Task-03

Group 27

Nizbath Ahsan : 23341119/20101227
Saiyeda Sabiha: 20201022
Mahdi Hossain: 20301194

15. Chatbots and dialogue systems

In addition to highlighting the value of language and communication in human contact, the talk addresses how intelligently inanimate objects are depicted in literature. It also describes the two types of conversational agents: chatbots and task-oriented dialogue agents.

15.1 Properties of human conversation

Turns: Computers must be able to grasp the turn structure in a dialogue in order to know when to talk, pause, and restart, which might be difficult owing to background noise and pauses.

Speech acts: Speech acts are the activities that speakers take when conversing, and the four main types of speech acts that indicate the speaker's purpose are constatives, directions, commissives, and acknowledgments.

Grounding: Humans constantly use grounding to ensure that participants in a conversation have something in common and to improve communication by ensuring that the listener has understood the speaker's message.

Sub Dialogues and dialogue structure: Adjacency pairs, which consist of a first and second pair section and can be split by a side sequence or sub-dialogue, are how conversations are organized. These pairs frequently include predefined replies, such as in questions and answers or offers and acceptances.

Initiative: Mixed initiative is a typical occurrence in human-to-human communication, but conversational systems find it challenging to display, and system-initiative designs may be grating. User-initiative systems are preferable because they let users ask questions and get replies.

Inference and implicature: In dialogue, the speaker is expected to draw certain conclusions beyond what the listener says, and Grice's aphorisms serve as guidelines for evaluating dialogue statements, generating twists, turns, acts, justifications, and dialogue structures specific to the dialogue system. , initiative and hidden, but research actively seeks solutions.

15.2 chatbots

The most basic sort of conversation system that imitates casual human-human interaction is the chatbot. Like Facebook's Blender-Bot and Microsoft's XiaoIce system, they are entertaining and capable of holding lengthy discussions.

15.2.1 Rule based chatbots: ELIZA and PARRY: ELIZA was a chatbot designed to mimic a Rogerian psychotherapist who repeats a patient's statements to get more information based on an algorithm. PARRY was the first system to pass the Turing test, convince psychiatrists that they were talking to a real person during an interview, and included a model of mental states to study schizophrenia.

15.2.2 Corpus-based chat gpt: This topic defines Corpus Chatgpt bot which is basically defined as human-to-human conversations instead of using hand-built rules by requiring intensive data with a collection of millions or even billions of words for training. The majority of chatbots that rely on a corpus of data for their responses generate replies to a user's input by either retrieving relevant information from the corpus using information retrieval techniques or by generating responses using a language model or encoder-decoder model based on the contextual cues in the dialogue. So two types of methods are used retrieval method and generation methods for a corpus-based chat get System.

Retrieval system: It mostly generates a single response turn for the conversations and for that, it is called a response generation system. In retrieval method is to respond according to the turn of the user and retrieve the appropriate turn form a corpus-based system.

Generation system: Thinking of response generation as an encoder-decoder task—transducing from the user's prior turn to the system's turn—is another way to use a corpus to generate dialogue which is basically called a generation method.

15.2.3 Hybrid Architecture: The architecture of chatbots can be of different types mentioning rule based and neural/corpus architecture and frame based structure.In this architecture bit responses are generated using bot generators like fine tuned neural language, ruled based generators.

Fine Tuned Neural Language: In this method a unique GPT-2 language model that improves via practice to respond to queries by paraphrasing information from Wikipedia.

Ruled based response generator: These generators use regexes and sentiment classifiers to classify user responses and handwritten templates to generate bot utterances.

15.3 GUS: Simple Frame-based Dialogue Systems

Task-based dialogue systems help users complete specific tasks by providing prompts and directions. The GUS architecture is a simple framework for task-based dialogue, and modern task-based dialogue systems use frames and domain ontologies to help the system understand user sentences.

15.3.1 Control structure for frame-based dialogue: The frame-based dialogue system uses frames to store information, and it uses pre-defined question templates to skip questions that have already been answered. There are also condition-action rules attached to slots, which can automatically fill in related frames. The system must be able to determine which frame a given input is supposed to fill and switch control to that frame. The GUS architecture is a production rule system that dynamically switches control according to factors such as the user's input and some simple dialogue history. Once enough information is gathered, the system performs the necessary action and returns the result to the user.

15.3.2 Determining Domain, Intent, and Slot fillers in GUS: A frame-based system is able to understand what a user is saying by taking into account the context of the speech. This is done using rules based on context-free grammars and parsers, as well as more advanced rules if the system is equipped with machine learning. Machine learning is used to fill in slots automatically in a system that uses frame-based understanding.

15.3.3 Other components of frame-based dialogue: This chapter discusses how the ASR component in a dialogue system can take audio input and output transcribed words, and how it can be constrained based on the dialogue state. A restrictive grammar is a language model that is dependent on dialogue state, and it can be trained on specific questions or hand-written using finite-state or context-free grammars. Natural language generation modules in dialogue systems tend to use template-based generation, in which prespecified sentences or prompts are created by the designer, and can include variables that are filled in by the generator. Grounding can be added to templated responses to make them more natural. The rule-based GUS approach is common in industrial applications and can provide high precision and coverage if the domain is narrow and experts are available, but it can be expensive and slow to create, and may suffer from recall problems.

