applied sciences

MDPI

Article

Conversational AI over Military Scenarios Using Intent Detection and Response Generation

Hsiu-Min Chuang * and Ding-Wei Cheng

Department of Computer Science and Information Engineering, Chung Cheng Institute of Technology, National Defense University, Taoyuan City 335, Taiwan; dinwei0108@gmail.com
* Correspondence: showmin1205@gmail.com

Abstract: With the rise of artificial intelligence, conversational agents (CA) have found use in various applications in the commerce and service industries. In recent years, many conversational datasets have becomes publicly available, most relating to open-domain social conversations. However, it is difficult to obtain domain-specific or language-specific conversational datasets. This work focused on developing conversational systems based on the Chinese corpus over military scenarios. The soldier will need information regarding their surroundings and orders to carry out their mission in an unfamiliar environment. Additionally, using a conversational military agent will help soldiers obtain immediate and relevant responses while reducing labor and cost requirements when performing repetitive tasks. This paper proposes a system architecture for conversational military agents based on natural language understanding (NLU) and natural language generation (NLG). The NLU phase comprises two tasks: intent detection and slot filling. Detecting intent and filling slots involves predicting the user's intent and extracting related entities. The goal of the NLG phase, in contrast, is to provide answers or ask questions to clarify the user's needs. In this study, the military training task was when soldiers sought information via a conversational agent during the mission. In summary, we provide a practical approach to enabling conversational agents over military scenarios. Additionally, the proposed conversational system can be trained by other datasets for future application domains.

Keywords: conversational AI; intent detection; slot filling; retrieval-based question answering; query generation

Citation: Chuang, H.-M.; Cheng, D.-W. Conversational AI over Military Scenarios Using Intent Detection and Response Generation. *Appl. Sci.* **2022**, *12*, 2494. https://doi.org/10.3390/app12052494

Academic Editors: Arturo Montejo-Ráez and Salud María Jiménez-Zafra

Received: 31 January 2022
Accepted: 24 February 2022
Published: 27 February 2022

1. Introduction

Breakthroughs in artificial intelligence and natural language processing (NLP) have made it possible for conversational agents to provide appropriate replies in various domains, helping to reduce labor costs [1–4]. Task-oriented conversational agents, in particular, are of great interest to many researchers. According to a 2018 VentureBeat article[5] over 300,000 chatbots are operating on Facebook. In addition, a 2021 Userlike survey showed that 68% of consumers liked that chatbots can provide fast answers or responses [6].As a result, text-based conversational systems or chatbots have become increasingly common in everyday life. Task-oriented conversational AI use NLP and NLU to perform intent detection and response generation based on domain-specific information, and are mainly used in entertainment [7], finance [8], medicine [9,10], law [11,12], education [13], etc.

Combat training emphasizes timeliness, coupled with the ever-changing battlefield. As a result, effectively predicting the combat information required by soldiers has become one of the key technologies on the frontline battlefield. Operators need to send and receive the type of data they want to enhance their situational awareness [14]. However, it is costly and practically difficult to provide a human assistant to every operator [15,16]. At the 2017 National Training and Simulation Association (NTSA) conference held in Florida, AI experts and military officials discussed valuable applications of AI in military training [15]. Considering that future battlefields and combat scenarios will be increasingly complex and difficult to navigate, the ability to use AI to design extremely realistic, intelligent entities

that can be immersed in simulations will be an invaluable weapon for the Navy and Marine Corps. To reduce the risk to personnel in practice, since 2021, the U.S. The Navy has been planning to develop virtual assistants to assist in submarine hunting (https://voicebot.ai/2021/02/10/the-us-navy-wants-a-virtual-assistant-to-help-hunt-submarines/ (accessed on 31 January 2022)). For example, sonar operators on ships must manage the complexity of sonar technology and set settings based on weather, location, etc. Hence, the Navy wants to utilize artificial intelligence to enhance the operating system, improving sonar detection and reducing training costs. These information-processing AI systems can be tailored to specific industries.

