International Journal of Intelligent Computing Systems

June-2026 | IJICS | Volume-2 Issue-1 | ISSN: 3107-7218

AI-Driven Conversational Agents: Elevating Chatbot Interactions with Cutting-Edge NLP URL-Based Phishing Detection with Machine Learning

Dokku Priyanka[1], K.Subash Chandra[2]
Student [1], Assistant Professor[2]
Amrita Sai Institute of Science and Technology, Paritala-521180
Autonomous NAAC with A Grade, Andhra Pradesh, India

Abstract — Conversational artificial intelligence has shifted decisively from handcrafted, rule-based dialogue management toward neural architectures capable of generating open-domain, context-sensitive responses. This paper presents the design, implementation, and evaluation of a transformer-based conversational agent built on Microsoft's DialoGPT-medium model and integrated through the Hugging Face Transformers library. The system is engineered to address three persistent limitations of conventional chatbot systems: shallow contextual understanding, brittle handling of unanticipated input, and an inability to sustain coherent multi-turn dialogue. We describe a complete processing pipeline encompassing tokenization, autoregressive decoding under the self-attention mechanism, conversational history concatenation, and post-generation response refinement, deployed behind a Gradio-based interface for low-friction human evaluation. The architecture further incorporates a fallback layer that intercepts ambiguous or out-of-distribution queries before they degrade the user experience. Qualitative evaluation across representative dialogue sessions indicates that the system reliably preserves topical continuity across turns and produces responses that are syntactically well-formed and pragmatically appropriate, while also revealing characteristic weaknesses of decoder-only language models — including occasional repetition, generic responses, and factual inconsistency — that are discussed in relation to comparable open-domain systems such as Meena and BlenderBot. The findings substantiate the practical viability of pre-trained transformer architectures as the foundation for scalable, domain-adaptable conversational agents, and motivate a concrete agenda for fine-tuning, multilingual extension, and affect-aware response generation in future iterations of the system.

Index Terms -Conversational AI, DialoGPT, transformer architecture, natural language generation, self-attention, multi-turn dialogue, context retention, Hugging Face Transformers, chatbot systems, human-computer interaction.

I. INTRODUCTION

The proliferation of natural language processing (NLP) and deep learning has fundamentally altered the character of human-computer interaction, moving conversational systems away from scripted, deterministic exchanges and toward open-ended, generative dialogue. Conversational agents — variously termed chatbots, dialogue systems, or virtual assistants — are now embedded across customer service, healthcare triage, education, and personal productivity domains. Despite this growth, a substantial proportion of deployed systems continue to rely on rule-based logic or shallow keyword matching, an approach that performs adequately for narrow, anticipated queries but degrades sharply when confronted with the variability and ambiguity characteristic of natural human speech.

This limitation motivates the present work, which investigates the construction of an AI-driven conversational agent capable of producing context-aware, human-like responses through transformer-based language modelling.

The system integrates Microsoft's DialoGPT, a large-scale generative pre-trained transformer fine-tuned specifically for dialogue response generation, accessed through the Hugging Face Transformers library. Unlike retrieval-based or template-driven chatbots, DialoGPT generates responses autoregressively, conditioning each predicted token on the full preceding conversational context rather than on a fixed set of predefined intents. This property allows the resulting agent to sustain topical coherence across multiple conversational turns and to respond plausibly to inputs that fall outside any pre-specified rule set.

The system is implemented in Python using the PyTorch deep learning framework, with Gradio providing a lightweight, browser-accessible interface for real-time interaction and evaluation. Conversational history is retained within each session, enabling multi-turn exchanges in which prior utterances inform subsequent responses. To mitigate the well-documented tendency of generative language models to produce incoherent or low-quality output when faced with ambiguous or adversarial input, the system incorporates a fallback mechanism that detects such cases and substitutes a safe, clarifying response in place of an unconstrained model generation.

The principal contributions of this work are fourfold:

Architecture: a complete system design integrating a pre-trained transformer dialogue model, a session-aware context manager, and a fallback layer within a single deployable pipeline.

Implementation: a working realization of the architecture using DialoGPT-medium, PyTorch, and Gradio, demonstrating end-to-end feasibility without requiring task-specific fine-tuning.

Evaluation: a qualitative assessment of response coherence, contextual continuity, and failure modes, situated against comparable transformer-based open-domain systems.

