knowledge

MDPI

Review

Do You Ever Get Off Track in a Conversation? The Conversational System's Anatomy and Evaluation Metrics

Sargam Yadav [1,†] and Abhishek Kaushik [2,*,†]

[1] School of Computing, Dublin Business School, D02 WC04 Dublin, Ireland; sargam.yadav@edhec.com
[2] CeADAR, School of Computer Science, University College Dublin, D04 V2N9 Dublin, Ireland
[*] Correspondence: abhishek.kaushik@ucd.ie
[†] These authors contributed equally to this work.

Abstract: Conversational systems are now applicable to almost every business domain. Evaluation is an important step in the creation of dialog systems so that they may be readily tested and prototyped. There is no universally agreed upon metric for evaluating all dialog systems. Human evaluation, which is not computerized, is now the most effective and complete evaluation approach. Data gathering and analysis are evaluation activities that need human intervention. In this work, we address the many types of dialog systems and the assessment methods that may be used with them. The benefits and drawbacks of each sort of evaluation approach are also explored, which could better help us understand the expectations associated with developing an automated evaluation system. The objective of this study is to investigate conversational agents, their design approaches and evaluation metrics. This approach can help us to better understand the overall process of dialog system development, and future possibilities to enhance user experience. Because human assessment is costly and time consuming, we emphasize the need of having a generally recognized and automated evaluation model for conversational systems, which may significantly minimize the amount of time required for analysis.

Keywords: dialogue system; human-centered computing; chatbots; user simulation; user evaluation; evaluation metrics

check for updates

Citation: Yadav, S.; Kaushik, A. Do You Ever Get Off Track in a Conversation? The Conversational System's Anatomy and Evaluation Metrics. *Knowledge* **2022**, *2*, 55–87. https://doi.org/10.3390/knowledge2010004

Academic Editor: Jerry Chun-Wei Lin

Received: 27 October 2021
Accepted: 23 December 2021
Published: 14 January 2022

1. Introduction

A dialog system is any conversational agent whose purpose is communicate with a human through a text or voice interface. Dialog systems are extensively used in a number of industries for a number of different sectors such as customer service [1], information seeking [2], conversational therapy [3], diagnosis [4], and more. Dialog systems are beneficial as they can support multiple users without any decrement in performance. They are also robust and easily adaptable. Providing an automated response reduces the chances of human error. Users may prefer to interact with a dialog system as it does not judge or misuse the information disclosed by the user [5]. Dialog systems also eliminate the factor of human frustration in repeated use of the system. Thus, they prove beneficial for both the business and the end user in terms of increasing full-time availability. Rule-based dialog systems such as ALICE [6] and ELIZA [7] make of use of matching pairs to output dialog. Both machine learning and deep learning tools are also widely used in natural language processing to implement such systems [8,9]. Dialog systems are considered valuable because they are approachable, improve customer experience, manage large number of customers, and reduce operational costs. Companies are investing a lot into dialog systems because of this reason. Dialog systems are available round the clock to improve overall customer experience. People also usually prefer to interact with dialog systems over humans [10]. McTear et al. [11] explain that dialog systems usually construct the dialog in turns. Each turn can be defined by one or more responses from each user. Two consecutive turns between each user can be referred to as an exchange, and multiple exchanges could be referred to as a dialog. Each turn taken by a user contribute a part to the dialog. A turn can consist of a single word, as well as multiple sentences. Depending on the task at hand, the dialogues may be short or long. For example, shorter dialogues may be used with mobile assistants, longer dialogues may be needed where a lot of information is required to be verified, such as travel management and financial agents.

Some studies provide a broad classification of dialog systems [12] as task-oriented dialog systems and non-task oriented dialog systems. Task-oriented dialog systems assist the user in completing a task or achieving a goal. Conversational search-based systems [13], which provide an agent to assist in search tasks, are a subcategory of task-oriented systems. Non-task oriented dialog systems emulate a more human-like conversational flow. Question-answering systems are trained on a large corpus of domain-specific knowledge so that they can provide responses [2]. Task-oriented dialog systems are employed by businesses for purposes such as interactive self-service, automatic diagnosis [14], scheduling appointments, ordering food, technical support, educational help, etc. Hoy [15] mention popular examples of widely available task-oriented virtual assistants such as Apple Siri [16], Microsoft Cortana [17], Alexa [18], and more. Conversational systems can aim to answer user queries on websites or carry out 'chats' with the users. The focus rests on providing a natural conversational pattern so the system appears human-like. Kaushik et al. [19] introduce a conversational search system with a dialog agent that assists users in complex information seeking processes. QA systems can be used for tasks such as answering questions about university admissions [20], quiz generation [21], customer service [22], and more. Search-based conversational systems provide assistance to the user in completing a search task.

Figure 1 shows classification of dialog systems. The categorization is done on several features of the system such as:

1. Interaction mode: The dialog system could receive input from the user in the form of text or voice.
2. Design approach: The system may be modeled to provide answers about a particular topic (closed domain) or a range of topics (open-domain).
3. Search-based: Search-based systems assist the user in fulfilling their information needs.
4. Goals: The system may be designed for task completion (task-oriented), or carrying on conversations (non-task oriented).
5. Knowledge domain: The response may be selected by following a pre-defined rule (rule-based), retrieving pre-defined responses (retrieval based), or generating new responses entirely (generative models).

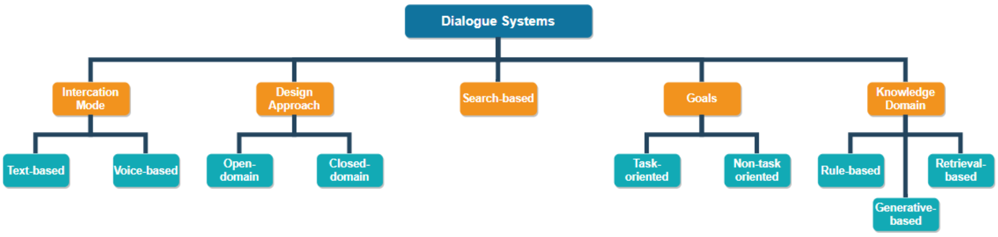


Figure 1. Classification of Dialog Systems.

Irrespective of the type of dialog system, the most important step in its development is evaluation. Evaluation provides feedback on factors of the system such as success rate, user satisfaction, contextual validity, etc. thus allowing us to grow and develop a better system to provide adaptability. Evaluation of dialog systems is a crucial step in their development, deployment, and improvement. There is no standard way to define the efficiency of a dialog system. As discussed in the previous paragraph, task-oriented dialog systems may score efficiency by successful completion of tasks. Non-task oriented systems aim to provide a human-like conversations, so the naturalness of dialogues may be prioritised. A reliable factor to measure the efficiency of a dialog system is user satisfaction. A system may perform well on empirical evaluation, but might ultimately not meet the user's expectations. Higher user satisfaction will result in higher acceptance and use of the system. The metrics for evaluating dialog systems also vary based on different application domains. Chatbots that provide services such as reservations, bookings, etc. can easily be evaluated with quantitative metrics such as task success rate [12], BLEU [23], and ROUGE [24]. Conversational agents, such as therapy bots, may require human evaluation [3].

A broad classification of evaluation metrics is empirical evaluation metrics and usability-based evaluation metrics. Empirical evaluation metrics are quantitative and calculated on factors such as correctness of the response based on a ground truth response [25]. Examples of empirical evaluation metrics are provided by Gunasekara et al. [26] and include Success rate, Precision, Recall, BLEU [23], ROUGE [24], etc. Usability metrics, on the other hand calculate the efficiency of the system based on user satisfaction. Hara et al. [27] propose an estimation method of user satisfaction for a spoken dialog system. The model was trained on a huge dataset and user evaluation was conducted in a real environment. The model achieved a classification accuracy of 94.7%. Yang et al. [28] make use of collaborative filtering to predict user satisfaction in spoken dialog systems. The evaluation process varies in accordance with the type of dialogue system. Task oriented systems can be evaluated with empirical metrics. QA systems can also be evaluated with quantitative metrics such as precision and recall. Conversational search systems require both qualitative and quantitative evaluation methods. This study aims to discuss the various methods available for evaluation of dialog systems. We discuss the various criteria that are considered for evaluation and an approach for automation of the evaluation process.

2. Motivation

The motivation of this study is to highlight evaluation methods for dialogue systems as they vary according to their task. As there is no widely accepted evaluation metric, the evaluation task is designed specially for the particular projects. This is a time-consuming process and the need for a well-designed evaluation metric which generalizes well to unknown systems is apparent. As mentioned in the previous section, evaluation is categorized into empirical methods and usability metrics. There is a significant trade-off between the application of empirical metrics and the metrics based on user satisfaction. Empirical metrics are quick and easy to use, with very little human effort required. However, they do not provide an accurate estimation of human judgement of the system. Empirical evaluation also involves training large models on large datasets, thus increasing the computational use and costs. User evaluation, on the contrary, provides a better idea of the user satisfaction with the system. However, it is very time-intensive and costly. Evaluation of task-oriented chatbots can be performed using empirical metrics and human evaluation. Empirical evaluation methods are usually automated and do not require human intervention. Only an initial evaluation script is required. However, to be able to provide an accurate picture of the performance of the chatbot, the empirical metric must provide results that correlate strongly to human judgement. This has shown not be the case for several metrics, thus greatly impacting their usability.

Human evaluation is a more reliable alternative to empirical methods. However, there are certain other shortcomings with human evaluation. These include the costs associated with obtaining human judgements and lack of a standardized protocol for human evaluation. The development of datasets for benchmark evaluation and analysis of the evaluation are tasks that depend on human effort. These responses can then be used for comparative analysis with model-generated responses. Manually annotating responses takes up a lot of time and human effort, thereby increasing costs. It may also be difficult to find specialists for the field in which the dialog system is applicable. The lack of standardization makes it difficult to compare different systems. There is also a gap between the usability of empirical evaluation metrics available for task-oriented and conversational systems. An evaluation metric that might work well for task oriented systems may not translate well for conversational agents [29,30]. Empirical evaluation metrics provide an objective and quantitative measure of performance of dialog systems. They represent the various functions of a system through mathematical equations. They are straightforward and easy to implement, but do not approximate well to human judgements [25]. Human evaluation provides a more subjective interpretation of the system's performance. The evaluation is carried out from the user's perspective by collecting responses about the various aspects of the dialogue system. In this paper, we explore the options in evaluation methods which do not require too much human interaction but provide a better picture of the system's performance. This approach attempts to find some middle ground which satisfies both evaluation types.

For this study, Google Scholar was used to perform searches of relevant keywords. The following keywords were searched: Dialogue systems, Evaluation metrics, Discourse models, Conversational AI, Chatbots, User evaluation dialog systems, Task-oriented dialog system, Search-based dialog system, Conversational dialog system, Dialog metric, Conversational search, Human centered computing. The citations for the top five results of various keywords is shown in Figure 2. We noted and averaged the citations for each keyword, and selected the results with a citation number greater than the average. The subsequent searches were done pertaining to the specific evaluation methods. The following research questions aim to be answered by this review study:

1. What are the evaluation methods available for Dialog systems based on the structure of the dialog?
2. What is the requirement of an automated evaluation method for testing the usability of dialog systems?

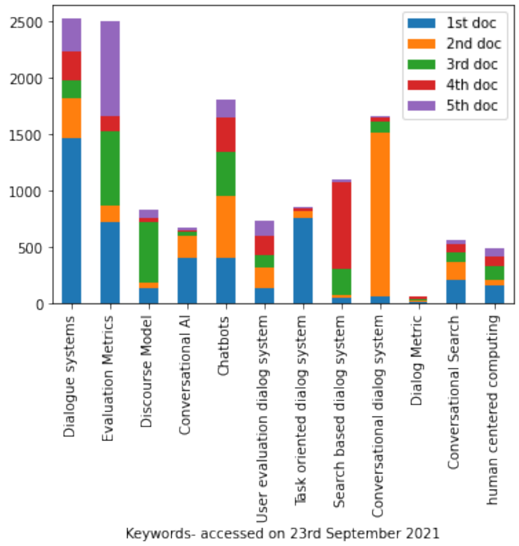


Figure 2. Number of Citations per keyword.

3. Structural Flow of the Article

This review paper aims to encompass the various types of dialog systems and the evaluation methods that can be used to better quantify and improve the systems. A brief overview of structure of the article is given in Figure 3. The article is structured as follows: The components of dialog systems are discussed in Section 4. Sections 5–7 provide a brief discussion of task-oriented, conversational, and question-answering dialog systems respectively. Section 8 classifies dialog systems on the basis of the logic component, that is, rule-based systems and AI based systems. Section 9 covers the evaluation of dialog systems. The section is divided into empirical metrics and user-based evaluation. The usability based metrics are discussed separately for task-oriented and conversational agents. Section 10 provides an overview of the various datasets and evaluation challenges currently being conducted in the field of dialog systems.

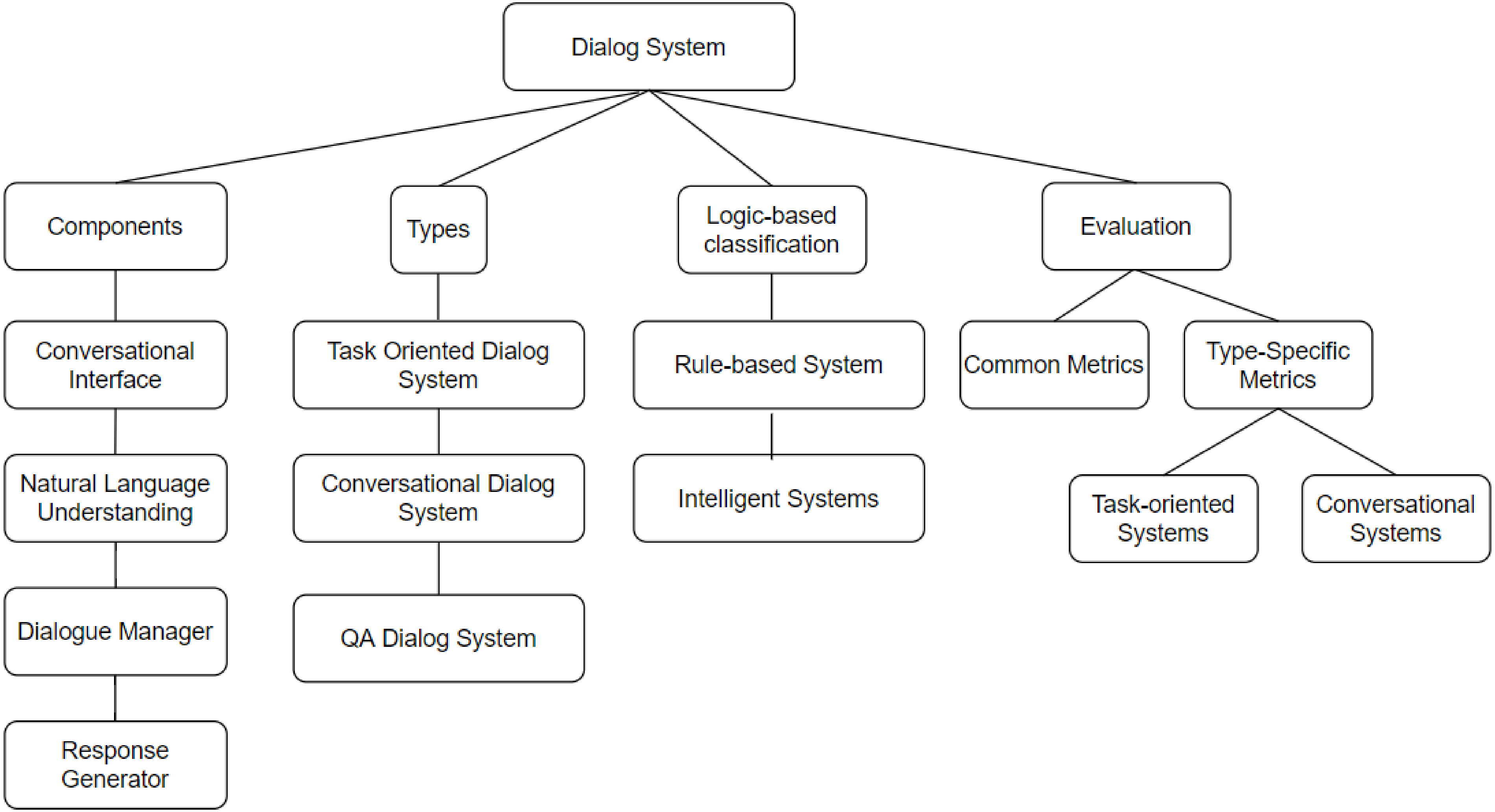


Figure 3. Structure of the paper.

