

Article

ArRASA: Channel Optimization for Deep Learning-Based Arabic NLU Chatbot Framework

Meshrif Alruily

Faculty of Computer and Information Sciences, Jouf University, Sakaka 72388, Saudi Arabia; mfalruily@ju.edu.sa

Abstract: Since the introduction of deep learning-based chatbots for knowledge services, many research and development efforts have been undertaken in a variety of fields. The global market for chatbots has grown dramatically as a result of strong demand. Nevertheless, open-domain chatbots' limited functional scalability poses a challenge to their implementation in industries. Much work has been performed on creating chatbots for languages such as English, Chinese, etc. Still, there is a need to develop chatbots for other languages such as Arabic, Persian, etc., as they are widely used on the Internet today. In this paper, we introduce, ArRASA as a channel optimization strategy based on a deep-learning platform to create a chatbot that understands Arabic. ArRASA is a closed-domain chatbot that can be used in any Arabic industry. The proposed system consists of four major parts. These parts include tokenization of text, featurization, intent categorization and entity extraction. The performance of ArRASA is evaluated using traditional assessment metrics, i.e., accuracy and F1 score for the intent classification and entity extraction tasks in the Arabic language. The proposed framework archives promising results by securing 96%, 94% and 94%, 95% accuracy and an F1 score for intent classification and entity extraction, respectively.

Keywords: Arabic; natural language understanding; deep learning; chatbot



Citation: Alruily, M. ArRASA: Channel Optimization for Deep Learning-Based Arabic NLU Chatbot Framework. *Electronics* **2022**, *11*, 3745. <https://doi.org/10.3390/electronics11223745>

Academic Editors: Grzegorz Dudek and Rui Pedro Lopes

Received: 8 September 2022

Accepted: 7 November 2022

Published: 15 November 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2022 by the author. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

The Arabic language is among the five most commonly used online languages. It is used for conversion among large communities around the world. Technological innovation has a significant impact on people's lives. In the field of computer science, a great deal has been accomplished in recent years. Artificial intelligence (AI) has produced remarkable and important achievements [1]. As AI interacts with various scientific fields, it has an increased in importance in recent years. Many aspects of artificial intelligence are covered, including natural language processing (NLP). A human language is a natural language (English, Arabic, or Spanish). In AI, machine learning and chatbots, intent classification is classifying someone's intent by analyzing their language. Intent classification is a kind of NLP that focuses on categorizing text into different categories to interpret text better.

Intents are generic characteristics that link a user's text to a bot action (prediction workflow). For example, the whole sentence "What is the weather today?" will map to the 'weather inquiry' intent, not just a portion of it. [2]. In the last decade, intent classification and entity extraction have been widely used to develop chatbots for various languages. Since Arabic is the fourth-largest language used on the Internet, intent classification for Arabic is necessary. Very little work has been performed on Arabic intent classification and entity extraction. Moreover, there are a few articles on the development of chatbots for the Arabic language.

Machine learning and natural language processing are used in intent classification to automatically match specific sentences and words with a suitable intent [3]. For example, a machine learning model could discover that words, such as buy and acquire, are often linked with the urge to buy. On the other hand, intent classifiers must be trained using text samples, often referred to as training data. Tags such as Interested, Need Information,

Unsubscribe, Wrong Person, Email Bounce, Autoreply and others may be useful when going through customer emails.

NLP is a subfield of AI that uses natural language to allow human–computer interaction and communication. NLP has spawned a slew of new applications. A chatbot is one of the most intriguing natural language artificial intelligence applications [4]. A chatbot is software that uses natural language to conduct a human–computer interaction through auditory or textual means. As a result, it functions as a virtual assistant that uses artificial intelligence to mimic conversational abilities and human-like behavior [5]. It also contains embedded information that aids in identifying and comprehending the phrase and the generation of the right answer. Many research articles have been published in this area due to the importance of sentiment analysis. However, this research has concentrated on English and other Indo-European languages. In morphologically rich languages such as Arabic, there has been very little research on sentiment analysis [6]. Despite this, many academics have focused on sentiment analysis in Arabic due to the growing number of Arabic internet users and the exponential development of Arabic online content in the past decade. Pipelines can be used to streamline a workflow for machine learning. Pre-processing, extraction of features, categorization and post-processing are all possible steps in the pipeline. Many other necessary phases in this pipeline may be added according to the complexity of applications. By optimization, we intend to modify the model for optimum performance. Any learning model’s effectiveness depends on choosing the parameters that produce the greatest outcomes. The concept of optimization can be compared to a search algorithm that explores a range of parameters and picks out the best among them.

Because Arabic is such a complicated language, developing Arabic chatbots has posed a significant challenge to the academic community. Only a few works have tried to create Arabic chatbots so far. ArabChat [7] is one such project, as follows: a rule-based chatbot capable of pattern matching and delivering appropriate responses to user inquiries. BOTTA [8], another project, is a retrieval-based model that supports the Egyptian dialect. Ollobot is a rule-based chatbot that provides health monitoring and assistance in the medical sector [9]. However, because of its restricted functional scalability, an open-domain chatbot cannot be successfully used in every business. Furthermore, since most chatbot frameworks are written in English, building an efficient and multi-objective chatbot for Arabic is necessary. This research proposes an ArRASA, a pipeline optimization approach based on a deep learning-based open-source chatbot system that understands Arabic, to solve this issue.

The proposed approach consists of the following steps: Tokenization, feature extraction, specific intent classification and suitable entity extraction are the four phases of this closed-domain chatbot. A closed-domain chatbot, also known as a domain-specific chatbot, focuses on a certain range of issues and provides limited replies based on the business issue. For instance, a food delivery chatbot can only let users place, monitor, or cancel an order. Such straight-shooting discussion is kind of bumping into an acquaintance as follows: you expect people to be likely to inquire about your work and maybe comment on the environment. You have prepared answers to every topic, and the idea is just to satisfy the enquiries. While an open-domain chatbot is required to grasp any matter and provide appropriate answers. The proposed model can be scaled by adding more intents and entities. Open-domain chatbots are less effective in the industry, so the proposed study focuses on developing closed-domain chatbots. Moreover, a handsome amount of work is performed using traditional machine and deep learning approaches, while the proposed study uses transformers-based techniques for the development of a more effective and reliable chatbot.

This paper discusses ArRASA, a pipeline optimization approach based on a deep learning-based open-source chatbot system that understands Arabic. To cope with this topic, first, we discuss the Arabic language and its different perspectives and challenges, as it has some special characteristics and rules to deal with a problem. We discuss the related approaches in the literature review section. The proposed methodology discusses

the complete operations of the ArRASA. The proposed model can be scaled by adding more intents and entities. In terms of Arabic language understanding, an optimization experiment is carried out at each step. The prime contributions of the proposed solution can be summarized as follows:

- ArRASA is a channel optimization strategy proposed based on a deep-learning platform to create a chatbot that understands Arabic;
- ArRASA is a novel approach for a closed-domain chatbot using RASA (an open-source conversational AI platform) that can be used in any Arabic industry;
- Tokenization, feature extraction, specific intent classification and suitable entity extraction are the four phases of the proposed approach;
- The performance of ArRASA is evaluated using traditional assessment metrics, i.e., accuracy and F1 score for the intent classification and entity extraction tasks in the Arabic language;
- The performance is also compared with the existing approaches regarding accuracy and F1 score.