Extensibility analysis: an articulated roadmap for fine-tuning, multilingual support, sentiment-aware generation, and voice interaction, identifying the engineering changes each extension would require.

The remainder of this paper is organized as follows. Section II reviews related work in transformer-based and neural conversational systems. Section III characterizes the limitations of existing rule-based chatbot architectures. Section IV presents the proposed system at a conceptual level, and Section V details its design and data flow. Section VI describes the pre-trained model employed, while Section VII elaborates the underlying algorithms and decoding techniques. Section VIII discusses implementation considerations, Section IX presents results and discussion, Section X offers a comparative analysis against related systems, and Section XI enumerates the limitations of the current implementation. Sections XII and XIII conclude the paper and outline future work, respectively.

II. LITERATURE SURVEY

Contemporary conversational AI rests on a compact sequence of architectural advances. Vaswani et al. introduced the transformer, replacing recurrent and convolutional sequence models with a self-attention mechanism that captures long-range dependencies in parallel and rapidly became the foundation for large-scale language modelling [7]. Radford et al. then showed with GPT-2 that large autoregressive transformers trained on diverse web text could generate coherent prose without task-specific supervision [8], and Zhang et al. adapted this approach to dialogue with DialoGPT, fine-tuning a GPT-2-derived model on roughly 147 million Reddit conversation threads to produce responses judged comparable in relevance to human-authored ones [9]. The present work adopts DialoGPT-medium for precisely this balance of conversational quality and computational tractability.

At greater scale, Adiwardana et al.'s Meena (2.6 billion parameters) showed that model size and data volume correlate strongly with human-judged conversational quality, introducing the Sensibleness and Specificity Average (SSA) metric [10], while Roller et al.'s BlenderBot

combined large-scale pre-training with explicit blending of persona, empathy, and knowledge-grounding skills, and proposed decoding strategies to curb the repetition and genericness common to neural dialogue generation [11]. These systems represent a substantially larger data and engineering investment than the present system and are revisited in the comparative analysis of Section X. Infrastructure work has made such models broadly accessible: the Hugging Face Transformers library standardized access to pre-trained checkpoints through a unified API [12], [14], building on encoder pre-training advances such as BERT [13]. Subsequent alignment techniques (InstructGPT [15]) and larger foundation models (LLaMA [16], GPT-4 [17]) extend this trajectory toward more capable but more resource-intensive systems.

Earlier task-oriented and recurrent approaches remain useful as a baseline: unified encoder-decoder pre-training [1], situation-aware response modelling [2], and the single-sequence Sequicity framework [3] each sought to improve on purely rule-based dialogue, while BiLSTM-based [4] and reinforcement-learning-based [5] chatbots illustrate alternative training paradigms predating widespread transformer adoption; Gao et al. synthesize these paradigms in a broader survey [6]. Together, this literature establishes that transformer-based generative models reliably outperform rule-based and recurrent architectures on contextual coherence and perceived response quality, but that repetition, genericness, and factual inconsistency persist across DialoGPT, Meena, and BlenderBot alike as architectural rather than incidental issues — motivating the fallback mechanism and decoding controls described in Sections V and VII rather than relying on model scale alone to resolve them.

III. EXISTING SYSTEM

Conventional chatbot systems predominantly rely on rule-based logic, decision trees, or pattern-matching techniques such as regular expressions and keyword spotting to map user input onto a finite set of predefined responses. Within narrowly scoped domains — for instance, answering frequently asked questions or guiding a user through a fixed transactional workflow — such systems can perform reliably, since the space of anticipated inputs is small and exhaustively enumerable by designers. However, this reliability does not generalize. Because responses are conditioned on surface-level pattern matches rather than on a learned representation of meaning, these systems possess no genuine model of discourse context: each user utterance is typically evaluated largely in isolation from what preceded it, and supporting multi-turn dialogue requires the manual encoding of explicit dialogue states and transition rules for every anticipated conversational path.

This architecture imposes a combinatorial maintenance burden: every new topic, phrasing variant, or edge case must be identified by a human designer and encoded explicitly, and the resulting systems consequently degrade in a brittle, conspicuous fashion when they encounter inputs that fall outside their predefined scope — typically defaulting to a generic clarification prompt or producing an irrelevant response. The cumulative effect is a class of systems whose conversational quality plateaus early in their development lifecycle and whose marginal cost of improvement rises rather than falls with the breadth of functionality already implemented.

