Review Paper of Flight Booking Using A Hybrid Rule Based And Data Driven Approach

Prof. Vibha D. Kumar[1], Gopal Sanjay Zalte[2], Roshan Bharat Zalte[3], Gopal Kadu Zalte[4], Pradip Gajanan Pandav[5], Aniket Jayant Patil[6]

[1, 2, 3, 4, 5, 6] Dept of Computer Science Engineering

[1, 2, 3, 4, 5, 6] Padm. Dr. V. B. Kolte College of Engineering Malkapur , Maharashtra, India

Abstract- *The paper explores the development of a text-based flight booking Dialogue System (DS) using a hybrid approach that combines rule-based and data-driven techniques. Traditionally, Dialogue Systems are categorized into rule-based and data-driven approaches, with data-driven systems requiring substantial training data and rule-based systems relying on predefined rules and keywords. While data-driven approaches often yield better results, the lack of training data for Arabic task-oriented DSs has led to the prevalent use of rule-based approaches in their development. In response, the authors propose a hybrid approach for an Arabic task-oriented DS for flight booking. This hybrid approach leverages the Wit.ai natural language interface, configuring conversation flow through the Wizard of Oz technique, and developing DS intents and entities via crowdsourced training examples. Evaluation results demonstrate the system's proficiency in understanding user utterances and its ability to self-improve efficiently. Overall, the study presents a novel approach to developing Arabic task-oriented DSs, bridging the gap between rule-based and data-driven methodologies to enhance the effectiveness of flight booking systems.*

Keywords- Arabic booking bot, dialogue system, flight booking, hybrid approach.

I. INTRODUCTION

The advent of digital technology has revolutionized various aspects of modern life, including the way people travel and book flights. In this context, the development of effective flight booking systems is crucial to ensure seamless travel experiences for passengers. Traditionally, the development of Dialogue Systems (DSs) for flight booking has been approached using either rule-based or data-driven techniques. Rule-based systems rely on predefined sets of rules and keywords to interpret user inputs, while data-driven systems require large quantities of training data to learn and improve over time. However, both approaches have their limitations: rule-based systems may lack adaptability and scalability, while data-driven systems often require extensive resources and may struggle with limited data availability, particularly for specialized domains such as Arabic task-oriented DSs.

In response to these challenges, this paper proposes a novel hybrid approach that combines elements of both rule-based and data-driven methodologies for developing a text-based flight booking DS. By leveraging the strengths of each approach, the proposed system aims to overcome the limitations of traditional rule-based and data-driven systems, thereby enhancing the efficiency and effectiveness of flight booking processes.

The focus of this study lies in addressing the specific challenges associated with developing an Arabic task-oriented DS for flight booking, where the scarcity of training data poses a significant obstacle. To overcome this limitation, the proposed hybrid approach integrates rule-based components with data-driven techniques, thereby leveraging the benefits of both approaches while mitigating their respective shortcomings. The utilization of the Wit.ai natural language interface and crowdsourcing training examples facilitates the development of DS intents and entities, enabling the system to understand and respond to user utterances effectively.

Overall, this paper contributes to the advancement of flight booking systems by introducing a novel hybrid approach that combines rule-based and data-driven techniques. By addressing the challenges inherent in developing Arabic task-oriented DSs, the proposed approach demonstrates potential applicability and effectiveness in improving the user experience for flight booking processes.

II. PROBLEM FORMULATION

The contemporary landscape of flight booking systems faces challenges that necessitate innovative solutions for enhanced efficiency and user experience. The existing approaches to developing Dialogue Systems (DSs) for flight booking primarily fall into two categories: rule-based and data-driven. Rule-based systems rely on predefined rules and keywords, exhibiting limitations in adaptability and scalability. On the other hand, data-driven systems, while promising, encounter challenges associated with the need for substantial training data, especially in specialized domains like Arabic task-oriented DSs.

The specific problem at hand is the development of an effective flight booking DS tailored for Arabic users, where the insufficiency of training data poses a significant hurdle. Existing data-driven approaches, commonly sophisticated for English DSs, may not be directly applicable due to the scarcity of Arabic-specific training data. Rule-based approaches, often utilized due to data limitations, might lack the sophistication necessary for a seamless and user-friendly flight booking experience.