In a recent study, the three main types of Human–Machine Interfaces (HMI) were text-based systems, voice-based systems, and interactive interface systems [17–19]. For example, Dr. Felix Gervits from the Army Research Laboratory worked with the U.S. Army Combat Capabilities Development Command and the University of Southern California's Institute for creative technologies to develop autonomous systems (https://eandt.theiet.org/content/articles/2021/04/military-bots-could-become-teammates-with-real-time-conversational-ai/ (accessed on 31 January 2022) [20] to derive intent from a soldier's speech via a statistical language classifier. By combining NLU with dialogue management and having the classifier learn the patterns between verbal commands, responses, and actions, they created a system that could respond appropriately to new commands and knew when to request extra information. In addition, Robb et al. [21] proposed a conversational multimodal interface by combining visual indicators with a conversational system, providing a natural way for users to gain information on vehicle status/faults and mission progress and to set reminders. The system can be used for operations in remote and hazardous environments.

During military training missions, soldiers must follow guidelines or personnel instructions. However, the overloading information may not be understood and completed effectively. In addition, traditional retrieval systems may delay user action. Hence, we constructed a conversational agent over a set of military scenarios that enables users to operate on constantly evolving battlefields and to obtain the information they need through conversation. Based on the survey of conversational systems in [22], we aim to design task-oriented dialogue systems for application in military scenarios, focusing on question answering with martial training intent and relevant entity information. Therefore, conversational goals for social and entertainment purposes, such as greetings, entertainment, and advertising, are outside the focus of our system. Therefore, the ability of a military conversational AI to correctly detect its user's intent and identify entities in a sentence will determine whether it can successfully reply to users.

One challenge for intent detection is that the questions of military users can be terse and ambiguous [23]. Furthermore, the answer often depends on the context of the conversations. To narrow down the range of possible intent types, we first defined the range of applications for the types of intent it was meant to detect, and then classified and annotated the conversational data. In general, although the users' queries are short, most will mention the important entities. The role of slot filling is to identify and annotate the entities in the sentence, e.g., persons, events, times, locations, and weapons. As for the response provided by the system to the user, the challenges are to choose the most appropriate answer and to generate questions that require explicit information when missing the primary entity from the user's query.

To enable the task-oriented conversational system, the architecture comprises four modules based on a pipeline strategy: (1) slot filling, (2) intent detection, (3) retrieval-based answering, and (4) query generation. For the (1) and (2) modules, we trained by our prepared dataset by named-entity recognition (NER) models and a support vector machine (SVM) classifier [24], respectively. The (3) module is used by the BM25 algorithm [25] to retrieve the Military List and then rank the most appropriate solution using the Learning to Rank (LTR) model [26]. For the final module, we adopted the template-based question generation for the database of the Army Joint Task List (AJTL) according to the user's intent.

The performance of our military conversational system was experimentally evaluated in terms of the performance of its intent detection, slot filling, LTR modeling, and question

generation modules. The system performed well based on both quantitative and qualitative metrics. Therefore, this study established a new approach for the development of military conversational systems. The proposed architecture could also be trained using other domain-specific datasets to expand its scope of applicability. The contributions of this study may be summarized as follows.

- A task-oriented conversational system was designed based on the practical needs of military tasks. As its module functions and datasets are mutually independent, it is possible to use this architecture to accelerate the training of domain-specific conversational systems in other domains, as one simply has to replace the dataset.
- This study defined the four core tasks of a conversational system and used machine-learning technologies to enable the realization. They included using NER models for slot filling, a classifier for intent detection, answering by the retrieval-based and learning-to-rank (LTR) model, and generating new queries by the template-based method.
- The experimental results highlight the performance of the intention detection, slot filling, sentence ranking, and the overall user satisfaction for the conversational system. The result can serve as a promising direction for future studies.

The remainder of this paper is organized as follows. Section 2 describes related work and technologies. Section 3 introduces the proposed architecture and functions. Section 4 presents the experimental results evaluating. Section 5 summarizes the tasks and discusses future directions.

2. Related Work

This section reviews related work of conversational AI and military conversational systems. As well as tasks and models for NLU, emphasizing intent detection and slot filling, we conclude with a review of response generation methods.

2.1. Conversational AI

The rapid development of AI technologies has increased academic interest in human–computer interfaces, with applications ranging from domain-specific settings to open-domain conversations. In the business world, personalized AI assistants such as Siri (Apple), Assistant (Google), Cortana (Microsoft), Messenger (Facebook), and Alexa (Amazon) have become increasingly common. Owing to the extensive labeling of conversational databases and the application of deep-learning and NLP techniques, conversational systems have made considerable progress in understanding the semantics of natural language and contextual reasoning. We divide conversational AI into several categories according to their purposes as the following.