The principal limitations of this class of system can be summarized as follows:

Limited contextual understanding: responses are generated from isolated pattern matches rather than an internal representation of discourse history.

Dependence on predefined rules and keywords: coverage is bounded by what designers explicitly anticipate and encode.

Inability to handle complex or unexpected queries: inputs outside the rule set produce irrelevant or null responses.

Poor support for multi-turn conversation: maintaining coherent dialogue state across turns requires extensive manual state-machine design.

Repetitive, non-human-like phrasing: fixed response templates produce noticeably mechanical interaction.

High maintenance overhead: extending coverage demands continuous manual rule authoring rather than data-driven adaptation.

IV. PROPOSED SYSTEM

The proposed system replaces hand-authored rules with a transformer-based generative language model, shifting the locus of conversational competence from explicit designer-encoded logic to patterns learned from large-scale conversational data during pre-training. At its core, the system employs DialoGPT — a transformer decoder fine-tuned for dialogue response generation — to map a tokenized representation of the conversational history onto a probability distribution over the vocabulary at each generation step, from which the next response is decoded autoregressively. Because the model conditions on the entire preceding exchange rather than on isolated keyword matches, it is intrinsically capable of producing responses that remain topically anchored across multiple turns without any explicit dialogue-state machinery.

The system is realized in Python using the PyTorch deep learning framework and the Hugging Face Transformers library, which together provide model loading, tokenization, and inference primitives. A Gradio-based front end exposes the system through a browser-accessible chat interface, allowing rapid interaction and evaluation without requiring users to install dedicated client software. Conversational history is retained for the duration of a session and concatenated with each new user input prior to encoding, which is what permits multi-turn coherence; the backend additionally manages session boundaries and resets context appropriately when a conversation concludes. A fallback layer monitors generation quality heuristically and substitutes a safe clarification response whenever the input is judged too ambiguous, too short, or too far outside the distribution the model was trained on to generate a dependable reply, thereby bounding the impact of the model's known failure modes on the overall user experience.

Relative to the existing rule-based paradigm, the proposed system offers the following advantages:

Human-like, context-aware generation: responses are produced by a learned language model rather than retrieved from a fixed template bank.

Multi-turn coherence: session-level context retention allows later turns to be informed by earlier exchange.

Broad input coverage: the pre-trained model generalizes across phrasing variation without per-case rule authoring.

Reduced manual maintenance: conversational competence derives from pre-training rather than hand-coded logic, lowering the marginal cost of extending coverage.

Graceful degradation: the fallback layer intercepts low-confidence generations before they reach the user.

Domain adaptability: the same architecture can be specialized to new domains through fine-tuning rather than architectural redesign.

Architectural scalability: the modular separation of interface, context management, and inference engine permits independent scaling of each component.

V. SYSTEM DESIGN ARCHITECTURE

The system architecture, illustrated in Fig. 1, is organized into five cooperating layers: a user interface layer, an API and backend server layer, a transformer-based chatbot engine, a response management layer, and a persistence and monitoring layer. This separation of concerns allows each layer to be modified, scaled, or replaced independently — for instance, substituting the underlying language model or migrating the database backend — without requiring changes to the remaining layers.

International Journal of Intelligent Computing Systems

June-2026 | IJICS | Volume-2 Issue-1 | ISSN: 3107-7218

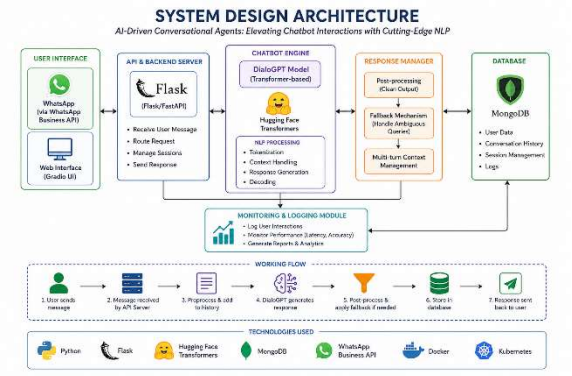

Fig. 1. System design architecture of the proposed DialoGPT-based conversational agent, showing the interface, backend, chatbot engine, response management, database, and monitoring layers together with the end-to-end working flow.