4. Components of Dialog Systems

Dialogue systems serve as an interface with which the user can interact in a natural manner. There are several components in the dialog system architecture that help model the dialog as a sequence of actions. It is very helpful to understand the functions of the various components of a dialog system, as the impact of making changes in the components can be mapped to the final product. Arora et al. [31] propose the following components of a dialogue system: Input interface, Natural language understanding, Dialogue manager, Domain specific component, and Response generator. Some of these components vary for different dialogue systems, depending on application area. Each component in the system provides an input for the next one. The query provided by the user serves as the input of the NLU unit. The NLU unit attempts to understand the input and corresponds with the dialogue manager. The response generator corresponds with the domain-specific component to provide an output to the user query. The conceptual workflow of a dialogue system and its various components is shown in Figure 4.

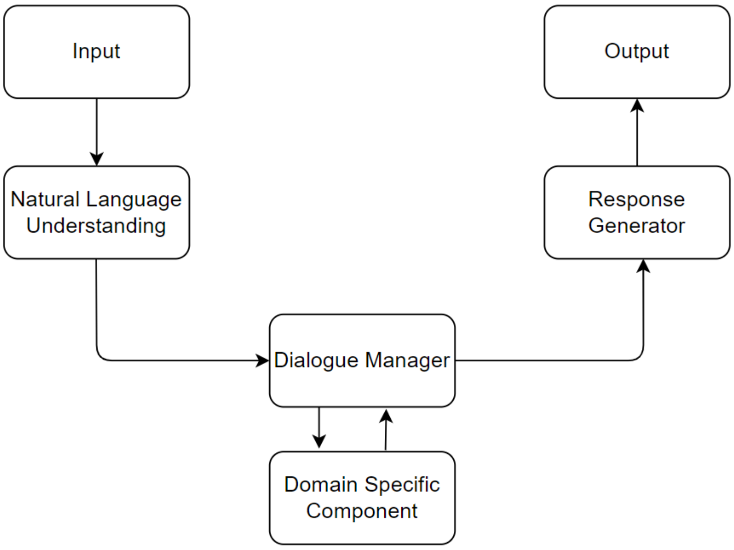


Figure 4. Components of a Dialog System.

4.1. Conversational Interface

The input to the dialog system could be in the form of text or speech. A conversational interface serves to provide the user with the ability to interact with the dialogue system in a natural manner. A dialog system may be text-based, speech-based or multimodal. In a text-based system, an interface with a chat option will be available. Spoken dialog systems accept input in the form of speech over one or multiple turns [32]. A multimodal dialog system will accept input in more than one form, such as speech, text, pen, and body movement [33]. A conversational user interface (CUI) is a digital interface that facilitates interaction with a dialog system in a convenient manner. Klopfenstein et al. [34] demonstrate several areas in which conversational interfaces are being studied, such as natural language processing, human-computer interaction, usability of the interface, and so on. A CUI should be as user-friendly as possible, as it follows the principles of human to human conversation [11]. The user does not need to worried about the model used or other functionality. The user is provided a simple interface to answer and ask questions. There are several methods available to develop a user interface. Some of the most relevant parameters from an evaluation perspective would be user experience, time taken, and user frustration while using the system [35].

4.2. Natural Language Understanding

Natural Language Understanding (NLU) is a tool-kit to help an assistant to understand the input. It helps to extract structured data in the form of intents and entities from unstructured human language. Intents can be understood as labels that represent the overall goal of the user's input. Entities are pieces of information that an assistant may need in a certain context. NLU consists of three sub-tasks which are intent classification, domain classification, and slot identification [36]. Slot-pair pairs are extracted from the current user utterance. Some of the common functions associated with NLU are listed as follows:

1. Intents: Intent classification is an important component of natural language understanding (NLU) which classifies and labels the input given by the user to assign it a specific goal [37]. It enables the dialog system to understand user requests. For example, the user may input a question such as "What time does the store open?". The intent for this particular request is most likely "Opening Times", which can then allow the system to answer with the opening hours of the store. An intent represents a mapping between what a user says and what action should be taken by the software.
2. Named entity recognition: Named entity recognition (NER) is the task of identifying named entities in the text [38]. Entities are pre-defined categories like names, locations, quantities, expressions of times, etc. Entity extraction enables the system to extract information from the text and helps in organising in text. For example, the name of a store, location of a venue, and the fee of a particular service could be considered entities. An entity represents concepts that are often specific to domain as a way of mapping natural language phrases to canonical phrases that capture their meaning.
3. Pattern Matching: Pattern matching helps to match the input obtained from the user with the database and try to obtain an appropriate response [39]. Lee et al. [40] list the algorithms available for pattern matching such as fuzzy string matching, regular expressions [41], rule-based matching, token-based matching, etc.
4. Parsing: Text parsing is the process of determining the syntactic structure of a text. It separates a given corpus into smaller components based on some rules [42]. Parsing algorithms parse the text in accordance to the predefined rule of algorithms such as left-right and bottom-up algorithms. These algorithms learn to recognize strings and assign syntactic structures to the strings [43]. Figure 5 shows an example of a parsing in action. The syntactic structure of the sentence 'The message is the answer' is obtained by dividing its constituent words according to their grammatical function.
5. TF-IDF: Term frequency- inverse document frequency (TF-IDF) weight measures the importance of a word to the document it belongs to in a corpus. It converts user input into vectors and finds similarity with documents. The system returns high priority results from the chatbot by comparing higher cosine similarity with TF-IDF. Cosine Similarity measures the content-based similarity between two vectors which represent the text summary and reference system summary in the vector space. TF-IDF models can also serve as similarity metrics for evaluation of dialogue [44,45].
6. Word2Vec: Word2Vec is a two-layered neural network used to produce word embedding [46–48]. It is a similar approach to TF-IDF as it also processes the text corpus into vectors. The text corpus is turned into a numerical form and plotted into a vector space to create a knowledge base.

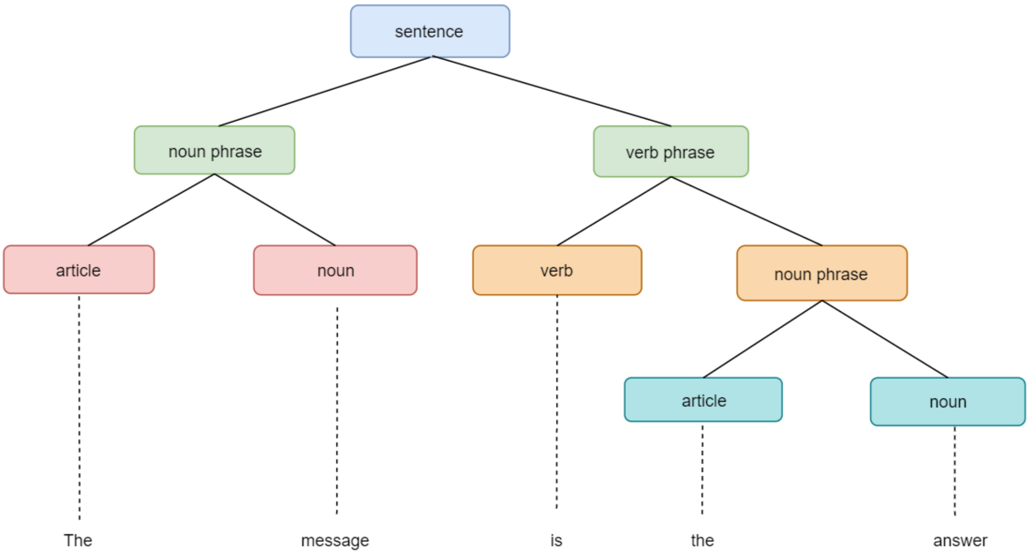


Figure 5. Parsing.

4.3. Dialog Manager

A dialog is a written or spoken conversational exchange between two or more entities. A dialog manager keeps track of history in memory with states. It is also responsible for generating responses and providing a conversational flow [49]. Dialog planning enables the system to be able to manage the conversation by understanding the context. The system should be able to manage a conversation even when there is a switch of context in the dialog. The dialog manager also takes into account dialog history while selecting responses that feel more natural and user-friendly. For example, the user might say "I need to order a copy of ′ To kill a mockingbird′ in e-book" and the system might record the order. The user might then say "change it to hard copy". The system should be able to correctly interpret from context that the user requires a hard copy of the same book. The dialog manager receives its input from the natural language understanding components, corresponds with the logic or knowledge component, and passes the output to the natural language generation component [31]. Dialog state tracking (DST) is used to estimate the current belief system of a dialog taking the preceding conversation into account [50]. The current belief state encodes the user′s intent and the preceding dialogues to better handle misunderstandings. Xu and Hu [51] explain the use of end-to-end models to better handle unknown slot values in dialog state tracking. A higher probability may be assigned to an entity based on the dialogues in the previous turns. The dialog manager learns the strategy. The input of the strategy module is the current belief as computed by the DST module. The dialog manager then generates the next dialog or action in the system. The dialog control decides which action needs to be taken.

McTear [52] identify two main steps in the dialogue management process:

1. Dialogue modeling: It keeps track of the state of the dialogue.
2. Dialogue control: It makes decision about the next step to be taken by the system.

4.4. Domain Specific Component

This component is the external database or expert system that contains the domain knowledge. Conversational search agents can work with search and information retrieval from the internet [53]. Abdul-Kader and Woods [54] propose the development of a dialog system which does not require a knowledge base, but searches the world wide web, with the help of chatterbot Python library and NLTK. Text-matching is performed to find the answers for queries and this new knowledge is then added to the structured database. Maroengsit et al. [55] thoroughly explain several methods for information extraction and processing of semantic information. The processing step makes extensive use of natural language processing (NLP). The data that is used to train the system is collected externally or from conversational dialogues, and then converted to a suitable format.

A publicly available corpus of data could be also be used to train a conversational system. The dialog system may vary significantly depending on the data on which it is trained. Knowledge that the dialog system is trained on is the heart of a dialog system. Some of the largest datasets present are the Reddit Corpus [56], with over 1.7 billion comments. The Twitter corpus [25] consists of 1.3 million conversations. The Ubuntu Corpus could be used to train a task-oriented or information retrieval system [57]. Datasets from several evaluation challenges are also made available [58,59]. However, creating a knowledge base for a very specific task would require a lot of human time and effort in engineering and annotation. The corpus might need to be annotated by human judges that could be crowd-sourced at Amazon mechanical Turk [60]. An external component may also interface with natural language query processing to extract SQL queries from natural queries [61].

4.5. Response Generator

The response generation step provides the output to the user. Natural language generation (NLG) transforms the structured data into natural language to provide an output to the user [62,63]. In order to successfully generate appropriate responses to the user query, it is important that the user intent and context of the conversation is understood comprehensively.

4.5.1. Rule-Based Systems

Rule-based systems follow a set of pre-defined rules or a tree-like structure to generate responses. One of the earliest rule-based systems is ELIZA [7] created in 1986 at MIT. ELIZA used pattern matching and keywords to process a set of pre-programmed rules to allow the system to give proper responses. PARRY [64] on the other hand attempts to model the mind of an actual paranoid patient with results that clear the Turing test. Section 6.1 discusses rule-based systems in more detail.

4.5.2. Corpus-Based Systems

Corpus based systems usually work with employing information retrieval (IR) to datasets of human-human interaction. IR can copy a user's response from a previous conversation to the current conversation. Machine learning and neural network models can also be trained to map a user utterance to a system response [65]. There is no strict adherence to hand-written rules and the functions of the system depend on the corpora [66]. Several available corpora, such as movie dialogues, conversations on chat platforms, and tweets, can serve as the training set for the model. Corpus based systems are also known as response generation systems [67], as the primary focus is the generation of a response appropriate to the user's query.

5. Types of Dialog Systems

In this section, we discuss the various types of dialog systems with their characteristics. The section provides an overview of task-oriented, conversational and question-answering systems.

5.1. Task-Oriented Dialog Systems

Task-oriented systems are conversational agents developed for solving a particular task in the most efficient way possible. Task-oriented systems have a narrow focus of performing a certain task specified by the user. Zhang et al. [68] illustrate that dialog systems could be designed for specific tasks such as making reservations, inquiring about a particular business, customer support [22], teaching aid [69], etc. Task-oriented dialog systems could be text-based [70], speech-based [71], or multi-modal [72]. Task-oriented systems are trained on restricted domain knowledge and do not handle trivia questions well. The system could be trained with knowledge specific to any domain and follows a clear structure.

Dialog Structure

Task-oriented systems are usually based on domain ontology, which defines the terms in a specific area, the relationships between the terms, and an expression of the relationships in an hierarchical structure. Task-oriented systems usually encompass a narrow domain. The models for developing these bots are usually retrieval-based systems. As we discussed in Section 5.1, retrieval-based models use a repository of predefined responses.

5.2. Conversational Dialog Systems

Conversational agents are a type of non-task oriented systems which are developed either for normal chit-chatting or for performing conversational search [73,74]. Conversational agents are designed to have extended and unstructured conversations, which try to replicate the 'naturalness' of human to human interactions. Conversational dialog systems follow the structure of human conversations instead of focusing on the completion of specific tasks. These systems could be used for entertainment and other purposes such as providing psychological counseling. There is no specific task to solve so the dialog is more open-domain. The dialogues are usually longer as in case of human interactions. Cahn [75] recommend that it is preferable for a social conversational system to have a personality as they attempt to mimic human behavior. The Turing test [76] is a good judge of the effectiveness of conversational agents in imitating humans. The solution proposed by Kenny et al. [77] makes use of virtual humans as a solution for monitoring and providing healthcare for the elderly. The virtual humans are endowed with a personality, speech recognition, and other NLP functionalities. Vision and other sensors could also be added to absorb certain variables from the person's environment. This could provide beneficial assisted healthcare for the person. Tavarnes et al. [78] introduced the idea of Virtual Patient Learning (VPL), which is used in the providing training aid for healthcare students and professionals.

Modelling Conversational Dialog Systems

There are currently two main types of modelling approaches for conversational dialog systems: rule-based systems and corpus-based systems. Rule-based systems are further discussed in logic based classification Section 6.1. We will further discuss the corpus-based approaches used to model conversational agents. Corpus-based approaches could either be based on a retrieval model or a generative model. Retrieval-based models try to extract the appropriate response from its knowledge corpus. Generative models attempt to generate appropriate responses during the conversation. Intelligent conversational agents are usually build using generative models. Generative models can generate new responses from scratch. They are more likely to make mistakes, as the responses are longer. These dialog systems often attempt to pass various versions of the Turing test.

1. Generative models: Generative models are an unsupervised approach to generate responses for the dialog system. The response generation step provides the output to the user. Natural language generation (NLG) transforms the structured data into natural language to provide an output to the user. In order to successfully generate appropriate responses to the user query, it is important that the user intent and context of the conversation are understood comprehensively. Generative models make use of deep neural networks. The dialog is generated by training the model on a large corpus of dialogues, and the most appropriate response to a given user utterance is returned. The experiment done by Serban et al. [79] to create a generative model involved two models which are Recurrent neural networks (RNNs) [80] and Hierarchical recurrent encoder decoded architecture (HRED). Vinyals et al. [81] and Serban et al. [82] demonstrate the application in dialog response generation for both LSTM and HRED architectures respectively.

