Jordanian Journal of Computers and Information Technology (JJCIT), Vol. 10, No. 01, March 2024.

MULTI-DOMAIN MACHINE LEARNING APPROACH OF NAMED ENTITY RECOGNITION FOR ARABIC BOOKING CHATBOT ENGINES USING PRE-TRAINED BIDIRECTIONAL TRANSFORMERS

Boshra Sadder, Rahma Sadder, Gheith Abandah and Iyad Jafar

(Received: 12-Sep.-2023, Revised: 4-Nov.-2023, Accepted: 20-Nov.-2023)

ABSTRACT

Chatbots have recently become essential in various fields, ranging from customer service and information acquisition to entertainment. The use of chatbots reduces operational costs and human errors while providing services at any time. This work presents a Named Entity Recognition (NER) model for the Arabic booking chatbot, focusing on booking tickets and appointments across multiple domains. This research paves the way for the development of chatbots that can support multiple booking domains, contributing to the advancement of the Arabic language in this field. We adopt deep machine-learning and transfer-learning approaches to solve this task. Specifically, we utilized and fine-tuned the AraBERTv0.2 base model to develop the Named Entity Recognition for Booking Queries (NERB) model. Furthermore, we extended it to the Domain-Aware Named Entity Recognition for Booking Queries (DA-NERB) model by adding an additional input for domain type and an embedding layer. The input to our proposed model consists of text sequences of reservation requests, while the output includes sequences of tags representing entities within the input sequences. For training and testing, we synthesized the Arabic Booking Chatbot-Synthetic Dataset (ABC-S Dataset), comprising 76,117 reservation samples that span seven different domains and encompassing 26 categories of named entities. Additionally, we collected the Arabic Booking Chatbot-Collected Dataset (ABC-C Dataset) from volunteers to evaluate our model using various samples. It's worth noting that these datasets are written in informal Arabic, specifically the Levantine dialect. The proposed model achieves 100% and 96.9% accuracy scores on ABC-S (test set) and ABC-C, respectively. Both the datasets and the code for our model are publicly available to support research in the field of Arabic chatbots.

KEYWORDS

Chatbot,Arabic booking chatbot, Named entity recognition, AraBERT, Arabic booking dataset.

1. INTRODUCTION

Human-computer interaction is a technology that enables the interaction between humans and computers using natural languages, such as English and Arabic. The complexity of human language has necessitated a technology capable of understanding human language [1]. In the past decades, artificial intelligence (AI) and deep learning (DL) have shown significant development in various areas, including computer vision, natural-language processing (NLP) and speech processing [2]. NLP is a branch of AI research that explores computers' ability to understand human language.

A chatbot, also known as a conversation agent (CA), represents an example of an AI application that facilitates interaction between a person and a computer device using natural language. Chatbot agents have garnered the attention of numerous researchers aiming to make the conversation between humans and machines more rational [3].

Thanks to the significant advancements in AI and machine learning, users view chatbots as a promising alternative to traditional customer-service channels. They are perceived by customers as natural and function as virtual assistants, offering a more interactive experience compared to mobile applications. Chatbots help perform tasks, address customer inquiries and make purchase recommendations. Over the past few years, there has been a growing interest in chatbots. In general, chatbots offer instant responses, can serve an unlimited number of users simultaneously and provide 24-hour service [4]. This has a direct impact on businesses in terms of reducing the cost of delivering services to customers. In summary, chatbots have the potential to save money, manpower and possibly

B. Sadder, R. Sadder, G. Abandah (Corresponding Author) and I. Jafar are with Department of Computer Eng., The University of Jordan, Amman, Jordan. Emails: sadder@outlook.com, rahma_sadder@yahoo.com, abandah@ju.edu.jo and iyad.jafar@ju.edu.jo

increase customer satisfaction [5].

Building chatbots that support the English language has received more attention and is technically more advanced than those designed for Arabic. The scarcity of Arabic chatbots can be attributed to several factors. Primarily, these include: 1) The lack of training data in Arabic due to the widespread use of English in comparison to Arabic. 2) The complexity and characteristics of the Arabic language, including the fact that it is a Semitic language rich in morphology and featuring variations in orthography. 3) The presence of diverse Arabic dialects [6]-[7].

Numerous chatbots specialize in booking services, such as those for booking airline tickets, hotels, …*etc.* However, most of the current research on modeling booking chatbots focuses on a single domain, with only a few supporting multiple domains. Furthermore, when it comes to the Arabic language, there is a shortage of efficient booking chatbots. In general, published studies on Arabic chatbots are scarce, highlighting a significant knowledge gap [7].

Chatbots expect users to submit booking queries in the form of unstructured text and they are supposed to process these queries and respond accordingly. One crucial step in this process is the chatbot's ability to recognize critical and important entities within the query to formulate an appropriate response. The main objective of this work is to develop a named-entity recognition (NER) model for Arabic booking chatbots, which can be applied across multiple domains to extract and classify reservation entities in booking queries with high accuracy. To build our model, we used and fine-tuned the state-of-the-art AraBERTv0.2 model.

A significant contribution of this work is the building of the Arabic Booking Chatbot-Synthetic Dataset (ABC-S Dataset), which contains 76,117 samples and encompasses 26 entity categories. Additionally, we collected reservation samples from volunteers to evaluate our model on various samples; this dataset is referred to as the Arabic Booking Chatbot-Collected Dataset (ABC-C Dataset). Also, we propose a solution for the Arabic booking chatbot that is dedicated to booking tickets and appointments for seven different domains, which are flights, hotels, cinemas, football matches, cars, restaurants and clinics. The main characteristic of the proposed model when compared to similar models lies in the ability to perform the NER task when different booking domains are considered. In other words, we present a single NER model that is domain-aware, eliminating the need for multiple models.

The rest of the paper is organized as follows, In Section 2, we provide a review of related work on chatbots and NER models. Section 3 discusses the process of building and collecting the datasets, as well as describing the two NER models. In Section 4, we present and discuss the evaluation results and outline the limitations of this work. Finally, we conclude the paper in Section 5.

2. LITERATURE REVIEW

There are many works on Arabic chatbot applications. The first part of the literature review presents papers on chatbot applications in Arabic. The second part of the literature review presents the papers that include NER models, as well as the main papers on which this research is based.

2.1 Chatbot Applications

Chatbot systems are basically software applications used to conduct a conversation between humans and computers using natural languages. These systems have recently become essential in our lives, as there are many organizations that use them to provide customer service.

The Dialogue System Development Framework is a set of services, tools and software-development kits that provide a rich foundation or framework for AI-chatbot development [8]. For example, it provides web interfaces to create a knowledge base. With the advent of these development frameworks, the implementation of the dialogue system has become easier and many dialogue systems (DSs) have been introduced to serve different tasks. Examples of such frameworks are: Wit.ai, Rasa and IBM Watson Assistant [9].

In the past few years, many chatbots have been developed for various languages to serve several domains. Unfortunately, there is limited research on Arabic chatbots due to the characteristics of the Arabic language. Al-Hammoud et al. [3] surveyed the available research conducted in the field of

Arabic chatbots and concluded that there is a scarcity of research available on Arabic chatbots and that all the available works are retrieval-based [3].

Al-Ajmi and Al-Twairesh [10] proposed an Arabic dialogue system for flight booking tasks using a hybrid rule-based and data-driven approach utilizing the Wit.ai framework, since it has a set of predefined trained entities that support Arabic [10]. This system is supposed to process the text provided by the user, understand it and then extract a set of entities, such as location, route type, date, time and ticket class. In addition, the system must be able to recognize whether the user's input contains an intention to book a flight to proceed with the reservation process.

The evaluation of the developed system was conducted holistically, without focusing on individual components. The assessment involved 21 participants, with 90.48% of them having prior experience in booking flight tickets. The evaluation of the Dialogue System (DS) took place in two stages, aiming to assess both ease of use and system effectiveness. In the initial stage, participants provided feedback to enhance the system, which informed improvements made to the developed models for testing in the subsequent stage. Overall, users had a positive experience when booking airline tickets using the developed flight-booking DS.