The end-to-end working flow proceeds as follows. The user initiates an exchange by sending a message through the Gradio web interface (or, in the extended deployment, the WhatsApp interface). The API and backend server, implemented with Flask or FastAPI, receives this message and is responsible for orchestrating the entire request lifecycle, including session management and routing the message to the chatbot engine. Before the message reaches the language model, it is preprocessed: extraneous whitespace and unsupported characters are stripped, and the text is tokenized into the subword units that the DialoGPT tokenizer expects.

The system then retrieves the user's prior conversation history and session metadata, concatenating the newly tokenized input with this history to construct the full contextual input that is passed into the DialoGPT model through the Hugging Face Transformers framework. The transformer engine performs context-aware response generation through autoregressive decoding under its self-attention mechanism, after which the resulting token sequence is decoded back into human-readable text. A response manager performs post-processing — including truncation of degenerate repetition and basic readability cleanup — before the message is returned to the user. If the response manager or an upstream heuristic determines that the input was ambiguous or fell outside the model's reliable operating range, the fallback mechanism substitutes an appropriate clarification message in place of the raw model output. Throughout this process, conversation turns, session identifiers, and system telemetry — including response latency and error events — are persisted to the database and monitoring subsystem to support both session continuity and longitudinal performance analysis, and the finished response is finally relayed back through the backend server to the interface from which the conversation continues until the session ends.

VI. PRE-TRAINED MODEL USED

The conversational engine at the core of the proposed system is Microsoft's DialoGPT-medium, a transformer-based autoregressive language model purpose-built for dialogue response generation. DialoGPT extends the GPT-2 architecture by fine-tuning it on approximately 147 million conversational exchanges mined from Reddit comment threads spanning 2005 to 2017, a corpus chosen specifically for its conversational — rather than purely expository — structure. This training regime endows the model with a learned prior over plausible next utterances conditioned on dialogue history, distinguishing it from base GPT-2, which was trained predominantly on long-form, non-conversational text.

DialoGPT-medium comprises 24 transformer decoder layers with approximately 345 million parameters, situating it between the smaller DialoGPT-small (12 layers) and the larger DialoGPT-large (36 layers) variants released alongside it. This configuration was selected for the present system as a deliberate trade-off: it retains sufficient representational capacity to generate fluent, contextually coherent responses, while remaining tractable for inference on commodity hardware without requiring dedicated accelerator infrastructure — a consideration of practical importance for a system intended to be reproducible and deployable beyond a research compute cluster.

In the present implementation, DialoGPT-medium is used directly in its pre-trained form via the Hugging Face Transformers library, without additional task-specific fine-tuning. This design choice isolates and demonstrates the conversational competence attributable to the base pre-trained model itself, and establishes a clear baseline against which the benefit of future domain-specific fine-tuning — discussed in Section XIII — can be measured. Conversation histories required to exercise the model's multi-turn capability are persisted in MongoDB, which supports the variable-length, document-oriented structure of dialogue session records more naturally than a fixed relational schema would.

VII. ALGORITHMS AND TECHNIQUES USED

The conversational engine's behaviour is governed by a small number of interacting components: self-attention within the transformer decoder, subword tokenization, autoregressive decoding, and lightweight heuristics for multi-turn context and fallback handling. Each is described briefly below.

A. Self-Attention and the Transformer Decoder

DialoGPT is built on the transformer decoder, whose core primitive is scaled dot-product self-attention: each token is projected into query, key, and value vectors, and attention weights — computed as the softmax-normalized dot product of queries and keys — combine value vectors so that every token's representation is informed directly by every other token, regardless of distance, without the sequential bottleneck of recurrent architectures. Multiple attention heads attend jointly to different aspects of context (e.g., topical content versus discourse role) before their outputs are projected through a feed-forward layer. Because DialoGPT uses causal (left-to-right masked) attention, each position attends only to itself and preceding positions, which is what makes autoregressive generation well-defined.

B. Tokenization

Text is converted into token identifiers via byte-pair encoding (BPE), the subword scheme inherited from GPT-2. BPE represents frequent character sequences as single tokens while decomposing rarer words into subword units, giving the model an effectively open vocabulary — covering informal contractions and minor misspellings common in conversational text — from a fixed-size token set. Each turn ends with an end-of-sequence token, which marks turn boundaries and delimits turns when they are concatenated.