The remainder of the paper is organized as follows: Section 2 discusses the related work and Arabic language, and its challenges are elaborated in Section 3. The proposed solution is developed in Section 4, while the system structure is discussed in Section 5. The performance evaluation is presented in Sections 6 and 7 concludes the paper.

2. Related Work

BOTTA [8] is the first Arabic dialect chatbot developed for Egyptian Arabic to work as a conversational agent to mimic user-friendly chats. Various components of the BOTTA chatbot are defined, and it presents various solutions. Researchers working on Arabic chatbot technology can access the BOTTA database files for free and with public access.

Shawar et al. [10] show how machine-learning techniques were used to create an Arabic chatbot that accepts user feedback in Arabic and responds with Qur'anic quotes. A device that learned conversational patterns from a corpus of transcribed conversations was used to create various chatbots that spoke English, French and Afrikaans. Because the Qur'an is not a copy of a dialogue, the learning method has been altered to accommodate the Qur'an's format in terms of sooras and ayyas.

Bashir et al. [11] propose a method for using named entity recognition and text categorization using deep learning methods in the Arabic area of home automation. To do this, we provide an NLU module that can be further combined with an ASR, a conversation manager and a natural language generator module to create a fully functional dialogue system. The process of gathering and annotating the data, constructing the intent classifier and entity extractor models, and ultimately the assessment of these techniques against various benchmarks are all included in the study.

AlHumoud et al. [12] summarize published Arabic chatbot studies to recognize information gaps and illustrate areas that need more investigation and research. This research found a scarcity of Arabic chatbots and that all available works are retrieval-based. The experiments are divided into the following two classes, depending on the method of chatbot communication interaction: text and voice conversational chatbots. The study was presented and assessed according to the deployment method, the duration and breadth of the presentations, and the model used for the chatbot dataset. According to the study, all the assessed chatbots used a retrieval-based dataset model.

Nabiha [13] is a chatbot that uses the Saudi Arabic dialect to converse with King Saud University Information Technology (IT) students. As a consequence, Nabiha will be the first Saudi chatbot to communicate in the Saudi dialect. Nabiha is accessible on several platforms, including Android, Twitter and the Web, to make it simpler to use. Students may contact Nabiha by downloading an app, tweeting her, or visiting her website. According to the students in the IT department who tested Nabiha, the results were acceptable, given the general challenges of the Arabic language and the Saudi dialect.

The study in [14] is the first Arabic end-to-end generative model for task-oriented DS (AraConv). It makes use of various parameters for the multilingual transformer model mT5. We also provide the Arabic-TOD discourse dataset, which was utilized to train and evaluate the AraConv model. Compared to research employing identical monolingual conditions, the findings obtained are fair. We propose joint training, in which the model is jointly trained on Arabic conversation data with data from one or two high-resource languages such as English and Chinese, to minimize issues related to a short training dataset and enhance the AraConv model's outcomes.

Many authors worked on the development of Arabic chatbots, but most of them worked on the development of open-domain chatbots. Open-domain chatbots are less effective in the industry, so the proposed study focuses on developing closed-domain chatbots. Moreover, a handsome amount of work is performed using traditional machine and deep learning approaches, while the proposed study uses transformer-based techniques for the development of a more effective and reliable chatbot. The previous studies focused on the development of chatbots, but few worked on optimizing the proposed techniques. The proposed technique also worked on optimizing the proposed architecture for enhancing the accuracy and efficiency of the proposed Arabic chatbot; Table 1 presents the comparison of previous techniques.

Table 1. Comparison of previous techniques.

Ref.	Domain	Technique	Data	Accuracy
[8]	Open	CNN & RNN	3173 Arabic Questions	94%
[10]	Islamic Hadith	SVM	Islamic hadith	87%
[11]	Open	LSTMs and CNNs	AQMAR	92%
[12]	Open	CNN & SVM	TALAA AFAQ Data	82%
[13]	Open	Hybrid Learning	600 FAQ Pages	85%
[14]	Open	AraConv	Arabic-TOD	68%

Rasa is a platform for building AI-powered, quality chatbots in the industry. It is used by developers all around the world to build chatbots and contextual assistants. ArRASA is a channel optimization strategy based on a deep-learning platform to create a chatbot that understands Arabic. ArRASA is a closed-domain chatbot that can be used in any Arabic industry. As we proposed an optimized Arabic language chatbot using RASA, so we named it ArRASA.

3. Arabic Language

With about 300 million people speaking Arabic, it is one of the world's most commonly spoken languages. It is also widely spoken as the primary language in the Central African Republic of Chad, a non-Arab nation, and a minority language in Afghanistan and Israel (where Arabic and Hebrew are both official languages). Iran and Nigeria are among those countries [15]. Arabic, along with Chinese, English, French, Russian and Spanish, became one of the six official languages of the United Nations in 1974. In India, Indonesia, Pakistan and Tanzania, about one billion Muslims study Arabic as a foreign or second language for liturgical and scholastic reasons. Several Muslim and Arab populations in the United States use Arabic in their everyday contacts and for religious reasons.

Challenges of Arabic Language

Dialectal Variation: Arabic has various dialects, all of which are very distinct from one another, as follows: the official written and reading language is Modern Standard Arabic (MSA), and many dialects are spoken versions of the language [15]. Unlike the MSA, which has an official standard orthography and a plethora of resources, Arabic dialects lack such norms and resources. In terms of phonology, morphology and vocabulary, dialects

differ from MSA and each other. Dialects are not recognized as languages in the Arab world and are not taught in schools. Dialectal Arabic, on the other hand, is widely used in online chats. This is why we believe it is more fitting to concentrate on dialectal Arabic in the sense of a chatbot.

Orthographic Ambiguity and Inconsistency: Arabic orthography uses optional diacritical marks to represent short vowels and consonantal doubling, which are seldom used in the text. As a consequence, there is a lot of uncertainty. Furthermore, Arabic writers often misspell various difficult letters, including the Alif-Hamza forms and Ta-Marbuta [16]. Regarding Arabic dialects, orthography is compounded by the lack of standard orthographies [17].

Geomorphological Richness of Arabic: Gender, number, person, voice, aspect and other characteristics are all inflected in Arabic words, as well as taking a variety of connected clitics [8]. This is especially challenging in the context of a chatbot system. Due to the gender-specific nature of verbs, adjectives and pronouns, the chatbot must answer in the following two ways: for male and female users. The ArRASA solves the geomorphological richness of Arabic issue by using the data about gender, number, person, etc., in the dataset. Moreover, the proposed model has various advantages, i.e., is easy to integrate because it is developed using the open-source RASA platform and supports single and multiple intents.

