J. Integr. Sci. Technol. 2025, 13(6), 1130

Journal of Integrated
SCIENCE & TECHNOLOGY

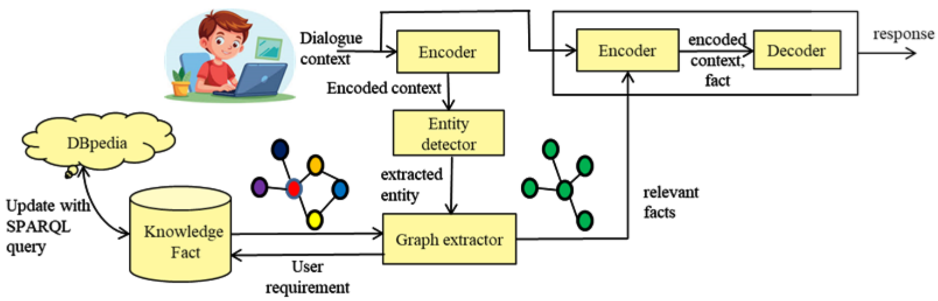
Knowledge graph based multi domain Dialogue System using Large Language models

Archana Patil,[1*] Shashikant Ghumbre,[2] Vahida Attar[1]

[1]Department of Computer Engineering, College of Engineering Pune, Savitribai Phule Pune University, Maharashtra, India.
[2]Computer Engineering, GCOE&R, Avasari, Savitribai Phule Pune University, Maharashtra, India.

Submitted on: 28-Dec-2024, Accepted and Published on: 21-Apr-2025

Article



ABSTRACT

Recently, with the rapid advancement of Large Language Models (LLMs), various applications such as Natural Language Generation (NLG) have garnered significant attention. NLG has found use across multiple domains, including machine translation, text summarization, and dialogue systems. Dialogue systems facilitate communication between humans and machines, and they are typically classified into task-oriented and open-domain systems. To enable dialogue systems to interact in a human-like manner, large volumes of training data are necessary. However, these systems often lack the reasoning capabilities needed for more intelligent interactions. To address this limitation, researchers have explored integrating external knowledge into dialogue systems, which has led to notable improvements in performance. Nevertheless, external knowledge sources are frequently static. As a result, when relevant information is missing, the system may fail to generate appropriate responses. This paper proposes the design of a multi-domain dialogue system that leverages publicly available knowledge graphs, such as DBpedia, to enhance response generation within predefined domains. When the system encounters a query for which it lacks the necessary knowledge, it dynamically retrieves the relevant information from DBpedia, updates its external knowledge base, and incorporates this data into the response generation process. The system utilizes transformer-based LLMs for generating responses and has been evaluated using two standard benchmark datasets as well as a custom dataset derived from DBpedia. Experimental results demonstrate notable improvements in performance.

Keywords: Natural language generation, Knowledge enhanced, Multi domain, Dialogue system, DBpedia.

INTRODUCTION

Machines can be made to converse with humans like how humans converse with each other, these applications are called dialogue system. Dialogue systems can broadly be divided as task oriented dialogue system and open domain dialogue system[1]. Completion of specific task is a goal of task-oriented dialogue system while open domain dialogue system aims at long term communication with user. Task oriented dialogue system can be single domain or multi domain dialogue system. Single domain dialogue system can handle conversation in single domain while multi domain dialogue system can handle both single and multiple domain conversation effectively aiming for task completion spanning to single or multiple domains.

*Corresponding Author: Archana Patil, College of Engineering, Pune, SPPU.
Tel: +91-020-25507323; Email: abp.comp@coeptech.ac.in

Cite as: *J. Integr. Sci. Technol., 2025, 13(6), 1130.*
URN:NBN:sciencein.jist.2025.v13.1130
DOI:10.62110/sciencein.jist.2025.v13.1130

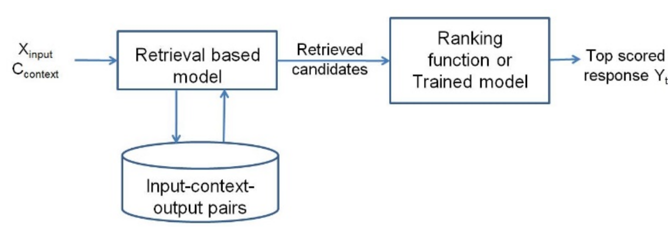


Figure 1. Retrieval based system.

Frameworks for building Dialogue system can be categorized as retrieval-based systems, generative systems and hybrid systems [1]. As shown in Figure 1, retrieval-based system stores input-context-output pairs. It takes X_{input} and $C_{context}$ as a query and retrieves a matching candidates list from a database using the scoring function and top-scored candidate response is then selected as model response Y_{output}. Classical learning-to-rank algorithms or neural matching models can be used for implementing the ranking function, ultimate aim of the system is to reduce the loss function and get required response. Response generated using retrieval-based system is fluent, grammatically correct and of high quality as systems make use of already stored responses and generates response for most matching input, but this technique does not work well with open domain dialogue system.