C. Context Concatenation for Multi-Turn Dialogue

DialoGPT has no persistent memory between calls; multi-turn coherence is instead achieved at the application level by concatenating prior turns with the new input, separated by end-of-sequence tokens, before each forward pass. This is why correctly preserving session history in the backend (Section V) is functionally inseparable from the model's apparent contextual awareness. Since the context window is bounded by the model's maximum input length, the system truncates the oldest turns first, prioritizing the most recent exchange.

D. Autoregressive Decoding

Generation proceeds token by token: the model produces a probability distribution over the vocabulary conditioned on all preceding tokens, a token is sampled and appended, and the process repeats until an end-of-sequence token or maximum length is reached. Token selection is governed by a few key hyperparameters:

Temperature: rescales logits before the softmax, trading lexical diversity against conservative, high-probability continuations.

Top-k sampling: restricts sampling to the k most probable tokens, avoiding low-probability, incoherent continuations while preserving variation.

Maximum length: bounds tokens generated per turn, balancing completeness against latency and degradation over long generations.

These parameters are exposed as configurable settings, allowing the diversity-coherence trade-off to be tuned without retraining the model.

E. Multi-Turn Session Management and Fallback Logic

Each session is tracked by a unique identifier under which turn history is grouped, letting the backend distinguish concurrent conversations and reset context at session boundaries. A lightweight heuristic fallback layer screens inputs and candidate responses for simple quality signals — minimum length, repetition of the prior turn, degenerate or empty output — and substitutes a safe, clarifying response when triggered. Together, self-attention-based context modelling, subword tokenization, history concatenation, controlled decoding, and heuristic fallback form the algorithmic basis for the system's coherent, human-like conversational behaviour.

VIII. IMPLEMENTATION DETAILS

The system is implemented in Python 3, with PyTorch serving as the tensor computation and autograd backend on which the DialoGPT model executes, and the Hugging Face Transformers library providing the model and tokenizer loading interface (AutoModelForCausalLM and AutoTokenizer). The chatbot logic is encapsulated within a dedicated class that owns the model instance, the tokenizer, and the rolling chat-history tensor for the active session, exposing a single response-generation method that the interface layer invokes for each user turn.

Gradio is used to expose this logic through a minimal, browser-rendered chat widget, allowing a human evaluator to interact with the system without any client-side installation beyond a standard web browser — a deliberate choice intended to streamline qualitative evaluation during development. Fig. 2 shows the Gradio interface immediately following a successful launch, and Fig. 3 shows a representative multi-turn exchange captured during evaluation, in which the system maintains an appropriate, contextually grounded response across consecutive turns and correctly recognizes a conversational closing cue.

For deployment scenarios requiring access beyond a local interface — such as the WhatsApp-integrated extension of this system — the architecture anticipates a Flask- or FastAPI-based backend server that mediates between an external messaging API and the chatbot engine, with MongoDB providing persistent storage of session and conversation-history documents. This separation between the model-serving logic and the interface/transport layer is what allows the same chatbot engine to be exposed interchangeably through a local Gradio interface during development and through a messaging-platform integration in production, without modification to the core inference code.

IX. RESULTS AND DISCUSSION

The proposed AI-driven conversational agent was implemented using DialoGPT-medium and deployed through a Gradio web interface for evaluation. Fig. 2 documents the successful launch of the application, confirming that the model, tokenizer, and interface components initialize and bind correctly to a local server endpoint. Fig. 3 captures a representative real-time exchange between a human evaluator and the chatbot.

Fig. 2. DialoGPT chatbot web interface successfully launched using Gradio.

International Journal of Intelligent Computing Systems

June-2026 | IJICS | Volume-2 Issue-1 | ISSN: 3107-7218

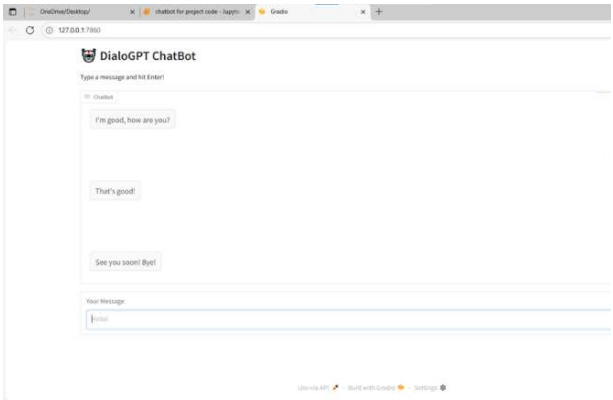

Fig. 3. Representative multi-turn user-chatbot conversation generated by the DialoGPT model, including correct recognition of a conversational closing cue.