Language models are probabilistic or statistical models that determine the probability of the occurrence of a word from a given corpus. Contrary to rule-based algorithms, language models attempt to understand the contextual relationships between different words in a sentence. Responses generated from a language model are thus more relevant in the given context. One of the most popularly used language models is the Bidirectional encoder representations from transformers (BERT). Devlin et al. [83] introduce BERT, which is a specific, large transformer masked language model. For a masked language model (MLM), you train the model by removing words and having the model fill in the correct word. Masked language models are useful because they are a type of contextual word embedding. Contextual word embedding allows you to have different word representations for different contextual meanings of the word. The BERT architecture uses a stack of either 12 (BASE) or 24 (LARGE) Encoders. It can be used as a general-purpose pre-trained model that is fine-tuned for specific tasks. BERT is a transferable model, thus it can be used as input to smaller, task specific models. With successful fine tuning of the model, we can achieve a very high accuracy. There are pre-trained BERT models available in over 100 languages. There are certain extensions of the BERT architecture such as RoBERTa [84], DistilBERT [85], AlBERT [86], and more. BERT models are also present in different languages such as CamemBERT (French) [87], AraBERT (Arabic) [88] and mBERT (multilingual) [89]. Drawbacks are that the model is very big, slow to train, computationally expensive, and needs to be fine-tuned for downstream tasks.

2. Utterance Selection: The modeling of the dialog system is done as an information retrieval task in utterance selection. Candidate responses are ranked according to their relevance. The most appropriate response is retrieved from the database according to a given utterance. Figure 6 shows the extraction of values from an utterance. A probability could be calculated which then ranks the candidate utterances according to their relevance. The utterances in a dialog database help define the dialog structure [90], as the system learns to map the semantically relevant responses to the user utterances. An utterance selection model as defined by Baxter et al. [91], is a set of rules that help in filling slots for response generation. The similarity between the dialog history and candidate utterances can be measured by a similarity measure. Surface form similarity measures the similarity based on token level. Examples of these measures include METEOR [92] and Term Frequency-Inverse Document Frequency (TF-IDF) models, discussed in Section 4.2 [44,45]. The recurrent surface text pattern approach is proposed by Duplessis et al. [93] which involves a database of recurrent surface text patterns and the utterance retrieval from the database through a generalised vector-space model.

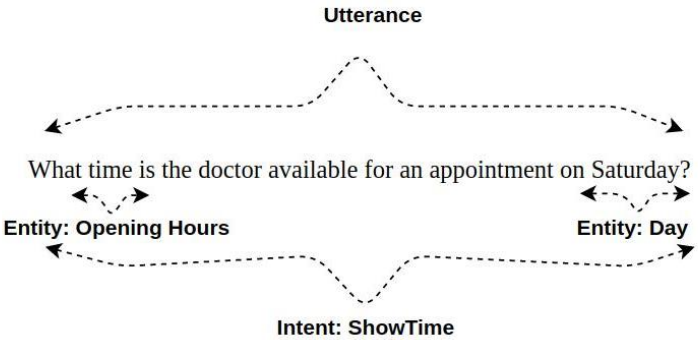

Utterance
What time is the doctor available for an appointment on Saturday?
Entity: Opening Hours
Entity: Day
Intent: ShowTime

Figure 6. Example of an Utterance.

5.3. Question Answering Dialog Systems

Question answering systems are used to answer a specific question asked by a user in natural language [2,94]. Unlike rule-based systems which can be used to perform a multitude of tasks, question answering systems do not have to be particularly designed for a domain [95]. The system is usually initiated by a user prompt. The system can take several turns to gather all the information needed to answer the question. QA systems are not very strictly structured, but the answers depend on the knowledge domain the system is trained on. QA systems can be build to provide answers for multiple domains. Unlike conversational agents, QA systems do not attempt to mimic human-like interaction. The main focus lies on correctly answering the question in a short amount of time. The responses provided by the system are generally short and follow a natural language format. Chandra and Suyanto [20] present a question-answering dialog system which is trained on a dataset of conversations about university admission. The chatbot was designed using a sequence-to-sequence and an attention mechanism technique. Sreelakshmi et al. [21] propose a QA system which performs both question-answering and quiz-generation. Neural networks are used for the question-answering module and the quiz generation module generates question-answer pairs. The approach takes input in form of a book which is converted to the knowledge base, making it adaptable to a variety of subjects.

Evaluation

Due to the nature of multi-turn QA systems, it is quite difficult to design an accurate and automated evaluation metric. A significant amount of human intervention is required in the evaluation process. Evaluation metrics used in information retrieval systems are often employed in the evaluation of QA systems as well. Information retrieval systems respond to a user's query turn with a reply retrieved from the corpus it is trained on. The relevance of the dialog system depends on the sector and the corpus it is trained on. In addition to the training corpus, the system can also append the conversational data it obtains from interacting with users. There are several algorithms available to develop a model for an IR system. The response can be generated by using words from the prior response or using full IR ranking algorithms. Non-dialog text can also serve as the training data for IR systems. For example, dialog systems can be trained on articles to generate informative responses. Oniani et al. [96] perform subjective evaluation of language models for a COVID-19 question-answering system. The responses generated by the four chosen models were annotated by two medical experts on different relevance scores. The bidirectional encoder representations from transformers (BERT) model provided the best results out of the four.

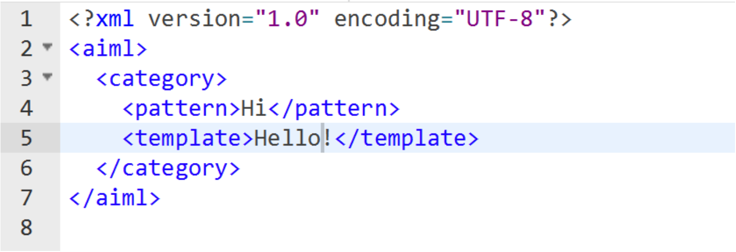
6. Classification Based on Logic

Dialogue systems can be of the following two types: Rule-based and Intelligent. Rule-based systems follow a set of predefined rules and cannot answer questions that are not hard-coded. They map out conversations through the implementation of a decision tree. Intelligent systems, on the other hand are able to learn from interacting with users. They implement artificial intelligence (AI) techniques such as machine learning, deep learning, and reinforcement learning. There are certain advantages and disadvantages to both of these. Rule-based systems are quick and easy to train. The evaluation process is also simpler and they are usually more reliable. The downside is that they have a very restricted conversational range and do not evolve with interactions. AI-based dialog agents attempt to understand the meaning of the text so they are able to generate relevant responses. This requires a lot of data for training. Evaluation also becomes more difficult.

6.1. Rule-Based Systems

Piccinini [97] discusses the "imitation game" developed by Alan Turing. It is a test between human and machine subjects and the task is to differentiate between the two. If a machine is able to fool the "interrogator", it is said to have passed the Turing Test and thus has the ability to think. The first conversational system to even get close to passing the Turing Test was ELIZA introduced by Weizenbaum [7] and developed in MIT in 1960. It was trained on 'scripts'. The most popular script was DOCTOR, which was modeled as a Rogerian psychotherapist. ELIZA follows keyword identification and pattern matching to generate rule-based responses. The next to come in line was PARRY introduced by

Colby [64], which was modeled as a mind of a paranoid schizophrenic. Another significant rule-based system is ALICE, introduced by AbuShawar and Atwell [6]. ALICE works on a knowledge base to generate responses based on <pattern> and <template> pairs. The knowledge base was created in AIML (Artificial Intelligence Mark-up Language), which is an extension of XML. Figure 7 provides a snippet of AIML code. FAQ agents could also be developed as a rule-based system. FAQ agents are a type of task-oriented dialog system that help businesses automate answering of frequently asked questions by users. Sethi [98] present a prototype for a simple FAQ agent that provides an easy user interface. FAQ agents are trained on domain-specific knowledge, making them highly efficient and knowledgeable. Rahman [99] introduce an ALICE chatbot presented as an FAQ agent for providing assistance to users looking for information about a university. Lee et al. [100] discuss the usefulness of chatbots in reducing administrative load by using the NASA-TLX questionnaire. FAQ agents prove to be better at providing responses as they can easily extract entities and understand intents. Van Rousselt [101] mention FAQ agents which are very commonly used now-a-days. They allow users to ask a simple questions and get a response. Follow-up questions may also be added. The industry standard to build FAQ assisted agents is by using a set of rules or a state machine.



```
<?xml version="1.0" encoding="UTF-8"?>
<aiml>
  <category>
    <pattern>Hi</pattern>
    <template>Hello!</template>
  </category>
</aiml>

```

Figure 7. AIML script example.

6.2. Intelligent Systems

Intelligent systems, or AI systems, make use of natural language processing and artificial intelligence tools to generate responses. Machine learning, deep learning, and reinforcement learning tools could be used to implement intelligent systems. Intelligent systems attempt to understand the intent behind the user input, rather than keyword matching. Conversations with intelligent systems is more natural and human-like, as the system can be trained to accommodate basic human errors such as typing errors and grammatical mistakes. AI-based systems provide an advantage over rule-based systems as they are better at extracting entities and understanding intents. With the use of natural language processing, it is possible to understand the context of the user input. For example, the word 'bank' may have a different meaning depending on the context of the sentence. It could be used in the context of a river bank or a financial institution. Intelligent systems should be able keep a context of what has been said before, handle unexpected conversation turns, drive the conversation when path is diverted, and improve over time. The current state-of-the-art dialog systems include Apple's Siri [16], Amazon's Alexa [18], and Microsoft Cortana [17].

6.2.1. Deep Learning and Machine Learning Based Systems

Nagarhalli et al. [102] present the current trends in the development of dialog system systems. Conversational artificial intelligent bots are one of the most important application of natural language processing. Su et al. [103] propose a dialog system to interact with elderly people, with the use of a structured database containing over 2000 message and response pairs, making use of LSTMs. Kuligowska [104] explain how to get a robot to have a conversation on a story narrated by it. The robot can be developed by using a CKIB toolkit for word segmentation, parts of speech tagging, and term frequency inverse document frequency for noun identification. Artificial intelligence markup language (AIML) is used for development of the dialog system. Legacy techniques are used for the dialog system, which may not be the most efficient technique. Baby et al. [105] propose to use a dialog system for home automation with the help of IoT domain. The paper proposes to use NLTK to identify keywords which are then passed on to the micro-controller which will control home appliances. Lee et al. [106] suggest the development of a companion robot for children. Question- answer pairs are generated and ranked from about 100 student's tales with the help of logistic regression (LR). Pichponreay et al. [107] propose to use the ApachePDFBCK for optical character recognition of text present in PDF documents. The questions can be generated and string matching can be used to generate answers. Dialog systems have been showed to be efficient for use in customer service time and time again. D'silva et al. [108] show the use of dialog systems for customer service, as the problems faced by customers are not always resolved by the companies.

Dialog systems are also being widely used in the health care sector. Madhu et al. [4] propose a dialog system which takes in symptoms as user input and provides disease predictions, along with the suggested medication and dosage. There are several limitations of using a dialog system for medical purposes. There is a lack of scalability as the diseases and symptoms have to be hard-coded. Choi et al. [109] propose the development of dialog system that helps users with queries about newly purchased electronic gadgets. Understanding the features of a new product by going through a manual can be a daunting task. The dialog system can be helpful to keep pace with the user's learning speed. Kuligowska [104] propose a dialog system for answering queries about banking procedures. Customer service may not be always available to answer user's queries, and this may be particularly important if a large sum of money is involved. The dialog system is trained on a dataset comprised of frequently asked questions from various banking platforms. The paper makes use of NLTK and bag-of-words for converting words into vectors. Classification is performed through different machine learning algorithms, and the question is mapped to the answer using cosine similarity. The accuracy is claimed to reach 87%.

Models such as Sequence to Sequence (Seq2Seq) demonstrated by Latif et al. [110] and Bidirectional encoder representations from Transformers (BERT) as demonstrated by Devlin et al. [83] are the current industry best practices. Kaushik et al. [111] introduce a multi-view conversational search interface with a dialogue-based agent which assists users in refining their searches to obtain relevant results. The system was enabled with image search and implemented with the help of RASA [112]. The dialog structure for the agent is defined in the following three steps: identification of information need of user, formation and presentation of results in the chat, and successful completion of the search task. The system proves to be an effective method of lowering cognitive load and frustration of the user while performing searches. The study by Aliannejadi et al. [113] highlights different conversational search strategies used by an agent that guide the search by providing query suggestions and clarifications. Kaushik et al. [19] introduce a conversational agent to assist users in performing search tasks. The multi-view search interface includes a graphical interface and a conversational agent to help the user to perform searches without increasing their cognitive load. The information retrieval bot provides assistance to the user to develop their query in a series of interactions.

6.2.2. Prototypes

1. Dialogflow: Dialogflow [114] provides a conversational experience powered by Artificial intelligence. Singh et al. [115] discuss the advantages of a conversation management framework such as Dialogflow in certain use cases. It can receive responses in both text and voice form, and can be integrated across virtually any platform such as speakers, home applications, wearable sensors, etc. The following components of Dialogflow are discussed in [116]: Intent matching, Entity extraction, and Dialog control. The intent matching step recognizes what the user wants. An intent is created for anything the user might request. For example, checking the price of a product, booking an appointment, checking opening hours, etc. A typical Dialogflow agent, which

represents a single conversational experience, might have a few to a thousand intents, which are each trained to recognize a specific user need. Entity extraction extracts relevant information from the user query. Dialogflow can extract information using system entities. For example, in Figure 8, the intent is identified as 'Opening_time' and 'time' entity is extracted. Dialog control shapes the flow of the conversation. The subsequent dialogues are interpreted in the context of the previous input.

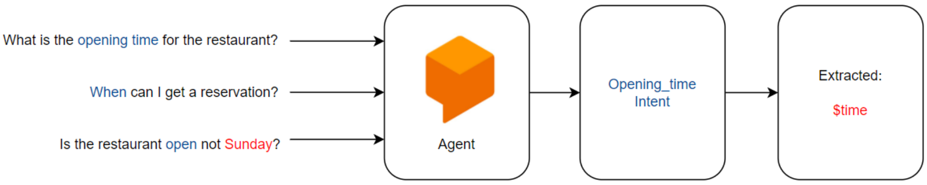

What is the opening time for the restaurant?
When can I get a reservation?
Is the restaurant open not Sunday?
Agent
Opening_time Intent
Extracted:
$time

Figure 8. Intent classification in Dialogflow [117].

2. Alexa: Amazon Alexa [18] is a virtual assistant incorporated with the Internet of Things (IoT). It can respond to any natural language query. One of the many outstanding features of Alexa is the developer tool. Every query given to it is parsed into a data structure and given to the user on AWS, where the user can then write their own custom code around it. Lopatovska et al. [118] explore user interactions with Alexa. Analysis of the data collected suggested that Alexa serves well as a virtual assistant for actions such as playing music, checking weather conditions, etc.
3. RASA: Bocklisch et al. [112] introduce Rasa, which is an open source machine learning framework for building contextual AI assistants and conversational agents. The model is transparent, which means we can observe exactly what is happening under the hood and customize things precisely. It is a state-of-the-art model and is the most effective and time efficient tool to build complex dialog systems quickly. In RASA, an action is an operation which can be performed by the bot. It could be replying something in return, querying a database, or any other task possible by code. Action could be just a hard coded reply or some API generated response. Stories are a sample interaction between the user and bot, defined in terms of intents captured and actions performed. Harms et al. [119] discuss the use of RASA for dialog management. RASA is made of two components: Natural language understanding (NLU) and dialog management. Natural language understanding is an open-source natural language processing tool for intent classification and entity extraction. It thus helps the bot to attempt to understand the user. Singh et al. [115] explain the implementation of RASA Core. The core is the framework for machine learning-based contextual decision making. It learns by observing the patterns from conversational data.
The RASA architecture follows the steps given:
   (a) The message is received and passed to an interpreter, which converts it into a dictionary including the original text, the intent, and any entities that were found. This part is handled by the NLU.
   (b) The Tracker is the object which keeps track of conversation state. It receives the info that a new message has come in.
   (c) The policy receives the current state of the tracker.
   (d) The policy chooses which action to take next.
   (e) The chosen action is logged by a tracker and a response is sent to the user.