Al-Ghadhban and Al-Twairesh [11] developed an Arabic chatbot called "Nabiha," designed to serve as an academic advisor for students in the Information Technology Department at King Saud University (KSU). Nabiha's primary objective is to engage with students, providing answers to their inquiries regarding academic progress and available courses. The chatbot's development relied on pattern matching and the utilization of the Artificial Intelligence Markup Language (AIML). To convert readable text into AIML format, a Java program was created for this purpose. Data was collected from websites containing student opinions and complaints and the Nabiha chatbot was launched on the Pandorabots platform [11].

Fadhil [12] introduced OlloBot, an Arabic conversational agent that helps physicians follow-up with their patients and provide 24/7 support to patients. This chatbot utilized the IBM Watson Conversation platform to manage dialogue flow and provide AI assistance for capturing various user intents and entities. Following the development of the chatbot system, it was integrated with the Telegram Bot Platform [12]. To evaluate the system's performance, a total of forty-three users participated. The evaluation involved a questionnaire comprising 30 questions, focusing on four main aspects: usefulness, ease of use, ease of learning and user satisfaction with OlloBot's reliability. The respondents rated the questionnaire on a scale from 1 (strongly disagree) to 5 (strongly agree). The results showed good indicators, with most users expressing interest in the bot's social intelligence as well.

Alshareef and Siddiqui [6] presented an open-domain conversational agent (CA) designed to communicate in the Gulf dialect. They employed a sequence-to-sequence architecture model incorporating the attention mechanism to handle long dependencies in the input. The conversational problem was defined as a machine-translation problem. Therefore, they collected their corpus from tweets in the post–reply format. Their proposed model underwent training both with and without the use of pre-trained word embeddings and FastText. The system's evaluation included the Bilingual Evaluation Understudy (BLEU) score and human assessment. The results demonstrated that utilizing embeddings led to more comprehensible and contextually relevant outputs. Specifically, BLEU scores of 25.1 and 11.4 were achieved with and without the use of pre-trained word embeddings, respectively [6].

2.2 Named Entity Recognition (NER) Models

One major task of goal-oriented chatbot systems is to fill a set of slots embedded in a semantic frame or automatically extract essential information, including intents and entities, to accomplish the objective of the dialogue [13]. Solving this task is usually done using the NER technique, which aims to identify and categorize named entities within the user's input into predefined classes, such as person names, organizations and locations. Since the task is considered a multi-class classification process, some methods of text classification are used to name the tokens [14].

Devlin *et al.* [15] introduced a deep bidirectional-language model called BERT, which stands for bidirectional encoder representations from transformers. Language-specific BERT models are very effective in understanding language, provided that they are pre-trained on a very large corpus. The

pre-trained BERT model can be fine-tuned with just one additional output layer to efficiently solve many NLP tasks, such as question answering and named entity recognition (NER) [15].

BERT is a multi-layer bidirectional transformer encoder. The BERT-Base model consists of 12 layers, 768-hidden size, 12 self-attention heads and 110M parameters, while the BERT Large model consists of 24 layers, 1024-hidden size, 16 self-attention heads and 340M parameters [15]. This bidirectional model can read the text input once, unlike the unidirectional model, which reads the text input sequentially from one direction. This feature allows the BERT model to learn the context of the word in both directions [16]. In the pre-training stage of the BERT transformer, the model is trained using an unsupervised approach on a very large corpus of data over two tasks, the masked language models (MLM) task and the next-sentence prediction (NSP) task.

This transformer has a very large number of parameters that can be fine-tuned for a certain NLP task by adding an additional layer or more on top of BERT that fits the shape of the output and then training the entire model together. In this stage, the model parameters are first initialized with the pre-training parameters and then are fine-tuned using the supervised approach by training the model using a labeled dataset associated with the NLP task [15]. The BERT model expects a text input; however, this text must be segmented into words (tokens). In addition, some special tokens are entered with the input tokens, such as the classification token (CLS) which is used at the beginning of BERT inputs and the separation token (SEP) which is used to separate the parts of the input. The output of BERT is a vector representation for each word as the output [17].

Antoun *et al*. [14] proposed the AraBERT model by pre-training BERT on a large-scale Arabic corpus. AraBERT was evaluated on three different natural-language processing tasks; namely, sentiment analysis, question answering and NER using eight different datasets. The results of AraBERT's experiments are compared with Google's multilingual BERT (mBERT) and other state-of-the-art approaches. The performance of the AraBERT outperformed all approaches on most of the tested datasets [14]. There are different versions of the AraBERT, which are AraBERTv0.1, AraBERTv0.2, AraBERTv1 and AraBERTv2 with the main differences between them being the size of the model and the size of the used training datasets, in addition to the need for pre-segmentation of the input text before it is fed to the model.

For the NER task, Antoun *et al.* [18] fine-tuned the AraBERTv0.1 model to solve the NER task on the Arabic Named Entity Recognition Corpus (ANERcorp), which encompasses 16.5 k entities categorized into four groups: person, organization, location and miscellaneous. Their approach involved customizing the AraBERTv0.1 model by adding two new layers. The first was a dropout layer with a rate of 0.3, while the second was a linear output layer. The model was trained using Adam optimization, a learning-rate scheduler, cross-entropy loss function and 5-fold cross-validation. They reported an F1-score of 84.2% [18].

Youssef *et al.* [19] proposed an end-to-end deep learning NER model for Arabic. This model consists of stacked embeddings that combine the pooled contextual embedding, pre-trained word embeddings (FastText) and BERT embeddings (pre-trained AraBERT model), that are fed into bidirectional long short-term memory with a conditional random field layer (BiLSTM-CRF). The results showed that this model outperformed all previously published results for deep learning and non-deep learning models on the AQMAR dataset with an F1-score of 77.62% [19].

Taher *et al.* [20] proposed a model for NER in Persian. In the architecture of their model, they used the BERT pre-trained model, a fully connected layer and a conditional random field (CRF) layer. These two layers were trained on the representation of tokens that are extracted from BERT. In the NSURL-2019 competition for the NER task for the Persian language, this model achieved second place with F1-scores of 83.5% and 88.4%, in phrase and word level evaluation, respectively [20].

Benali *et al.* [21] presented the BERT-BiLSTM-CRF model, which incorporated the BERT model as an embedding feature and integrated it with the BiLSTM-CRF architecture. The outcomes of this study demonstrated its model's superiority over six state-of-the-art Arabic NER models, achieving the highest performance on a tweet corpus with an F1-score of 67.40% [21].

Shaker et al. [22] proposed the utilization of Long Short-Term Memory (LSTM) units and Gated Recurrent Units (GRUs) for building named entity recognition models in the Arabic language.

Furthermore, they introduced a diverse Arabic NER dataset encompassing nine categories of named entities, comprising over thirty-six thousand records, with texts spanning seven different domains. Both the LSTM and GRU models produced commendable results, achieving an approximate precision of 80% in identifying entity names [22].

NER models play a pivotal role in the extraction and classification of named entities, providing significant utility in various chatbot applications, particularly in the domain of booking services. Despite the extensive attention that NER has received in English, a notable gap persists in its comprehensive exploration within Arabic. This gap can be attributed to the morphological complexity of the language and the limited availability of Arabic resources [23]. Additionally, most of the prior research on Arabic conversational systems primarily focused on Modern Standard Arabic (MSA), with relatively few incorporating Arabic dialects, even though numerous dialects are spoken across the Arabic-speaking world [7]. Furthermore, publicly accessible Arabic datasets often lack support for multiple reservation domains. Hence, the objective of this work is to advance the field of Arabic NER by developing an accurate, domain-aware NER model and building a specialized dataset encompassing seven different reservation domains for training our model.