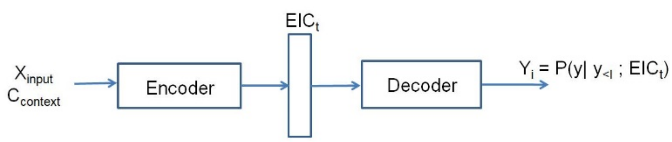


Figure 2. Generative system

Generative systems figure 2. generates response Y_{output} by generating y_i word by word conditioned on encoded input and context i.e. EIC_t and words generated till earlier timestep that is y_1 to y_{i-1}. A generative system can use functions like various attention mechanisms while generating response. Generative systems can produce novel responses unlike retrieval-based systems, but they suffer from the problem of grammatically incorrect sentences. Hybrid model combines both the earlier methods and uses two-stage method for response generation.

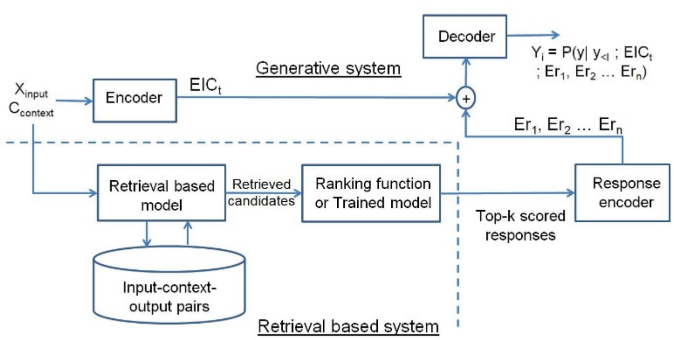


Figure 3. Hybrid system

Figure 3. hybrid system uses retrieval-based method to get the top-k sample response or relevant knowledge from the dataset for given query and context, then the generative system uses these top-k retrieved elements, input i.e. X_{input}, context i.e. $C_{context}$ and all words generated till earlier timestamp to generate word by word response for actual input. Hybrid systems combine advantages of both retrieval and generative based system.

Machines are not intelligent like humans to handle conversation efficiently and exhaustive training for handling conversation efficiently is not feasible. They can have effective conversation on the seen data as compared to unseen data. To boost the performance of dialogue system researchers have explored providing external relevant data which can be used by system for generating appropriate responses for the unseen input. Externally given data can be in structured or unstructured form.[2] Structured representation of data in triplet form as <sachin_tendulkar, birth_place, mumbai> generalized as < subject, predicate, object > where subject and object are two entities present in the system and predicate is relation between those entities. Triplet data can also have another structured representation called graph [3,4], [5]as shown in figure 4. Textual data like words, sentences, documents or pages is an example of unstructured data that can be given to dialogue system as external relevant knowledge for response generation. Data on Wikipedia [5] is continuously updated, DBpedia[6] dataset is extraction of data displayed on Wikipedia in Resource Description Framework (RDF) i.e. triplet data. DBpedia also facilitates firing queries to extracted data to get the data as per users requirement. This provides a good opportunity to use such dataset in dialogue system domain, where responses many times depend on factual data. Practically factual data can be updated as the time passes, DBpedia provides several ways of getting updated data. DBpedia data can be accessed using various methods like Linked Data, SPARQL Endpoint or RDF Dumps.[6]

Figure 4. Knowledge graph

Earlier designed dialogue systems are rule based systems, which have the problem of restricted or vague response generation. Machine learning techniques made diverse response generation possible in dialogue system, adding human characteristics like language understanding, contextual and fluent response generation to make dialogue system more intelligent was made possible due to development of deep learning technology. The beginning of this era was marked by seq-to-seq model, further advancement was done by incorporation of various techniques like attention, hierarchical encoder-decoder model using RNN, LSTM and GRU. Next benchmark was marked by transformer based pretrained large language model and are still dominating for designing of intelligent dialogue system. Our system proposes hybrid system which uses pretrained large language models for generating response for multi domain multi turn dialogue system. According to the given input and context, the designed system fetches information from externally provided knowledge fact and uses it for generating response using pretrained model for the given input.

LITERATURE SURVEY

Dialogue system are categorized as task oriented dialogue system and open domain dialogue system. The goal of the Task oriented dialogue system is to help user complete particular tasks like hotel bookings or taxi bookings etc., while the later system focuses on making conversations more engaging. Multi domain dialogue system allows user to accomplish composite task completion like taxi booking, restaurant booking etc. Conventional task-oriented dialogue systems are usually accompanied by extraneous task-specific knowledge fact which empowers them to generate informative conversation.[7–9] The knowledge fact in these systems can be structured in the form of triplets, knowledge graph, or unstructured documents.[2] Task oriented dialogue system uses traditional pipelined approach or end-to-end task-oriented dialogue system. Systems implemented with pipelined approach uses different modules for natural language understanding, dialogue state tracking, dialog policy learning and response generation which are trained separately and sequentially one after the other. But these systems require annotated data for training which is unavailable. The goal of end-to-end designed systems is to produce the response end-to-end while removing dependency on intermediate annotations [10]. It trains all components for natural language understanding, knowledge retrieval and natural language generation simultaneously making it difficult for all components to satisfy same goal of error reduction and it also makes the model heavy and difficult for training.