Figure 9 shows the functionality of RASA.

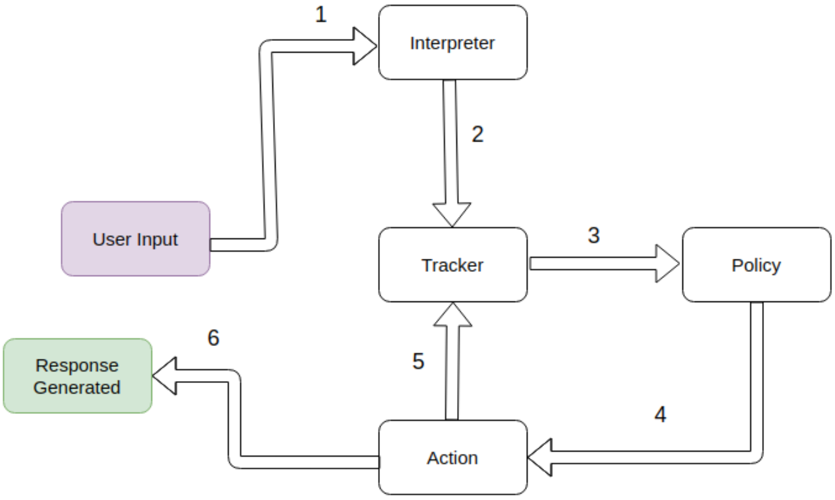


Figure 9. RASA Architecture [112].

6.2.3. Reinforcement Learning Based System

Reinforcement learning based systems learn from interacting with users and consequently develop their own control systems. The dialog exchange can be assumed to be a set of actions, which may be chatting or completing tasks. Previous dialog states are taken into account while performing next action, which ensures contextual awareness. Li et al. [120] apply a neural reinforcement learning method, which simulates conversation between two virtual agents, in order to explore all possible actions that can be taken while maximizing reward. The approach is a combination of Sequence to Sequence (Seq2Seq) model and reinforcement learning, which enables to it generate semantically appropriate responses while also optimizing future rewards. Zhao and Eskenazi [121] propose a novel end-to-end framework for a spoken task-oriented dialog system using deep reinforcement learning. A combination of reinforcement learning and supervised learning was used to provide better learning rates. The proposed model resulted in better results compared to a traditional spoken dialog system pipeline. Scheffler and Young [122] propose a method to generate human-computer dialog strategies using reinforcement learning. A simulation tool trained on a corpus of real user data is used to model user behavior. The results were promising as the learned dialog policies outperformed handcrafted policies given the same state space. Dhingra et al. [123] introduce KB-InfoBot, a search agent for searching knowledge bases without the use of complex queries. The study uses a soft retrieval process from the external database. The reinforcement learning process leads to higher success rate against both real world and simulated users.

7. Evaluation

Evaluation of dialog systems is a cumbersome task as there are no clear metrics on which the performance can be measured. Dialog systems exist for a number of application sectors, and each system needs to be evaluated accordingly. Usually human evaluations are considered to be more appropriate, but they can be very costly and time consuming. The effort for developing an automatic evaluation system is ongoing [25,124]. Evaluation of task-oriented and question answering systems can be relatively easy, as we have a clearly defined and unambiguous task to be performed. For example, answering a frequently asked question correctly and booking a reservation are both measurable tasks. For more complex system, a user satisfaction rating could prove to be useful. Deriu et al. [12] define three important characteristics of an efficient evaluation metric. The evaluation process needs to be automatic, as human labour is time and cost intensive. The process should be repeatable, that is it should yield similar results when used on a dialog system under similar circumstances. The ratings yielded by the evaluation should correlate to human judgements. There is also a need for explainability to figure out which features of the dialog system show correlation to the quality of the dialog. There is a trade-off of comparability to real world scenarios because the environment is very controlled. A large number of users can be recruited via crowd sourcing, for example, Amazon Mechanical Trunk [125]. The quality of the evaluation is shown to be comparable to laboratory conditions, as there is a high variability of user behavior present. In the Loebner prize contest [126], participants develop dialog systems which compete on their ability to fool the judge in a restricted chat session. The contest also evaluates dialog systems in terms of naturalness. Evaluation of spoken dialog systems (SLDs) can be done using the glass-box and black-box dialog quality metrics along with user satisfaction feedback [127]. The glass box metrics evaluate individual components such as sentence understanding and sentence recognition. Black box metrics evaluate system as whole based on user satisfaction and acceptance, evaluation of performance of the system in terms of achieving the task in terms of time taken, and number of turns. Table 1 lists a number of evaluation models implemented with the specific criteria evaluated and corresponding results.

Table 1. Evaluation.

Evaluation Method	Model Used	Results	Evaluation Criteria
Survey of WeightMentor App [128]	SUS, UEQ and CUQ Questionnaires	Median SUS score 84.83 ± 12.03, Participant scores above +0.8, CUQ mean score 76.20 ± 11.46.	Usability study on WeightMentor implemented through DialogFlow
Automatic dialog Evaluation Model (ADEM) [129]	RNN	Pearson's correlation = 0.41 on the utterance level and at 0.954 on the system level.	Quality of dialog responses
Adversarial Evaluation [130]	Data generated artificially by the generator and real data; GAN; Adversarial training where model tries to discriminate whether the data is real or artificial	Discriminator accuracy of 62.5%	Naturalness of dialog
Evaluation by Next Utterance Classification [131]	Datasets: Ubuntu Corpus, SubTle Corpus, Twitter Corpus. Dual-Encoder(DE), includes RNNs and LSTMs. ANN trained to find human correlation.	*Recall*@1 and *Recall*@2 values were calculated. The highest *R*@1 value was 88.4 ± 7.0 percent for the Twitter Corpus. The highest R@2 was 98.4 ± 2.7% for Twitter Corpus.	Dialog strategies and next utterance
Lexical Diversity [132]	Maximum Mutual Information (MMI) Models that generate more diverse and appropriate responses	Improved quality as measured by BLEU and human evaluation.	Diversity of responses for conversational system.
Distance Weighted Cosine Similarity Measure [133]	Vector Space Model (VSM), Centroid classification algorithm. Datasets used were 20 Newspapers, Reuters52c, Sector	Micro-average of the F1 score (MicF1) is highest for Reuters52c (90.3816).	Cosine similarity for text classification
RADDLE [134]	Fine-tuned GPT-2, Domain Aware Multi-Decoder (DAMD), SOLOIST	SOLOIST performs the best with 10 more average points than GPT-2.	Generalization ability of Task-oriented dialog systems
Ruber [135]	Seq2Seq with attention, Retrieval from Human Judgements	Ruber metric shows to have near human correlation.	Correlation with human judgement for Open domain dialog systems

For this study, we have classified the available metrics into two categories: automated empirical metrics and user-experience evaluation. User-based evaluation tools are discussed separately for task-oriented and conversational dialogue systems. Figure 10 gives a few examples of both empirical evaluation metrics and user-based evaluation tools.

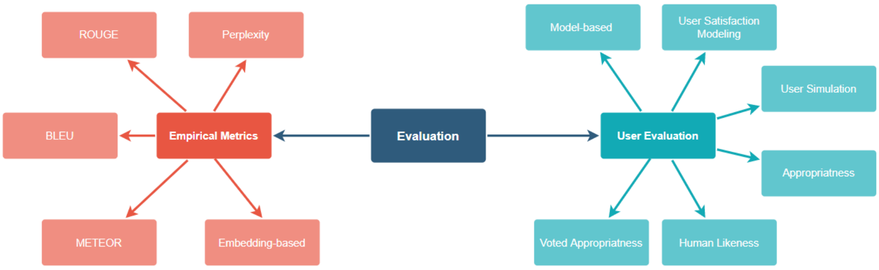


Figure 10. Categorization of Evaluation Metrics.

7.1. Empirical Evaluation Metrics

Empirical evaluation metrics can be calculated easily through the use of a mathematical formula or algorithmic script. They are quantitative in nature and do not require much human intervention to calculate. However, they do not provide a very accurate picture of the system's performance. Empirical metrics can be used for evaluating certain aspects of both task-oriented and conversational dialogue systems. For task-oriented systems, the task completion success can be evaluated by gauging the correctness of the solution provided. Efficiency costs are measured by testing the system's ability to help users. Factors such as total turns taken, time elapsed in a dialog, and the total number of queries can be used to measure the efficiency cost. These metrics provide feedback on various aspects of system responses such as similarity to ground truth, understanding of context, diversity in response, etc. Metrics such as BLEU [23], Coherence [136], ROUGE [24], Perplexity [137] have also been used in dialog system research [138–140]. The limitation is that in the context of dialog systems, more than one response may be considered valid. Some of the commonly used metrics are elaborated in this section.

7.1.1. Embedding-Based Metrics

Word embeddings are commonly used to represent the words in a document [141]. They are vector representations of a particular word and help to capture the syntactic and contextual functions of the word. Liu et al. [25] highlight several available metrics for evaluation of machine translation responses. The responses generated by a dialog system can also be thought of as an MT response, thereby allowing the use of the same evaluation metrics. Common word-embedding techniques include Word2Vec [46], Bag-of-words [142] and TF-IDF. Methods like Word2Vec approximate the meaning of a word by examining the frequency of its co-occurence with other words in the corpus. Word-embeddings are thereby calculated using distributional semantics. Vectors of individual words in a sentence are then approximated into sentence-level embeddings using some heuristic. The sentence level embedding of candidate and target response are then compared using a measure such as cosine distance. Cosine similarity is the cosine of the angle between the two vectors to approximate how similar the two vectors are.

1. Greedy Matching: Based on the cosine similarity of their word embeddings, the tokens in two sequences are greedily matched [143]. The total score is then calculated by taking an average of all words. The greedy matching approach generally favors responses with key words that are semantically similar to those in the ground truth response [144]. Rus and Lintean [145] compare greedy and optimal matching methods for two intelligent tutoring systems. The greedy method does not obtain the global maximum similarity score between the candidate and ground responses.
2. Embedding Average: Embedding average metric calculates sentence-level embeddings using additive composition. Additive composition computes the meanings of phrases by averaging the vector representations of their constituent words [146–148]. The embedding average is defined as the mean of the word embeddings of each token in a sentence. The cosine similarity between the respective sentence level embeddings is computed to compare a ground truth response and retrieved response [25].

3. Vector Extrema: Vector extrema is another method to calculate sentence-level embeddings [149,150]. For each dimension of the word vectors, take the most extreme value amongst all word vectors in the sentence, and use that value in the sentence-level embedding [25].

7.1.2. BLEU

Papineni et al. [23] introduce Bilingual evaluation understudy (BLEU), which is developed from a precision method that automatically evaluates machine translation to measure the quality of the translation. A good quality corpus of human translation is used as a reference. The BLEU score ranges the machine translation from 0 to 1. An n-gram is simply a sequence of N words. Precision score works better with shorter responses. For example, the dialog system may provide two different responses to “Hello”: “Hi” or “How are you?”. The response with the highest BLEU score is considered to be more appropriate and the dialog system should ideally return this response. The assumption in evaluation through BLEU is that there is only one ground truth response. Dhyani and Kumar [151] made use of a bidirectional recurrent neural network to implement an assistant conversational agent and the BLEU score was calculated for the agent. Li et al. [152] analyze the performance of an information retrieval chatbot using the BLEU-N algorithm. BLEU-1, BLEU-2, BLEU-3, and BLEU-4 scores were calculated for the system.

Suppose there is a single candidate ground truth response and is denoted as r and the proposed response is denoted as $\hat{r}$. The j token in the ground truth response r is denoted by w_j.

$$P_n(r + \hat{r}) = \frac{\sum_k min(h(k,r), h(k,\hat{r}_i)}{\sum_k h(k, r_i)} \tag{1}$$

where k indexes all possible n-grams of length n and $h(k,r)$ is the number of n-grams k in r.

BLEU has several shortcomings when it comes to evaluating dialog systems. Callison et al. [153] explore the effectiveness of a BLEU score in machine translation. There is no correlation to human judgements in machine translated responses. Two different replies generated by the system may be equally appropriate but have no n-gram overlap. It is also biased and incapable of considering the semantic similarity between responses.

7.1.3. ROUGE

Recall-Oriented Understudy for Gisting Evaluation (ROUGE), as explained by Lin et al. [24], performs evaluation by comparing ideal summaries created by a user in order to determine the quality as a summary [154]. The precision and recall is calculated using the overlap of words between the system generated summary and the reference summary. The recall in this case refers to how much the system summary captures the reference summary. Dutta and Klakow [155] explain that the ROUGE metric can be used in dialog systems to measure the similarity of the system generated output with the reference response. The precision determines how much of the system summary was relevant. Considering the number of individual words that overlap, precision can be calculated as:

$$ROUGE - N_{Precision} = \frac{m^n_{overlap}}{m^n_{output}} \tag{2}$$

where m^n_{output} is the number of n-grams in the system output, and $m^n_{overlap}$ is the number of overlapping n-grams between system response and reference response.

$$ROUGE - N_{Recall} = \frac{m^n_{overlap}}{m^n_{ref}} \tag{3}$$

where m^n_{ref} is the number of n-grams in the reference response, and $m^n_{overlap}$ is the number of overlapping n-grams between system response and reference response.

ROUGE scores are then calculated as the granularity of texts being compared between the system summaries and reference summaries. The ROUGE-N measures the unigram, bigram, trigram and higher order n-gram overlaps. It is computed as:

$$ROUGE - N = \frac{2 \times ROUGE - N_{Precision} - ROUGE - N_{Recall}}{ROUGE - N_{Precision} + ROUGE - N_{Recall}} \tag{4}$$

7.1.4. METEOR

Banerjee and Lavie [156] introduce Metric for Evaluation of Translation with Explicit ORdering (METEOR), which is an automatic metric which evaluates machine translation results by conducting unigram matching of machine produced and human reference translations. METEOR attempts to overcome some limitations of the BLEU score. Unlike BLEU, METEOR takes into account the recall, which is the proportion of matched n-grams out of the total number of n-grams in the reference translation. METEOR works by aligning the candidate and target responses. Denkowski et al. [92] explain the application of the METEOR score in the evaluation of dialog system.

First, the harmonic mean of the precision (P) and recall (R) of the responses, F_{mean}, is calculated:

$$F_{mean} = \frac{10PR}{R + 9P} \tag{5}$$

To account for longer matches of a given sequence of n-grams, a penalty if calculated as follows:

$$Penalty = 0.5 * \left(\frac{no.of chunks}{no.of unigrams_matched} \right)^3 \tag{6}$$

The METEOR score is then calculated as:

$$METEOR = F_{mean} * (1 - Penalty) \tag{7}$$

7.1.5. Perplexity

Perplexity is the inverse likelihood of predicting the responses of the test set, that is, how accurately can the model predict the next dialogue. Chen et al. [137] discuss the use of perplexity in evaluating spoken dialogue systems. It is shown to have poor correlation to word-rate error and thus is not a good metric to evaluate language models. Adiwardana et al. [157] compare perplexity and a human evaluation metric, Sensibleness and Specificity Average (SSA). SSA is a human evaluation metric proposed by Google. Perplexity and SSA are shown to have a very strong correlation for the evaluation of an open-domain chatbot. Jena et al. [158] evaluate the chatbot with perplexity and overlap between the domain-specific and human conversation dialogues.