3. MATERIALS AND METHOD

3.1 The Developed Datasets

Training deep machine-learning models for NLP tasks typically requires thousands of samples to achieve satisfactory results. In our case, this means the need to gather a substantial number of booking samples for various domains and manually tag the words in these samples, which can be a challenging task. Alternatively, we have proposed the creation and synthesis of a dataset for Arabic booking queries automatically using programs. We refer to this dataset as the Arabic Booking Chatbot-Synthetic Dataset (ABC-S Dataset). Additionally, for a more realistic evaluation and results, we have also collected another dataset of real Arabic booking queries from volunteers, which we named the Arabic Booking Chatbot-Collected Dataset (ABC-C Dataset). These datasets are used to train and test the proposed models. Both datasets are publicly available on GitHub [24].

3.1.1 Building the ABC-S Dataset

The dataset to be designed is intended to support multi-domain NER in Arabic booking queries. There are many domains in which booking service is required. In the ABC-S dataset, we proposed populating it with booking queries in the flight, hotel, cinema, football match, car rental, restaurant and clinic domains.

Next, for each of these domains, we listed the possible entities that may be present in the queries. For example, when booking a flight ticket, the query may include the departure and arrival cities, departure and return dates and times, the number of tickets and the ticket class. To facilitate the learning process of the NER model, the identified entities were assigned numeric labels or tags. A total of 25 entities were defined for the seven domains. Table 1 lists the names of these entities and their corresponding tags. Note that some entities are common between different domains, such as "DateAndTime 1," while others are exclusive to one domain, like the "DoctorName" entity. The entity name 'Others' has been added to the list and given a tag of 0. This entity is used to label any entity in the booking query that is not among the 25 predefined entities.

With these entities and their defined tags, we created a set of various reservation templates for each domain. These templates are written in informal Arabic (Levantine dialect) and essentially consist of sentences for possible reservation queries that contain general text and the names of entities. Table 2 displays some of the booking templates in the flight domain. The entity names, represented by the colored underlined words in the template, serve as variables and are later filled with actual values, which are then concatenated with the remaining text when all templates are generated.

To generate complete reservation samples in the dataset, a list of values for each of these entities was defined. For instance, the "Flight Ticket Class" entity has five different values, as listed in the first column of Table 3. A list of tag sequences was created to label each word in the value with the appropriate entity tag. Table 3 illustrates how the various possible values for the “Ticket Class” entity are tagged, with each word assigned a tag value of 7, corresponding to the “Ticket Class” tag, as shown in Table 1.

"Multi-domain Machine Learning Approach of Named Entity Recognition for Arabic Booking Chatbot Engines Using Pre-trained Bidirectional Transformers", B. Sadder, R. Sadder, G. Abandah and I. Jafar.

Table 1. Entities in the ABC-S dataset.

Tag	Entity Name	Tag	Entity Name
0	Others	13	First Team
1	City 1 (Departure City)	14	Second Team
2	City 2 (Arrival City)	15	League (League Name)
3	Date and Day 1 (Start Date)	16	Hotel (Hotel Name)
4	Time 1 (Start Time)	17	Room (Number of Rooms)
5	Date and Day 2 (End Date)	18	Restaurant (Restaurant Name)
6	Time 2 (End Time)	19	Movie (Movie Name)
7	Ticket Class	20	Cinema (Cinema Name)
8	Round Trip	21	Car Name
9	Number of Tickets	22	Car Model
10	Name (Customer Name)	23	Period
11	Doctor Name	24	Stadium Name
12	Phone (Phone Number)	25	Clinic Name

Table 2. Sample of reservation templates in the flight domain.

Template No.		Template Text
1	Arabic	"أريد حجز رحلة ذهاب من " + City 1 + " إلى " + City 2 + " " + Numbers + " " + TicketClass + " على " + Time 1 + " الساعة " + DateAndDay 1
	English	"*I want to book one-way ticket from* " + City 1 + " *to* " + City 2 + " " + Numbers + " " + DateAndDay 1 + " *at* " + Time 1 + " *on* " + TicketClass
2	Arabic	"رحلتي من " + City 1 + "الى " + City 2 + "وبدي احجز تذاكر " + Numbers + " وبدي انزل بفندق " + Hotel + "خلال فترة الرحلة واريد توفير وجبة فطور وعشاء "
	English	"*My flight is from* "+ City 1 +" *to* "+ City 2 +" *and I want to book tickets* "+ Numbers +" *and I want to stay at* "+ Hotel +" *during the trip and I want to provide breakfast and dinner*"
3	Arabic	"اريد تذكرة ذهاب فقط من " + City 1 + "الى " + City 2 + "على " + DateAndDay 1 + " يوم " + TicketClass + "
	English	"*On*" + DateAndDay 1 + "*I want to book one-way ticket from* " + City 1 + " *to* " + City 2" + " *on* "+ TicketClass

Table 3. Tag sequences of possible values for the ticket-class entity.

Ticket Class Value	Tag Sequence
درجة اقتصادية (Economic Class)	7 7
الدرجة الاولى (The First Class)	7 7
درجة اولى (First Class)	7 7
درجة رجال الاعمال (Business Men Class)	7 7 7
درجة اعمال (Business Class)	7 7

To generate the reservation samples, the entity names in the reservation templates were randomly filled with values from the defined lists and then concatenated with the remaining text in the template. Subsequently, each of the generated samples was converted into a sequence of tags. In this process, the words corresponding to entities' values were replaced with their respective tag sequences, while the remaining words were replaced with a tag of 0. Table 4 illustrates the steps taken to generate a sample and its corresponding sequence of tags, using the first reservation template from Table 2. The sentence is read from right to left (Arabic), while the sequence of tags is read from left to right. For instance, "أريد حجز" from Table 4 takes the first two tags from the left, which are "0 0", while the last word in the sentence, "الأولى," is associated with the last tag, "7."

Following this approach, we generated a total of 76,117 samples for the seven domains in the ABC-S dataset. Table 5 provides an overview of the main attributes of the ABC-S dataset. The samples in this dataset were divided into training and test sets, with 90% for training and validation and 10% for

testing. The training set was further divided into training and validation sets, with percentages of 90% and 10%, respectively. Each of these sets is stored in a separate file with the structure shown in Table 6. The first column contains the sentence used to make the reservation, the second column contains a sequence of tags representing the entities in the sentence, the third column contains the domain ID to which the sample belongs and the fourth column contains the sample's label in the format (domain ID-template ID).

Table 4. Steps for generating a reservation sample in the flights domain.

Step 1. Select a template	"أريد حجز رحلة ذهاب من " + City 1 + " إلى " + City 2 + " " + Numbers + " " + DateAndDay 1 + " الساعة " + Time 1 + " على " + TicketClass "*I want to book one-way ticket from* " + City 1 + " *to* " + City 2 + " " + Numbers + " " + DateAndDay 1 + " *at* " + Time 1 + " *on* " + TicketClass
Step 2. Fill template with values	"أريد حجز رحلة ذهاب من " + الرياض + " إلى " + الدوحة + " " + لطفلين و ثلاث بالغين + " يوم الخميس 2022/6/26 الساعة " + 9 مساء + " على " + الدرجة الأولى "*I want to book one-way ticket from* " + **Riyadh** + " *to* " + **Doha** + " " + **for two kids and three adults** + " " + **on Thursday 26/6/2022** + " *at* " + **9 PM** + " *on* " + **first class**
Step 3. Generate sample text	"أريد حجز رحلة ذهاب من الرياض إلى الدوحة لطفلين وثلاثة بالغين يوم الخميس 2022/6/26 الساعة 9 مساء على الدرجة الأولى" *I want to book one-way ticket from Riyadh to Doha for two kids and three adults on Thursday 22/6/2022 at 9 PM on first class*
Step 4. Represent sample as a sequence of tags	"0 0 0 8 0 1 0 2 9 0 9 9 3 3 3 0 4 4 0 7 7 "

Table 5. Summary of the ABC-S dataset.