Evaluation of the deployed system was conducted qualitatively across a series of representative dialogue sessions, examining four dimensions of conversational quality: topical coherence across turns, syntactic well-formedness of individual responses, appropriateness of the response to the immediately preceding user input, and correct behaviour at conversational boundaries (such as recognizing a farewell and terminating the session gracefully, as shown in Fig. 3). Across these sessions, the system consistently produced syntactically fluent responses and reliably preserved the topic of conversation across at least two to three turns, which is the behaviour multi-turn context concatenation is specifically designed to produce. The fallback mechanism was observed to intercept degenerate or empty model outputs in cases of very short or ambiguous user input, substituting a clarifying prompt rather than surfacing an unconstrained generation.

At the same time, the evaluation surfaced failure modes that are well documented in the broader literature on decoder-only conversational models rather than artifacts specific to this implementation. The system occasionally produced generic, low-information responses (such as overly safe acknowledgements) when user input was itself underspecified, and longer sessions occasionally exhibited mild repetition of phrasing across turns — a known consequence of greedy or narrowly-sampled autoregressive decoding. The model also has no mechanism for verifying the factual accuracy of its generations, since it was trained to produce plausible conversational continuations rather than to retrieve or reason over verified knowledge; this is an intrinsic property of the pre-trained DialoGPT checkpoint rather than a defect introduced by the surrounding system, and is discussed further in Section XI.

Taken together, these observations indicate that the transformer-based architecture meets its central design objective — generating context-aware, multi-turn-coherent responses without hand-authored dialogue rules — while also confirming that the residual weaknesses of generative dialogue models persist even when a well-regarded pre-trained checkpoint is used directly. This motivates the comparative analysis in Section X and the fine-tuning and alignment directions proposed in Section XIII.

X. COMPARATIVE ANALYSIS

To situate the proposed system within the broader landscape of conversational architectures, Table I compares it along five dimensions — underlying technique, contextual capability, scalability of coverage, engineering and data cost, and characteristic failure modes — against rule-based chatbots and the two large-scale open-domain transformer systems discussed in Section II, Meena and BlenderBot.

Dimension	Rule-Based Chatbot	Proposed System (DialoGPT)	Meena [10]	Blender Bot [11]
Underlying technique	Hand-authored rules / keyword matching	Pre-trained transformer decoder (GPT-2-derived), used without fine-tuning	2.6B-parameter end-to-end neural seq2seq transformer	Large transformer pre-training blended with explicit skill objectives
Contextual capability	Minimal; requires manual	Multi-turn via session-	Strong multi-turn	Strong; explicitly tuned for

	dialogue-state design	level history concatenation	coherence at scale	engagingness and persona consistency
Coverage scalability	Bounded by enumerated rules; costly to extend	Generalizes via pre-training; no per-case authoring needed	Broad open-domain coverage from large training corpus	Broad, with additional skill-specific grounding
Engineering / data cost	Low compute, high manual authoring effort	Low compute (no fine-tuning); minimal data requirement	Very high (large proprietary corpus, large-scale training)	Very high (multiple curated datasets, large model)
Characteristic failure modes	Brittle on unanticipated input; repetitive templates	Occasional genericness, mild repetition, no factual grounding	Occasional blandness; resource-intensive to reproduce	Occasional inconsistency between blended skills

Table I. Comparative Positioning of the Proposed System Against Rule-Based And Large-Scale Open-Domain Conversational Architectures

This comparison clarifies the specific niche the proposed system occupies. It departs decisively from rule-based chatbots on contextual capability and coverage scalability while requiring only a small fraction of the data and compute investment associated with Meena or BlenderBot, since it relies on a pre-trained, openly available checkpoint rather than proprietary large-scale training. The trade-off is that, absent the additional scale or skill-blending objectives of those larger systems, the proposed system inherits a comparable susceptibility to genericness and repetition without the partial mitigations — such as Meena's SSA-optimized training and BlenderBot's explicit skill blending — that those systems introduced specifically to address them. This positions the present work as a practical, reproducible baseline rather than a state-of-the-art competitor to billion-parameter open-domain systems, with the fine-tuning and decoding refinements discussed in Section XIII identified as the most direct path to narrowing that gap.