Conversational AI, which are also known as chatbots, may be divided into task-oriented or non-task-oriented dialogue systems [27]. Task-oriented dialogue systems are meant to help users perform a specific task, e.g., intelligent food ordering, legal queries [28], and smart customer service. In addition, they usually have domain-specific conversational dialogues and knowledge bases.

Non-task-oriented dialogue systems (e.g., chatbots for the elderly or children) are meant to provide reasonable responses to users and thus provide entertainment and have open-domain dialogues that are not specifically constrained in scope. The first chatbot in the world was the Eliza chatbot in 1996 [29], which used simple dialogue to mimic a psychologist conversing with a patient. In 2017, Fitzpatrick et al., developed Woebot [30], a cognitive-behavioral therapeutic (CBT) chatbot, which was able to converse with patients and provide CBT assistance. Zhang et al. [31] proposed a unified conversational search/recommendation framework called "System Ask—User Respond," which was trained using a large collection of user reviews in e-commerce. They then evaluated the performance of this framework using metrics such as the Normalized Discounted Cumulative Gain (NDCG).

Task-oriented dialogue systems are typically designed with a "pipeline" consisting of four modules: a NLU module, dialogue state tracker, dialogue policy learning module, and NLG module [27]. Recently, some workers have proposed end-to-end frameworks

to expand the expressiveness of the state space and support dialogue beyond domain-specific corpora [32]. For example, Zhao and Eskenazi proposed an end-to-end framework that used reinforcement learning and policy learning to optimize the dialogue system for dialogue state tracking. They tested this framework using a 20-question game, where the conversational system asked the user a series of yes-or-no questions to find the answer to a specific question.

There are three main approaches for conversational response generation [4]: rule-based approaches, retrieval-based approaches, and generative approaches. A rule-based system often requires a large amount of manual design and labeling work and therefore has the highest costs. Retrieval-based conversational AI uses keyword matching with machine learning or deep learning to determine an optimal predefined response [33]. Finally, generative conversational AI can be trained in multiple stages using supervised/unsupervised learning, reinforcement learning, or adversarial learning. Recently, Zhang et al. [34] presented a graph-based self-adaptive conversational AI; it used a knowledge graph whose nodes and links represented key entities and semantic relationships, respectively, as a dynamic knowledge base. It allowed the system to gain knowledge through conversations with end-users. Based on the definition above and the category of conversational AI, the system presented in this work may be characterized as a domain-specific (military) task-oriented dialogue system. To ensure that the dialogues produced by the conversational AI are compatible with the expectations of military tasks, we used a pipeline design and retrieval-based response generation method.

Military-domain task-oriented conversational systems can be divided into three types according to their mode of interaction: voice-type, text-type, and interactive interface-type systems. A few successful examples are described below. The Siri chatbot developed by Apple in 2011 began as a part of the "Cognitive Assistant that Learns and Organizes" project funded by the Defense Advanced Research Projects Agency (DARPA); by using perceptual and experiential learning, Siri reduced the information overload faced by battlefield commanders. It has since become a virtual voice assistant of national importance. With the development of expert assistant systems and the application of text-based conversational AI in military applications, IBM's Watson system came to be used to provide occupational information to US military members and help them transition from active duty into civilian life. In 1998, DARPA presented a dialogue system based on conversational multimodal interfaces, which allowed its users to perform operational tasks more efficiently [28].

Due to the lack of relevant literature on military dialogue systems, this study summarizes relevant research on military dialogue systems in different periods in the past, as shown in Table 1. For example, Roque et al. [35] in 2006 proposed a spoken dialogue system that can engage in Call For Fire Radio dialogues (Radiobot-CFF) to help train soldiers in proper procedures for requesting artillery fire missions. They provided three modes: fully-automated, semi-automated, and passive mode, as the radio operator in a simulated Fire Direction Center (FDC) takes calls from a forward observer for artillery fire in training exercises.