Recent advancement in transformer based pre-trained language model has evolved the development of task oriented dialogue system. Researchers have explored adding external knowledge fact to dialogue system to enhance performance of these systems, but knowledge retrieval which is a foundation of task-oriented dialogue system is a challenging task because there are no gold labels accessible to train a retriever. Static external knowledge fact creates hindrance in generation of appropriate response if corresponding knowledge is missing in knowledge fact. Also, availability of annotated data is an issue for designing a dialogue system. We collected dialogue dataset using entity, relation, objects present in DBpedia [6], which is a standard repository for storing the data from Wikipedia in a structured format. DBpedia was used for extracting data for six different domains, details given in upcoming section and then conversations based on entities, relations and objects extracted were created, and these conversations were labeled for existence of the entities, relations and objects present in conversation using a script. Script used extracted data from DBpedia for creating annotated data available for training designed system.

This work proposes a model consisting of three modules: Named entity recognizer (NER), graph extractor and response generator. Named entity recognizer is used to extract entity from dialogue history, extracted entity is used to retrieve relevant knowledge from external knowledge fact and these retrieved relevant entities along with the dialogue context are used for fine tuning large language models using knowledge augmentation method to generate appropriate response for the given input. Our system decouples the independent graph extractor from NER model and response generator model and encodes the dynamic knowledge extracted by graph extractor in multi domain settings for generating appropriate response for given input. This paper investigates the effectiveness of Large Pre-trained transformer-based model T5 [11]and BART [12] for generating responses in dialogue systems with external knowledge fact. The performance evaluation of the proposed model is done using metrics BLEU, F1-score on standard dataset MultiWOZ, Camrest and our curated dataset using DBpedia. To address the knowledge gap issue in our DBpedia curated data, we update it by firing a dynamic SPARQL query to DBpedia and update the existing knowledge fact. Results demonstrate that the performance of the proposed systems is comparable with existing models with the additional feature of updating knowledge.

Proposed model aims to solve the following listed issues:

- Multi domain dialogue system requires annotated data for training the model. The proposed system automates the annotation process by using data information extracted from DBpeida.
- Static external fact creates problems for response generation if knowledge is missing, model provides knowledge fact updating mechanism to solve the issue.
- The proposed model has an independent graph extractor decoupled from NER model and response generator model, making model training easy.

METHODOLOGY

Given a dialogue context and knowledge graph our model GraphConvAI aims to generate informative response. In a multi turn conversation system, dialogue context C_t at timestep 't' is a set of utterances between user and system till timestep 't' and is represented as $C_t = (U_0, R_0,..., U_t)$ where U_i (user utterance) and R_i (system response) represent sequence of words in i^{th} turn of conversation. $KG_d(K, V)$ represents complete knowledge graph for the domain which stores the data in the form of key value pairs for every entity in the domain.

We have extracted structured dataset from DBpedia belonging to six distinct interrelated domains: books(name, author, genre, publisher, year), films(name, genre, director, producer, cast), theatre(name, language, genre, writer, characters), music(name, artist, genre, writer, year), painting(name, artist, ownedby, year), and sculpture(name, madeof, location, artist). We have chosen a variety of domains so that the performance of the built system can be tested within the domain and cross domain as well. These domains were specifically chosen due to their interconnected nature and appropriateness for extraction from DBPedia. Data for predefined entities is extracted from DBpedia in the form of triplets (Resource Description format) with the help of SPARQL query. Extracted data is then converted into key-value to represent each distinct entity for every domain in the dataset. Extracted data was compiled and used for creating conversations manually. Conversations contained both user utterance and system response based on extracted entities and their interrelated data. Statistics for created conversations are shown in table 1.

Table 1. Conversation statistics

DBpedia extracted dataset	Number of samples
Train conversation pair	3181
Validation conversation pair	746
Test conversation pair	744
Avg. Turns / dialogue	5.32
Avg. Tokens / turn	9.52
Total number of turns	9342
Vacabulary	5334

Overall system architecture of proposed system is as shown in Figure 5, which mainly consists of NER model, Graph extractor and response generator. NER model takes dialogue history as input, from input it extracts entity and their corresponding labels. These extracted entities are used for extracting relevant facts from the externally provided knowledge fact and these extracted knowledge facts are dynamically used for generating relevant response for the given input. Ensuring precision in entity detection and information extraction and generating relevant response generation are the aim of the system. Proposed model was evaluated on the data extracted from DBpedia as well as MultiWOZ 2.1 and Camrest dataset.

reduce the cross-entropy loss for entity detection and the best version of the trained model was used to identify correct entities for unlabeled dataset. The result of NER model on different dataset is shown in table. 2, which ensures that the finely tuned model recognizes relevant entities and their labels to well understand the requirement of the user which is crucial task for extracting relevant knowledge from external knowledge.