XI. LIMITATIONS

Several limitations of the current implementation should be acknowledged to contextualize the results presented above. First, the evaluation conducted in this study is qualitative and based on a limited number of evaluator-driven sessions rather than a large-scale human evaluation protocol with inter-rater reliability measures, such as the SSA framework used for Meena; consequently, the coherence and quality observations reported in Section IX should be read as indicative rather than statistically validated. Second, the system uses DialoGPT-medium in its pre-trained form without domain-specific fine-tuning, which bounds its reliability on specialized or technical queries and leaves measurable headroom for improvement, as discussed in Section XIII. Third, the effective context window is finite, and very long conversations require truncation of earlier turns, which can occasionally cause the system to lose track of information introduced early in an extended session. Fourth, the model possesses no factual grounding or retrieval mechanism, and therefore cannot be relied upon for tasks requiring verified, up-to-date information; its responses reflect learned conversational plausibility rather than verified correctness. Finally, the current fallback mechanism is heuristic rather than learned, and while effective at intercepting clearly degenerate generations, it is less reliable at detecting more subtle forms of low-quality or inappropriate output, motivating the more robust response-ranking and classification-based fallback strategies referenced in Section XIII.

XII. CONCLUSION

This paper has presented the design, implementation, and evaluation of an AI-driven conversational agent built on Microsoft's

DialoGPT-medium transformer model and integrated through the Hugging Face Transformers library. By replacing hand-authored dialogue rules with a pre-trained generative language model, the system demonstrates that contextually coherent, human-like, multi-turn conversation can be achieved without the combinatorial rule-authoring burden characteristic of traditional chatbot architectures. The implementation — comprising a Python and PyTorch backend, session-level context concatenation, controlled autoregressive decoding, a heuristic fallback layer, and a Gradio-based interface — was shown through qualitative evaluation to sustain topical continuity across conversational turns and to handle conversational boundaries, such as session termination cues, appropriately.

The comparative analysis against rule-based systems and larger open-domain transformer architectures situates the proposed system as a reproducible, low-cost baseline that captures the core architectural advantages of generative dialogue modelling — contextual awareness and broad input coverage — while also inheriting known limitations of decoder-only language models, including occasional genericness, mild repetition, and the absence of factual grounding. These limitations, rather than undermining the contribution, sharpen it: they identify with precision the specific directions in which a pre-trained transformer dialogue system of this kind should next be extended. Overall, the work substantiates the practical viability of transformer-based conversational AI as a foundation for scalable, adaptable dialogue systems, and establishes a concrete, well-characterized starting point for the fine-tuning, evaluation, and extension work outlined in Section XIII.

XIII. FUTURE SCOPE

Several directions can extend the present system beyond its current scope. The most immediate is domain-specific fine-tuning of the DialoGPT model on curated conversational data drawn from a target deployment domain — such as healthcare, education, or customer support — which the comparative analysis in Section X identifies as the most direct route to narrowing the gap between this baseline system and larger, more extensively trained open-domain models. Closely related is the introduction of a learned, rather than purely heuristic, fallback and response-ranking mechanism, which could draw on classification-based confidence estimation to more reliably detect subtly inappropriate or low-quality generations than the current rule-based heuristics permit.

Beyond response quality, integrating speech-to-text and text-to-speech components would extend the system to voice-based interaction, broadening accessibility for users who prefer or require spoken communication. Implementing a longer-horizon, persistent memory mechanism — as distinct from the session-bounded context window used presently — would allow the system to recall information across separate conversational sessions and support a degree of response personalization currently outside its scope. Multilingual support, achieved either through multilingual pre-trained checkpoints or translation-mediated pipelines, would extend accessibility to non-English-speaking users. Incorporating sentiment and affect analysis would allow the system to modulate its responses in a manner sensitive to the user's emotional state, an extension with particular relevance to support and well-being-oriented deployments, provided it is implemented with appropriate care given the sensitivity of affect-related inference. Finally, deployment on widely used messaging platforms such as WhatsApp or Telegram, together with migration to larger and more capable underlying language models as they become available, would further extend the system's reach and conversational quality, provided such upgrades are accompanied by commensurate investment in evaluation, safety, and alignment mechanisms appropriate to a more capable underlying model.