Domain	**D1**	**D2**	**D3**	**D4**	**D5**	**D6**	**D7**	**Total**
Name	Flight Booking	Hotel Booking	Cinema Booking	Football Match	Car Booking	Restaurant Booking	Clinic Booking	
Number of Used Entities	15	12	10	13	10	8	8	26
Number of Templates	12	5	5	7	6	7	10	52
Number of Samples	30,437	22,476	8,360	5,599	3,716	2,891	2,638	76,117

Table 6. The structure of the ABC-S dataset file.

Sentences	**Tags**	**Domain**	**Domain-Template**
مرحبا انا علي عوده الله بدي اسافر من البرازيل الى الصين بيوم 6 الشهر بتذكرة طيران من درجة اقتصادية *Hi, I'm Ali Odet Allah I want to travel from Brazil to China on the 6th on the economy class*	0 0 10 10 10 0 0 0 1 0 2 3 3 3 9 0 0 7 7	1	D1T5
بدي انزل بفندق لو رويال في الإسكندرية واحجز 4 غرف نوم ل 3 اشخاص كبار *I want to stay in the LaRoyal hotel in Alexandria and book four rooms for three adults*	0 0 0 16 16 0 1 0 17 17 0 0 9 9 9	2	D2T5
لو سمحت احجزلي تذاكر طيران ذهاب وعودة من الدمام الى القاهرة لشخصين بالغين و 3 اطفال على درجة اولى *Please book two-way tickets for me from Dammam to Cairo for two adults and 3 kids on the first class*	0 0 0 9 0 8 8 0 1 0 2 9 9 0 9 9 0 7 7	1	D1T7

3.1.2 Collecting the ABC-C Dataset

To test the proposed model on real reservation samples, we proposed building the ABC-Collected

Dataset (ABC-C Dataset) which contains reservation samples collected from 38 volunteers. The volunteers were asked to freely write reservation samples in different domains without having prior knowledge about the generated synthetic samples in the ABC-S dataset. A total of 200 reservation samples were collected from the volunteers, with the distribution provided in Table 7. The tags for these sequences were generated manually. Table 8 displays some examples of these samples along with their corresponding tags.

Table 7. Distribution of reservation samples in the ABC-C dataset

Domain	**D1**	**D2**	**D3**	**D4**	**D5**	**D6**	**D7**
Number of Samples	30	24	29	29	27	30	31
Total	200						

Table 8. Structure of the ABC-C dataset file.

Sentences	**Tags**	**Domain ID**
مطلوب حجز موعد لعيادة الاسنان عند الدكتور نزار غنام الساعة الخامسة مساءً.	0 0 0 0 25 0 0 11 11 0 4 4	D1
مرحباً بدي حجز طاولة في مطعم جبري ع التراس لشخصين بدون ارجيلة.	0 0 0 0 0 0 18 0 0 9 0 0	D6
لو سمحت اريد حجز سيارة ميتسوبيشي باجيرو لتلاتة أيام وانكم تحضروها لموقعي في جبل الحسين شارع المجدل عمارة 30 ، و يتم تسليمها من قبلي في المكتب عندكم.	0 0 0 0 0 21 21 23 23 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	D5

3.2 The Proposed Models

As we mentioned earlier, the main objective of this work is to develop and evaluate a named entity recognition model for Arabic booking chatbots that can be used with multiple domains. For this purpose, we propose two models that customize the AraBERT transformer as discussed in the following sub-sections.

3.2.1 The NERB Model

The first proposed model is referred to as the Named Entity Recognition Model for Booking Queries (NERB). The architecture of this model is shown in Figure 1. At the core of this model lies the AraBERTv0.2-base model. We chose to use this version of AraBERT, because it is trained on a larger dataset and has demonstrated better performance in the NER task [14]. We customized the AraBERT model by adding two new layers: a dropout layer with a rate of 0.3 to prevent overfitting and a linear output layer with 26 outputs corresponding to the number of entity categories in the ABC-S dataset.

The input to the AraBERT transformer is represented using three torch tensors: Input IDs, Input Mask and Segment IDs. Input IDs are tokens mapped into IDs. The Input Mask is a sequence of 1s and 0s, where 1s correspond to real tokens and 0s represent padding tokens. This mask is used to prevent the model from considering the padded tokens. Segment IDs utilize 1s and 0s to distinguish between two sentences. While BERT expects sentence pairs, in our case, the input query belongs to a single sentence, so the segment ID is essentially a sequence of 0s. AraBERT transforms these three torch tensors into a torch tensor with the dimensions of (Max Sequence Length, Token Vector Size = 60 and 768). The Max Sequence Length is set to 60 in NERB, while it is typically 512 in BERT. The value 768 corresponds to the hidden size dimensions of the encoder layer in the AraBERT model.

The output from the AraBERT transformer is subsequently used as input for the following layers, which include a dropout layer and a linear layer with 768 inputs and 26 outputs. The final output from this model is a torch tensor with dimensions of (Max Sequence Length, Number of Classes = 60 and 26). Following this, the output dimensions of the NERB model are reduced by selecting the classes with the highest probability and removing padding from the sequence, so that its length is equal to the length of the input vector.

The compilation of the proposed model was carried out using the same configuration as the base model. We employed the Adam optimizer with a learning-rate scheduler and utilized the cross-entropy

loss function. The model was trained using the ABC-S Dataset and we evaluated its performance using the accuracy score.

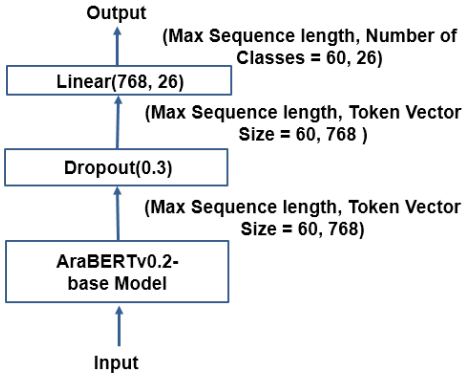

Figure 1. The architecture of the proposed NERB model.

For training, we used a maximum of 100 epochs for the training process. In each epoch, the model trains using the training set with a batch size of 32 and then predicts the tags of the validation set with a batch size of 8. We monitored the validation loss value and implemented early stopping if the validation loss did not improve for five consecutive epochs. The model with the lowest validation loss was adopted and utilized for testing. The model was trained on a GPU for 14 epochs, taking approximately 10 minutes per epoch and was then stopped due to the application of early stopping with a patience value set to 5. The ninth epoch recorded the lowest validation loss of $1.31e^{-06}$.

3.2.2 The DA-NERB Model

The second model that we propose is the Domain-aware Named Entity Recognition for Booking Queries (DA-NERB). The basic idea of this model is to extend the NERB model such that some prior knowledge about the domain is incorporated in the input to the model to investigate the effect of such knowledge on the overall performance. Figure 2 shows the architecture of the DA-NERB model. Technically, the DA-NERB model has an additional input which is the domain ID; a value between 1 and 7 as shown in Table 5. This input is fed into an embedding layer with 8 cells that transforms the single-value domain ID into an 8-element. The output of the embedding layer is then reshaped to match the size of the dropout layer and then fed into a concatenation layer that merges the embedding-layer output with the dropout layer to obtain a tensor of size (Max Sequence Length, Token Vector Size = 60 and 776). A linear output layer with 776 inputs and 26 outputs receives the concatenated tensor and outputs a tensor with size (Max Sequence Length, Number of Classes = 60 and 26). Also, the final-output dimensions will be reduced as we mentioned previously in the NERB model.

Compiling and training this model were performed using the same configuration of the NERB model using the ABC-S dataset. This model was trained using GPU for 12 epochs with approximately 10 minutes per epoch and then stopped due to the use of early stopping with the patience value set to 5. The seventh epoch has the lowest validation loss of 8.44×10^{-07}.