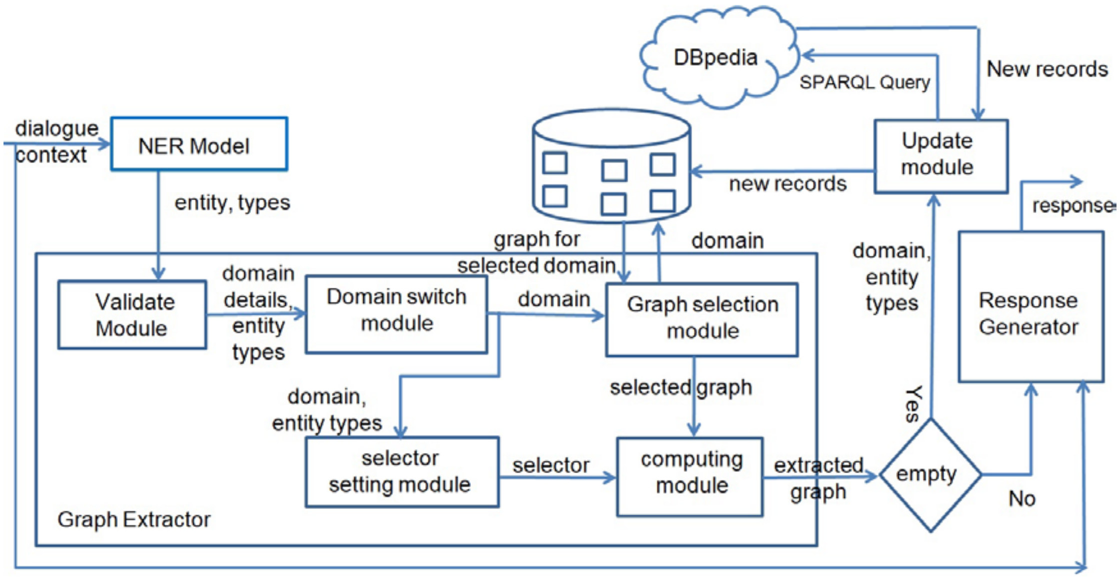


Figure 5. Overall system architecture.

I. Named Entity Recognition (NER) Model

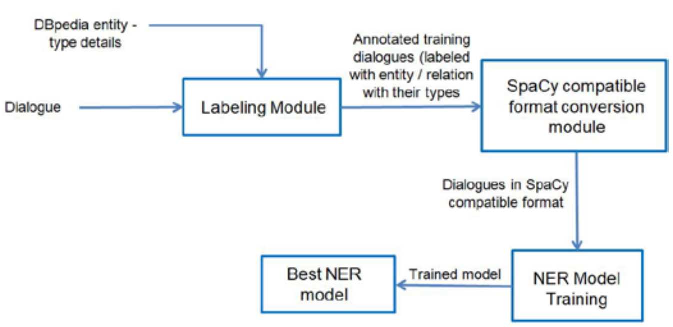


Figure 6. Conversation labeling and training of NER model.

Named entity recognition is important tasks in various NLP applications like question answering system, dialogue system etc., as it is necessary to understand what the requirement of user is before generating relevant response to the given input. As shown in Figure 6, to understand the requirement of user, conversations were carefully labeled for entities or their keys or values and their corresponding types via a labeling module which used structure of extracted data from DBpedia for labeling. These labeled conversations were pre-processed and converted to a format compatible with spaCy's NER model, and SpaCy's NER model was fine-tuned with these labeled conversations for extracting entities present in given input. The NER model was trained to

Table 2. Performance of NER model on various datasets.

Metrics	MultiWOZ	Camrest	DBpedia extracted dataset
Precision	0.7964	0.9456	0.9385
Recall	0.8619	0.9512	0.9302
F1-score	0.8278	0.9483	0.9343

II. Graph Extractor

The proposed system uses NER extracted entities for extracting relevant facts from the external knowledge graph. External knowledge for the domain is represented using entity and their key-value pairs. The designed system converts the external knowledge in two-dimensional matrix structure which helps simplifying information retrieval of the required data for the given input. Main steps involved in graph extractor are creating the graph, extracting the subgraph and graph computation. Detailed working of these components is given in the coming section. Extracted relevant facts are used by the response generation model for generating response for the given input.

a) Creating the graph

The process starts with initializing the graph for every domain, which is represented as a matrix structure as depicted in Figure. 7. Initially, a 2D matrix is created by considering distinct key-value pairs present in the external knowledge for the given domain. The matrix is then populated with zeros and ones. Each entity in external

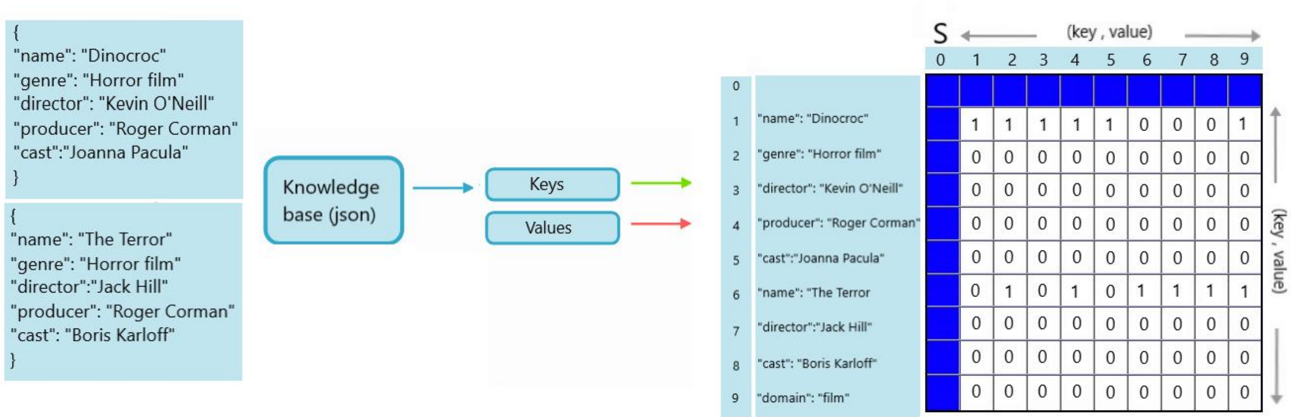


Figure 7. An example of graph creation where both rows and columns are distinct key-value pairs, where first row is padded and first column is 1 for values that are selected otherwise 0.