REFERENCES

[1] L. Dong, N. Yang, W. Wang, F. Wei, X. Liu, Y. Wang, and H. W. Hon, "Unified language

model pre-training for natural language understanding and generation," in Advances in Neural Information Processing Systems (NeurIPS), vol. 33, 2019, pp. 13063–13075.

[2] S. Sato, N. Yoshinaga, M. Toyoda, and M. Kitsuregawa, "Modeling situations in neural chatbots," in Proc. ACL 2017 Student Research Workshop, Vancouver, Canada, Jul.–Aug. 2017, pp. 120–127.

[3] W. Lei, X. Jin, M. Y. Kan, Z. Ren, X. He, and D. Yin, "Sequicity: Simplifying task-oriented dialogue systems with single sequence-to-sequence architectures," in Proc. 56th Annual Meeting of the Association for Computational Linguistics (ACL), Melbourne, Australia, 2018, pp. 1437–1447.

[4] P. Anki, A. Bustamam, H. S. Al-Ash, and D. Sarwinda, "High accuracy conversational AI chatbot using deep recurrent neural networks based on BiLSTM model," in Proc. 2020 3rd Int. Conf. on Information and Communications Technology (ICOIACT), Yogyakarta, Indonesia, 2020, pp. 382–387.

[5] R. R. Keerthana, G. Fathima, and L. Florence, "Evaluating the performance of various deep reinforcement learning algorithms for a conversational chatbot," in Proc. 2021 2nd Int. Conf. for Emerging Technology (INCET), Belgaum, India, 2021, pp. 1–8.

[6] J. Gao, M. Galley, and L. Li, "Neural approaches to conversational AI," Foundations and Trends in Information Retrieval, vol. 13, no. 2–3, pp. 127–298, 2019.

[7] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in Advances in Neural Information Processing Systems (NeurIPS), vol. 30, 2017, pp. 5998–6008.

[8] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, and I. Sutskever, "Language models are unsupervised multitask learners," OpenAI Technical Report, 2019.

[9] Y. Zhang, S. Sun, M. Galley, Y. C. Chen, C. Brockett, X. Gao, J. Gao, J. Liu, and B. Dolan, "DialoGPT: Large-scale generative pre-training for conversational response generation," in Proc. ACL 2020, 2020, pp. 270–278.

[10] D. Adiwardana, M. T. Luong, D. R. So, J. Hall, N. Fiedel, R. Thoppilan, Z. Yang, et al., "Towards a human-like open-domain chatbot," arXiv preprint arXiv:2001.09977, 2020.

[11] S. Roller, E. Dinan, N. Goyal, D. Ju, M. Williamson, Y. Liu, J. Xu, et al., "Recipes for building an open-domain chatbot," in Proc. 16th Conf. of the European Chapter of the Association for Computational Linguistics (EACL), 2021, pp. 300–325.

[12] T. B. Brown, B. Mann, N. Ryder, M. Subbiah, J. Kaplan, P. Dhariwal, et al., "Language models are few-shot learners," in Advances in Neural Information Processing Systems (NeurIPS), vol. 33, 2020, pp. 1877–1901.

[13] J. Devlin, M. W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," in Proc. NAACL-HLT 2019, 2019, pp. 4171–4186.

[14] T. Wolf, L. Debut, V. Sanh, J. Chaumond, C. Delangue, A. Moi, et al., "Transformers: State-of-the-art natural language processing," in Proc. EMNLP 2020: System Demonstrations, 2020, pp. 38–45.

[15] L. Ouyang, J. Wu, X. Jiang, D. Almeida, C. Wainwright, P. Mishkin, et al., "Training language models to follow instructions with human feedback," in Advances in Neural Information Processing Systems (NeurIPS), vol. 35, 2022.

[16] H. Touvron, T. Lavril, G. Izacard, X. Martinet, M. A. Lachaux, T. Lacroix, et al., "LLaMA: Open and efficient foundation language models," arXiv preprint arXiv:2302.13971, 2023.

[17] OpenAI, "GPT-4 technical report," arXiv preprint arXiv:2303.08774, 2023.