Since DA-NERB is aware of the domain to which the input query belongs, we have introduced a simple post-processing step to enhance its output predictions. The output of DA-NERB consists of a sequence of predicted tags for each word in the input. Essentially, if the domain is known, the predicted tags should contain values for entities that belong to that specific domain, alongside tags that correspond to 'Others'. Figure 3 illustrates the seven reservation domains in the ABC-S dataset, with entities that must be present (highlighted in red) in any input query within these domains, as well as entities that may be present (highlighted in black). Based on this information, the post-processing step involves replacing incorrectly predicted tags with '0', which is the tag for the 'Others' entity. For instance, if the input query belongs to the 'flight' domain, a tag corresponding to a 'Restaurant Name' entity should not be present, so it is replaced with '0'. This post-processing step can also be employed to identify any missing entities in the input. This functionality can be valuable when the model is deployed to interactively gather these entities from the user.

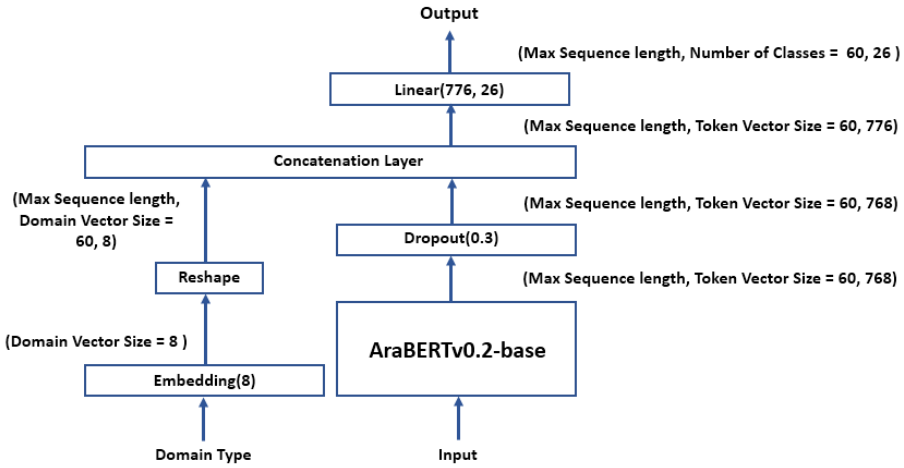

Figure 2. The architecture of the proposed DA-NERB model.

Domain Type \ Classes	Other	City_1	City_2	Date_And_Day_1	Time_1	Date_And_Day_2	Time_2	Ticket_Class	Round_Trip	Numbers_Of_Tickets	Names	Doctor_Names	Phone	Team_1	Team_2	League	Hotel	Room	Restaurant	Movie	Cinema	Car_Name	Car_Model	Period_Time	Stadium	Clinic
Domain Type	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25
Airline Ticket Booking	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓		✓				✓	✓						✓		
Hotel Booking	✓	✓		✓	✓	✓	✓			✓	✓		✓				✓	✓						✓		
Cinema Booking	✓			✓	✓		✓	✓		✓	✓		✓							✓	✓					
Football Match Booking	✓	✓		✓	✓		✓	✓		✓	✓		✓	✓	✓	✓									✓	
Car Booking	✓			✓	✓	✓	✓				✓		✓									✓	✓	✓		
Restaurant Booking	✓			✓	✓		✓			✓	✓		✓						✓							
Clinic Appointment Booking	✓			✓	✓		✓				✓	✓	✓													✓

Figure 3. The expected and required entity tags based on domain type.

4. Results And Discussion

This section discusses the experimental platform used and presents the results obtained from experiments conducted on both the ABC-S and ABC-C datasets across various scenarios. The outcomes of these experiments encompass the performance of both the NERB and DA-NERB models. Furthermore, we provide a comparative analysis of the DA-NERB model's performance *versus* that of the NERB model when trained on a single reservation domain. We also compare our work with some previous studies. Finally, we outline the limitations of this work.

4.1 Experimental Setup

In our experiments, we utilized Kaggle notebooks to build, test the models, as well as to synthesize the ABC-S Dataset. The Kaggle platform offers a cost-free, browser-based environment that enables users to discover and share datasets, write, save and execute codes on Jupyter notebooks while utilizing powerful computing resources, including CPUs, GPUs and TPUs [25]. For our implementation and testing, we employed Python 3.7.10, Transformers 4.5.1 and PyTorch 1.9.1 libraries. The test set from the ABC-S along with the ABC-C datasets, was used to assess the proposed models. We evaluated the overall performance of the models using the accuracy metric.

4.2 The NERB Model Results

When evaluating the NERB model using the ABC-S dataset (test set), a classification accuracy score of 100% is achieved. This high accuracy is anticipated, because the AraBERT model is a transformer designed for the Arabic language and has undergone extensive training on a large corpus. However, the perfect accuracy score may indicate a notable similarity between the test set and the training set. To address this concern, we conducted an evaluation of the model using the ABC-C dataset, which we collected from 38 volunteers to ensure high diversity. This dataset comprises 200 booking tickets distributed across the seven domains.

Table 9 displays the classification report for evaluating the NERB model on the ABC-C dataset. This report includes precision, recall and F1-score metrics for the 25 entity categories. It's important to note that the "Phone" entity with a tag of 12 is not present, as none of the volunteers provided it in the collected reservation queries within the ABC-C dataset. The last column shows the support or the frequency of each entity in the dataset. In this table, we can see that three entities that have an F1-score value of 1.000, while many entities have a high value of F1-score regardless of their support in the dataset. For example, the “Stadium Name” entity has a support of 11 with an F1-score of 1.000 and the “City 1” entity has a support of 37 with an F1-score of 0.829. Overall, the NERB model achieved a classification accuracy of 95.5% on the ABC-C dataset.

Table 9. The classification report of the NERB model for the ABC-C dataset.

Label	Entity Name	Precision	Recall	F1-Score	Support
0	Others	0.985	0.964	0.974	1,798
3	Date and Day 1	0.984	0.966	0.975	262
9	Number of Tickets	0.927	0.953	0.940	214
4	Time 1	0.894	0.988	0.939	163
7	Ticket Class	0.916	0.950	0.933	80
23	Period	0.927	0.900	0.913	70
1	City 1	0.756	0.919	0.829	37
19	Movie	0.941	0.889	0.914	36
21	Car Name	1.000	0.853	0.921	34
16	Hotel	0.952	0.769	0.851	26
17	Room	0.677	0.920	0.780	25
25	Clinic Name	0.846	0.880	0.863	25
11	Doctor Name	0.950	1.000	0.974	19
5	Date and Day 2	0.864	1.000	0.927	19
13	First Team	1.000	1.000	1.000	17
14	Second Team	0.944	1.000	0.971	17
2	City 2	0.923	0.800	0.857	15
18	Restaurant	0.684	0.929	0.788	14
10	Name	0.619	1.000	0.765	13
20	Cinema	0.818	0.692	0.750	13
6	Time 2	1.000	1.000	1.000	13
24	Stadium Name	1.000	1.000	1.000	11
8	Round Trip	1.000	0.750	0.857	8
15	League	1.000	0.667	0.800	6
22	Car Model	0.333	1.000	0.500	1
Macro-average		0.878	0.912	0.881	Total: 2,936
Accuracy					**95.5%**

To better understand where the model fails in classification, Figure 4 shows the confusion matrix for the NERB model when evaluated using the ABC-C dataset.