4. Proposed Solution

As we know, the number of services and products is growing rapidly around the globe. Due to this, the number of queries to the producers is also increasing. To solve this problem, companies hire individuals to serve as customer support for their products and services. However, this procedure of responding to consumers' questions is expensive for the company and quite slow for the users. There must be an effective and accessible approach to addressing this issue. Various researchers presented automated chatbots that give responses to customer queries instead of humans in various languages. Researchers presented different Arabic chatbots to work on the Arabic language. These chatbots have their own limitations. We propose a framework for optimizing Arabic chatbots by using the RASA framework, which is one of the current leading open-source platforms for chatbot development. The reason behind using the RASA is that it has not been used for Arabic chatbots in the past. ArRASA is a channel optimization method that uses a deep-learning model to develop an Arabic chatbot. ArRASA is a closed-domain chatbot that may be utilized in any Arabic industry.

4.1. Natural Language Understanding

Natural language processing (NLP) is concerned with how machines interpret language and promote "natural" back-and-forth contact between humans and computers. On the other hand, natural language comprehension is concerned with a machine's capacity to comprehend human language [17]. NLU is the process of rearranging unstructured data so that computers can "understand" and interpret it. For example, Machine Translation, Automatic Ticket Routing and Questioning Answer are established based on the concept of NLU. Natural language understanding (NLU), which translates natural language speech into a machine-readable format, is the primary technology used by chatbots. Figure 1 depicts the NLUs design, and the NLUs preprocessing phase is split into two parts. Tokenization is the initial step, in which a corpus is divided into tokens, which are grammatically intangible language units. The second stage is featurization, which involves extracting the properties of each token. Following the preprocessing step, the purpose is classified to properly comprehend the user's request, and an object is extracted to give a suitable answer [18]. As a result, a user-friendly chatbot framework can be created. Intent classification is the technique of determining a user's intent based on a user's statement. Entities are preset collections of items that make sense, such as names, organizations, time expressions, numbers and other groups of objects. Various sets of entities are needed to be collected for each chatbot. Intents and entities are used to understand what a user wants and how to generate the correct answer to the user's query. The RASA NLU can be used in

chatbots and AI assistants for language understanding, emphasizing intent categorization and entity extraction. RASA NLU can interpret data using these two features. We formed a training dataset to recognize the intent and extract the entity using the RASA NLU pipeline. Rasa is a platform for building AI-powered, quality chatbots in the industry. Developers use it all around the world to build chatbots and contextual assistants. As described in Section 5, the training set contains a variety of intents and entities, and there may be no entity or multiple entities in any sentence.

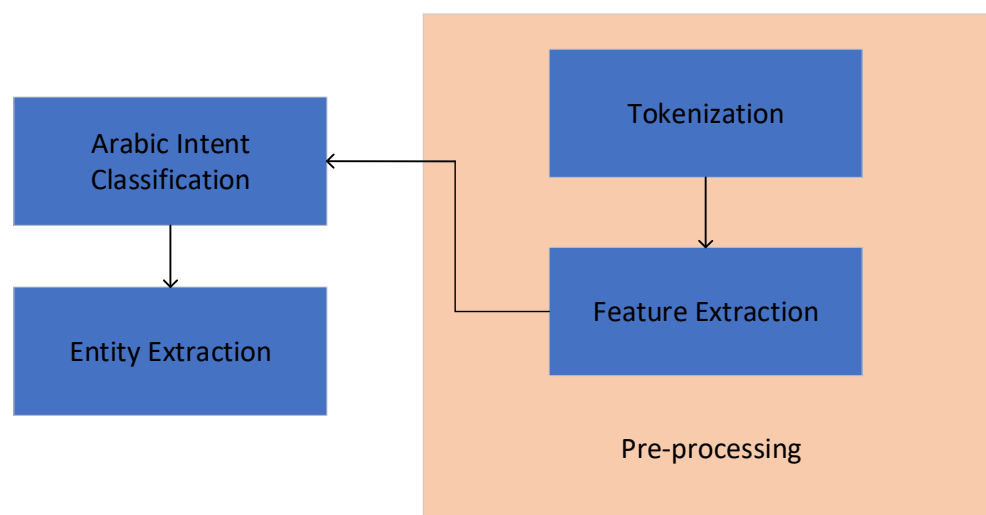


Figure 1. Structure of Arabic NLU.

4.2. Pre-Training Setup

We utilize the masked language modeling (MLM) task to accomplish the first pre-training objective, which involves whole-word masking and replacing 15% of the N input tokens. The [MASK] token is used 80 percent of the time, a random token 10% of the time and the original token 10% to replace those tokens. By forcing the computer to anticipate the whole word rather than only parts of it as whole word masking raises the pre-training challenge. The next sentence prediction (NSP) task, which allows the model to identify the relationship between two phrases and is helpful for a range of language comprehension tasks such as question answering and machine translation, is one that we often employ. We mask a particular percentage of words or phrases in MLM, and the system is intended to guess such masked words based on other words in the text. Since the representation of the masked term is learned based on the words that appear on both sides of the masked term. Moreover, The MLM systems are bidirectional in structure. As MLM is the process of masking words in a series with a masked token and training the model to populate that mask with a suitable token. As a result, the model will concentrate on both the right and left contexts. So, using the MLM for the proposed model enables the proposed model to understand the context of terms both at the beginning and at the end of a phrase.

4.3. DIET (Dual Intent and Entity Transformer)

DIET is a multi-task transformer that conducts intent classification and entity recognition at the same time. It allows to plug-and-play pre-trained embeddings such as BERT, GloVe, ConveRT and others [19]. No one collection of embeddings consistently performs well across various datasets in the tests. As a result, a modular architecture is particularly essential.

DIET is a software development process that incorporates a modular design. In terms of accuracy and performance, it is comparable to large-scale, pre-trained language models. It is 6X quicker to train than the present state of the art. On language comprehension benchmarks such as GLUE [20] and SuperGLUE [21], large-scale pre-trained language models have demonstrated promising results, with significant gains over previous pre-training techniques such as GloVe [22] and supervised approaches. These embeddings

are well suited to generalize across tasks since they were trained on large-scale natural language text corpora.

Input phrases are interpreted as a series of tokens, either words or sub-words, based on the featurization pipeline. We add a special classification token for the Arabic language at the end of each phrase. Each input token is characterized using sparse and/or dense characteristics. At the token level, one-hot encodings of character n-grams ($n \leq 5$) and multi-hot encodings of character n-grams ($n \leq 5$) are scarce. Because character n-grams contain much redundant information, we use dropout for these sparse features to prevent overfitting. Figure 2 presents the proposed optimized pipeline architecture for the Arabic chatbot.