7.2. User Evaluation Methods

User-based evaluation methods are focus on the usability aspect of the dialog system. They provide a more comprehensive picture of the system's performance in real world scenarios. In this section, usability evaluation methods are discussed for task-oriented and conversational dialog systems.

7.2.1. Task-Oriented Dialog Systems

Task-oriented systems have highly structured dialogues and quantifiable tasks. For example, If the venue of an event was requested and the correct information was provided by the dialog system, the task can be considered complete. Thus, it becomes easier to evaluate these systems based on the efficiency of the task completion. User evaluation may also be implemented to make the system more adaptable to real-world scenarios. Common evaluation methodologies used for task-oriented systems are discussed in this section:

1. User Satisfaction Modeling: The user satisfaction for any dialog system is a good indicator of it's usability. User satisfaction modeling can be conducted in three steps [12].

There is a requirement for explainability, which quantifies the impact that different properties of the dialog system have on user satisfaction. The evaluation process also has to be automated based on the properties of the dialog system. Differentiability requirement evaluates different dialog strategies using models. Two factors need to be considered while user satisfaction modeling. The agent evaluating the dialog system and the granularity at which the system is being evaluated are the major factors. The dialog could be evaluated by the user or by objective judges. The granularity of the dialog lies on two extremes: the evaluation could take place at the dialog level or exchange level. Engelbrecht et al. [159] present a method to model user satisfaction with the use of hidden markov models. Mean squared error in predicting the most probable state at each turn was calculated for the optimized model. This approach provided a way to analyze the models and features that affect the quality ratings, making it comparable to empirical ratings.

2. User Simulation: User simulators are tools designed to simulate user behaviors. Georgila et al. [160] make use of n-gram user simulation models to evaluate spoken dialog systems. Schatzmann et al. [161] discuss the development of user simulation techniques with reinforcement learning. Two evaluation strategies can be deployed for user simulators: direct and indirect. Direct evaluation is performed on the basis of various metrics, such as precision and recall on the dialog acts, perplexity, etc. Indirect evaluation attempts to evaluate the trained dialog manager. It measures the utility of the user simulation. Kreyssig et al. [162] propose the neural user simulator (NUS), which is a neural network based evaluation approach.
3. User Experience (UX): An important exploration was done by Holmes et al. [163] to test the applicability of conventional methods to assess conversational user interface. There are important questions raised in the paper such as how the evaluation results will correlate when applied to conversational agent usability. The usability of the WeightMentor App conducted by Holmes et al. [128], is calculated by using three metrics. The System Usability Scale (SUS) developed by Brooke [164] is one of the most popular and commonly used means of assessing usability. There are a total 10 questions, 5 covering positive aspects and 5 covering negative aspects of the dialog system. Each question is scored out of five. Final scores are then calculated out of 100. The mean WeighMentor score places the system in the 96th ± 100th percentile. Schrepp et al. [165] applied another metric called the User Experience Questionnaire (UEQ), which thoroughly assesses the UX. UEQ tells us to what level does the system meets the user expectation and how it tests against other dialog systems. The system performed well in all UEQ scales. The final metric is the Chatbot Usability Questionnaire (CUQ). Participants were given 16 items relating to the positive and negative aspects of the system. The questions were ranked out of 5, a scale of "Strongly Agree" and "Strongly Disagree". The CUQ test provided a mean score of 76.20 with the highest score of 100. Sharma et al. [166] introduce Atreya Bot, which is developed to facilitate chemical students and researchers in performing drug related queries from the ChEMBL database. The study aims to simplify the process of performing a successful search, outlining the challenges present in fulfilling a query. User frustration and mental workload is relatively low while searching on a conversational agent as compared to a traditional search engine.
4. PARADISE Framework: The Paradigm for Dialog System Evaluation (PARADISE), proposed by Walker et al. [29], is one of the best-known evaluation frameworks proposed for task-oriented systems. PARADISE framework is based on user ratings on the dialog level and also allows for evaluations of sub dialogues. Figure 11 explores the structure of objectives in the evaluation of a dialog system according to the PARADISE framework. The utterances are compared with a reference answer to perform automatic evaluation. This method has certain limitations, such as the evaluation process cannot discriminate between different strategies, the approach does not always generalize well, and the dialog performance cannot be attributed

to system specific properties [167]. The user interacts with the dialog system and proceeds to complete a questionnaire [168]. Responses in the questionnaire are then used to compute a user satisfaction score. This score can be used as the target variable for a linear regression model. Linear regression models can then be trained with the logged conversations serving as input variables. The model can then be fitted to predict the user satisfaction for the given input variables. Variables such as task-success can be extracted automatically, whereas variables such as inappropriate repair utterances need to be manually extracted by experts. The system also performs well with differing user populations and is good at performing predictions for new systems.

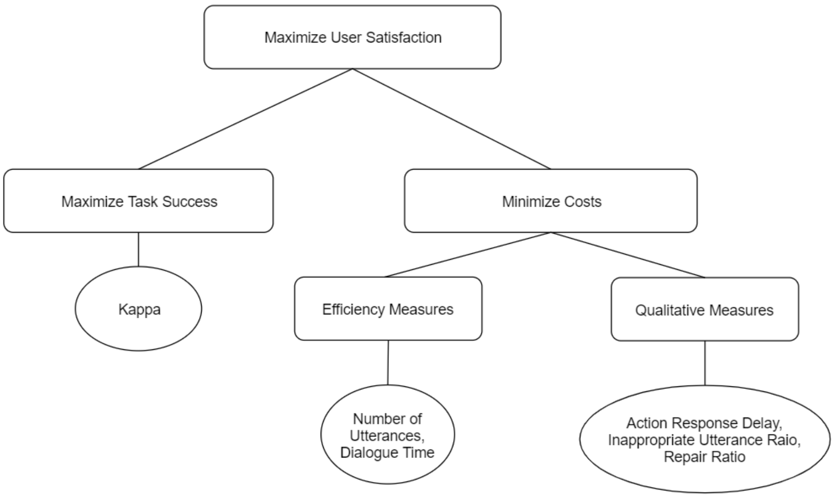

Maximize User Satisfaction
Maximize Task Success
Minimize Costs
Kappa
Efficiency Measures
Qualitative Measures
Number of Utterances, Dialogue Time
Action Response Delay, Inappropriate Utterance Raio, Repair Ratio

Figure 11. Structure of Objectives in the PARADISE Framework [29].

7.2.2. Evaluation of Conversational Dialogue Systems

It is trickier to evaluate conversational dialog systems due to the lack of dialogue structure and a clearly defined task to perform. The definition of high quality dialog isn't always clear and depends on the application of the conversational system. Even if there is a clear defined criteria for a high quality dialog, the measurement is not always valid. There is some discussion on whether the conversational agent should pass the Turing test. The consensus is that there should be an emphasis on developing the dialog system to appear and interact more human-like. Similar to the character development of a fictional character, people expect to suspend their disbelief while interacting with a conversational agent. The emphasis should be placed on establishing a rapport with the user during the interaction. Kaushik et al. [124] provide a comprehensive framework for evaluation of conversational agents. The paper identifies the following important criteria while performing evaluation on conversational agents: cognitive load, cognitive engagement, search as learning, knowledge gain, user experience, and software usability. The quality of conversational systems may be measured by the appropriateness of its responses and likeness to human conversations. There are several shortcomings to these approaches. There are generally two levels to evaluate conversational dialog systems: coarse-grained evaluations and fine-grained evaluations. Coarse-grained evaluations focus on the appropriateness of the responses provided by the dialog system. It includes two main concepts: appropriateness of the responses and human likeness. Fine-grained evaluations, on the other hand, focus on specific aspects of the dialog system's behavior. The system's ability to remain coherent and maintain a topic of conversation are measured. Quality of interaction is measured in terms of user satisfaction, that is whether the user gets the information they want, if they are comfortable with the system, and in which form they receive the information.

1. Appropriateness: It is a coarse-grained concept of dialog evaluation. Many fine-grained concepts, such as coherence, relevance, and correctness are also encapsulated

within it. The main approach employed includes the word-overlap metrics, originally used in machine translation and summarization. Word-overlap metrics have been discussed in Section 7.1. Word-overlap based scores include BLEU score and ROUGE score, which can serve as an approximation for the appropriateness of an utterance. One drawback of these metrics is that they show no correlation to human judgements. Galley et al. [140] propose ΔBLEU, which incorporates human judgements into the BLEU score. The reference responses in the test set are rated by human judges in relevance to the context. In the model based evaluation approach, such as ADEM [129], user behavior is modeled. A broad array of behavioral aspects need to be considered for the model to prove effective. The impact of using various dialog strategies should be explained by the model. The different types of users and the typical errors made by them while using the system are encapsulated by the model.

2. Human Likeness: The quality or a conversational agent can be measured by the Turing Test [76]. A conversational system is said to pass the Turing Test if it convincing to a human that it is human as well. The generative adversarial model as proposed by Xu et al. [169] can be used to evaluate dialog systems. The evaluation framework is made up of a generator for generating data and a discriminator to distinguish between real data and artificially generated data. The naturalness of a generated dialog response can be directly calculated by adversarial loss. As explained by Kannan and Vinyals [130], the encoder-decoder architecture employing recurrent neural networks has shown to be particularly helpful in dialog generation for dialog systems. The user query is input by the encoder. The decoder generates a response based on the final state of the first network. The method is based on Generative Adversarial Networks (GANs).
3. Fine-grained Metrics: Topic based evaluation measures the ability of the conversational agent to coherently talk about different topics. Two dimensions for topic-based evaluation are considered: topic breadth and topic depth [170]. The ability of the system to learn about a large variety of topics is measured by topic breadth, whereas topic depth measures if the system can sustain a long and cohesive conversation about one topic. A deep averaging network (DAN) can be trained to perform topic classification and detection of topic-specific keywords. A large amount of conversational data and questions are used to train the DAN model. There could be multiple utterances that could be considered as acceptable [171]. If at least one annotator marks the response as appropriate, the response is considered appropriate by the weak agreement metric [172]. The weak agreement has certain limitations. It relies heavily on human annotations and is not applicable to large amounts of data. Voted appropriateness takes into account the number of votes an utterance received for a given context, thus overcoming the limitations of weak agreement [172]. Each utterance is weighted uniquely. Voted appropriateness also has a higher correlation to human judgement as compared to weak agreement.

8. Evaluation Datasets and Challenges

We have discussed the two broad categories of evaluation methods for dialog systems. It is crucial to understand the implementation of a metric as they enable us to explore the shortcomings of a system. However, conversational dialogue is usually unstructured, making it difficult to obtain a reliable evaluation. It is necessary to train the models on structured datasets for reliable evaluation and comparison of different systems. This will also increase the relevance of results to make them more comparable to real-world scenarios. The public availability of datasets has made it easier to evaluate dialog systems. Datasets could be used for several evaluation procedures and are not restricted to one metric. Serban et al. [173] have already carried out evaluation of dialog systems and provided a survey of publicly available datasets. This section covers some popularly used datasets and research challenges that focus on improving the current state of the art in dialogue systems.

8.1. Datasets for Task-Oriented Systems

Since task-oriented systems are build to complete a very specific task, it is difficult to find pre-existing datasets for these systems. The dataset for a system might be difficult to re-use. To re-use them, several changes need to be made which requires a lot of human effort. A POMDP-based dialog system mentioned by Gasic et al. [174] makes use of crowd-sources on the Amazon mechanical turk service. The paper also talks about the transferability of datasets to a new domain by creating and keeping the unknown slot as a constraint. Wen et al. [70] developed an end-to-end dialog system and also used Amazon mechanical turk service to find users for user evaluation. The Dialog State Tracking challenge (https://www.microsoft.com/en-us/research/publication/the-dialog-state-tracking-challenge-series-a-review/) (accessed on 3 October 2021), which is focused on the development of the dialog state, has also led to the release of several datasets.

8.2. Data for Question Answering Dialog Systems

Data for QA dialog systems could be derived from chats on forums or online message boards. For instance, Lowe et al. [131] created the Ubuntu dialog Corpus, which contains close to a million multi-turn dialogues extracted from the chat logs that provided support for Ubuntu related problems. Qu et al. [175] present the Microsoft counterpart, known as the MSDialog. Recently there have been several datasets released publicly for evaluating QA dialog systems. Choi et al. [176] introduce QuAC, or Question Answering in Context. It consists of 14 K information-seeking QA dialog, or 100 K questions in total. Reddy et al. [177] present another such dataset called the CoQAwhich consists of 127 K questions and answers, 8 K of which were obtained from seven different domains.

8.3. Data for Conversational Dialog Systems

Micro-blogging or social media websites provide most of the data for training and evaluating conversational dialog systems. Fortunately, we do have publicly available datasets for conversational dialog systems. The Twitter corpus, which inspired the work of Ritter et al. [138], consists of 1.3 million conversations that were made publicly available. Another Twitter based dataset available (https://www.microsoft.com/en-us/download/details.aspx?id=52375) (accessed on 3 October 2021) is a collection of 4232 three-step conversational snippets. Sordoni et al. [139] utilize the latter Twitter corpus to train an end-to-end system. The Twitter logs were labeled by crowd-sourcing to allow annotators to measure the quality of the response, given the context. Another corpus worth a mention is the Reddit Corpus which consists of 1.7 billion comments.

8.4. Evaluation Challenge

Conversational systems become more and more intelligent everyday. Evaluation challenges for dialog systems have become important for setting the current benchmark for state-of-the-art dialog systems. These evaluation challenges release the datasets used in the competition for further research. Kim et al. [178] present one of the popular challenges called Dialog state tracking challenge (DSTC). It has now reached its sixth edition and released DSTC (1–6). Pavlopoulos et al. [179] present another popular challenge which is the Conversational intelligence challenge (ConvAI). The initiative aims to unify the efforts towards building intelligent dialog systems. The tasks to performed be at the challenge include submitting a dialog system to carry on natural conversations with a human based on certain news articles. One of the most popular challenge is the Alexa Prize (https://developer.amazon.com/alexaprize) (accessed on 3 October 2021). Ram et al. [180] provide an overview of the plethora of techniques employed to build the dialog systems such as NLU, context modeling, dialog management, etc. The research and the work done is usually used to advance the current Alexa technology.