	Label		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25
1	0	Others	1734	5	6	1	0	0	0	1	0	2	2	1	2	0	2	5	0	4	2	13	0	0	7	0	11
2	1	City1	3	34	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	10	Name	0	0	13	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	11	Doctor Name	0	0	0	19	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
5	13	First Team	0	0	0	0	17	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
6	14	Second Team	0	0	0	0	0	17	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
7	15	League	2	0	0	0	0	0	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8	16	Hotel	0	1	0	0	0	0	0	20	5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
9	17	Room	1	0	0	0	0	0	0	0	23	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
10	18	Restaurant	1	0	0	0	0	0	0	0	0	13	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
11	19	Movie	3	0	1	0	0	0	0	0	0	0	32	0	0	0	0	0	0	0	0	0	0	0	0	0	0
12	2	City2	0	2	0	0	0	1	0	0	0	0	0	12	0	0	0	0	0	0	0	0	0	0	0	0	0
13	20	Cinema	0	2	0	0	0	0	0	0	0	2	0	0	9	0	0	0	0	0	0	0	0	0	0	0	0
14	21	Car Name	3	0	0	0	0	0	0	0	2	0	0	0	0	29	0	0	0	0	0	0	0	0	0	0	0
15	22	Car Model	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
16	23	Period of Time	1	0	0	0	0	0	0	0	2	0	0	0	0	0	0	63	0	0	2	0	0	0	0	0	2
17	24	Stadium Name	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	11	0	0	0	0	0	0	0	0
18	25	Clinic Name	0	0	0	0	0	0	0	0	0	2	0	0	0	0	0	0	0	22	0	0	0	0	0	0	1
19	3	Date And Day1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	253	6	3	0	0	0	0
20	4	Time1	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	161	0	0	0	0	0
21	5	Date and Day2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	19	0	0	0	0
22	6	Time2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	13	0	0	0
23	7	Ticket Class	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	76	0	0
24	8	RoundTrip	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	6	1
25	9	Number Of Tickets	6	1	1	0	0	0	0	0	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	204

Predictions

Figure 4. Confusion matrix for the NERB-model predictions on the ABC-C dataset.

From this matrix, we can see that a total of 64 instances of the “Others” entity were classified as one

of the defined entities in the dataset. Also, the low precision of the “Name” entity, which has a support of 13, is due to the fact that 6 of the “Others” entities are classified as “Name” entities; despite the fact that the “Name” entity has a recall of 1. On the other hand, the “League” entity with a support of 6 has a recall value of 0.667 and a precision value of 1.000.

4.3 The DA-NERB Model Results

Like the evaluation results of the NERB model on the ABC-S dataset, the DA-NERB model achieved a classification accuracy score of 100%. Therefore, we turned to using the ABC-C dataset for evaluation. In Table 10, you can observe the classification report for evaluating the DA-NERB model on the ABC-C dataset, where it achieved a classification accuracy of 96.4%, surpassing the NERB model's performance. Furthermore, the table highlights that the DA-NERB model has improved F1-score values for many entities in comparison to the NERB model. For example, the 'Name' entity, which had an F1-score of 0.765 for the NERB model, reached a perfect F1-score of 1.000 with the DA-NERB model. Nevertheless, this model obtained lower F1-scores for a few entities, such as the 'First Team' and 'Cinema' entities.

Table 10. The classification report of the DA-NERB model for the ABC-C dataset.

Label	Entity Name	Precision	Recall	F1-Score	Support
0	Others	0.983	0.973	0.978	1,798
3	Date and Day 1	0.960	1.000	0.979	262
9	Number of Tickets	0.928	0.963	0.945	214
4	Time 1	0.988	0.969	0.978	163
7	Ticket Class	0.948	0.913	0.930	80
23	Period	0.986	0.971	0.978	70
1	City 1	0.897	0.946	0.921	37
19	Movie	0.897	0.972	0.933	36
21	Car Name	1.000	0.647	0.786	34
16	Hotel	1.000	0.808	0.894	26
17	Room	0.759	0.880	0.815	25
25	Clinic Name	0.880	0.880	0.880	25
11	Doctor Name	0.905	1.000	0.950	19
5	Date and Day 2	0.950	1.000	0.974	19
13	First Team	0.944	1.000	0.971	17
14	Second Team	1.000	1.000	1.000	17
2	City 2	0.929	0.867	0.897	15
18	Restaurant	0.778	1.000	0.875	14
10	Name	1.000	1.000	1.000	13
20	Cinema	0.526	0.769	0.625	13
6	Time 2	1.000	0.923	0.960	13
24	Stadium Name	0.917	1.000	0.957	11
8	Round Trip	0.857	0.750	0.800	8
15	League	0.714	0.833	0.769	6
22	Car Model	0.500	1.000	0.667	1
Macro-average		0.890	0.923	0.899	Total: 2,936
Accuracy					**96.4%**

Figure 5 displays the confusion matrices for the ABC-C dataset of the DA-NERB model. According to this figure, the "Car Name" entity has a precision value of 1.000 while this entity has a low recall value of 0.647, because the model classifies the "Car Name" entity as the other entities. The DA-NERB model has notably enhanced precision values for many entities, as seen in the case of the 'City 1' entity, with a precision value of 0.756 for the NERB model and an improved value of 0.897 for the DA-NERB model. Additionally, the DA-NERB model has improved recall values for various entities, such as the 'Movie' entity, which has a recall value of 0.889 for the NERB model and an improved value of 0.972 for the DA-NERB model. However, there were a few instances where the results were opposite, as observed in the case of the 'Car Name' entity, where the NERB model achieved a recall value of 0.853, while the DA-NERB model had a lower recall value of 0.647. Overall, the DA-NERB model demonstrated an improvement in the F1-score for numerous entities.

As we mentioned in Sub-section 3.2.2, the DA-NERB model is aware of the reservation domain of the input query; hence, we can apply the post-processing step that we discussed earlier to refine the classified tags and identify the missing tags in the query based on the domain. Effectively, applying the post-processing step improved the DA-NERB model and increased the accuracy to 96.9% when the ABC-C dataset was considered.

	Label		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	
1	0	Others	1750	2	0	0	0	0	2	0	1	2	2	1	9	0	0	1	0	3	4	2	0	0	4	1	14	
2	1	City1	1	35	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	
3	10	Name	0	0	13	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
4	11	Doctor Name	0	0	0	19	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
5	13	First Team	0	0	0	0	17	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
6	14	Second Team	0	0	0	0	0	17	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
7	15	League	0	0	0	0	1	0	5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
8	16	Hotel	0	0	0	0	0	0	0	21	5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
9	17	Room	3	0	0	0	0	0	0	0	22	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
10	18	Restaurant	0	0	0	0	0	0	0	0	0	14	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
11	19	Movie	1	0	0	0	0	0	0	0	0	0	35	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
12	2	City2	0	2	0	0	0	0	0	0	0	0	0	13	0	0	0	0	0	0	0	0	0	0	0	0	0	
13	20	Cinema	0	0	0	0	0	0	0	0	0	2	1	0	10	0	0	0	0	0	0	0	0	0	0	0	0	
14	21	Car Name	11	0	0	0	0	0	0	0	0	0	0	0	0	22	1	0	0	0	0	0	0	0	0	0	0	
15	22	Car Model	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	
16	23	Period of Time	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	68	0	0	2	0	0	0	0	0	0	
17	24	Stadium Name	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	11	0	0	0	0	0	0	0	0	
18	25	Clinic Name	0	0	0	2	0	0	0	0	0	0	0	0	0	0	0	0	0	22	0	0	0	0	0	0	1	
19	3	Date And Day1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	262	0	0	0	0	0	0	
20	4	Time1	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	4	158	0	0	0	0	0	
21	5	Date and Day2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	19	0	0	0	0	
22	6	Time2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	12	0	0	0	
23	7	Ticket Class	7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	73	0	0	
24	8	RoundTrip	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	6	1	
25	9	Number Of Tickets	6	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	206	
			Predictions																									Actuals

Figure 5. Confusion matrix for the DA-NERB model predictions on the ABC-C dataset.

Table 11. Sample from the ABC-C dataset and the outputs of the NERB, DA-NERB and the post-processing step.

