CAGE: A Hybrid Framework for Closed-Domain Conversational Agents

Edward Burgin ✉, Sourav Dutta, Haytham Assem, and Raj Nath Patel

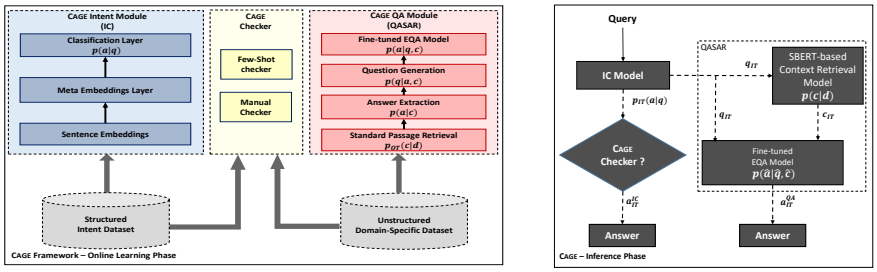
Huawei Research, Ireland
{edwardburgin, sourav.dutta2, raj.nath.patel}@huawei.com,
hithsala@amazon.co.uk

Abstract. Current *conversational agents* are primarily designed to answer user queries based on structured pre-defined utterance-response pairs. While *question-answering (QA) systems* extracts potential answers, to queries, from unstructured texts. However, in domain-specific settings, manual creation of query-response pairs is expensive, and domain adaptation of QA platforms is crucial. To this end, we propose CAGE, a "hybrid" conversational framework seamlessly integrating structured and unstructured data to obtain precise answers for user queries – improving *user experience* and *quality-of-service*. We describe the different components combining *query matching* and *extractive question answering*, and demonstrate the multi-lingual chatbot interface provided to a user.

1 Introduction

Chatbots or "virtual agents" provide a natural dialogue interface to users, simplifying information search and assisting in domain-specific applications. As such, chatbots are increasingly used in healthcare [8], ecommerce [6], public administration [9], and education [1] – involving (i) domain understanding; (ii) anticipating question styles; (iii) query responses; and (iv) multi-linguality. This makes it more challenging than open-domain digital assistants like Google Voice, Alexa, Siri and Cortana.

Traditionally chatbots relied on IR [9] on curated FAQ utterance-responses [5] – depicting high precision, but poor recall due to vocabulary mismatch and domain specificity. Machine reading comprehension (MRC) extracts answer spans from unstructured texts [14], providing flexibility in terms of data and coverage, but lacks contextual answer generation. Light-weight chatbots using MRC [13] have been widely incorporated [10]. Unfortunately, limited efforts exist towards combining the above techniques [4], and separate channels are proposed like Google DialogFlow (chatbot and knowledge connector), Amazon Services (Lex and Kendra) and Microsoft Azure (LUIS and QnAMaker). This paper presents a *hybrid and unified* chatbot prototype for integration of both structured and unstructured domain-specific data, to seamlessly answer diverse user queries.



(a) (b)

Fig. 1. (a) System architecture of CAGE with *IC* Module, QASAR Module, and Checker Module. (b) Interaction flow diagram of CAGE at interaction (inference) time.

2 *CAGE* Framework

We now introduce our *hybrid* conversational framework, *Closed-Domain Conversational AGEnt (*CAGE*)* (Fig. 1(a)). CAGE comprises **3** components, as described next.

• **Intent Classification Module (*IC*)** – This module trains a supervised learning model for *User Question Classification* based on a pre-defined structured query-response dataset, specifying questions (with a few paraphrases) along with curated answers. The trained model is used to classify an end-user's query, during inference, to one of the pre-defined questions. Internally, the *IC* module utilizes several *multi-lingual sentence encoders* to map questions into high-dimensional dense vector representations. The embeddings from the encoders are concatenated to obtain *"query meta-embedding"*. Finally, a shallow Multi-Layer Perceptron (MLP) with two hidden layer with ReLU non-linear activations and a softmax layer is used for classification. Specifically, the *IC* module implements the approach of [11], with the underlying "frozen" sentence encoders (instead of fine-tuning) to achieve (1) few-shot learning and (2) inexpensive compute requirement – making CAGE suitable for deployment as an online cloud based chatbot service using Amazon Lex or Google DialogFlow.