knowledge along with its key-value pairs corresponds to a row in the matrix, while each distinct key-value pair is represented as a column. If a key-value pair exists as a property in a record of the external knowledge, the corresponding cell in the matrix identified by the row and column, is set to 1; otherwise, it's set to 0. System uses separate graphs for each domain. This approach aids in reducing the computational load and enhances the graphical representations fidelity.

domain is passed to graph selection module and domain, along with the entity-types are passed to selector setting module for setting selector as per user requirement. Once the domain is identified its corresponding graph is loaded by graph selection module for further processing. A selector which represents 1 dimensional matrix with dimensions matching a row length of the graph in the selected domain is then initialized. Entity-type pair extracted by NER model are used to set appropriate cells 1 in the selector and

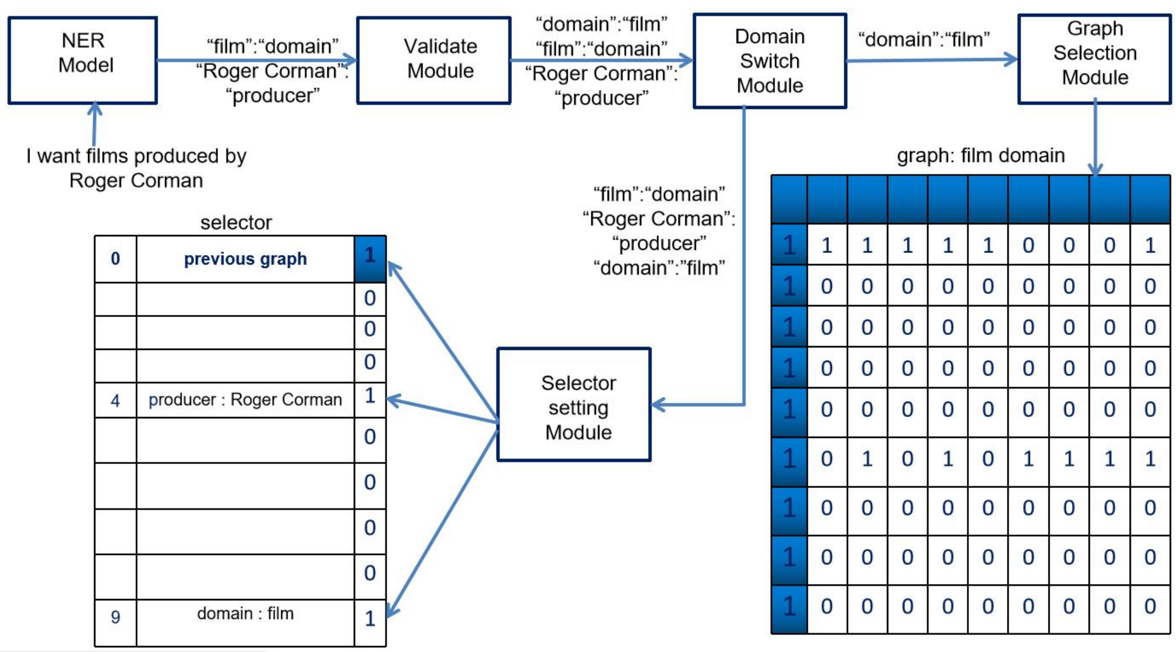


Figure 8. An overview of graph selection and setting selector for further computation.

b) Extracting the subgraph

Figure 8 illustrates, steps involved in choosing the correct graph and setting selector. First, entities and their corresponding types are extracted from the input using the trained NER model. Next step is to determine the domain of the extracted key-value pairs and to examine if domain switching is required or not, accordingly the now selector with extracted graph is ready for processing.

c) Graph Computation

Initially the first column in 2D graph matrix all are set to one as whole graph is selected and then model narrows down the graph to relevant entities according to the supplied input. As shown in Figure 9, the selector represented as a 1D matrix, and the graph represented as a 2D matrix and their product resultant 1D matrix helps to find entities present in knowledge fact and matching the given input requirement. Cell in the resulting 1D matrix represents an entity record. If the entity in a cell matches all the requirements in the selector, it indicates that this cell corresponds to a selected record, for example “the terror” and “the dinocroc” as in figure 9. The index of the selected record in the 1D matrix as shown in figure 9, corresponds to the row index in the graph as shown in figure 7. By using key-value pairs present in the selected row of the graph figure 7, the original record of the selected entity is reconstructed. As already discussed, the presence of a key-value pair in the row indicates that the corresponding property is present in the record and the required record is successfully extracted from the graph. If there are more than ‘k’ matching records, the top ‘k’ entities are selected using similarity and in cases where retrieved records are empty, indicating potential gaps in existing knowledge, model updates external knowledge graph as explained in update mechanism section. In short for extracting subgraph from larger graph, graph extractor of proposed model uses entities recognized by NER model, sets the selector accordingly and fetches all entities exactly matching the requirement in the given input. Performance of graph extractor for named entity linking is as shown in table 3. In case of entity missing in dataset, then proposed model prepares dynamic query using the extracted entities, fires it to DBpedia and fetches the required entity details fron DBpedia, updates the system knowledge graph accordingly.