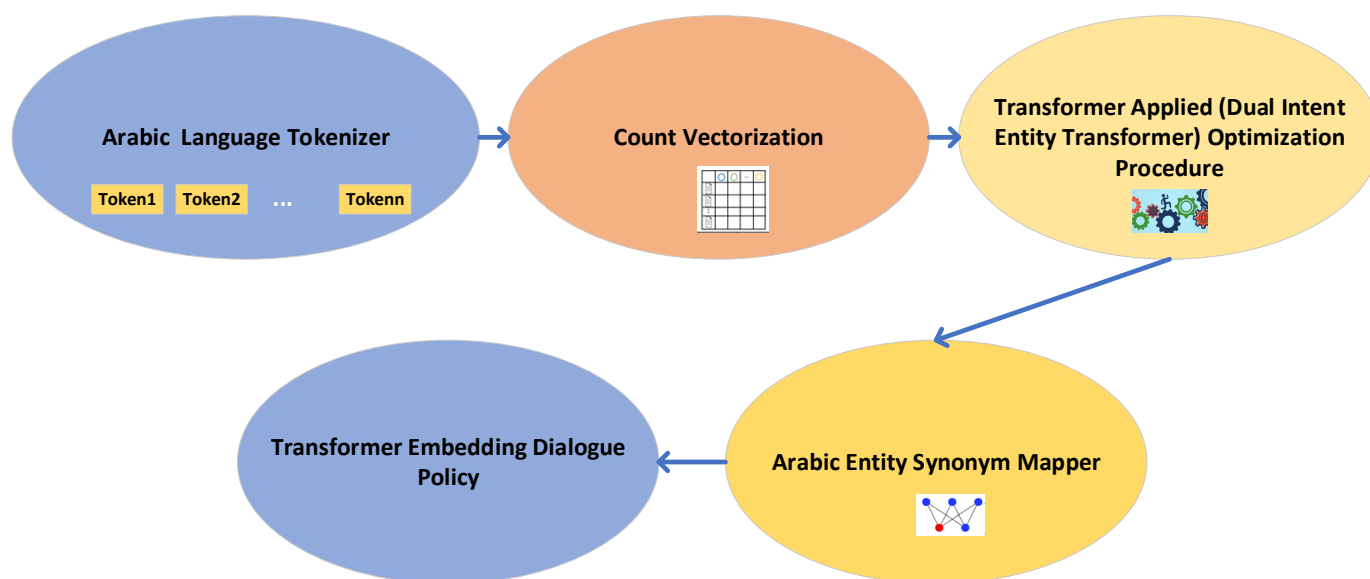


Figure 2. Proposed optimized pipeline architecture for Arabic chatbot.

A two-layer transformer with relative position attention is utilized for encoding context throughout the whole phrase. The input dimension of the transformer design must be the same as the transformer layers. The concatenated features are sent through another fully connected layer with shared weights across all sequence stages to match the dimension of the transformer layers, which is set to 256 for the proposed model.

5. System Architecture

Although the overall architecture of the proposed framework is identical to that of the DIET baseline model (DIET-Base), numerous tests were conducted to enhance intent classification and entity extraction performance. The ideal number of epochs was determined through a performance experiment. The ideal number of epochs was chosen as 100. The structure of the proposed framework is illustrated in Figure 3.

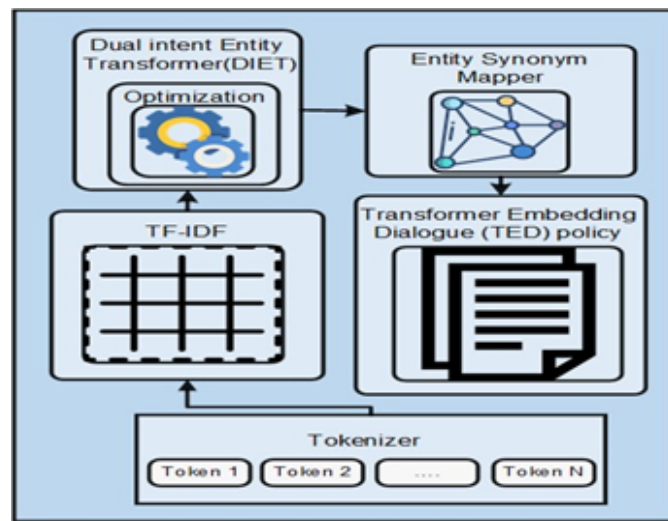


Figure 3. Proposed architecture.

The tokenizer extracts tokens from the data, and these tokens are sent to the proposed transformer layer through feed-forward layers of the LSTM model. The transformer layer provides a sequence (a) of entity labels, which is utilized as input data for the conditional random field (CRF) technique, which generates the sequence (X_{entity}) of entity labels. The loss of an entity can be computed by the negative value of the log-likelihood of CRF, as shown in Equation (1).

$$Entity = CRF(a, X_{entity}) \tag{1}$$

This mechanism reduces the weight sparsity from 0.85 to 0.75, and the number of transformer layers is raised from two to four. For parameter optimization, the number of embedding dimensions of the model increases from 25 to 35, and the hidden layer size increases from 256 to 512. The characteristics of the proposed architecture are listed in Table 2. There are now four total transformer layers, up from the two before.

Table 2. Parameter comparison.

Parameter	DIET-Base	Proposed-DIET
The Number of Transformer Layer	2	3
Masked Language Model	Yes	Yes
Transformer Size	256	512
Drop Rate	0.25	0.25
Embedding Dimension	20	35
Hidden Layer Size	256,128	512,128

6. Experiments and Results

There are various steps involves in the process of intent classification. These steps are given below as follows:

6.1. Data Gathering

The initial phase in the NLP life cycle is data collection. This step aims to find and collect relevant data about the subject. It is indeed among the most crucial stages in the life cycle of NLP [23]. The output’s efficiency will be determined by the quantity and quality of the data collected. The more data there is, the more precise the prediction will be. The dataset used in the study was created using different datasets. Those datasets include the dataset proposed by Bashir et al. [14]. We have used these datasets and some scraped data from Arabic websites and newspapers to create our data for the proposed framework. Datasets were created for the studies in this article based on typical activities that apply to

a broad range of industrial sectors. The data collection includes seven categories for intent categorization. Greeting (greet), closure (goodbye), emotions (happy, sad), food (food-related menu), departmental contact details (dept. contacts), division of labor works (Pers work) and calculator are examples of these categories (Calc). The dataset’s distribution is shown in Figure 4.