The Hassan system [36] proposed by Gandhe in 2009 is a set of tactical question-answering dialogue systems, including a management interface for creating dialogue content and a dialogue manager, which can be used to build multiple virtual characters for tactical questions. The experiment consisted of 19 dialogues and 296 utterances. Furthermore, the experts expanded the range of possible responses provided by the virtual character by annotating other candidate utterances according to need. However, the system lacks the capabilities of question generation.

MIRIAM is a conversational multimodal interface developed for command-and-control systems proposed by Robb et al. [21]. The system improved situational awareness by providing information in multiple modalities (including audio, images, and text) to clarify the textual ambiguities that often arise in natural language and improve understanding. Therefore, multimodal conversational AI could become an important trend in the design of military conversational systems. The recent example of a successful military conversational AI would be the human-robot navigation system of Gervits et al. [20].

Table 1. Summary of military conversational systems.

	Interface	NLU	Dialogue State Tracking	NLG
Radiobot-CFF [35]	Spoken	✔	✔	✕
Hassan [36]	Text	✔	✔	✕
MIRIAM [21]	Multimodal	✔	✔	✔
Gervits [20]	Spoken	✔	✔	✔
Our study	Text	✔	✔	✔

2.2. Intent Detection and Slot Filling

In a conversation, the queries provided by the user are usually relatively short. Therefore, the dialogue system must first determine the aim or intent of the question. However, the information within the query may be incomplete or stated implicitly. Suppose the dialogue system does not possess the knowledge or context necessary to answer the question. In that case, it may require several rounds of dialogue to redress these issues and confirm the user's intentions.

In 2018, Zhang et al. [31] proposed a "circular" conversational architecture, where the dialogue system clarified a user's intentions through several rounds of dialogue. Intent detection pertains to the determination of intent by analyzing the structure of the question [37], e.g., by "5W2H" analysis (why, what, where, when, who, how, and how much) [38], to improve the accuracy of the retrieved answer. Furthermore, as the information contained by the query is critical for determining the correct answer, NER techniques can be used to identify key 12 persons, events, times, places, and objects and narrow down the query's scope.

Intent detection may be performed using statistical or rule-based methods. For example, Setyawan et al. [39] used Naïve Bayes and logistic regression machine-learning methods to perform intent detection, with the term frequency-inverse document frequency being the classification feature. In 2012, Wang et al. [40] converted short snippets into vectors to perform intent (sentiment) classification and compared the SVM, Naïve Bayes, and continuous bag-of-words methods. The typical representative one of the supervised learning methods is the SVM. Over the past 10 years, many scholars have uses SVM as a comparative method for intent classification or sentiment analysis, and the summary references are as shown in Table 2. In addition, deep-learning models have also become commonplace in intent detection. Nigam et al. [41] used a recurrent neural network to perform multi-staged named-entity learning and then used the named entities as classification features.

Table 2. Summary of SVM Models used in NLU Applications.

Reference	CA [1]	Domain	Language	Data Size	Models [2]	Optimal
Chen, 2012 [42]	✔	Community Question Answering	English	1.5 K	SVM, C4.5, RF, NB, KNN	SVM
Bhargava, 2013 [43]	✔	Audiovisual Media	English	27.5 K	SVM, HMM, CRF	SVM
Sarikaya, 2016 [44]	✔	Personal Assistant	English	400 K	SVM	SVM
Gaikwad, 2016 [45]	✕	Sentiment Analysis	English	8 K	SVM, NB, KNN	SVM
Sullivan, 2018 [46]	✕	Booking flights/ Accommodation	English	8 K	SVM, CNN	≈

Table 2. *Cont.*

Reference	CA [1]	Domain	Language	Data Size	Models [2]	Optimal
Troussas, 2020 [47]	✗	Learning Styles	English	<1 K	SVM, NB, KNN, ensemble	ensemble
Rustamov, 2021 [48]	✔	Banking Services	Azerbaijani	161 K	LR, SVM, NN, DIET	DIET
Our study	✔	Military Training	Chinese	10 K	SVM	SVM

[1] The study is used to develop a conversational system (CA). [2] Model abbreviation: Random Forest (RF), Logistic Regression (LR), Naive Bayes (NB), *k* Neural Network (KNN), Convolutional Neural Network (VNN), Hidden Markov Model (HMM), Dual Intent and Entity Transformer (DIET).