Table 3. Named entity linking performance.

Metrics	MultiWOZ	Camrest	DBpedia extracted dataset
Precision	1	1	1
Recall	0.8619	0.9512	0.9302
F1-score	0.9258	0.9747	0.9638

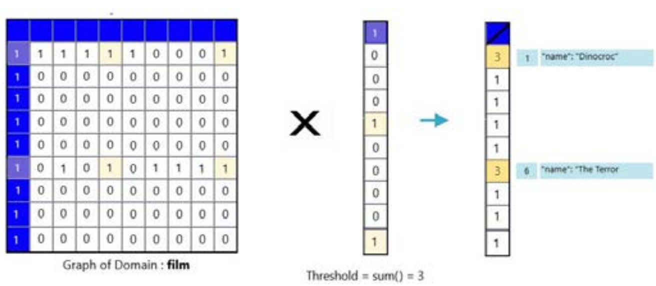


Figure 9. Computation Process of Graph, where 2D matrix is selected graph and 1D matrix is selector from Figure 8.

d) Update mechanism

During subgraph extraction from larger domain graph, system may face an issue of empty record extraction. In this case proposed model uses dynamic approach to update its domain knowledge graph. System uses the dialogue context to extract the domain of discourse which are retrieved by NER model and is used to design dynamic SPARQL query which is fired on DBpedia to get the matching data. After the query execution, relevant records are extracted, model selects 7 records and updates the current knowledge graph. The entities, features and values extracted from the retrieved records are added to domain graph enriching system’s understanding and enhancing its ability to comprehend similar contexts in the future. This iterative process of dynamic querying and knowledge integration serves to continually enhance external knowledge fact. By augmenting existing graph with new information, system creates an updated and enriched knowledge matrix, empowering proposed system to provide more accurate and comprehensive responses to user input in future.

III. Response generator

Model uses entity concatenation and knowledge augmentation approach for response generation using T5 and BART model. The response generation model uses an encoder-decoder architecture which encodes the dialogue context as well as relevant facts or extracted entities from graph extractor using encoder, while the decoder in response generation model is trained to generate informative response using the encoded context and relevant facts or entities. The dialogue context at the t^{th} turn in a multi-turn conversation system can be expressed as C_t, let KG_d represents complete knowledge graph and K_t represents graph extracted by graph extractor for the given dialogue context with ‘m’ entities. Each entity e_i has key-value attribute pairs (k_{ij}, v_{ij}) where ‘i’ is the i^{th} entity and ‘j’ is the j^{th} key-value attribute pair. Thus K_t can be represented as $K_t = (e_1, e_2, \ldots, e_m)$.

a) Encoder

Encoder of the response generator model concatenates the dialogue context with each retrieved entity separately to further enhance input representation for the response generation model as in equation 1. Hidden representations of retrieved entities are further concatenated to obtain final hidden state of encoder model H_t as shown in equation 2.

$$\boldsymbol{H_{t,i} = Transformer\_Enc([C_t\,; e_i])} \tag{1}$$

$$H_t = [H_{t,1};\ H_{t,2}; \ldots;\ H_{t,m}] \tag{2}$$

b) Decoder

Output of encoder H_t is taken as input by the decoder for generating response word by word. To generate the appropriate response, the decoder uses self attention for previously generated token and cross attention for dialogue context and all the retrieved entities. The probability distribution for each response token in R_t is defined using equation 3.

$$\boldsymbol{P(R_{t,i}) = Transformer\_Dec(R_{t,i} | R_{t,1}, R_{t,2}, \ldots, R_{t,i-1}, H_t)} \tag{3}$$

$$Loss_{generation} = -\sum_{i=1}^{|R_t|} log\ P(R_{t,i}) \tag{4}$$

Training objective of response generator is to reduce the standard cross-entropy loss calculated using equation 4. Overall model loss is cross entropy loss for entity detection and loss of words generated using response generation model.

EXPERIMENTS

I. Dataset

To evaluate the poposed models performance, it was tested on two standard datasets Camrest[8] and MultiWoZ 2.1[9] along with our corpus extracted from DBpedia. Camrest is a single domain multi-turn dialogue dataset containing information about restaurants. MultiWoz 2.1 is a multi-domain multi-turn dataset containing information about attraction, bus, hotel, hospital, restaurant, taxi and train that is seven different domains having both single domain and multi-domain dialogues. Data extracted from DBpedia belong to six different domains i.e. book, film, theater, music, painting, sculpture. This dataset contains both single domain and cross domain conversations.