• **EQA Module (QASAR)** – This module fine-tunes a QA model for *self-supervised domain adaptation* by automatically generating *context-question-answer* triples from domain-specific unstructured documents. We employ pre-trained T5 model [12] for *self-learning*, wherein extracted paragraphs from a document are used to generate possible questions and corresponding answer spans – providing a set of triples that forms the synthetic training dataset for CAGE. These triples are then used to fine-tune a pre-trained SpanBERT QA model [7] for adapting it to our application domain. This provides a fully self-supervised approach with enhanced performance, especially for closed-domain datasets [2]. Currently, multi-linguality is supported via machine translation, however a multi-lingual QA model can be easily incorporated.

• **Checker Module** – The final module drives the integration for seamless transition between the *IC* and QASAR modules to extract the best answer –

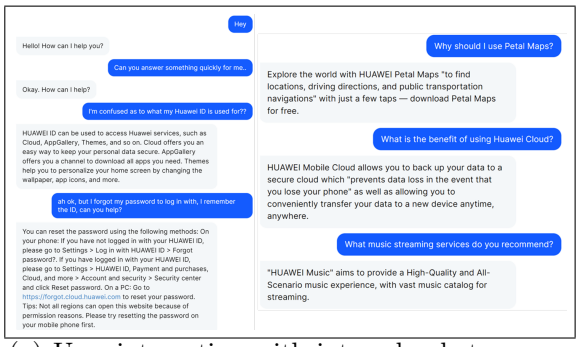
enabling the "hybrid" nature of our system. The appropriate selection/triggering threshold can be set either by (a) manually setting the module selection threshold based on application data, or (2) F1-score on a small validation data based on different confidence scores of both the modules. In our framework, we empirically set the default threshold to 0.65. That is, if the match confidence of *IC* module is 0.65 or more, the predicted response is returned, else the answer span obtained from the text by QASAR module is presented.

Inference: A user query is first passed to the *IC* module to obtain a matching question (as prediction on structured data typically depicts high precision) along with the matching probability. If the score is greater than the switching threshold, the matched answer is returned. Otherwise, the query is routed to QASAR (to extract a possibly answer) along with a set of sentences (i.e., context) from the text, that might have the answer – to obtain the answer text span from QASAR. As a fall-back policy, if the EQA module is also not confident, the chatbot requests the user to rephrase the query (or flags it as out-of-scope).

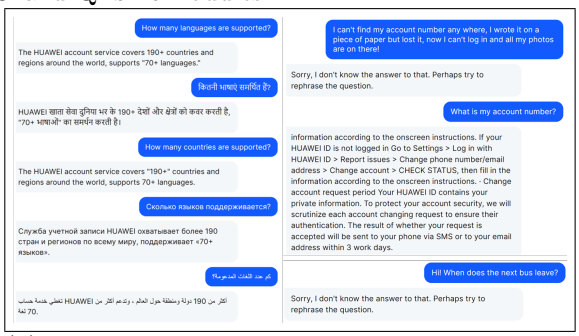
3 CAGE System Demonstration

We now present snapshots of user interaction for our multi-lingual CAGE chatbot platform. CAGE was integrated with the popular *BotFront* dialogue system interface (based on Meteor app) provided by Rasa [3]. We showcase on *three* data sources – (a) chitchat data with various "small talk" and greetings; (ii) structured FAQ data on Huawei Mobile Service (HMS) with 50 different questions (and paraphrasings); and (iii) unstructured text description of HMS applications obtained from the web.

In Fig. 2(a), we show a typical user interaction wherein the user initially *greets* the system followed by a domain pertinent question. We see that our system *correctly matches* the user question to the predefined FAQ, even for colloquially phrased user queries.



(a) User interaction with inter-play between *IC* and QASAR modules.



(b) Multi-lingual user query answering, rephrasing, and fall-back policy.

Fig. 2. System Demonstration of CAGE framework.

Further, we see factoid-based questions are efficiently answered by the QASAR module, wherein information present in the text are retrieved along with a *longer context* for readability.