9. Discussion

The review study has emphasized the need to develop an evaluation method for dialog systems which can be used repeatedly without any human intervention. We observed that there are no generally accepted evaluation metrics for dialog systems. The two research questions posed at beginning of the study were:

1. What are the evaluation methods available for Dialog systems based on the structure of the dialog?
   To answer this question, we have performed a thorough search of evaluation methods and categorized them into empirical and user-based methods. We highlighted several evaluation methods for both task-oriented dialog systems and conversational agents. There was also a distinction made between automated quantitative metrics [23,24] and user evaluation methodologies [124,129]. It is easier to evaluate task-oriented systems as they need to complete a specific task, the efficiency of which could be measured. User satisfaction can also be modeled, as we have covered before, and can be use to automate the evaluation process. Conversational agents are a little trickier to evaluate, especially when reducing human effort. While evaluating conversational agents, we do not have a particular set task or a correct answer. The focus is on the 'quality' of the dialog. Human judges annotate the 'appropriateness' of a dialog. The current state-of-the-art for modelling appropriateness is by means of latent representations such as ADEM [129].
2. What is the requirement of an automated evaluation method for testing the usability of dialog systems?
   To elaborate on this research question, it is necessary that empirical metrics show high correlation to human judgements, or the evaluation would be out of context. As these metrics do not correlate strongly to human judgements [25], the need for a more comprehensive evaluation metrics became apparent. Scoring high on an empirical evaluation method does not guarantee that the system performs well if users cannot navigate the system easily. On the other hand, user-based evaluation is very subjective. Thus, a gap exists in the evaluation methods available for dialogue systems. There is considerable amount of work in performing evaluation of dialog system with existing metrics. Several of the evaluation methods require human intervention to perform feature engineering, manually annotate data, and double-check the results provided by evaluation metrics. Human labor is too expensive to obtain in most cases. Certain other factors such as cognitive load on the users and human annotators also need to be considered while performing evaluation. An automated evaluation system can considerably reduce human effort, while also being able to provide insight on system quality to improve usability of the system. Existing evaluation methods depend on the type of dialog system they are evaluating, thereby reducing generalization. There is also a lack of well-defined metrics for evaluation of conversational search agents. There is a long way to go in developing evaluation methods for conversational search systems. Current evaluation methods are either very vague or very specific. The future scope of this study is to develop a comprehensive evaluation system by finding some middle ground between empirical and user-based methods. In this study, we attempt to understand the process of performing comprehensive dialog evaluation, which requires automating the process, making it repeatable, and increasing correlation to human judgements. A semi-automated evaluation approach is provided in the study done by Kaushik et al. [124], which presents the framework for an implicit evaluation method. The framework encompasses various information retrieval (IR) and user-based factors such as user satisfaction, knowledge gain, cognitive load, and so on. There is a possibility for such a standard framework accepted by different research paradigms by drafting it into an Application Programming Interface (API). The advantage of this approach is that all the data is collected together and comparative study can be performed. The data can also be stored and used in the future for further analysis. The limitations of this approach is that there is still some human

effort required in the form of expert knowledge. Another approach is to develop an in-built evaluation tool with an interactive system. The system is evaluated with each user interaction and the results are sent directly to the analyser. This approach is time-saving, but it makes one-on-one comparison of dialog systems difficult. The system may be updated with each evaluation which may result in a bias, which could be subject to further investigation. Further exploration of factors which affect the conversation in multiple dimensions would be interesting. The issues and challenges discussed in this paper can be discussed by the future research community to develop better dialog systems that provide enhanced user experience.

10. Conclusions

There are a number of approaches for evaluating dialog systems. However, as stated in the research, no scalable and automated solution exists that does not necessitate considerable human participation. Human assessment may not always be a realistic alternative due to the difficulty in finding enough evaluators to complete the assignment. Each of the assessment methods mentioned can be improved upon and modified to generalize well to dialog systems. It is thus critical to establish an evaluation procedure that can be carried out without requiring too much human input at each phase. Furthermore, the differences in the applications of the dialog systems must be considered when constructing the assessment techniques, since a single measure cannot be applied to all systems. The future scope of this study is to develop a standard evaluation framework for dialog systems. It is crucial that dialog systems designed with different approaches (rule-based or machine learning) can be evaluated within a standard framework with similar parameters so that comparative studies can be performed in a controlled environment. The cognitive load on the user while using the system needs to be kept in mind during evaluation. The results provided by an evaluation should also be understandable to an analyzer or researcher without much expert knowledge, so as to not further increase their cognitive load. Analysis of evaluation results also requires a lot of time and it is prone to human errors. A standard framework will significantly ease the effort and make the evaluation less susceptible to human errors. It will allow comparative studies to be performed between different dialog systems. Automation will also save time by eliminating manual tasks, leaving more room for further experimentation. Thus, we can safely conclude that it is essential to develop a standard evaluation framework acceptable by the dialog research community and adaptable to an automated or semi-automated paradigm. Future prospective research of standard conversational evaluation metrics will set a benchmark for upcoming research in dialog systems. Moreover, the standard evaluation framework can further be designed and developed based on factors such as user experience, interactive experience, cognitive load, and more.

Author Contributions: The following work has been categorized under different authors name. Conceptualization: A.K. and S.Y.; Methodology: S.Y.; Validation: A.K. and S.Y.; Formal analysis: S.Y.; Investigation: S.Y. and A.K.; Data curation: S.Y.; writing—original draft preparation: S.Y.; Writing—review and editing: A.K.; Visualization: S.Y.; Supervision: A.K.; Project administration: A.K.; Funding acquisition: NA. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Xu, A.; Liu, Z.; Guo, Y.; Sinha, V.; Akkiraju, R. A new chatbot for customer service on social media. In Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems, Denver, CO, USA, 6–11 May 2017; pp. 3506–3510.
2. Quarteroni, S.; Manandhar, S. A chatbot-based interactive question answering system. In Proceedings of the 11th Workshop on the Semantics and Pragmatics of Dialogue 2007, Rovereto, Italy, 30 May–1 June 2007; pp. 83–90.

3. Prochaska, J.J.; Vogel, E.A.; Chieng, A.; Kendra, M.; Baiocchi, M.; Pajarito, S.; Robinson, A. A Therapeutic Relational Agent for Reducing Problematic Substance Use (Woebot): Development and Usability Study. *J. Med. Internet Res.* **2021**, *23*, e24850. [CrossRef]
4. Madhu, D.; Jain, C.N.; Sebastain, E.; Shaji, S.; Ajayakumar, A. A novel approach for medical assistance using trained chatbot. In Proceedings of the International Conference on Inventive Communication and Computational Technologies (ICICCT), Coimbatore, India, 10–11 March 2017; pp. 243–246.
5. Følstad, A.; Nordheim, C.B.; Bjørkli, C.A. What makes users trust a chatbot for customer service? An exploratory interview study. In *International Conference on Internet Science, Proceedings of the 5th International Conference, INSCI 2018, St. Petersburg, Russia, 24–26 October 2018*; Springer: Cham, Switzerland, 2018; pp. 194–208.
6. AbuShawar, B.; Atwell, E. ALICE chatbot: Trials and outputs. *Comput. Sist.* **2015**, *19*, 625–632. [CrossRef]
7. Weizenbaum, J. ELIZA—A computer program for the study of natural language communication between man and machine. *Commun. ACM* **1966**, *9*, 36–45. [CrossRef]
8. Csaky, R. Deep learning based chatbot models. *arXiv* **2019**, arXiv:1908.08835.
9. Shawar, B.A.; Atwell, E.S. Using corpora in machine-learning chatbot systems. *Int. J. Corpus Linguist.* **2005**, *10*, 489–516. [CrossRef]
10. Haristiani, N. Artificial Intelligence (AI) chatbot as language learning medium: An inquiry. In *Journal of Physics: Conference Series*; IOP Publishing: Bristol, UK, 2019; Volume 1387, p. 012020.
11. McTear, M.F.; Callejas, Z.; Griol, D. Toward a Technology of Conversation. In *The Conversational Interface*; Springer: Berlin/Heidelberg, Germany, 2016; Volume 6, pp. 25–50.
12. Deriu, J.; Rodrigo, A.; Otegi, A.; Echegoyen, G.; Rosset, S.; Agirre, E.; Cieliebak, M. Survey on evaluation methods for dialogue systems. *Artif. Intell. Rev.* **2021**, *54*, 755–810. [CrossRef] [PubMed]
13. Radlinski, F.; Craswell, N. A theoretical framework for conversational search. In Proceedings of the 2017 Conference on Conference Human Information Interaction and Retrieval, Oslo, Norway, 7–11 March 2017; pp. 117–126.
14. Wei, Z.; Liu, Q.; Peng, B.; Tou, H.; Chen, T.; Huang, X.J.; Wong, K.F.; Dai, X. Task-oriented dialogue system for automatic diagnosis. In Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), Melbourne, Australia, 15–20 July 2018; pp. 201–207.
15. Hoy, M.B. Alexa, Siri, Cortana, and more: An introduction to voice assistants. *Med. Ref. Serv. Q.* **2018**, *37*, 81–88. [CrossRef]
16. Siri. Available online: https://www.apple.com/siri/ (accessed on 6 October 2021).
17. Cortana. Available online: https://www.microsoft.com/en-us/cortana (accessed on 6 October 2021).
18. Amazon Alexa. Available online: https://alexa.amazon.com (accessed on 3 October 2021).
19. Kaushik, A.; Bhat Ramachandra, V.; Jones, G.J. An interface for agent supported conversational search. In Proceedings of the 2020 Conference on Human Information Interaction and Retrieval, Vancouver, British Columbia, 14–18 March 2020; pp. 452–456.
20. Chandra, Y.W.; Suyanto, S. Indonesian chatbot of university admission using a question answering system based on sequence-to-sequence model. *Procedia Comput. Sci.* **2019**, *157*, 367–374. [CrossRef]
21. Sreelakshmi, A.; Abhinaya, S.; Nair, A.; Nirmala, S.J. A question answering and quiz generation chatbot for education. In Proceedings of the 2019 Grace Hopper Celebration India (GHCI), Bangalore, India, 6–8 November 2019; pp. 1–6.
22. Cui, L.; Huang, S.; Wei, F.; Tan, C.; Duan, C.; Zhou, M. Superagent: A customer service chatbot for e-commerce websites. In Proceedings of the ACL 2017, System Demonstrations, Vancouver, BC, Canada, 30 July–4 August 2017; pp. 97–102.
23. Papineni, K.; Roukos, S.; Ward, T.; Zhu, W.J. Bleu: A method for automatic evaluation of machine translation. In Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics, Stroudsburg, PA, USA, 7–12 July 2002; pp. 311–318.
24. Lin, C.Y. ROUGE: A Packagefor Automatic Evaluation of Summaries. In Proceedings of the Workshop on Text Summarization Branches Out, Post2Conference Workshop of ACL, Barcelona, Spain, 25–26 July 2004; pp. 74–81.
25. Liu, C.W.; Lowe, R.; Serban, I.V.; Noseworthy, M.; Charlin, L.; Pineau, J. How not to evaluate your dialogue system: An empirical study of unsupervised evaluation metrics for dialogue response generation. *arXiv* **2016**, arXiv:1603.08023.
26. Gunasekara, C.; Kim, S.; D'Haro, L.F.; Rastogi, A.; Chen, Y.N.; Eric, M.; Hedayatnia, B.; Gopalakrishnan, K.; Liu, Y.; Huang, C.W.; et al. Overview of the ninth dialog system technology challenge: Dstc9. *arXiv* **2020**, arXiv:2011.06486.
27. Hara, S.; Kitaoka, N.; Takeda, K. Estimation Method of User Satisfaction Using N-gram-based Dialog History Model for Spoken Dialog System. In Proceedings of the LREC, Valletta, Malta, 17–23 May 2010.
28. Yang, Z.; Levow, G.A.; Meng, H. Predicting user satisfaction in spoken dialog system evaluation with collaborative filtering. *IEEE J. Sel. Top. Signal Process.* **2012**, *6*, 971–981. [CrossRef]
29. Walker, M.A.; Litman, D.J.; Kamm, C.A.; Abella, A. PARADISE: A framework for evaluating spoken dialogue agents. *arXiv* **1997**, arXiv:cmp-lg/9704004.
30. Malchanau, A.; Petukhova, V.; Bunt, H. Multimodal dialogue system evaluation: A case study applying usability standards. In *9th International Workshop on Spoken Dialogue System Technology*; Springer: Singapore, 2019; pp. 145–159.
31. Arora, S.; Batra, K.; Singh, S. Dialogue system: A brief review. *arXiv* **2013**, arXiv:1306.4134.
32. Fraser, N.; Gibbon, D.; Moore, R.; Winski, R. Assessment of interactive systems. In *Handbook of Standards and Resources for Spoken Language Systems*; Mouton de Gruyter: Berlin, Germany, 1998; pp. 564–615.
33. Oviatt, S. Multimodal interfaces. In *The Human-Computer Interaction Handbook*; CRC Press: Boca Raton, FL, USA, 2007; pp. 439–458.
34. Klopfenstein, L.C.; Delpriori, S.; Malatini, S.; Bogliolo, A. The rise of bots: A survey of conversational interfaces, patterns, and paradigms. In Proceedings of the 2017 Conference on Designing Interactive Systems, Edinburgh, UK, 10–14 June 2017; pp. 555–565.

35. McTear, M.; Callejas, Z.; Griol, D. The dawn of the conversational interface. In *The Conversational Interface*; Springer: Berlin/Heidelberg, Germany, 2016; pp. 11–24.
36. Allen, J. *Natural Language Understanding*; Benjamin-Cummings Publishing Co., Inc.: San Francisco, CA, USA, 1988.
37. Ravuri, S.; Stoicke, A. A comparative study of neural network models for lexical intent classification. In Proceedings of the IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU), Scottsdale, AZ, USA, 13–17 December 2015; pp. 368–374.
38. Nadeau, D.; Sekine, S. A survey of named entity recognition and classification. *Lingvisticae Investig.* **2007**, *30*, 3–26. [CrossRef]
39. Aimin, F.C.H. Automatic recognition of natural language based on pattern matching. *Comput. Eng. Appl.* **2006**.
40. Lee, G.G.; Seo, J.; Lee, S.; Jung, H.; Cho, B.H.; Lee, C.; Kwak, B.K.; Cha, J.; Kim, D.; An, J.; et al. SiteQ: Engineering high performance QA system using lexico-semantic pattern matching and shallow NLP. In Proceedings of the TREC, Gaithersburg, MD, USA, 13–16 November 2001.
41. Chatterjee, N.; Kaushik, N. RENT: Regular expression and NLP-based term extraction scheme for agricultural domain. In *Proceedings of the International Conference on Data Engineering and Communication Technology*; Springer: Singapore, 2017; pp. 511–522.
42. Ranjan, N.; Mundada, K.; Phaltane, K.; Ahmad, S. A Survey on Techniques in NLP. *Int. J. Comput. Appl.* **2016**, *134*, 6–9. [CrossRef]
43. Huyck, C.R.; Lytinen, S.L. Efficient heuristic natural language parsing. In Proceedings of the AAAI, Washington, DC, USA, 11–15 July 1993; pp. 386–391.
44. Charras, F.; Duplessis, G.D.; Letard, V.; Ligozat, A.L.; Rosset, S. Comparing system-response retrieval models for open-domain and casual conversational agent. In Proceedings of the Second Workshop on Chatbots and Conversational Agent Technologies (WOCHAT@ IVA2016), Los Angeles, CA, USA, 20 September 2016.
45. Duplessis, G.D.; Letard, V.; Ligozat, A.L.; Rosset, S. Purely corpus-based automatic conversation authoring. In Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16), Portoroz, Slovenia, 23–28 May 2016; pp. 2728–2735.
46. Goldberg, Y.; Levy, O. word2vec Explained: Deriving Mikolov et al.'s negative-sampling word-embedding method. *arXiv* **2014**, arXiv:1402.3722.
47. McCormick, C. Word2vec Tutorial—The Skip-Gram Model. 2016. Available online: http://mccormickml.com/2016/04/19/word2vec-tutorial-the-skip-gram-model (accessed on 3 October 2021).
48. Mikolov, T.; Sutskever, I.; Chen, K.; Corrado, G.S.; Dean, J. Distributed representations of words and phrases and their compositionality. In Proceedings of the 26th International Conference on Neural Information Processing Systems (NIPS'13)-Volume 2, Lake Tahoe, NV, USA, 5–10 December 2013; pp. 3111–3119.
49. Schulert, A.J.; Rogers, G.T.; Hamilton, J.A. ADM—A dialog manager. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, San Francisco, CA, USA, April 1985; pp. 177–183.
50. Williams, J.D.; Henderson, M.; Raux, A.; Thomson, B.; Black, A.; Ramachandran, D. The dialog state tracking challenge series. *AI Mag.* **2014**, *35*, 121–124. [CrossRef]
51. Xu, P.; Hu, Q. An end-to-end approach for handling unknown slot values in dialogue state tracking. *arXiv* **2018**, arXiv:1805.01555.
52. McTear, M. The Role of Spoken Dialogue in User—Environment Interaction. In *Human-Centric Interfaces for Ambient Intelligence*; Elsevier: Amsterdam, The Netherlands, 2010; pp. 225–254.
53. Kobayashi, M.; Takeda, K. Information retrieval on the web. *ACM Comput. Surv. (CSUR)* **2000**, *32*, 144–173. [CrossRef]
54. Abdul-Kader, S.A.; Woods, J. Question answer system for online feedable new born Chatbot. In Proceedings of the Intelligent Systems Conference (IntelliSys), London, UK, 7–8 September 2017; pp. 863–869.
55. Maroengsit, W.; Piyakulpinyo, T.; Phonyiam, K.; Pongnumkul, S.; Chaovalit, P.; Theeramunkong, T. A Survey on Evaluation Methods for Chatbots. In Proceedings of the 7th International Conference on Information and Education Technology, Aizu-Wakamatsu, Japan, 29–31 March 2019; pp. 111–119.
56. Santhanam, S.; Shaikh, S. Towards best experiment design for evaluating dialogue system output. *arXiv* **2019**, arXiv:1909.10122.
57. Bartl, A.; Spanakis, G. A retrieval-based dialogue system utilizing utterance and context embeddings. In Proceedings of the 16th IEEE International Conference on Machine Learning and Applications (ICMLA), Cancun, Mexico, 18–21 December 2017; pp. 1120–1125.
58. Arora, P.; Kaushik, A.; Jones, G.J. DCU at the TREC 2019 Conversational Assistance Track. In Proceedings of the TREC, Gaithersburg, MD, USA, 13–15 November 2019.
59. Kaushik, A.; Ramachandra, V.B.; Jones, G.J. DCU at the FIRE 2020 Retrieval from Conversational Dialogues (RCD) task. In Proceedings of the FIRE 2020: 12th meeting of Forum for Information Retrieval Evaluation, Hyderabad, India, 16–20 December 2020; pp. 788–805.
60. Tetreault, J.; Filatova, E.; Chodorow, M. Rethinking grammatical error annotation and evaluation with the Amazon Mechanical Turk. In Proceedings of the NAACL HLT 2010 Fifth Workshop on Innovative Use, Los Angeles, CA, USA, 5 June 2010; pp. 45–48.
61. Satav, A.G.; Ausekar, A.B.; Bihani, R.M.; Shaikh, A. A proposed natural language query processing system. *Int. J. Sci. Appl. Inf. Technol.* **2014**, *3*. Available online: https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.458.8145&rep=rep1&type=pdf (accessed on 26 October 2021).
62. McDonald, D.D. Natural Language Generation. *Handb. Nat. Lang. Process.* **2010**, *2*, 121–144.
63. Bateman, J.; Zock, M. Natural language generation. In *The Oxford Handbook of Computational Linguistics*; Oxford University Press: Cambridge, UK, 2003.

