:
(external data about user and other things)
chatbot extracts these through API calls.

7> Natural Language Generation (NLG)
when chatbot knows what to say, NLG is responsible for crafting the actual response in natural-sounding language. (human like)

Types of Chatbots

1. Rule-Based
- follow predefined (IF con THEN action (response))
rules, keywords and decision tree.
- working: bot matches keywords or pattern from user input to database
eg FAQ bot, if user types shipping cost, bot might have rule that recognize shipping & cost and responds with 'Standard Shipping Cost $5'

2. AI powered (ML / DL)
- they use ML to learn, NLP to understand NLG to respond.

eg. The ChatGPT, Copilot etc

3. Generative - Based
- can generate novel responses on fly. They learn pattern and language from massive datasets and construct unique replies.
- eg. A customer support bot that can understand nuanced questions like 'My delivery for order #12345 is late, what should I do?' and provide a context-aware response automatically after making a API call itself

4. Retrieval - Based
- these use NLP to understand user ~~itp~~ intent and then retrieve the most appropriate pre-written answer from large library of potential answers.
- X generate new text but select best-fit.

eg. Airtel thanks app or and other type where you get same (exact) response on same (similar query) question

5. Button Based
- these are bot where it doesn't take totally free user input, but gives user button

damentals of Artificial Int...

type options for prompting, so the user's prompt is exactly what they want.
e.g. Many FAQs and other site bots.

Building Blocks And Concepts (Coding Perspective)

- Programming Languages: python (lib support)
- Frameworks / Platforms: Rasa, Google DialougeFlow, Microsoft Bot Framework
- Data Collection and Annotation: IDK LEAVE
- API Integration: writing code to make API request to backend services etc.
- Evaluation: : measure performance
- NLU evaluation: Accuracy, Precision, F1-score for intent classification & entity recogn
- Overall Dialouge evaluation: Task completion rate user satisfaction rating
- NLG evaluation: check response generated

→ Chat generative Pre-trained Transformer.

ChatGPT

- Sophisticated large language model (LLM) developed by OpenAI
- Core Tech: LLM and transformer Architecture
  - LLM: neural network with billions of parameters, trained on massive datasets of text and code.
  - Transformer Architecture: specific neural network architecture by Google in 2017, which revolutionized natural language processing.

    This uses Attention Mechanism:

    ↓

    models ability to weight the importance of different words in i/p sequence.

How it work?

1. Input: you provide txt / audio (voice) / doc
2. Tokenization: input text is broken down into smaller units called token (it can be words, parts of words, punctuations etc)
3. Neural Network Processing: token are broken into numeric representation and fed into the Transfer neural network.

   It processes it to get most appropriate sequence of tokens.

4. Output Generation: generates an o/p sequence of tokens, one by one, based on probabilities. these tokens again → human credeable texts.

Key Capabilities and Use Cases

1. Text Generation : doc generation
2. Code Generation & Debugging : img/video generation
3. Summarization : understanding.
4. Translation : notes making
5. Creative writing : email writing etc.

Limitations and Challenges

1. Hallucinations : can generate true sounding but factually incorrect data.
2. Bias : if training dataset if baised then outcome can be biased.
3. Bugging & lags : can lag and bug.
4. Ethical concerns : issues around misinformation, job displacement, copyright
5. Misuse
6. No control

Recommendation Algorithm

- an information filtering system predicting user preferences/ratings.
- goal is to suggest relevant items (personalized)

Need?

- Solve info overload
- ↑ user experience
- ↑ engagement, reach
- ↑ sales, content consumption
- enable discovery of new items

Core types

1. Content-Based Filtering.

- recommends items similar to those user ~~interacted~~ (liked) in past.
- uses item features + user profile.
- user profile: build from explicit (ratings) or implit (views, purchased, interacted item user +tively
- items features: each item is described by set of attributes (e.g. movie: genre, actors, director, etc)
- Matching: sys recommends items whose features match the user's profile
- Adv: nice; simple; addictive
- Dis: limited space of recommendation; req data about user likes...

2. Collaborative Filtering (CF)
- based on idea that users who agreed in past will agree in future.
- It finds patterns between user and item interaction

types

User-Based	Item based
finds similar users → recommend items liked by them	finds similar items liked by the user
eg A & B like x, y A like z also so B gets recommend z	eg Users like P & Q together → when A like P, Q is recommended.

3. Hybrid Methods
combine content-base + collaborative filtering + deep learning

Evaluation Matrics
how do we know recommendation sys is good
1. precision and recall : (how many recmd are correct)
2. Diversity → variety of recmds
3. Coverage → items explored from all db
4 F1-Score

Applications

1. E-commerce 2. Streaming (Netflix, yt)
3. Social media 4. News 5. food delivery

Challenges

- data sparsity (few interactions per user)
- cold-start (no history for new user)
- Scalability (millions of users and items)
- Serendipity (hard to suggest unexpected but relevant items)
- Shilling attacks (malicious attacks)
- Echo-chamber (overpersonalization limits diversity)

Digital Voice Assitance

- Software agent performing tasks via spoken commands
- uses AI techs like
  - Automatic Speech Recognition (speech → text)
  - Natural Language Understanding (entity and intent identification)
  - Dialouge Management (convo history and state)
  - Task Execution (actual action to respond)
  - Natural Language Generation (get text, human like response)
  - Text to Speech (text to speech)

Workflow

1. activation (wake word (hey siri) / button)
2. Audio capture
3. ASR → text (Automatic Speech Recognition)
4. NLU → intent + slot (Natural language understanding)
5. DM updates state (dialouge management)
6. Action determination
7. Action execution
8. NLG → text response (natural language generation)
9. TTS → text to speech (text to speech)
10. playback to user.

AI/ ML concepts used things like

ASR NLU DM NLG TTS

Examples: Alexa, Siri, google gemini, cortana etc.

Challenges

- contextual understanding : (sarcasm, implicit intent)
- multimode input : voice + gesture + vision
- privacy (Security · robustness · complexity
- ↓ personalization (more tailored response required)

Virtual Face Filters

- real time digital overlays on faces in photo/video
- detect face → overlays (CGI → blends seamlessly
- AR, Snap filters, BG12 etc

Core Techs

- Face Detection: locate face(s) in the frame
- Facial landmark detection: key points on face
  (68 total ⇒ (eyes, nose, mouth, jawline etc)
- AR overlay: Align CGI to face geometry
- Real time processing: smooth align with updates

Working

1. input capture (camera stream)
2. face detection
3. landmark extraction
4. 3D pose estimation (pitch, yaw, roll)
5. Filter application (CGI etc)
6. Rendering & output (video + filter combined)

AI / ML concepts

- Computer vision: base field to interpret visuals
- Deep Learning (CNN): detection + landmarks
- Training Data: labelled faces with CGI
- Pose estimation (orientation of face in 3D)

- real-time optimization

Application

- social media - filters - AR - amazon etc

Challenges

- robustness - privacy - misuse - unrealism
- filters changing as per mood, interactions, env, words
- custom filter creation