For example, for the question *"How many languages are supported?"*, CAGE is seen to report: `The Huawei account service covers 190+ countries ..., supports ''70+ languages''`. Here, the text span in quotes provides the direct answer, while the entire response presents a well-contexted human readable response. In fact, even seemingly *objective questions* like *" Why should I use Petal Maps?"* are well answered by CAGE (`to find locations, driving directions, and public transport navigations` in this case). In Fig. 2(b), we depict the multi-linguality and *out-of-scope* scenarios of our framework. Overall, we showcase our *usability, performance and quality-of-service*. The inference time was typically less than 500ms.

Note, a standalone question matching or question answering system would fail for many of the above queries. Thus, we empirically compare the performance on a small annotated HMS data sample; with questions half of which are answerable from the text, while the others are related to pre-defined questions. We use *F1 score* to gauge the performance, with: *True Positive* (TP) for correct answer, *True Negative* (TN) for null returned on unanswerable question, *False Positive* (FP) for incorrect matching, and *False Negative* (FN) if null response is given to a true answer.

Table 1. Accuracy results on small HMS dataset.

Method	P	R	F1
EQA	0.78	1.00	0.88
Intent	0.89	1.00	0.94
CAGE	**0.93**	1.00	**0.97**

From Tab. 1, we observe that our "hybrid" CAGE framework performs better than the classification and EQA system individually, precisely answering both types of user questions. For detailed results of *IC* and QASAR modules on other datasets, please refer to [11, 2]. A short demo of CAGE can be found at https://youtu.be/PIzwbrmM4UU.

4 Conclusion

This paper presented CAGE, a novel *multi-lingual "hybrid"* deployable conversational system seamlessly coupling both question matching from *structured* data as well as extractive answering from *unstructured* data. CAGE combines *few-shot classification* with *domain-adapted answering* to provide high efficiency, improving quality-of-service.

References

1. Adamopoulou, E., Moussiades, L.: An Overv. of Chatbot Tech.. In: AIAI. pp. 373–383 (2020)

2. Assem, H., Sarkar, R., Dutta, S.: QASAR: Self-supervised learning framework for extractive question answering. In: IEEE Big Data. pp. 1797–1808 (2021)
3. Bocklisch, T., Faulkner, J., Pawlowski, N., Nichol, A.: Rasa: Open Source Language Understanding and Dialogue Management. In: NIPS Workshop on Conversational AI (2017)
4. Gapanyuk, Y., Chernobrovkin, S., Leontiev, A., Latkin, I., Belyanova, M., Morozenkov, O.: A Hybrid Chatbot System Combining QA and Knowledge-Base Approaches. In: AIST (2018)
5. Hussain, S., Sianaki, O., Ababneh, N.: A Survey on Conversational Agents/Chatbots Classification and Design Techniques, pp. 946–956. Springer (2019)
6. Manzano, M.D.I., Lopez, N.V., Gonzalez, N.A., Rodriguez, C.C.: Impl. of Chatbot in Online Commerce, and Open Innov.. J. of Open Innovation: Tech., Market, and Complx. **7**(2) (2021)
7. Joshi, M., Chen, D., Liu, Y., Weld, D.S., Zettlemoyer, L., Levy, O.: Spanbert: Improving pre-training by representing and predicting spans. TACL **8**, 64–77 (2020)
8. Jovanovic, M., Baez, M., Casati, F.: Chatbots as Conversational Healthcare Services. IEEE Internet Computing **25**(3), 44–51 (2021)
9. Lommatzsch, A., Katins, J.: An Information Retrieval-based Approach for Building Intuitive Chatbots for Large Knowledge Bases. In: LWDA (2019)
10. McTear, M.: Conversational AI: Dialogue Systems, Conversational Agents and Chatbots. Morgan and Claypool (2021)
11. Patel, R.N., Burgin, E., Assem, H., Dutta, S.: Efficient multi-lingual sentence classification framework with sentence meta encoders. In: IEEE Big Data. pp. 1889–1899 (2021)
12. Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y., Li, W., Liu, P.J.: Explr. the limits of transfer learning with a uni. t2t transformer. arXiv:1910.10683 (2019)
13. Yan, Z., Duan, N., Bao, J., Chen, P., Zhou, M., Li, Z., Zhou, J.: DocChat: An IR Approach for Chatbot Engines Using Unstructured Documents. In: ACL. pp. 516–525 (2016)
14. Zhang, Z., Zhao, H., Wang, R.: Machine Reading Comprehension: The Role of Contextualized Language Models and Beyond. Computational Linguistics **1**(1), 1–51 (2020)