Words in Query	True Label		NERB Output		DA-NERB Output		Post-processing	
	Tag	Entity	Tag	Entity	Tag	Entity	Tag	Entity
مسا	0	Other	**4**	Time 1	0	Other	0	Other
الخير،	0	Other	0	Other	0	Other	0	Other
بأدر	0	Other	0	Other	0	Other	0	Other
احجز	0	Other	0	Other	0	Other	0	Other
الجي	21	Car Name	**17**	Room	**0**	Other	0	Other
كلاس	21	Car Name	**17**	Room	**0**	Other	0	Other
لثلاث	23	Period	23	Period	23	Period	23	Period
أيام	23	Period	23	Period	23	Period	23	Period
بس	0	Other	0	Other	0	Other	0	Other
ضروري	0	Other	0	Other	0	Other	0	Other
كثير	0	Other	0	Other	0	Other	0	Other
تكون	0	Other	0	Other	0	Other	0	Other
لبلاك	0	Other	**10**	Customer Name	**9**	Number of Tickets	**0**	Other
لأنها	0	Other	0	Other	0	Other	0	Other
طلب	0	Other	0	Other	0	Other	0	Other
المدام،	0	Other	**10**	Customer Name	0	Other	0	Other
ممنونكم	0	Other	0	Other	0	Other	0	Other

To illustrate the concept of the post-processing step, Table 11 provides the output of the NERB, DA-NERB and the post-processing step for a sample from the ABC-C dataset. The sample is " مسا الخير، بأدر احجز الجي كلاس لتلات ايام بس ضروري كتيير تكون لبلاك لأنها طلب المدام، ممنونكم." (Good Evening, I want to book the G class for three days, but it has to be black, because my wife asked for it, thank you.), which is a query for a car rental. The NERB output has five errors, while the DA-NERB has three errors, which are shown with underlines in the table. Both models failed to correctly tag the car model "الجي كلاس" (G Class) and the car color "لبلاك" (black). However, applying the post-processing step to the DA-NERB output corrected the classification of the car color to 0, which matches the true label of "Others". Furthermore, the post-processing step identified missing tags in the query, such as tags 3 (Start Date), 4 (Start Time), 5 (End Date), 6 (End Time), 21 (Car Name) and 22 (Car Model). These missing tags are expected to be acquired by prompting the user interactively once the model is deployed.

4.4 Comparison with the Single-Domain

For a more comprehensive and fair comparison, we conducted an experiment to evaluate the performance of the DA-NERB model when compared to the performance of the NERB when it is trained on a single domain. For this purpose, we trained seven versions of the NERB model, each tailored to one of the seven domains. Technically, each of these single-domain models was trained using a sub-set of the samples from the ABC-S dataset specific to the domain of that model. For testing, we assessed these individual models alongside the DA-NERB model using the domain-specific samples from the ABC-C dataset. Table 12 provides a comparison between the single-domain models and the DA-NERB model with post-processing. It is evident that the accuracy of the DA-NERB model consistently surpasses that of each of the single-domain models.

Table 12. Comparison between the DA-NERB model with post-processing and the single-domain models.

Domain Type	**ABC-C Dataset Samples/Entities**	**Single-domain NERB Models (Without Post-processing)**	**DA-NERB Model (With Post-processing)**
		Accuracy	**Accuracy**
Flight Booking	30 / 494	96%	97%
Hotel Booking	24 / 393	89%	96%
Cinema Booking	29 / 391	95%	96%
Football Match Booking	29 / 375	94%	98%
Car Booking	29 / 401	93%	95%
Restaurant Booking	30 / 446	94%	98%
Clinic Appointment Booking	31 / 436	95%	98%

4.5 Comparison with Previous Works

Fadhil [12] focused on the patient follow-up domain, Al-Ajmi and Al-Twairesh [10] handled the flight booking domain and most of the Arabic NER works focused on a single domain, whereas this work addressed seven different domains.

Shaker *et al.* [22] introduced an Arabic NER dataset encompassing 9 categories of named entities, comprising over 36,000 records, with texts spanning 7 different domains. This work presented the ABC-S Dataset, which comprises 76,117 reservation samples spanning 7 different domains and encompassing 26 categories of named entities.

We fine-tuned AraBERTv0.2 on the ABC-S dataset that supports 26 different entity classes; in contrast, Antoun *et al.* [18] fine-tuned AraBERTv0.1 on ANERcorp which contains entities belonging only to 4 different classes.

4.6 Limitations

One of the constraints in this work is the utilization of the DA-NER model to recognize novel entities that fall outside the reservation domains on which it was previously trained. Another constraint in this work is the dialect used, especially when the reservation request is entered in a dialect other than the Levantine dialect. In such cases, the model may be able to recognize certain entities, such as time and date, but it may face difficulty in recognizing other entities that rely on the text context.

5. CONCLUSION

In this paper, we proposed the design and evaluation of a named entity recognition model for Arabic chatbots. We achieved this by fine-tuning and extending the state-of-the-art AraBERT to recognize booking entities and information from unstructured Arabic reservation queries. Due to the scarcity of datasets for reservation queries in Arabic, we proposed the synthesis of the ABC-S dataset and the collection of the ABC-C datasets for training and testing the proposed models. These datasets contain reservation queries for seven domains.

Effectively, we proposed two models: the NERB and DA-NERB. The NERB model is basically the AraBERT model that is modified by adding an output layer of 26 outputs and trained on the ABC-S

dataset. On the other hand, the DA-NERB extends the NERB model by including an additional input for the domain type and an embedding layer to incorporate prior knowledge about the domain.

Experimental evaluation of these models proved the ability of the NERB and DA-NERB models to recognize the entities in the queries in the ABC-C dataset with an accuracy of 95.5% and 96.4%, respectively. Furthermore, we proposed improving the DA-NERB classification accuracy using a post-processing step utilizing the fact that the domain type is known. The classification accuracy of the DA-NERB increased to 96.9% using this post-processing.

One major contribution of this work is building a specialized dataset for Arabic unstructured reservation texts. Specifically, we built the Arabic Booking Chatbot Dataset (ABC-S Dataset), which contains 76,117 reservation samples encompassing 26 entity types for seven different reservation domains. Also, we collected samples for booking tickets from 38 volunteers. These datasets are publicly available to other researchers to contribute to the development of Arabic chatbots.

In the future, we may eliminate the need for post-processing by utilizing other pre-trained language models, such as MARBERT [27] and AraT5 [26]. Additionally, it is possible to train the model on additional dialects. Moreover, we are looking to integrate our model with an available library to build a chatbot reservation system and then enable the users to book tickets in Arabic text and set some conditions and restrictions for booking tickets, in order to avoid misunderstanding when using the booking chatbot.

REFERENCES

[1] E. H. Almansor and F. K. Hussain, "Survey on Intelligent Chatbots: State-of-the-art and Future Research Directions," Proc. of Conf. on Complex, Intelligent and Software Intensive Systems (CISIS 2019), vol. 993, pp. 534–543, 2020.

[2] M. Al-Ayyoub et al., "Deep Learning for Arabic NLP: A Survey," JOCSCI, vol. 26, pp. 522-531, 2018.

[3] S. AlHumoud, A. Al Wazrah and W. Aldamegh, "Arabic Chatbots: A Survey," IJSCSA, vol. 9, no. 8, pp. 535-541, 2018.

[4] M. Mnasri, "Recent Advances in Conversational NLP: Towards the Standardization of Chatbot Building," DOI: 10.48550/arXiv.1903.09025, Clermont-Ferrand, France, 2019.

[5] Infobip, "The Intelligent Chatbot Building Platform," [Online], Available: https://www.infobip.com, 2021.

[6] T. Alshareef and M. A. Siddiqui, "A seq2seq Neural Network Based Conversational Agent for Gulf Arabic Dialect," Proc. of the 2020 21st IEEE Int. Arab Conf. on Information Technology (ACIT), pp. 1-7, Giza, Egypt, 2021.