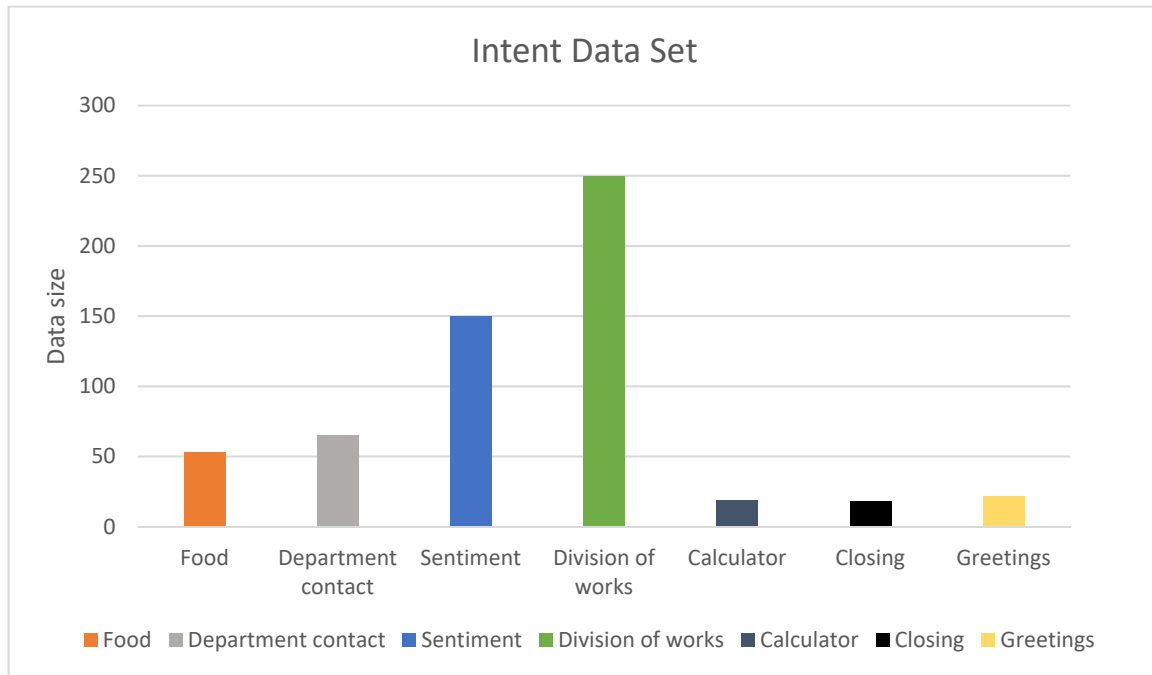


Figure 4. Dataset distribution.

A total of 2540 datasets containing the following seven entities were created for the entity extraction experiment. These include department (dept), date (date), work (work), place (place), company (company), name (name), number (num), time (time) and no entity. The division of these is illustrated in Table 3.

Table 3. Entity extraction data

Entity	Count
Date	120
Time	32
Name	73
Company	12
Work	720
Department	223
Number	37
No_Entity	1824

6.2. Data Preprocessing

When there is not enough data, oversampling can be used. By making rare samples better, it seeks to balance the dataset. Various new rare samples are produced using techniques such as repetition, bootstrapping, or SMOTE (synthetic minority over-sampling technique). We applied the SMOTE technique using the Python standard library “imblearn”. SMOTE initially selects a point at random from the rare class and calculates its k-nearest neighbors. Between the selected point and its neighbors, the synthesized points are placed. We need to prepare the data after it has been collected in order to move further. Data preparation refers to the process of organizing and preparing data for use in the machine learning process. In this step, unwanted phrases, words, whitespaces and other unnecessary data are removed

from the gathered data [23]. It is the technique for refining and turning raw data into the form used for ML/DL models. It is the procedure of cleaning the data, determining the variables to utilize and changing the data into a suitable format for analysis in the processing phase. It is among the most crucial steps in the entire procedure. We preprocess the data to convert it into a form so that it can be used for the deep learning models.

6.2.1. Tokenization

Tokenization is the most critical phase in data preprocessing, where all the words from the text are gathered and the number of times each word appears is counted. Five tokenizers were used for this purpose. These tokenizers include Takseem, Tf-idf, WhiteSpace, Arcab and ConveRT. With the help of these tokenizers, we determine how many times a single word appears in the text. In a dataset, we count words and create tokens for distinct words that occur. Each word is given a unique number when tokens are created. The token includes one-of-a-kind feature values that are used to create feature vectors. A tokenization library called tkseem has many methods for tokenizing and preprocessing Arabic text. This library is widely used for the tokenization of Arabic text. Arcab is a data-driven, unsupervised method for tokenizing subwords in Arabic phrases. There is no pre-tokenization phase where terms are extracted depending on the whitespaces; instead, it considers the training corpus as a sequence of raw Unicode characters. This allows the technique to be used for any string of characters, making it language-neutral. Transformer-backed dual-encoder networks serve as the foundation for ConveRT (conversational representations from transformers), a compact model of neural response selection for dialogue that has proven to perform at the cutting edge on various response selection tasks as well as in transfer learning for intent classification tasks. The performance of various tokenizers is listed in Table 4.

Table 4. Tokenization performance of tokenizers.

Task	WhiteSpace		Arcab		ConveRT	
	Train	Test	Train	Test	Train	Test
Entity Extraction	0.86	0.75	0.972	0.954	0.842	0.763
Intent Classification	1.00	0.962	1.00	0.978	1.00	0.973

6.2.2. Stop Word Removal

Stop words are unnecessary words, i.e., *لذا, لا, اجل, مع*, that exist in phrases but have no significance or clues about their contents. Examples of such words are so, for and with. Articles, conjunctions, pronouns (i.e., *هو, هي, هم*), prepositions (i.e., *الى, في, حول*, *من*), demonstratives and interrogatives (i.e., *اين, متى, لمن*) are further instances of these unimportant words. Once the dataset has been transformed into unique tokens, the next step is to delete every unnecessary word with no significance, e.g., whitespaces, commas, full stops, colons, semicolons and punctuation marks. Stop word elimination is the name given to the method of eliminating unnecessary words. Table 5 presents some stop words used in the study.

Table 5. Few stop words used in the study.

	Arabic Word	English Meaning
Prepositions	في	in
	على	on
	إلى	to
Pronouns	أنا	I
	نحن	we
Adverbs	تحت	below
	فوق	above
	الآن	now
Question	منذ	since
	لماذا	what
	متى	when
Articles	إذا	if
	ثم	then
	عدا	except

6.2.3. Featurization

Methods for feature selection are used in machine learning techniques. An attribute of a system or process that has been built from the initial input variables is represented by a feature. Due to the enormous magnitude of the data, it is challenging to train effective classifiers before deleting the undesirable characteristics. A real-world classification challenge may be better understood by reducing the number of characteristics that are redundant or unnecessary. Feature selection aids in data comprehension, lessens the impact of dimensionality, reduces processing needs and enhances prediction performance. To increase prediction accuracy, feature selection selects a subset of features. Featurization is the conversion of words into meaningful numbers (or vectors) that the deep learning algorithm can use for its training. For this purpose, we use the features of the count vector featurizer, lexical syntactic featurizer and Tf-idf featurizer. For feature extraction, the TF-IDF technique uses word statistical data. This solely evaluates the expressions for terms that are the same throughout the texts, such as ASCLL, despite considering that synonyms may replace them.

The performance of used featurizers is compared for analysis purposes. The count vectorizer outperforms all three featurizers for both tasks, i.e., intent classification and entity extraction. The performance of these featurizers is listed in Table 6.

Table 6. Performance comparison of featurizers.