64. Colby, K.M. Modeling a paranoid mind. *Behav. Brain Sci.* **1981**, *4*, 515–534. [CrossRef]
65. Lemon, O.; Pietquin, O. Machine learning for spoken dialogue systems. In Proceedings of the European Conference on Speech Communication and Technologies (Interspeech'07), Antwerp, Belgium, 27–31 August 2007; pp. 2685–2688.
66. Inui, N.; Koiso, T.; Nakamura, J.; Kotani, Y. Fully corpus-based natural language dialogue system. In Proceedings of the Natural Language Generation in Spoken and Written Dialogue, AAAI Spring Symposium, Stanford, CA, USA, 24–26 March 2003; pp. 1–3.
67. Oh, A.; Rudnicky, A. Stochastic language generation for spoken dialogue systems. In Proceedings of the ANLP-NAACL 2000 Workshop: Conversational Systems, Washington, DC, USA, 4 May 2000; pp. 27–32.
68. Zhang, Z.; Takanobu, R.; Zhu, Q.; Huang, M.; Zhu, X. Recent advances and challenges in task-oriented dialog systems. In *Science China Technological Sciences*; Springer: Berlin/Heidelberg, Germany, 16 September 2020; pp. 1–17.
69. Chen, P.; Lu, Y.; Peng, Y.; Liu, J.; Xu, Q. Identification of Students' Need Deficiency Through a Dialogue System. In *International Conference on Artificial Intelligence in Education, Proceedings of the 21st International Conference, AIED 2020, Ifrane, Morocco, 6–10 July 2020*; Springer: Cham, Switzerland, 2020; pp. 59–63.
70. Wen, T.H.; Vandyke, D.; Mrksic, N.; Gasic, M.; Rojas-Barahona, L.M.; Su, P.H.; Ultes, S.; Young, S. A network-based end-to-end trainable task-oriented dialogue system. *arXiv* **2016**, arXiv:1604.04562.
71. Chiba, Y.; Nose, T.; Kase, T.; Yamanaka, M.; Ito, A. An analysis of the effect of emotional speech synthesis on non-task-oriented dialogue system. In Proceedings of the 19th Annual SIGdial Meeting on Discourse and Dialogue, Melbourne, Australia, 12–14 July 2018; pp. 371–375.
72. Niculescu, A.I.; Jiang, R.; Kim, S.; Yeo, K.H.; D'Haro, L.F.; Niswar, A.; Banchs, R.E. SARA: Singapore's automated responsive assistant, a multimodal dialogue system for touristic information. In *International Conference on Mobile Web and Information Systems, Proceedings of the 11th International Conference, MobiWIS 2014, Barcelona, Spain, 27–29 August 2014*; Springer: Cham, Switzerland, 2014; pp. 153–164.
73. Zhang, Y.; Chen, X.; Ai, Q.; Yang, L.; Croft, W.B. Towards conversational search and recommendation: System ask, user respond. In Proceedings of the 27th ACM International Conference on Information and Knowledge Management, Torino Italy, 22–26 October 2018; pp. 177–186.
74. Vtyurina, A.; Savenkov, D.; Agichtein, E.; Clarke, C.L. Exploring conversational search with humans, assistants, and wizards. In Proceedings of the 2017 CHI Conference Extended Abstracts on Human Factors in Computing Systems, Denver, CO, USA, 6–11 May 2017; pp. 2187–2193.
75. Cahn, J. *CHATBOT: Architecture, Design, & Development*; University of Pennsylvania School of Engineering and Applied Science Department of Computer and Information Science: Philadelphia, PA, USA, 2017.
76. Turing, A.M. Mind. *Mind* **1950**, *59*, 433–460. [CrossRef]
77. Kenny, P.; Parsons, T.; Gratch, J.; Rizzo, A. Virtual humans for assisted health care. In Proceedings of the 1st International Conference on PErvasive Technologies Related to Assistive Environments, Athens, Greece, 16–18 July 2008; pp. 1–4.
78. Tavarnesi, G.; Laus, A.; Mazza, R.; Ambrosini, L.; Catenazzi, N.; Vanini, S.; Tuggener, D. Learning with Virtual Patients in Medical Education. In Proceedings of the EC-TEL (Practitioner Proceedings), Leeds, UK, 3–6 September 2018.
79. Serban, I.V.; Sordoni, A.; Bengio, Y.; Courville, A.; Pineau, J. Hierarchical neural network generative models for movie dialogues. *arXiv* **2015**, arXiv:1507.04808.
80. Medsker, L.R.; Jain, L. Recurrent neural networks. *Des. Appl.* **2001**, *5*, 64–67.
81. Vinyals, O.; Le, Q. A neural conversational model. *arXiv* **2015**, arXiv:1506.05869.
82. Serban, I.; Sordoni, A.; Bengio, Y.; Courville, A.; Pineau, J. Building end-to-end dialogue systems using generative hierarchical neural network models (2015). *arXiv* **2016**, arXiv:1507.04808.
83. Devlin, J.; Chang, M.W.; Lee, K.; Toutanova, K. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv* **2018**, arXiv:1810.04805.
84. Liu, Y.; Ott, M.; Goyal, N.; Du, J.; Joshi, M.; Chen, D.; Levy, O.; Lewis, M.; Zettlemoyer, L.; Stoyanov, V. Roberta: A robustly optimized bert pretraining approach. *arXiv* **2019**, arXiv:1907.11692.
85. Sanh, V.; Debut, L.; Chaumond, J.; Wolf, T. DistilBERT, a distilled version of BERT: Smaller, faster, cheaper and lighter. *arXiv* **2019**, arXiv:1910.01108.
86. Lan, Z.; Chen, M.; Goodman, S.; Gimpel, K.; Sharma, P.; Soricut, R. ALBERT: A Lite BERT for Self-supervised Learning of Language Representations. *arXiv* **2020**, arXiv:cs.CL/1909.11942.
87. Martin, L.; Muller, B.; Suárez, P.J.O.; Dupont, Y.; Romary, L.; de la Clergerie, É.V.; Seddah, D.; Sagot, B. Camembert: A tasty french language model. *arXiv* **2019**, arXiv:1911.03894.
88. Antoun, W.; Baly, F.; Hajj, H. AraBERT: Transformer-based model for Arabic language understanding. *arXiv* **2020**, arXiv:2003.00104.
89. Gonen, H.; Ravfogel, S.; Elazar, Y.; Goldberg, Y. It's not Greek to mBERT: Inducing Word-Level Translations from Multilingual BERT. *arXiv* **2020**, arXiv:2010.08275.
90. Lee, C.; Jung, S.; Kim, S.; Lee, G.G. Example-based dialog modeling for practical multi-domain dialog system. *Speech Commun.* **2009**, *51*, 466–484. [CrossRef]
91. Baxter, G.J.; Blythe, R.A.; Croft, W.; McKane, A.J. Utterance selection model of language change. *Phys. Rev. E* **2006**, *73*, 046118. [CrossRef]
92. Denkowski, M.; Lavie, A. Meteor 1.3: Automatic metric for reliable optimization and evaluation of machine translation systems. In Proceedings of the Sixth Workshop on Statistical Machine Translation, Scotland, UK, 30–31 July 2011; pp. 85–91.

93. Duplessis, G.D.; Charras, F.; Letard, V.; Ligozat, A.L.; Rosset, S. Utterance retrieval based on recurrent surface text patterns. In *European Conference on Information Retrieval, Proceedings of the 39th European Conference on IR Research, ECIR 2017, Aberdeen, UK, 8–13 April 2017*; Springer: Cham, Switzerland, 2017; pp. 199–211.
94. Bouziane, A.; Bouchiha, D.; Doumi, N.; Malki, M. Question answering systems: Survey and trends. *Procedia Comput. Sci.* **2015**, *73*, 366–375. [CrossRef]
95. Yang, Y.; Yih, W.t.; Meek, C. Wikiqa: A challenge dataset for open-domain question answering. In Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, Lisbon, Portugal, 19–21 September 2015; pp. 2013–2018.
96. Oniani, D.; Wang, Y. A qualitative evaluation of language models on automatic question-answering for COVID-19. In Proceedings of the 11th ACM International Conference on Bioinformatics, Computational Biology and Health Informatics, Virtual Event, USA, 21–24 September 2020; pp. 1–9.
97. Piccinini, G. Turing's rules for the imitation game. *Minds Mach.* **2000**, *10*, 573–582. [CrossRef]
98. Sethi, F. FAQ (Frequently Asked Questions) ChatBot for Conversation. *Authorea Prepr.* **2020**, *8*. [CrossRef]
99. Rahman, J. Implementation of ALICE Chatbot as Domain Specific Knowledge Bot for BRAC U (FAQ Bot). Ph.D. Thesis, BRAC University, Dhaka, Bangladesh, 2012.
100. Lee, K.; Jo, J.; Kim, J.; Kang, Y. Can Chatbots Help Reduce the Workload of Administrative Officers?-Implementing and Deploying FAQ Chatbot Service in a University. In *International Conference on Human-Computer Interaction, Proceedings of the 21st International Conference, HCII 2019, Orlando, FL, USA, 26–31 July 2019*; Springer: Cham, Switzerland, 2019; pp. 348–354.
101. Van Rousselt, R. Natural language processing bots. In *Pro Microsoft Teams Development*; Springer: Berlin/Heidelberg, Germany, 2021; pp. 161–185.
102. Nagarhalli, T.P.; Vaze, V.; Rana, N. A Review of Current Trends in the Development of Chatbot Systems. In Proceedings of the 6th International Conference on Advanced Computing and Communication Systems (ICACCS), Coimbatore, India, 6–7 March 2020; pp. 706–710.
103. Su, M.H.; Wu, C.H.; Huang, K.Y.; Hong, Q.B.; Wang, H.M. A chatbot using LSTM-based multi-layer embedding for elderly care. In Proceedings of the International Conference on Orange Technologies (ICOT), Singapore, 8–10 December 2017; pp. 70–74.
104. Kuligowska, K. Commercial chatbot: Performance evaluation, usability metrics and quality standards of embodied conversational agents. *Prof. Cent. Bus. Res.* **2015**, *2*, 1–16. [CrossRef]
105. Baby, C.J.; Khan, F.A.; Swathi, J. Home automation using IoT and a chatbot using natural language processing. In Proceedings of the Innovations in Power and Advanced Computing Technologies (i-PACT), Vellore, India, 21–22 April 2017; pp. 1–6.
106. Lee, C.H.; Chen, T.Y.; Chen, L.P.; Yang, P.C.; Tsai, R.T.H. Automatic question generation from children's stories for companion chatbot. In Proceedings of the IEEE International Conference on Information Reuse and Integration (IRI), Salt Lake City, UT, USA, 6–9 July 2018; pp. 491–494.
107. Pichponreay, L.; Kim, J.H.; Choi, C.H.; Lee, K.H.; Cho, W.S. Smart answering Chatbot based on OCR and Overgenerating Transformations and Ranking. In Proceedings of the Eighth International Conference on Ubiquitous and Future Networks (ICUFN), Vienna, Austria, 5–8 July 2016; pp. 1002–1005.
108. D'silva, G.M.; Thakare, S.; More, S.; Kuriakose, J. Real world smart chatbot for customer care using a software as a service (SaaS) architecture. In Proceedings of the International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud)(I-SMAC), Tamil Nadu, India, 10–11 February 2017; pp. 658–664.
109. Choi, H.; Hamanaka, T.; Matsui, K. Design and implementation of interactive product manual system using chatbot and sensed data. In Proceedings of the IEEE 6th Global Conference on Consumer Electronics (GCCE), Nagoya, Japan, 24–27 October 2017; pp. 1–5.
110. Latif, S.; Cuayáhuitl, H.; Pervez, F.; Shamshad, F.; Ali, H.S.; Cambria, E. A Survey on Deep Reinforcement Learning for Audio-Based Applications. *arXiv* **2021**, arXiv:2101.00240.
111. Kaushik, A.; Loir, N.; Jones, G.J. Multi-view conversational search interface using a dialogue-based agent. In *European Conference on Information Retrieval, Proceedings of the 43rd European Conference on IR Research, ECIR 2021, Virtual Event, 28 March–1 April 2021*; Springer: Cham, Switzerland, 2021; pp. 520–524.
112. Bocklisch, T.; Faulkner, J.; Pawlowski, N.; Nichol, A. Rasa: Open source language understanding and dialogue management. *arXiv* **2017**, arXiv:1712.05181.
113. Krasakis, A.M.; Aliannejadi, M.; Voskarides, N.; Kanoulas, E. Analysing the effect of clarifying questions on document ranking in conversational search. In Proceedings of the 2020 ACM SIGIR on International Conference on Theory of Information Retrieval, Virtual Event, Norway, 14–17 September 2020; pp. 129–132.
114. Google Dialogflow. Available online: https://dialogflow.cloud.google.com/ (accessed on 3 October 2021).
115. Singh, A.; Ramasubramanian, K.; Shivam, S. Introduction to Microsoft Bot, RASA, and Google Dialogflow. In *Building an Enterprise Chatbot*; Springer: Berlin/Heidelberg, Germany, 2019; pp. 281–302.
116. Dialogflow. Available online: https://dialogflow.com/docs (accessed on 3 October 2021).
117. Intent. Available online: https://cloud.google.com/dialogflow/es/docs/intents-overview (accessed on 26 October 2021).
118. Lopatovska, I.; Rink, K.; Knight, I.; Raines, K.; Cosenza, K.; Williams, H.; Sorsche, P.; Hirsch, D.; Li, Q.; Martinez, A. Talk to me: Exploring user interactions with the Amazon Alexa. *J. Librariansh. Inf. Sci.* **2019**, *51*, 984–997. [CrossRef]
119. Harms, J.G.; Kucherbaev, P.; Bozzon, A.; Houben, G.J. Approaches for dialog management in conversational agents. *IEEE Internet Comput.* **2018**, *23*, 13–22. [CrossRef]