Table 4. Comparison of graphConvAI on Camrest and MultiWoZ dataset

Model	MultiWOZ		Camrest	
	BLEU	Entity-F1	BLEU	Entity-F1
DSR [13]	9.10	30.00	18.30	53.60
KB-Retriever [14]	-	-	18.50	58.60
GLMP [10]	6.90	32.40	15.10	58.90
DF-Net [15]	9.40	35.10	-	-
GPT-2+KE [16]	15.05	39.58	18.00	54.85
EER [17]	13.60	35.60	19.20	65.70
FG2Seq [18]	14.60	36.50	20.20	66.40
CDNET [19]	11.90	38.70	21.80	68.60
GraphMemDialog [20]	14.90	40.20	22.30	64.40
ECO [21]	12.61	40.87	18.42	71.56
DialoKG [22]	12.60	43.50	23.40	**75.60**
Q-TOD [23]	17.62	50.61	23.75	74.22
MAKER (T5 Base) [24]	17.23	53.68	25.04	73.09
MAKER (T5 Large) [24]	18.77	54.72	25.53	74.36
GraphConvAI (T5)	20.33	56.51	27.31	75.37
GraphConvAI (BART)	**22.55**	**59.11**	**28.80**	74.91

II. Hyper-parameter settings

SpaCy's NER model was fine tuned for 40 epoch with Adam optimizer, learning rate = 0.001 and drop out = 0.2. For response generation model, model was fine tuned for 15 epoch with Adam optimizer, learning rate = 0.0001. Top-n retrieved entities used for MultiWOZ, Camrest and DBpedia dataset are 7, 5 and 5 respectively. These numbers were choosen after evaluation of model for different 'k' values as mentioned in ablation study section.

III. Evaluation metrics

Evaluation of a response generation in dialogue system is very important phase of system. It is necessary that metrics should be robust and reliable which can evaluate the quality, relevancy and coherence of the generated response. Most commonly used metrics are Bilingual Evaluation Understudy (BLEU) score and the Entity F1 score. BLEU is a metrics based on precision that uses n-gram overlap between the response produced by model and expected response to find how much they match with each other. Entity F1 score evaluates the model's capability to produce knowledge based response for the given input. It calculates the F1 score for the set of entities present in the target response and model produced response. Model with high BLEU and F1 score is expected. The performance of the proposed model on the benchmarked dataset with other existing models are given in table 4. Proposed model is also test on DBpedia extracted dataset and model performance is shown in Table 5.

Table 5. Performance of graphConvAI on DBpedia extracted dataset.

Model	DBpedia extracted dataset	
	BLEU	Entity-F1
GraphConvAI (T5)	21.60	57.15
GraphConvAI (BART)	23.44	59.47

IV. Ablation study

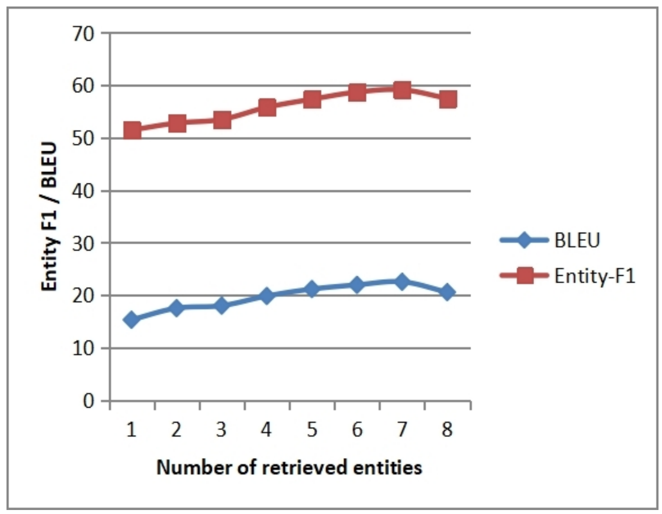


Figure 10. Performance of GraphConvAI for different retrieved entities on MultiWOZ dataset.

To evaluate the proposed model, we experimented with different top-k entities extracted from knowledge graph for the given input. Results of the experimentation for MultiWOZ dataset are shown in figure 10. Different number of entities from 1 to 8 were extracted and model was evaluated, result shows that number of entities extracted from the graph affects the BLEU score and entity-f1 score of the model. For MultiWOZ dataset as the number of retrieved entities increases, performance of the model also increases but upto 7 retrieved entities only, and after that the performance of the model starts decreasing as model generates noisy response. We used 'k' value for dataset where model performance was highest as mentioned in hyper parameter settings i.e. k is 7 for MultiWOZ dataset.

CONCLUSION

Multi domain dialogue system faces annotation data problem, proposed system automates the annotation process by using data extracted from DBpedia. Paper proposes a model consisting of three modules: named entity recognizer, graph extractor and the response generator. Named entity recognizer extracts the dialogue state from the dialogue history, extracted data is used to obtain relevant fact from external knowledge base and finally response generator model uses the dialogue history and extracted entities to generate the relevant response for the given input. In case of missing knowledge in external knowledge, system proposes an upate mechanism for firing SPARQL query to DBpedia for updating missing fact in external knowledge base. Model proposed an independent graph extractor isolated from named entity recognizer model and response generation model. Model was tested on two standard datasets and our own extracted corpus from DBpedia, experimentation shows that proposed model outperforms the existing models.