Task	Count Vector		Lexical Syntactic		Tf-Idf Featurizer	
	Train	Test	Train	Test	Train	Test
Entity Extraction	0.983	0.963	0.972	0.944	0.952	0.923
Intent Classification	1.00	0.976	1.00	0.972	1.00	0.973

The first phase of the proposed model was trained on 100 epochs to evaluate intent classification and entity extraction performance. The accuracy of the trained transformer-based model over 100 epochs is illustrated in Figure 5 below.

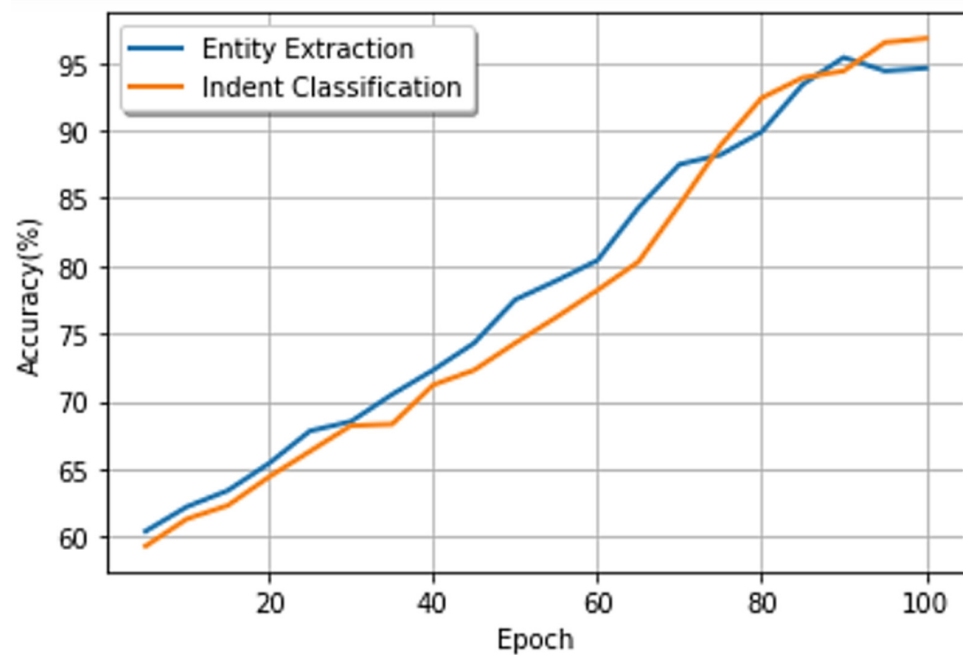


Figure 5. Accuracy of the proposed model.

As shown in the figure above, the proposed model achieves 97% accuracy for intent classification and 95% for entity extraction tasks. The performance of the model was also analyzed with the F1 score metric. The F1 score of the proposed model on both tasks.

As shown in Figure 6, the proposed model gains an F1 score of 95% for entity extraction and 96% for intent classification. This research compared the suggested intent classifier to existing intent classifiers in terms of intent classification and entity extraction. A performance assessment experiment was conducted to compute the DIET-base classifiers, keyword classifiers and fallback classifiers. According to the performance evaluation results, the proposed model's F1 score was 17.8%, 0.2% and 0.3% in intent classification from a given number of sentences. For entity extraction, the value of the F1 score was 3.1, 2.3 and 2.9% higher than the traditional DIET-base, keyword and fallback classifier. The results comparison of the proposed model for intent classification with other models is illustrated in Figure 7.

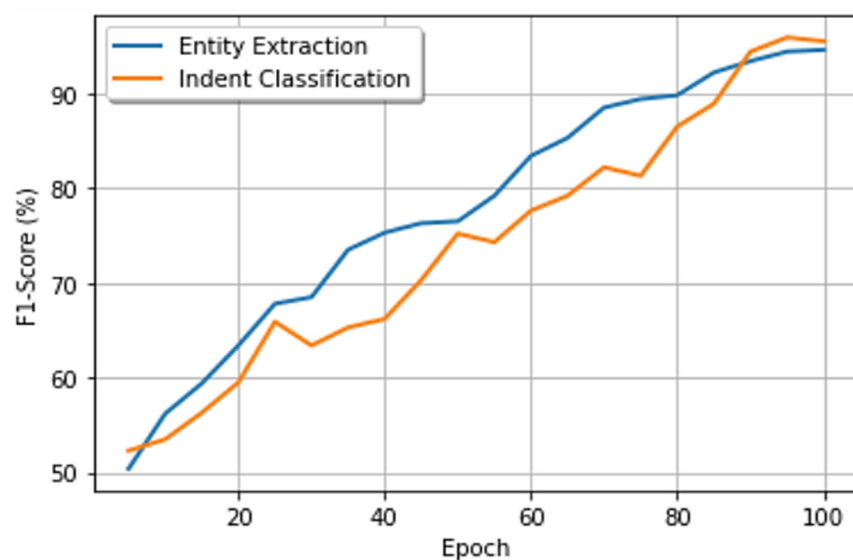


Figure 6. F1 score of the proposed model.

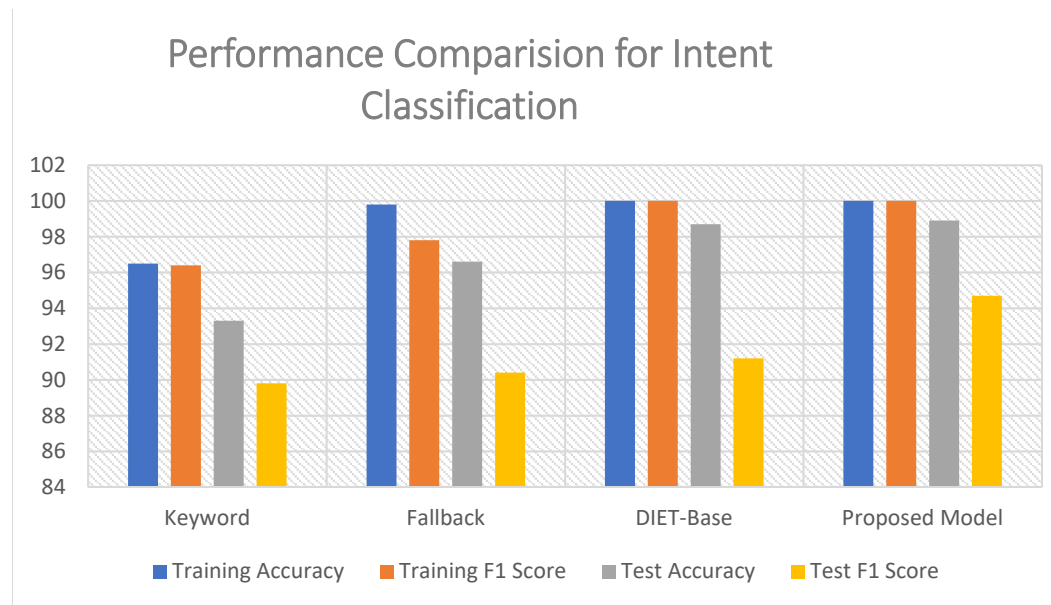


Figure 7. Performance comparison for intent classification.