15.4 Dialogue State Architecture

The dialogue-state or belief-state architecture used by contemporary task-based dialogue systems has six components. In order to extract slot fillers, keep the dialogue in its current state, choose the system's response, and produce natural language output, four of these components deal with spoken language processing. The dialogue-state design, as opposed to the straightforward GUS systems, provides for a more sophisticated dialogue policy that can aid the system in determining when to respond to the user's queries, seek clarification, or offer ideas. Many dialogue-state systems are being developed in research labs, despite the fact that the majority of commercial systems use architectural hybrids.

15.4.1 Dialogue Acts

The interactive role of a turn or sentence in a conversation is represented by dialogue acts, which are used by dialogue-state systems. Act labels for various types of dialogue systems must be varied, and the tagset was created for a particular purpose. Examples were provided of dialogue acts and their filler words in a restaurant recommendation scenario where users inform or confirm information with the system.

15.4.2 Slots Filling

Through supervised semantic parsing, the tasks of slot-filling and domain/intent classification can be accomplished. On a training set that matches each sentence with the appropriate set of slots, domain, and intent, a sequence model is trained. For each token location over potential BIO tags, the architecture contains a contextual embedding model, feedforward layer, and softmax output. In a semi-supervised learning approach, machine learning-based systems are frequently bootstrapped from rule-based ones. Homonym dictionaries can be used to extract filler strings from the tags and normalize them. Rule-based systems may still have superior precision for handling difficult scenarios like negation even when classifiers can surpass them.

15.4.3 Dialogue State Tracking

Dialogue state tracking is difficult and requires taking into account both the user's previous remarks and their most recent statement. In either straightforward or more complex models, this is accomplished utilizing supervised classification and embeddings encoding for dialogue act interpretation and slot-filling.

15.4.4 Dialogue Policy

The dialogue policy determines the system's next course of action based on the user's declared set of slot-fillers, with reinforcement learning and slot accuracy rewarded and penalties for incorrect slots and pointless queries. It uses a neural classifier to estimate probabilities from neural representations of slot fillers and utterances.

To make sure they have understood the user's input correctly, dialogue systems employ confirmation and rejection strategies. Explicit confirmation involves the system asking a straight yes-or-no question, while implicit confirmation involves the system repeating the user's input in the following inquiry. Rejection may indicate a lack of comprehension, and progressive prompting may be used to elicit further information from the user if their input is rejected. Additional characteristics like confidence and error penalties are used in confirmation.

15.4.5 Natural Language Generation in the Dialogue-State Model

After the dialogue act has been chosen, natural language generation (NLG) in dialogue systems generates the user response. The two processes of NLG modeling are content planning and sentence realization. When there is a lack of training data, delexicalization is typically utilized to produce NLG algorithms, and encoder-decoder models are frequently employed to convert frames to delexicalized phrases. A major aspect is the ability to build NLG algorithms that are tailored to a certain discourse act, such generating explanation questions. Targeted clarification questions are provided by rules or classifiers.

15.5 Evaluating Dialogue Systems

Evaluation is essential for dialogue system design.

15.5.1 Evaluating Chatbots

<u>Adversal Evaluation:</u> In order to evaluate dialogue systems beyond straightforward word similarity metrics, a novel paradigm called adversarial evaluation was developed. The effectiveness of a response generating system is determined by how successfully it can deceive this evaluator, which includes training a "Turing-like" evaluator classifier to distinguish between human- and machine-generated responses.

<u>Acute Eval:</u> Participant assessment, in which a human assessor converses with the model and scores it on several conversational quality aspects, and observer evaluation, in which outside annotators read the text of a whole discussion and offer a high-level score, are two methods used to assess chatbots. Due to the wide range of possible dialogue outcomes, automatic evaluations like BLEU or ROUGE are not appropriate for chatbots because they do not correspond well with human judgments.

15.5.2 Evaluating Task Based Dialogue

The effectiveness of task-based dialogue systems can be assessed using a variety of techniques. Monitoring overall task completion and measuring user satisfaction through surveys are the two basic evaluation techniques. To gauge how well a system is performing, performance evaluation heuristics like optimizing task success and minimizing costs can be applied. The correctness of the entire solution can be used to judge how well a conversation system does a task, and efficiency costs, which can be measured as the total length of dialogue time, can show how well the system supports users. Another crucial factor to take into account is quality cost, which assesses characteristics of the interaction that affect how the user views the system.

15.6 Dialogue System Design

The study of dialogue systems adheres to user-centered design principles and is strongly related to Human-Computer Interaction (HCI). This entails researching users and tasks, creating simulations and prototypes with Wizard-of-Oz technologies, and putting the design through iterative user testing. System design requires an iterative design cycle with embedded user testing. It's also crucial to value sensitive design, which takes into account the advantages, disadvantages, and potential stakeholders of the final system. On the subject of conversational interface design, there are several excellent books accessible.

15.6.1 Ethical Issues in Dialogue System Design

This text explores ethical problems with conversational agent design, including privacy issues, the perpetuating of gender stereotypes, and safety risks when users rely on them for information in life-or-death situations. Examples of offensive language and prejudiced conversation system answers are given. The need of value-sensitive design and adhering to best practices are emphasized in the text as ways to reduce potential damages. To protect test subjects, researchers are urged to collaborate with institutional review boards.