Conflict of Interest Statement

Author shows no conflict of interest.

References

1. M. Huang, X. Zhu, J. Gao. Challenges in Building Intelligent Open-domain Dialog Systems. *ACM Trans Inf Syst* **2020**, 38 (3).
2. J.I. Heng, W. Yu, C. Zhu, et al. A Survey of Knowledge-Enhanced Text Generation. *ACM Comput. Surv* **2022**, 1 (1).
3. D. Vrandečić, M. Krötzsch. Wikidata. *Commun ACM* **2014**, 57 (10), 78–85.
4. T. Pellissier Tanon, G. Weikum, F. Suchanek. YAGO 4: A Reason-able Knowledge Base. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* **2020**, 12123 LNCS, 583–596.
5. R. Schroeder, L. Taylor. Big data and Wikipedia research: social science knowledge across disciplinary divides. *Inf Commun Soc* **2015**, 18 (9), 1039–1056.
6. S. Auer, C. Bizer, G. Kobilarov, et al. DBpedia: A Nucleus for a Web of Open Data. In *international semantic web conference* **2007** (pp. 722-735). Berlin, Heidelberg: Springer Berlin Heidelberg.
7. M. Eric, L. Krishnan, F. Charette, C.D. Manning. Key-Value Retrieval Networks for Task-Oriented Dialogue. In *Proceedings of the 18th Annual SIGdial Meeting on Discourse and Dialogue*; pp 37–49.
8. T.-H. Wen, D. Vandyke, N. Mrkši´cmrkši´c, et al. A Network-based End-to-End Trainable Task-oriented Dialogue System. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1*; Vol. 1, pp 438–449.
9. M. Eric, R. Goel, S. Paul, et al. MultiWOZ 2.1: A Consolidated Multi-Domain Dialogue Dataset with State Corrections and State Tracking Baselines. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*; **2020**; pp 11–16.
10. C.-S. Wu, R. Socher, C. Xiong. Global-to-local Memory Pointer Networks for Task-Oriented Dialogue. In *International Conference on Learning Representations (ICLR 2019)*.
11. RaffelColin, ShazeerNoam, RobertsAdam, et al. Exploring the limits of transfer learning with a unified text-to-text transformer. *The Journal of Machine Learning Research* **2020**, 21, 1–67.
12. M. Lewis, Y. Liu, N. Goyal, et al. BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension. *Proceedings of the Annual Meeting of the Association for Computational Linguistics* **2020**, 7871–7880.
13. H. Wen, Y. Liu, W. Che, L. Qin, T. Liu. Sequence-to-Sequence Learning for Task-oriented Dialogue with Dialogue State Representation. In *Proceedings of the 27th International Conference on Computational Linguistics, COLING 2018*; pp 3781–3792.
14. L. Qin, Y. Liu, W. Che, et al. Entity-Consistent End-to-end Task-Oriented Dialogue System with KB Retriever. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*; pp 133–142.
15. L. Qin, X. Xu, W. Che, Y. Zhang, T. Liu. Dynamic Fusion Network for Multi-Domain End-to-end Task-Oriented Dialog. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*; pp 6344–6354.
16. A. Madotto, S. Cahyawijaya, G. Indra Winata, et al. Learning Knowledge Bases with Parameters for Task-Oriented Dialogue Systems. In *Findings of the Association for Computational Linguistics: EMNLP 2020*.
17. Z. He, J. Wang, J. Chen. Task-Oriented Dialog Generation with Enhanced Entity Representation. In *Proceedings of Interspeech 2020*; **2020**.
18. Z. He, Y. He, Q. Wu, J. Chen. Fg2seq: Effectively Encoding Knowledge for End-To-End Task-Oriented Dialog. In *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*; Institute of Electrical and Electronics Engineers Inc., **2020**; Vol. 2020-May, pp 8029–8033.
19. D. Raghu, A. Jain, M. -, S. Joshi. Constraint based Knowledge Base Distillation in End-to-End Task Oriented Dialogs. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*; Association for Computational Linguistics (ACL), **2021**; pp 5051–5061.
20. J. Wu, I.G. Harris, H. Zhao. GraphMemDialog: Optimizing End-to-End Task-Oriented Dialog Systems Using Graph Memory Networks. In *Proceedings of the AAAI Conference on Artificial Intelligence, 36*; **2022**.
21. G. Huang, X. Quan, Q. Wang. Autoregressive Entity Generation for End-to-End Task-Oriented Dialog. In *Proceedings of the 29th International Conference on Computational Linguistics*; **2022**; pp 323–332.
22. R. Al, H. Rony, R. Usbeck, J. Lehmann. DialoKG: Knowledge-Structure Aware Task-Oriented Dialogue Generation. In *Findings of the Association for Computational Linguistics: NAACL 2022*; pp 2557–2571.
23. X. Tian, Y. Lin, M. Song, et al. Q-TOD: A Query-driven Task-oriented Dialogue System. *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing, EMNLP 2022* **2022**, 7260–7271.
24. F. Wan, W. Shen, K. Yang, X. Quan, W. Bi. Multi-Grained Knowledge Retrieval for End-to-End Task-Oriented Dialog. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*; Long Papers; Vol. 1, pp 11196–11210.