Confusion matrices are a widely used metric when attempting to solve classification issues. Both binary and multiclass classification tasks can be performed with it. The confusion matrix for the entity extraction task is illustrated in Table 7.

Table 7. Confusion matrix for entity extraction.

	Date	Time	Name	Company	Work	Number	Department	No_Entity
Date	70	0	0	0	0	0	2	0
Time	0	15	0	0	0	1	0	0
Name	0	0	31	3	0	0	2	0
Company	2	0	0	8	0	1	0	0
Work	0	1	0	1	524	0	13	21
Number	0	0	2	1	0	19	0	0
Department	0	0	0	0	1	0	104	1
No_Entity	0	2	23	3	0	0	3	782

The confusion matrix for the indent classification tasks is also calculated in the evaluation phase of the study. Table 8 presents the indent classification confusion matrix.

Table 8. Confusion matrix for indent classification.

	Food	Department	Sentiment	Div. of Works	Calulator	Closing	Greetings
Food	18	1	0	0	0	0	3
Department	0	34	0	3	0	0	0
Sentiment	0	0	81	5	0	0	3
Div. of Works	1	0	0	127	3	0	1
Calulator	0	0	0	0	7	0	0
Closing	0	0	1	0	0	9	0
Greetings	0	0	0	0	1	0	12

Furthermore, the suggested model’s entity extractor was evaluated with existing entity extraction models. A performance evaluation occurs for a conditional random field (CRF) and DIET-Base. The F1 scores of the suggested model were 1.4, 0.9 and 0.3% higher in intent

classification and 4.2, 3.1 and 2.6% higher in the entity extraction, as per the performance evaluation findings. The performance of the entity extractor of the proposed model is illustrated in Figure 8. The main reason for the high accuracy and performance of the proposed model is the use of Dual Intent and Entity Transformer (DIET) for the Arabic language. The previous studies used different techniques for developing Arabic NLU, but Rasa provides that DIET is a multi-task transformer framework that simultaneously performs intent categorization and entity recognition. It allows us to plug-and-play a variety of pre-trained embeddings. Moreover, in terms of accuracy and stability, it is comparable to large-scale, pre-trained language models.

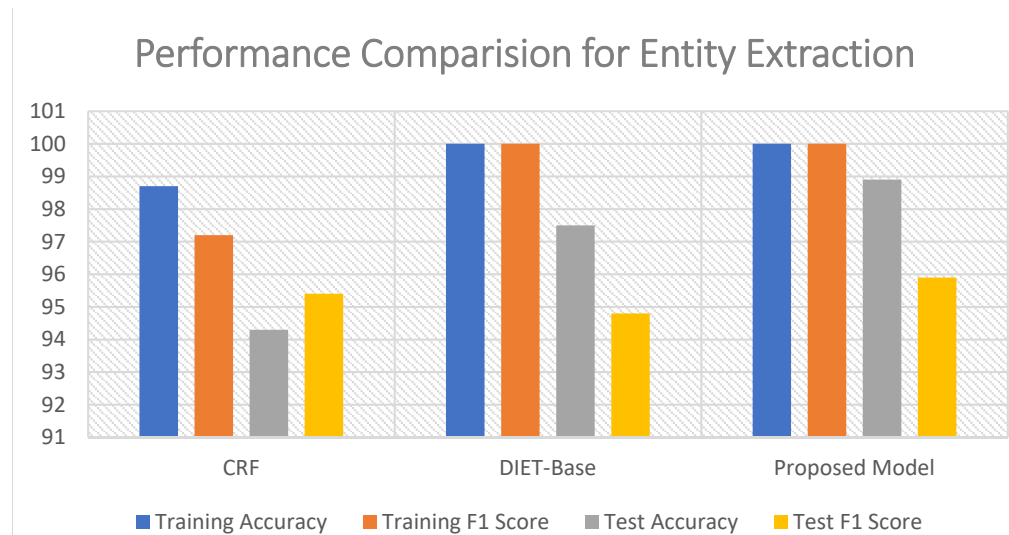


Figure 8. Performance of proposed model for entity extraction.

The comparison is performed to check the effectiveness of the proposed system. Some recent models for Arabic NLU are studied for this purpose, and a comparison has been made on the basis of accuracy and F1 score. Table 9 presents the comparison of the proposed system with previous studies. I have compared the performance of the proposed RASA-based Arabic chatbot with some existing studies, i.e., Faud et al. [17] and Bashir et al. [14] According to the statistics obtained, the proposed scheme outperforms both existing approaches in terms of accuracy and F1 score.

Table 9. Comparison with previous studies.

Model	Intent Classification		Entity Extraction	
	Train	Test	Train	Test
	Accuracy	F1	Accuracy	F1
Fuad et al. [17]	0.90	0.80	0.92	0.86
Bashir et al. [14]	0.92	0.85	0.93	0.91
DIET-Base	0.86	0.75	0.972	0.954
ArRASA	1.00	0.962	1.00	0.978

We have performed a comparison of the proposed dataset with another dataset proposed by Fuad [17], as shown in Figure 9. The proposed model achieved an efficient accuracy level on the dataset provided by Fuad [17]. Figure 9 presents the comparison of accuracy and F1 score on both datasets.

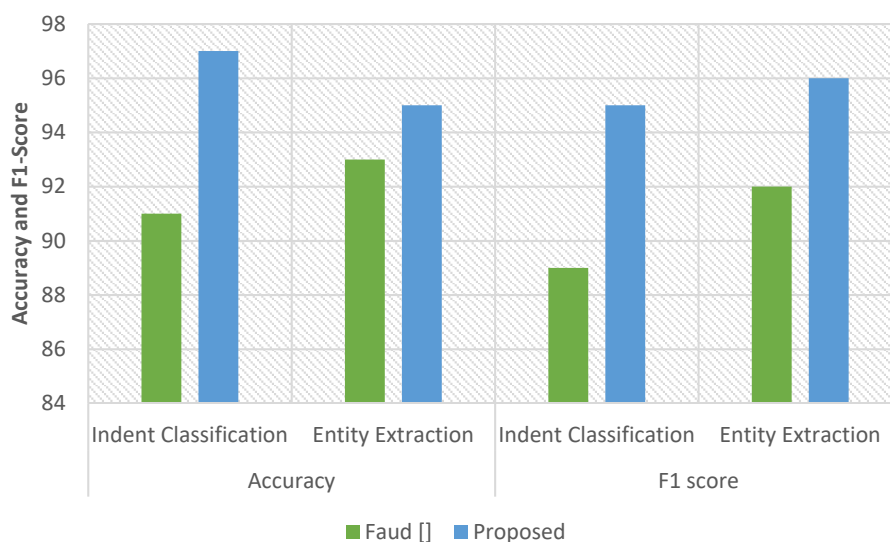


Figure 9. Comparison of accuracy and F1 score with an existing approach.