Slot filling is another critical task in dialogue systems, as it provides semantic information and helps the conversation system determine which bits of information in a sentence should be searched for. In previous studies, slot filling has often been performed using generative models (such as the hidden Markov model) or discriminative models (such as the conditional random field (CRF) model) to estimate the conditional probabilities of slot labels in a sequence. However, with the emergence of deep-learning models, bidirectional recurrent neural network (RNN) models and long short-term memory (LSTM) models trained with contextually annotated sentences are now used for slot filling. In recent studies, CRF models have been combined with RNNs to train slot-filling models for unseen semantic labels and multi-domain tasks, enhancing their performance. Yang et al. [49] presented an intent-aware neural ranking model, which used the "Transformer" architecture to perform language representation learning and to analyze user intent patterns in information-seeking conversations.

Intent detection is usually viewed as a supervised classification problem, that is, mapping a sentence to some class within a finite set of classes. Slot filling, in contrast, is viewed as a token sequence labeling problem. Traditionally, intent detection and slot filling are performed separately or in a pipeline. Recently, some studies have investigated the use of joint models for simultaneously performing intent detection and slot filling [50], and have proven that these tasks are closely related to each other. Compared to the pipeline approach, joint models are less susceptible to error propagation between the intent detection and slot-filling models. In addition, they can be trained and tuned as a single model. However, joint models cannot be easily generalized to unseen data due to variations in natural language expressions for the same intent.

Furthermore, domains and label sets can change over time in real-world applications. There is a lack of publicly available task-oriented datasets among the conversational corpora used for training. Moreover, the available datasets are limited to a few specific domains. These corpora may be divided into two types: the first type comprises user–system conversations, such as the Air Travel Information System [51] and WOZ2.0 [52]; the second type includes simulated human–system dialogues subsequently and manually converted into natural language, such as the machine-to-machine dataset [53].

2.3. Response Generation

NLG is the phase in NLP where task-oriented dialogs are completed to meet user needs. Response generation by the system can be performed using retrieval-based or generation-based models. Retrieval-based answers generate dialogue by retrieving the best responses from the corpus through a ranking function and often have highly fluent and informative answers. However, it tends to be repetitive and cannot handle semantics outside its corpus. On the other hand, generation-based conversational systems use logic to infer spoken responses and are therefore not bound by response templates. Traditionally, NLG always involves sentence planning, where the input semantic symbol is mapped to an intermediary representation of the utterance (e.g., a tree-like or template structure). The intermediary process is then converted into the final response through surface realization.

In 2002, Sneiders [54] presented a template-based automated answering model. The model considered four entities (human names, locations, organizations, and times) and

used NER to extract information (keywords). The keywords were then matched to question templates to create answers. However, answer generation alone may not produce an adequately correlated response with the original question. To address this problem, reference [55] used the co-occurrence of technical terms in a corpus to infer whether they were correlated. In 2003, Fiszman et al. [56] presented the SemRep system, which used manually-listed template rules for verbs to identify potential semantic relationships in a sentence and used identification criteria to select relevant technical terms and phrases.

Bhoir and Potey [57] proposed a heuristic retrieval-based conversational system for retrieving the most relevant answers from a predefined corpus based on a user's input and used complete sentences as candidate answers. It also considered the type of answer to select an appropriate response to the user. Choosing a proper reply is the most critical problem in question-answering systems, and the reaction must also be clear and concise. Therefore, selecting only the most critical information when formulating a reply is necessary. In another study [58], a similarity approach was used to predict whether a message was a reply to another message. This approach was validated by comparisons with a trained bidirectional encoder representation from the transformers model, with conversations generated by a bidirectional LSTM and RNN. The authors found that this approach helps to improve the understanding of the context.

Recently, there has been increasing interest in finding distributed vector representations (embeddings) for words, that is, by encoding the meanings of text into vectors. The text may range from words, phrases, and documents to human-to-human conversations. In 2017, Bartl and Spanakis [59] used a locality-sensitive hashing forest, an approximate nearest neighbor model, to generate context embeddings and find similar conversations in a corpus. The candidate answers were then ranked.