To address this problem, a hybrid rule-based and data-driven approach is proposed, aiming to combine the strengths of both methodologies while mitigating their respective weaknesses. This hybrid model seeks to improve adaptability, scalability, and efficiency in processing user inputs in Arabic, ultimately providing an enhanced flight booking experience. The integration of the Wit.ai natural language interface and the utilization of crowdsourced training examples represent strategic steps in overcoming the challenges associated with developing Arabic task-oriented DSs.

III. PROPOSE SYSTEM METHODOLOGY

The proposed system methodology outlines a comprehensive approach to developing a flight booking system, considering the specified operating system, coding language, tool, and database. The proposed methodology aligns with the given parameters:

Windows 10: The chosen operating system for development, ensuring compatibility with a wide range of software and tools.

Coding Language:

Python: Utilizing the Python programming language for its versatility, readability, and extensive libraries, which is advantageous for web development projects.
Development Framework:

Django: Leveraging the Django web framework for Python to streamline the development process. Django offers a high-level, clean, and pragmatic design, making it suitable for building robust web applications.
Database:

CSV: Opting for a CSV (Comma-Separated Values) file-based database system. While unconventional for large-scale applications, CSV can be suitable for smaller projects or prototyping due to its simplicity and ease of use.

WORKING ON LANGUAGES

The inclusion of technology details, both front-end and back-end, adds transparency to the study. It's essential to acknowledge the technological foundation supporting the research. The specifications on Bootstrap, Python, HTML, CSS, and Django versions, provide a clear picture of the development environment.

WORKING

The section discussing the technology stack and its application in the study outlines a comprehensive set of tools and frameworks for both front-end and back-end development, emphasizing their collective role in facilitating the effective implementation of the proposed methodology.

Front-End Technology:

Bootstrap (Version 5.1.3): The use of Bootstrap ensures a responsive and visually appealing user interface. Its grid system and pre-designed components enhance the efficiency of front-end development, providing a consistent and adaptable design across different devices.

HTML (Version 5) and CSS (Version 4.15): The combination of HTML5 and CSS4.15 provides the fundamental structure and styling for the web application. HTML5 introduces new semantic elements, while CSS4.15 enables advanced styling features, contributing to a modern and visually appealing user experience.

JavaScript (Version ES13): The use of JavaScript, particularly ES13, adds interactivity to the web application. JavaScript's capabilities enable dynamic content updates and user interactions, enhancing the overall user experience.

Back-End Technology:

The back-end technology for the flight booking system is centered aroundDjango, a powerful web framework written in Python. Django provides a structured and efficient environment for developing web applications by enforcing the Model-View-Controller (MVC) architecture. This architectural approach promotes code organization and separation of concerns, enhancing the maintainability of the system. With Django, developers can leverage built-in tools for URL routing, database interaction through Object-Relational Mapping (ORM), form handling, and more. The framework's batteries-included philosophy significantly accelerates the development process, allowing the team to focus on implementing specific functionalities related to flight booking.

Python, being the primary programming language, complements Django seamlessly. Known for its simplicity and readability, Python facilitates rapid development, making it an ideal choice for prototyping and building complex systems efficiently. Its extensive standard library and third-party package ecosystem offer developers a wide range of tools to enhance productivity. The integration of Python and Django empowers the back-end of the flight booking system with the versatility needed to handle various tasks, from defining data models to implementing authentication and authorization mechanisms.

Django's key features play a crucial role in shaping the back-end of the flight booking system. Models are defined to represent essential entities such as flights, bookings, and users, allowing for seamless interactions with the underlying database. Views and URL routing are employed to manage incoming requests, while forms and validation ensure the integrity and security of user inputs. The built-in authentication system, session management, middleware components, and database integration further contribute to the development of a robust and secure flight booking system. Through comprehensive testing and debugging, the back-end technology stack aims to deliver a reliable and scalable solution for efficient management of flight-related data and user interactions.

REFERENCES

[1] H. Trung, ‘‘Multimodal dialogue management-state of the art,’’ Dept. Hum. Media Interact., Univ. Twente, Enschede, The Netherlands, Tech. Rep. 0601, 2006, vol. 2.

[2] K. Nagao and A. Takeuchi, ‘‘Speech dialogue with facial displays: Multimodal human-computer conversation,’’ in Proc. 32nd Annu. meeting Assoc. Comput. Linguistics, 1994, pp. 102–109.