[7] Y. Saoudi and M. M. Gammoudi, "Trends and Challenges of Arabic Chatbots: Literature Review," Jordanian Journal of Computers and Information Technology (JJCIT), vol. 9, no. 3, pp. 261-286, 2023.

[8] Class Central, "Microsoft Bot Framework and Conversation as a Platform," [Online], Available: https://www.classcentral.com/course/edx-microsoft-bot-framework-and-conversation-as-a-platform-11325, 2021.

[9] Chatbots Life, "Best Chatbot Development Frameworks | RASA | IBM Watson | Dialogflow," [Online], Available: https://chatbotslife.com/best-chatbot-development-frameworks-rasa-ibm-watson-dialogflow-e2792f9363eb, 2019.

[10] A. H. Al-Ajmi and N. Al-Twairesh, "Building an Arabic Flight Booking Dialogue System Using a Hybrid Rule-based and Data Driven Approach," IEEE Access., vol. 9, pp. 7043-7053, Jan. 2021.

[11] D. Al-Ghadhban and N. Al-Twairesh, "Nabiha: An Arabic Dialect Chatbot, " IJACSA, vol. 11, no. 3, pp. 452-459, 2020.

[12] A. Fadhil, "Ollobot-Towards a Text-based Arabic Health Conversational Agent: Evaluation and Results," Proc. of the Int. Conf. on Recent Advances in Natural Language Processing (RANLP 2019), pp. 295-303, Varna, Bulgaria, 2019.

[13] G. Mesnil et al., "Using Recurrent Neural Networks for Slot Filling in Spoken Language Understanding," IEEE/ACM Transactions on Audio, Speech and Language Processing, vol. 23, no. 3, pp. 530-539, DOI:10.1109/TASLP.2014.2383614, Mar. 2015.

[14] W. Antoun, F. Baly and H. Hajj, "AraBERT: Transformer-based Model for Arabic Language Understanding," Proc. of the 4th Workshop on Open-Source Arabic Corpora and Processing Tools, with a Shared Task on Offensive Language Detection (LREC2020), pp. 9-15, Marseille, France, 2020.

[15] J. Devlin, M. W. Chang, K. Lee and K. Toutanova, "Bert: Pre-training of Deep Bidirectional Transformers for Language Understanding," Proceedings of NAACL-HLT2019, pp. 4171–4186, Minneapolis, Minnesota, June2-June7, 2019.

[16] R. Horev, "BERT Explained: State of the art Language Model for NLP," Towards Data Science,

[Online], Available: https://towardsdatascience.com/bert-explained-state-of-the-art-language-model-for-nlp-f8b21a9b6270, 2018.

[17] D. Shulga, "BERT to the Rescue," Towards Data Science, [Online], Available: https://towardsdatascience.com/bert-to-the-rescue-17671379687f, 2019.

[18] W. Antoun, "Aub-mind/arabert: Pre-trained Transformers for the Arabic Language Understanding and Generation (Arabic Bert, Arabic GPT2, Arabic Electra)," GitHub, Edited by M. Al Salti et al. AUB MIND, Beirut, Lebanon, [Online], Available: https://github.com/aub-mind/arabert, 2020.

[19] A. Youssef, M. Elattar and S. R. El-Beltagy, "A Multi-embeddings Approach Coupled with Deep Learning for Arabic Named Entity Recognition," Proc. of the 2020 2nd IEEE Novel Intelligent and Leading Emerging Sciences Conf. (NILES), pp. 456-460, Giza, Egypt, 2020.

[20] E. Taher, S. A.Hoseini and M.Shamsfard, "Beheshti-NER: Persian Named Entity Recognition Using BERT," Proc. of the 1st Int. Workshop on NLP Solutions for Under Resourced Languages (NSURL 2019) Co-located with ICNLSP 2019, pp. 37-42, Trento, Italy, 2020.

[21] B. A. Benali, S. Mihi, N. Laachfoubi and A. A. Mlouk, "Arabic Named Entity Recognition in Arabic Tweets Using BERT-based Models," Procedia Computer Science, vol. 203, pp. 733-738, 2022.

[22] A. Shaker, A. Aldarf and I. Bessmertny, "Using LSTM and GRU with a New Dataset for Named Entity Recognition in the Arabic Language," arXiv: 2304.03399, DOI: 10.48550/arXiv.2304.03399, 2023.

[23] X. Qu, Y. Gu, Q. Xia et al., "A Survey on Arabic Named Entity Recognition: Past, Recent Advances and Future Trends," arXiv: 2302.03512, DOI: 10.48550/arXiv.2302.03512, 2023.

[24] B. Saddar and R. Saddar, "Boshra-sadder/Arabic Booking Chatbot," GitHub, [Online], Available: https://github.com/Boshra-sadder/Arabic-Booking-Chatbot, Amman, Jordan, 2021.

[25] G.Yufeng, "Introduction to Kaggle Kernels," Towards Data Science, [Online], Available: https://towardsdatascience.com/introduction-to-kaggle-kernels-2ad754ebf77, 2017.

[26] E. M. B. Nagoudi, A. Elmadany and M. Abdul-Mageed, "TURJUMAN: A Public Toolkit for Neural Arabic Machine Translation," arXiv: 2206.03933, DOI: 10.48550/arXiv.2206.03933, 2022.

[27] B. AlKhamissi et al., "Adapting MARBERT for Improved Arabic Dialect Identification: Submission to the NADI," arXiv: 2103.01065, DOI: 10.48550/arXiv.2103.01065, 2021.

ملخص البحث:

لقد أصبحت أنظمة التَّحاور أساسية في الآونة الأخيرة في العديد من المجالات، بدءاً من خدمة الزّبائن والحصول على المعلومات وانتهاءً بالتّرفيه، وممّا يجدر ذكره أنّ استخدام أنظمة التَّحاور من شأنه أنْ يخفّض الكلفة التّشغيلية ويقلّل الأخطاء البشرية بالإضافة الى توفيره للخدمة في جميع الأوقات.

يُقدّم هذا البحث نموذجاً لتمييز الكينونات المسمّاة (NER) لتصميم وتطبيق وتقييم نظامٍ للتَّحاور باللّغة العربية يتعلّق بتقديم خدمات الحجْز، ويُركّز على حجز التذاكر والمواعيد في سبعة مجالات مختلفة، وهي الرحلات الجوية، والفنادق، ودور السينما، ومباريات كرة القدم، والسيارات، والمطاعم، والعيادات. ويمهّد هذا البحث الطّريق لتطوير المزيد من أنظمة التَّحاور باللّغة العربية التي بإمكانها أن تدعم العديد من مجالات الحجز. وتستخدم هذه الورقة تعلُّم الآلة العميق وتعلُّم النقل لإنجاز هذه المهمّة. وعلى وجه التّحديد، يستخدم النّموذج المقترح في هذه الدّراسة النموذج الاساسي AraBERTv0.2 لتدريب وتقييم أداء النّموذج. اعتمدت الدّراسة على إنشاء مجموعة بيانات باللّغة العربية تشمل (76117) عيّنة حجْز، وتغطي سبعة مجالات مختلفة، وتتضمّن (26) صنفاً من الكينونات المسمّاة. بالإضافة الى ذلك، تمّ جمْع مجموعة بياناتٍ من مُتطوّعين لتقييم النّموذج المقترح بعد تدريبه. ويُشار إلى أنّ مجموعتي البيانات المذكورتين مكتوبتان باللّغة العربية المتداولة في المشْرق. وقد بلغت دقّة النّموذج المقترح 100% و 96.9% عند تطبيقّه على مجموعتي البيانات المشار إليهما أعلاه. والمتوقّع أن يدعم النّموذج المقترح البحث العلمي المتعلّق بأنظمة التَّحاور باللُّغة العربية.