120. Li, J.; Monroe, W.; Ritter, A.; Galley, M.; Gao, J.; Jurafsky, D. Deep reinforcement learning for dialogue generation. *arXiv* **2016**, arXiv:1606.01541.
121. Zhao, T.; Eskenazi, M. Towards end-to-end learning for dialog state tracking and management using deep reinforcement learning. *arXiv* **2016**, arXiv:1606.02560.
122. Scheffler, K.; Young, S. Automatic learning of dialogue strategy using dialogue simulation and reinforcement learning. In Proceedings of the HLT, San Diego, CA, USA, 24–27 March 2002; Volume 2.
123. Dhingra, B.; Li, L.; Li, X.; Gao, J.; Chen, Y.N.; Ahmed, F.; Deng, L. Towards end-to-end reinforcement learning of dialogue agents for information access. *arXiv* **2016**, arXiv:1609.00777.
124. Kaushik, A.; Jones, G.J. A Conceptual Framework for Implicit Evaluation of Conversational Search Interfaces. *arXiv* **2021**, arXiv:2104.03940.
125. Jurcıcek, F.; Keizer, S.; Gašic, M.; Mairesse, F.; Thomson, B.; Yu, K.; Young, S. Real user evaluation of spoken dialogue systems using Amazon Mechanical Turk. In Proceedings of the INTERSPEECH, Florence, Italy, 27–31 August 2011; Volume 11.
126. Bradeško, L.; Mladenić, D. A survey of chatbot systems through a loebner prize competition. In Proceedings of the Slovenian Language Technologies Society Eighth Conference of Language Technologies, Ljubljana, Slovenia, 8–12 October 2012; Institut Jožef Stefan Ljubljana: Ljubljana, Slovenia, 2012; pp. 34–37.
127. Simpson, A.; Eraser, N.M. Black box and glass box evaluation of the SUNDIAL system. In Proceedings of the Third European Conference on Speech Communication and Technology, Berlin, Germany, 21–23 September 1993.
128. Holmes, S.; Moorhead, A.; Bond, R.; Zheng, H.; Coates, V.; McTear, M. WeightMentor: A new automated chatbot for weight loss maintenance. In Proceedings of the 32nd International BCS Human Computer Interaction Conference 32, Belfast, UK, 4–6 July 2018; pp. 1–5.
129. Lowe, R.; Noseworthy, M.; Serban, I.V.; Angelard-Gontier, N.; Bengio, Y.; Pineau, J. Towards an automatic turing test: Learning to evaluate dialogue responses. *arXiv* **2017**, arXiv:1708.07149.
130. Kannan, A.; Vinyals, O. Adversarial evaluation of dialogue models. *arXiv* **2017**, arXiv:1701.08198.
131. Lowe, R.; Serban, I.V.; Noseworthy, M.; Charlin, L.; Pineau, J. On the evaluation of dialogue systems with next utterance classification. *arXiv* **2016**, arXiv:1605.05414.
132. Li, J.; Galley, M.; Brockett, C.; Gao, J.; Dolan, B. A diversity-promoting objective function for neural conversation models. *arXiv* **2015**, arXiv:1510.03055.
133. Li, B.; Han, L. Distance weighted cosine similarity measure for text classification. In *International Conference on Intelligent Data Engineering and Automated Learning, Proceedings of the 14th International Conference, IDEAL 2013, Hefei, China, 20–23 October 2013*; Springer: Berlin/Heidelberg, Germany, 2013; pp. 611–618.
134. Peng, B.; Li, C.; Zhang, Z.; Zhu, C.; Li, J.; Gao, J. RADDLE: An Evaluation Benchmark and Analysis Platform for Robust Task-oriented Dialog Systems. *arXiv* **2020**, arXiv:2012.14666.
135. Tao, C.; Mou, L.; Zhao, D.; Yan, R. Ruber: An unsupervised method for automatic evaluation of open-domain dialog systems. In Proceedings of the AAAI Conference on Artificial Intelligence, New Orleans, LA, USA, 2–7 February 2018; Volume 32.
136. Xu, X.; Dušek, O.; Konstas, I.; Rieser, V. Better conversations by modeling, filtering, and optimizing for coherence and diversity. *arXiv* **2018**, arXiv:1809.06873.
137. Chen, S.F.; Beeferman, D.; Rosenfeld, R. *Evaluation Metrics for Language Models*; Carnegie Mellon University: Pittsburgh, PA, USA, 1980.
138. Ritter, A.; Cherry, C.; Dolan, B. Unsupervised modeling of twitter conversations. In Proceedings of the Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics, Los Angeles, CA, USA, 2–4 June 2010; pp. 172–180.
139. Sordoni, A.; Galley, M.; Auli, M.; Brockett, C.; Ji, Y.; Mitchell, M.; Nie, J.Y.; Gao, J.; Dolan, B. A neural network approach to context-sensitive generation of conversational responses. *arXiv* **2015**, arXiv:1506.06714.
140. Galley, M.; Brockett, C.; Sordoni, A.; Ji, Y.; Auli, M.; Quirk, C.; Mitchell, M.; Gao, J.; Dolan, B. deltaBLEU: A discriminative metric for generation tasks with intrinsically diverse targets. *arXiv* **2015**, arXiv:1506.06863.
141. Almeida, F.; Xexéo, G. Word embeddings: A survey. *arXiv* **2019**, arXiv:1901.09069.
142. Rudkowsky, E.; Haselmayer, M.; Wastian, M.; Jenny, M.; Emrich, Š.; Sedlmair, M. More than bags of words: Sentiment analysis with word embeddings. *Commun. Methods Meas.* **2018**, *12*, 140–157. [CrossRef]
143. Corley, C.; Mihalcea, R. Measures of text semantic similarity. In Proceedings of the ACL Workshop on Empirical Modeling of Semantic Equivalence, Ann Arbor, MI, USA, 30 June 2005.
144. Lintean, M.; Rus, V. Measuring semantic similarity in short texts through greedy pairing and word semantics. In Proceedings of the Twenty-Fifth International FLAIRS Conference, Marco Island, FL, USA, 23–25 May 2012.
145. Rus, V.; Lintean, M. An optimal assessment of natural language student input using word-to-word similarity metrics. In *International Conference on Intelligent Tutoring Systems, Proceedings of the 11th International Conference, ITS 2012, Chania, Crete, Greece, 14–18 June 2012*; Springer: Cham, Switzerland, 2012; pp. 675–676.
146. Foltz, P.W.; Kintsch, W.; Landauer, T.K. The measurement of textual coherence with latent semantic analysis. *Discourse Process.* **1998**, *25*, 285–307. [CrossRef]
147. Landauer, T.K.; Dumais, S.T. A solution to Plato's problem: The latent semantic analysis theory of acquisition, induction, and representation of knowledge. *Psychol. Rev.* **1997**, *104*, 211. [CrossRef]

148. Mitchell, J.; Lapata, M. Vector-based models of semantic composition. In Proceedings of the ACL-08: HLT, Columbus, OH, USA, 16–18 June 2008; pp. 236–244.
149. Forgues, G.; Pineau, J.; Larchevêque, J.M.; Tremblay, R. Bootstrapping dialog systems with word embeddings. In Proceedings of the Nips, Modern Machine Learning and Natural Language Processing Workshop, Montreal, QC, Canada, 12–13 December 2014; Volume 2.
150. Hardalov, M.; Koychev, I.; Nakov, P. Machine Reading Comprehension for Answer Re-Ranking in Customer Support Chatbots. *Information* **2019**, *10*, 82. [CrossRef]
151. Dhyani, M.; Kumar, R. An intelligent Chatbot using deep learning with Bidirectional RNN and attention model. *Mater. Today Proc.* **2021**, *34*, 817–824. [CrossRef]
152. Liu, Q.; Huang, J.; Wu, L.; Zhu, K.; Ba, S. CBET: Design and evaluation of a domain-specific chatbot for mobile learning. *Univers. Access Inf. Soc.* **2020**, *19*, 655–673. [CrossRef]
153. Callison-Burch, C.; Osborne, M.; Koehn, P. Re-evaluation the role of bleu in machine translation research. In Proceedings of the 11th Conference of the European Chapter of the Association for Computational Linguistics, Trento, Italy, 5–6 April 2006.
154. Lin, C.Y.; Och, F. Looking for a few good metrics: ROUGE and its evaluation. In Proceedings of the Ntcir Workshop, Tokyo, Japan, 2–4 June 2004.
155. Dutta, S.; Klakow, D. Evaluating a neural multi-turn chatbot using BLEU score. *Univ. Saarl.* **2019**, *10*, 1–12.
156. Banerjee, S.; Lavie, A. METEOR: An automatic metric for MT evaluation with improved correlation with human judgments. In Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization, Ann Arbor, MI, USA, 29 June 2005; pp. 65–72.
157. Adiwardana, D.; Luong, M.T.; So, D.R.; Hall, J.; Fiedel, N.; Thoppilan, R.; Yang, Z.; Kulshreshtha, A.; Nemade, G.; Lu, Y.; et al. Towards a human-like open-domain chatbot. *arXiv* **2020**, arXiv:2001.09977.
158. Jena, G.; Vashisht, M.; Basu, A.; Ungar, L.; Sedoc, J. Enterprise to computer: Star trek chatbot. *arXiv* **2017**, arXiv:1708.00818.
159. Engelbrecht, K.P.; Gödde, F.; Hartard, F.; Ketabdar, H.; Möller, S. Modeling user satisfaction with hidden Markov models. In Proceedings of the SIGDIAL 2009 Conference, London, UK, 11–12 September 2009; pp. 170–177.
160. Georgila, K.; Henderson, J.; Lemon, O. User simulation for spoken dialogue systems: Learning and evaluation. In *Interspeech*; Citeseer: Pittsburgh, PA, USA, 2006; pp. 1065–1068.
161. Schatzmann, J.; Weilhammer, K.; Stuttle, M.; Young, S. A survey of statistical user simulation techniques for reinforcement-learning of dialogue management strategies. *Knowl. Eng. Rev.* **2006**, *21*, 97–126. [CrossRef]
162. Kreyssig, F.; Casanueva, I.; Budzianowski, P.; Gasic, M. Neural user simulation for corpus-based policy optimisation for spoken dialogue systems. *arXiv* **2018**, arXiv:1805.06966.
163. Holmes, S.; Moorhead, A.; Bond, R.; Zheng, H.; Coates, V.; McTear, M. Usability testing of a healthcare chatbot: Can we use conventional methods to assess conversational user interfaces? In Proceedings of the 31st European Conference on Cognitive Ergonomics, Belfast, UK, 10–13 September 2019; pp. 207–214.
164. Lewis, J.R.; Sauro, J. The factor structure of the system usability scale. In *International Conference on Human Centered Design, Proceedings of the First International Conference, HCD 2009, Held as Part of HCI International 2009, San Diego, CA, USA, 19–24 July 2009*; Springer: Berlin/Heidelberg, Germany, 2009; pp. 94–103.
165. Schrepp, M. User experience questionnaire handbook. In *All You Need to Know to Apply the UEQ Successfully in Your Project*; UEQ: Weyhe, Germany, 2015
166. Sharma, M.; Kaushik, A.; Kumar, R.; Rai, S.K.; Desai, H.H.; Yadav, S. Communication is the universal solvent: Atreya bot—An interactive bot for chemical scientists. *arXiv* **2021**, arXiv:2106.07257.
167. Hajdinjak, M.; Mihelič, F. The PARADISE evaluation framework: Issues and findings. *Comput. Linguist.* **2006**, *32*, 263–272. [CrossRef]
168. Peras, D. Chatbot evaluation metrics. In Proceedings of the 36th International Scientific Conference on Economic and Social Development: Book of Proceedings, Zagreb, Hvatska, 14–15 December 2018; pp. 89–97.
169. Xu, Q.; Huang, G.; Yuan, Y.; Guo, C.; Sun, Y.; Wu, F.; Weinberger, K. An empirical study on evaluation metrics of generative adversarial networks. *arXiv* **2018**, arXiv:1806.07755.
170. Guo, F.; Metallinou, A.; Khatri, C.; Raju, A.; Venkatesh, A.; Ram, A. Topic-based evaluation for conversational bots. *arXiv* **2018**, arXiv:1801.03622.
171. DeVault, D.; Leuski, A.; Sagae, K. Toward learning and evaluation of dialogue policies with text examples. In Proceedings of the SIGDIAL 2011 Conference, Portland, OR, USA, 17–18 June 2011; pp. 39–48.
172. Gandhe, S.; Traum, D. A semi-automated evaluation metric for dialogue model coherence. In *Situated Dialog in Speech-Based Human-Computer Interaction*; Springer: Berlin/Heidelberg, Germany, 2016; pp. 217–225.
173. Serban, I.V.; Lowe, R.; Henderson, P.; Charlin, L.; Pineau, J. A survey of available corpora for building data-driven dialogue systems: The journal version. *Dialogue Discourse* **2018**, *9*, 1–49. [CrossRef]
174. Gasic, M.; Breslin, C.; Henderson, M.; Kim, D.; Szummer, M.; Thomson, B.; Tsiakoulis, P.; Young, S. POMDP-based dialogue manager adaptation to extended domains. In Proceedings of the SIGDIAL 2013 Conference, Metz, France, 22–24 August 2013; pp. 214–222.

175. Qu, C.; Yang, L.; Croft, W.B.; Trippas, J.R.; Zhang, Y.; Qiu, M. Analyzing and characterizing user intent in information-seeking conversations. In Proceedings of the 41st International ACM SIGIR Conference on Research & Development in Information Retrieval, Ann Arbor, MI, USA, 8–12 July 2018; pp. 989–992.
176. Choi, E.; He, H.; Iyyer, M.; Yatskar, M.; Yih, W.T.; Choi, Y.; Liang, P.; Zettlemoyer, L. Quac: Question answering in context. *arXiv* **2018**, arXiv:1808.07036.
177. Reddy, S.; Chen, D.; Manning, C.D. Coqa: A conversational question answering challenge. *Trans. Assoc. Comput. Linguist.* **2019**, *7*, 249–266. [CrossRef]
178. Kim, S.; D'Haro, L.F.; Banchs, R.E.; Williams, J.D.; Henderson, M. The fourth dialog state tracking challenge. In *Dialogues with Social Robots*; Springer: Berlin/Heidelberg, Germany, 2017; pp. 435–449.
179. Pavlopoulos, J.; Thain, N.; Dixon, L.; Androutsopoulos, I. Convai at semeval-2019 task 6: Offensive language identification and categorization with perspective and bert. In Proceedings of the 13th International Workshop on Semantic Evaluation, Minneapolis, MN, USA, 6–7 June 2019; pp. 571–576.
180. Ram, A.; Prasad, R.; Khatri, C.; Venkatesh, A.; Gabriel, R.; Liu, Q.; Nunn, J.; Hedayatnia, B.; Cheng, M.; Nagar, A.; et al. Conversational ai: The science behind the alexa prize. *arXiv* **2018**, arXiv:1801.03604.