[3] A. Dingli and D. Scerri, ‘‘Building a hybrid: Chatterbot–dialog system,’’ in Proc. Int. Conf. Text, Speech Dialogue, 2013, pp. 145–152.

[4] D. Jurafsky and J. H. Martin, Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition, 2nd ed. Upper Saddle River, NJ, USA: Prentice-Hall, 2009.

[5] M. Hijjawi and Y. Elsheikh, ‘‘Arabic language challenges in text based conversational agents compared to the english language,’’ Int. J. Comput. Sci. Inf. Technol., vol. 7, no. 3, pp. 1–13, Jun. 2015.

[6] M. Mnasri, ‘‘Recent advances in conversational NLP: Towards the standardization of Chatbot building,’’ 2019, arXiv:1903.09025. [Online]. Available: http://arxiv.org/abs/1903.09025

[7] TOBi—The Vodacom ChatBot. Accessed: Apr. 7, 2020.[Online]. Available: http://www.vodacom.co.za/vodacom/services/tobi

[8] Siri—Apple. Accessed: Apr. 7, 2020.[Online]. Available: https://www. apple.com/siri/

[9] S. AlHumoud, A. A. Wazrah, and W. Aldamegh, ‘‘Arabic Chatbots: A survey,’’ Int. J. Adv. Comput. Sci. Appl., vol. 9, no. 8, p. 7, 2018.

[10] R. S. Wallace, ‘‘The anatomy of A.L.I.C.E.,’’ in Parsing the Turing Test: Philosophical and Methodological Issues in the Quest for the Thinking Computer, R. Epstein, G. Roberts, and G. Beber, Eds. Dordrecht, The Netherlands: Springer, 2009, pp. 181–210.

[11] D. A. Ali and N. Habash, ‘‘Botta: An Arabic dialect Chatbot,’’ in Proc. 26th Int. Conf. Comput. Linguistics, Syst. Demonstrations (COLING), 2016, pp. 208–212.

[12] B. Abu Shawar, ‘‘A Chatbot as a natural Web interface to Arabic Web QA,’’ Int. J. Emerg. Technol. Learn. (iJET), vol. 6, no. 1, pp. 37–43, Mar. 2011.

[13] A. Shawar and E. S. Atwell, ‘‘An Arabic Chatbot giving answers from the Qur’an,’’ in Proc. 11th Conf. Sur le TraitementAutomatique des LanguesNaturelles (TALN), vol. 2, 2004, pp. 197–202.

[14] Pandorabots. (May 6, 2016). ROSIE: Customizable Base Content. Medium.Accessed: Apr. 8, 2020.[Online]. Available: https://medium. com/pandorabots-blog/rosie-customizable-base-content-efb925fa7d56

[15] J. Nielsen, Usability Engineering. Amsterdam, The Netherlands: Elsevier, 1994.

[16] S. S. Aljameel, J. D. O’Shea, K. A. Crockett, A. Latham, and M. Kaleem, ‘‘Development of an arabic conversational intelligent tutoring system for education of children with ASD,’’ in Proc. IEEE Int. Conf. Comput. Intell. Virtual Environ. Meas. Syst. Appl. (CIVEMSA), Jun. 2017, pp. 24–29.

[17] O. G. Alobaidi, K. A. Crockett, J. D. O’Shea, and T. M. Jarad, ‘‘Abdullah: An intelligent arabic conversational tutoring system for modern islamic education,’’ in Proc. World Congr. Eng., vol. 7, 2013, pp. 1–7.

[18] O. Alobaidi, ‘‘Arabic conversational agent for modern Islamic education,’’ Ph.D. dissertation, Dept. Comput., Math.Digit.Technol., Manchester Metropolitan Univ., Manchester, U.K., 2015.

[19] M. Hijjawi, Z. Bandar, K. Crockett, and D. Mclean, ‘‘ArabChat: An arabic conversational agent,’’ in Proc. 6th Int. Conf. Comput. Sci. Inf. Technol. (CSIT), Mar. 2014, pp. 227–237.

[20] M. Hijjawi, Z. Bandar, and K. Crockett, ‘‘A novel hybrid rule mechanism for the Arabic conversational agent ArabChat,’’ Global J. Technol., no. 8, pp. 185–194, 2015.