In 2018, Juraska et al. [60] presented a sequence-to-sequence natural language generator with an attentional mechanism and was able to produce accurate responses for a variety of conversational domains. In 2019, Song et al. [61] from Microsoft proposed the method based on the concept of pre-training and sequential neural networks. This method masked a segment of a random length and used an encoder–decoder attention mechanism to generate responses.

In 2020, Wang et al. [62] from Tencent proposed a deep-learning-based TransDG model for generating Chinese conversations. This model performed question–answer and semantics-named entity matching within its knowledge base and then selected the optimal strategy for response generation.

A recent study showed that template-based conversational AI faces two key challenges: (1) constructing a system grammar that balances the expressiveness necessary to conduct a task with the ability to infer parses from natural language correctly; and (2) dealing with parse ambiguities. Seungwhan et al. collected a new open-ended dialogue-KG parallel corpus called OpenDialKG [63], where each utterance from 15,000 human-to-human role-playing dialogues was manually annotated with a ground-truth reference to corresponding entities and paths from a large-scale knowledge graph with more than one million facts. They also proposed a DialKG Walker model for learning the symbolic transitions of dialogue contexts via structured traversals over the KG, and used an attention-based graph path decoder to predict the entities. Bockhorst et al. [64] addressed parse ambiguities by using a context-free grammar called episode grammar; the system constructed a semantic parse progressively for a multi-turn conversation, where the system's queries were derived from the parse uncertainty.

3. Methodology

This work proposes a conversational system architecture that uses machine-learning techniques through a Chinese corpus for military training missions. It includes the mission list of the joint training management system, the military dictionary, and the Army Joint Task List (AJTL). As the implementation of this system is independent of its domain and language, it can be used to enable conversational systems in other fields or languages by changing its corpus.

3.1. System Architecture

This study aims to develop a conversational AI for quickly answering soldiers' questions in the military training mission and supporting multiple conversation rounds. The architecture of our conversational system is shown in Figure 1. The user's query is first parsed by NLP, followed by a slot-filling module, which identifies important entities, and then the intent type is detected by the intent detection module. The system performs retrieval-based answer generation through the extracted entities and intents. The retrieval-based question-answering system ranked and selected the optimal responses. If the user confirms the answer is clear, take action; otherwise, the system will generate a new query to verify the user's intent. Slot filling and intent detection are the NLU stages for understanding, and retrieval-based answering and query generation are the NLG stages for responding. In the NLU stage, we use the CRF and SVM models to train slot-filling and intent-detection modules, respectively, which are practical and easy to implement due to the limited training information set. In addition, we make some summaries of the use of these models in related research. In the NLG stage, we use a learned ranking method to obtain retrieval-based answers. There are two basic types of generating sentences: extracted and abstract. This method is determined according to the greater probability of finding and querying within the existing corpus. The advantages of this method are that the grammar is relatively smooth and easy to understand and does not require a large number of training datasets—the main reason for responding to build mods. We use a template-based strategy in the query generation module. This template-based query generation method may propose a new query for the missing intent or entity and user to be confirmed with the user, that is, for the intent and entity to continue the dialogue, with the intention to avoid generating a new query and diverging context.

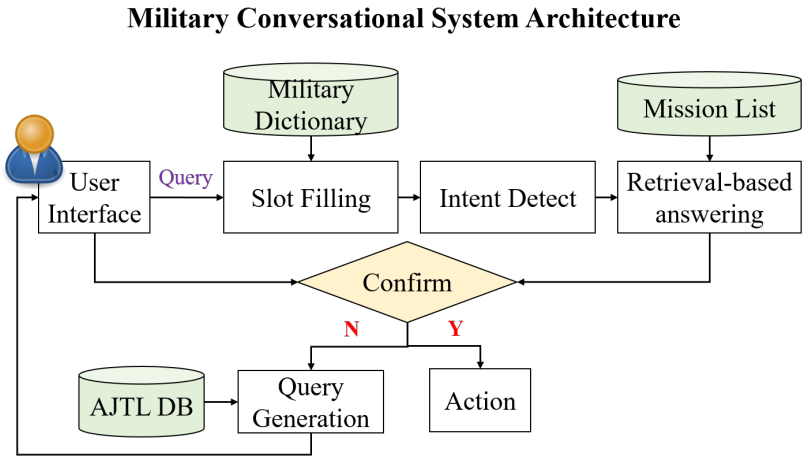

Figure 1. System architecture our conversation system.