7. Conclusions

This article presents ArRASA, a deep learning-based channel optimization framework for an open-source Arabic chatbot platform to interact with users remotely. ArRASA is a closed domain chatbot that can be used in nearly any Arabic industry. There are four phases, i.e., tokenization, feature extraction, suitable intent classification, and proper entity extraction, used in the proposed model and these are also tuned to interpret the Arabic language. The Taqseem tokenizer, specific for the Arabic language and a few others, was used for tokenization, while the count vector featurizer was utilized for featurization. This study created the DIET-based Arabic model (ArRASA) for the intent classification and proper entity extraction phases by tweaking and optimizing the DIET-Base model's parameters. The accuracy of the proposed system for intent classification is 96%, while it achieved 94% accuracy for the entity extraction task. The proposed system gained a 95% F1 score for intent classification and 94% for entity extraction.

Funding: This research received no external funding.

Conflicts of Interest: The authors declare that they have no conflict of interest to report regarding the present study.

References

- Rickli, J.M. *The Economic, Security and Military Implications of Artificial Intelligence for the Arab Gulf Countries*; Emirates Diplomatic Academy: Abu Dhabi, United Arab Emirates, 2018; pp. 1–13.
- Hahm, Y.; Kim, J.; An, S.; Lee, M.; Choi, K.S. Chatbot Who Wants to Learn the Knowledge: KB-Agent. In Proceedings of the 17th International Semantic Web Conference (ISWC 2018), NLIWod4, Monterey, CA, USA, 8–12 October 2018. 4p.
- Aleem, S.; Huda, N.u.; Amin, R.; Khalid, S.; Alshamrani, S.S.; Alshehri, A. Machine Learning Algorithms for Depression: Diagnosis, Insights, and Research Directions. *Electronics* **2022**, *11*, 1111. [\[CrossRef\]](#)
- Sarddar, D.; Dey, R.K.; Bose, R.; Roy, S. Topic modeling as a tool to gauge political sentiments from twitter feeds. *Int. J. Nat. Comput. Res.* **2020**, *9*, 14–35. [\[CrossRef\]](#)
- Wolf, T.; Debut, L.; Sanh, V.; Chaumond, J.; Delangue, C.; Moi, A.; Cistac, P.; Rault, T.; Louf, R.; Funtowicz, M.; et al. Transformers: State-of-the-art natural language processing. In Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations, Online, 3 June 2020; pp. 38–45.
- Iqbal, A.; Amin, R.; Iqbal, J.; Alroobaea, R.; Binmahfoudh, A.; Hussain, M. Sentiment Analysis of Consumer Reviews Using Deep Learning. *Sustainability* **2022**, *14*, 10844. [\[CrossRef\]](#)
- Hijjawi, M.; Bandar, Z.; Crockett, K.; Mclean, D. ArabChat: An arabic conversational agent. In Proceedings of the 2014 6th International Conference on Computer Science and Information Technology (CSIT), Piscataway, NJ, USA, 26–27 March 2014; IEEE: Piscataway, NJ, USA; pp. 227–237.
- Ali, D.A.; Habash, N. Botta: An arabic dialect chatbot. In Proceedings of the COLING 2016, the 26th International Conference on Computational Linguistics: System Demonstrations, Osaka, Japan, 11–16 December 2016; pp. 208–212.

9. Fadhil, A. OlloBot-towards a text-based arabic health conversational agent: Evaluation and results. In Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP 2019), Varna, Bulgaria, 2–4 September 2019; pp. 295–303.
10. Shawar, A.; Atwell, E.S. An Arabic chatbot giving answers from the Qur'an. In Proceedings of the TALN04: XI Conference sur le Traitement Automatique des Langues Naturelles, Fez, Morocco, 19–22 April 2004; ATALA: Monza, Italy, 2004; Volume 2, pp. 197–202.
11. Bashir, A.M.; Hassan, A.; Rosman, B.; Duma, D.; Ahmed, M. Implementation of a neural natural language understanding component for Arabic dialogue systems. *Procedia Comput. Sci.* **2018**, *142*, 222–229. [\[CrossRef\]](#)
12. AlHumoud, S.; al Wazrah, A.; Aldamegh, W. Arabic chatbots: A survey. *Int. J. Adv. Comput. Sci. Appl.* **2018**, *9*, 535–541. [\[CrossRef\]](#)
13. Al-Ghadhban, D.; Al-Twairesh, N. Nabiha: An Arabic dialect chatbot. *Int. J. Adv. Comput. Sci. Appl.* **2020**, *11*, 1–8. [\[CrossRef\]](#)
14. Fuad, A.; Al-Yahya, M. AraConv: Developing an Arabic Task-Oriented Dialogue System Using Multi-Lingual Transformer Model mT5. *Appl. Sci.* **2022**, *12*, 1881. [\[CrossRef\]](#)
15. Wilie, B.; Vincentio, K.; Winata, G.I.; Cahyawijaya, S.; Li, X.; Lim, Z.Y.; Soleman, S.; Mahendra, R.; Fung, P.; Bahar, S. Indonlu: Benchmark and resources for evaluating indonesian natural language understanding. *arXiv* **2020**, arXiv:2009.05387.
16. Bunk, T.; Varshneya, D.; Vlasov, V.; Nichol, A. Diet: Lightweight language understanding for dialogue systems. *arXiv* **2020**, arXiv:2004.09936.
17. Wang, A.; Singh, A.; Michael, J.; Hill, F.; Levy, O.; Bowman, S. GLUE: A multi-task benchmark and analysis platform for natural language understanding. *arXiv* **2020**, arXiv:1804.07461.
18. Habash, N.Y. Introduction to Arabic Natural Language Processing. In *Synthesis Lectures on Human Language Technologies*; Springer: Cham, Switzerland, 2010; Volume 3, pp. 1–187.
19. Al-Ayyoub, M.; Khamaiseh, A.A.; Jararweh, Y.; Al-Kabi, M.N. A comprehensive survey of arabic sentiment analysis. *Inf. Process. Manag.* **2019**, *56*, 320–342. [\[CrossRef\]](#)
20. Habash, N.; Eryani, F.; Khalifa, S.; Rambow, O.; Abdulrahim, D.; Erdmann, A.; Faraj, R.; Zaghouani, W.; Bouamor, H.; Zalmout, N.; et al. Unified guidelines and resources for Arabic dialect orthography. In Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018), Miyazaki, Japan, 7–12 May 2018.
21. Wang, A.; Pruksachatkun, Y.; Nangia, N.; Singh, A.; Michael, J.; Hill, F.; Levy, O.; Bowman, S. SuperGLUE: A multi-task benchmark and analysis platform for natural language understanding. *Adv. Neural Inf. Process. Syst.* **2019**, *32*, 3261–3275.
22. Pennington, J.; Socher, R.; Manning, C.D. Glove: Global vectors for word representation. In Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), Doha, Qatar, 25–29 October 2014; pp. 1532–1543.
23. Zelaya, C.V.G. Towards explaining the effects of data preprocessing on machine learning. In Proceedings of the 2019 IEEE 35th international conference on data engineering (ICDE), Macao, China, 8–12 April 2019; IEEE: Piscataway, NJ, USA, 2019; pp. 2086–2090.