3.2. Slot Filling and Intent Detection

Figure 2 illustrates the flow of a user's query. In the query "何時將完成後備部隊動員任務？(When will the reserve force complete the mobilization task?)," entities such as "何時 (when)" as B-time, "後備部隊 (reserve force)" as B-unit and I-unit, "動員任務 (mobilization task)" as B-event and I-event were annotated. Slot labels are labeled using the BIO format: B indicates the beginning of a slot span, I the middle of a span, and O indicates that the label does not belong to a slot. In addition, the query intent of this sentence is "when".

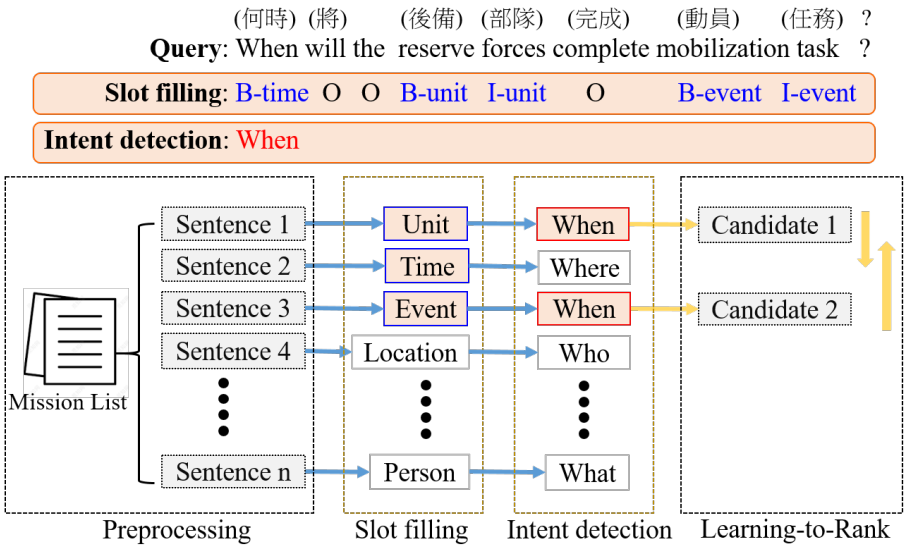

Figure 2. Flowchart of slot filling and intent detection.

To ensure the conversational system is able to deliver the correct intents and related entities, the system analyzes a soldier's utterances first to identify the entities mentioned and match them with intents stored in the mission list database, and then orders the appropriate response sentences. Because traditional retrieval systems rely on retrieving full-text search results for user queries, retrieval models are based on the similarity between the query and the text (e.g., vector space models). Therefore, the user may miss the correct answer because the intent of sentences with high similarity may not match the intent of the user's question. In other words, we prioritize intent and entity accuracy before evaluating similarity.

3.2.1. Slot Filling

The slot-filling task, in contrast, was defined as a sequence labeling problem, that is, an NER problem. We trained used five kinds of entities by the CRF model. Considering the amount of data and the implementation of integrating multiple modules, we choose the CRF-based method as the baseline for slot filling.

During the preprocessing phase, the CkipTagger (https://github.com/ckiplab/ckiptagger (accessed on 31 January 2022)) tool was used for Chinese word segmentation and part-of-speech (POS) tagging for the user's query. The CRF toolkit was performed to train five NER models. Five types of slots related to military missions were defined: the military unit and location, the name of military personnel (including job titles and ranks), the name of military event tasks, the name of the weapon, and time. As shown in Table 3, There are six types of features: POS tagging, vocabulary, specific terms, verbs, quantifiers, and punctuation. We match them to entities for vocabulary and specific terms, and the vocabulary source is the Military Dictionary. For verbs, quantifiers, and punctuation, we use them to determine the boundaries before and after entities. In summary, we trained five CRF models to predict five entities (i.e., location/unit, person, event, weapon, and time) for slot filling. The CRF model was then used to estimate the conditional probability of the sequence, as shown in Equation (1).

$$P(y|x) = \frac{1}{Z(x)} \prod_{t=1}^{T} exp \sum_{k=1}^{K} \theta_k f_k(y_t, y_{t-1}, x_t) \quad (1)$$

If it is assumed that x and y are random variables, given an observed sequence X, $P(y|x)$ is the conditional probability distribution of the hidden sequence Y, whose probability estimate in the state t depends on that in the state $t-1$. $Z(x)$ is a normalization function for normalizing the value of $P(y|x)$.

Table 3. Features of CRF models for slot filling.

	POS	Vocabulary	Specific Term	Verb	Quantifier	Punctuation
Location/unit	Y	Y	Address suffix	Y	N	Y
Person	Y	Y	Surname list	Y	Y	N
Event	Y	Y	Event suffix	Y	N	Y
Weapon	Y	Y	Digit/alphabet	Y	Y	Y
Time	Y	Y	Time suffix	Y	Y	Y

Note that the "Y" indicates that the feature is used, and the "N" indicates that the feature is not utilized.

3.2.2. Intent Detection

In this study, we adopted the SVM multi-class classification method [65] for intent detection. The algorithm constructs k SVM models, where k is the number of classes. All the examples in the mth class with positive labels are used to train the mth SVM and all the other examples with negative labels. Formally, given training data $(x_1, y_1), \ldots, (x_l, y_l)$, where $x_i \in R^n$, $i = 1, \ldots, l$, and $y \in 1, \ldots, k$ is the class of x_i, the mth SVM solves the following problem:

$$\begin{aligned} \min_{w^m, b^m, \xi^m} & \frac{1}{2}(w^m)^T w^m + C \sum_{i=1}^{l} \xi_i^m \\ & (w^m)^T \phi(x_i) + b_m \geq 1 - \xi_i^m \quad if\, y_i = m \\ & (w^m)^T \phi(x_i) + b_m \leq -1 + \xi_i^m \quad if\, y_i \neq m \\ & \xi_i^m \geq 0, i = 1, \ldots, l \end{aligned} \tag{2}$$

where the training data x_i are mapping to a higher dimension space by the function ϕ and C is the penalty parameter.

Here, intent detection is regarded as a multiclass classification problem, with the intent in a query consisting of four parts: who, where, when, and what. As we did not consider the possibility of multiple intents in one question, the hard classification performed the intent prediction with the highest probability of the query. SVM is one representative machine classifier for supervised learning methods. Many scholars use SVM as a comparison or combination method in recent studies, as discussed in Refs. [40,46,47,66,67]. However, despite these years of research, intent detection is still challenging. The classifier is used for intent detection by a SVM classifier, as SVMs are accurate for this task.

After extracting entities, there were 12 types of features for training user's intent, as shown in Table 4. Features 1–5 were the five types of entities extracted by NER models. Regarding Features 6–8, we used the Military Dictionary to match whether the query sentence contains military words and quantifiers. Features 9–12 were Common interrogative terms in Chinese. Formally, the multi-class classifier is used to predict an unseen sample x with labels 1 to k, which assigns the highest confidence score, as shown in Equation (4). We used a simple one-hot encoding to facilitate the classifier's training for nominal features, with matched features being one and unmatched features being 0.

$$y = argmax_{k \in \{1 \ldots K\}} f_k(x) \tag{3}$$

For a conversation system, there is difficulty remembering conversational intent from one sentence to the next. In other words, the procedure typically treats each query from the user as a new dialogue state. To alleviate this problem, our system stores intents and entities extracted from one round of conversations and intents and entities extracted from previous rounds of conversations. When the system confirms whether the response meets the user's information needs, if the user gives a negative reply, the intent and entity of the query are stored to avoid forgetting.

Table 4. Features of intent detection.

No.	Feature	Description
1	Location	Military location or organization
2	Person	Name, rank and title entities
3	Time	Time descriptor
4	Event	Military event
5	Weapon	Weapon or transport entity
6	Document	Military document name
7	Session	Phase name of combat missions
8	Unit	Commonly used quantifiers in military affairs
9	Who	An interrogative term about a person
10	Where	An interrogative term about a location
11	When	An interrogative term about time
12	What	An interrogative term about all other matters

3.3. Response Generation

After extracting the user intent and filling the slots, the second step is to answer the user using a retrieval-based response module. Suppose no intent or relevant entity was identified in the previous step. In that case, the system uses a template-based query generation module to ask the user for additional information to retrieve an appropriate answer. In practice, we use the Elasticsearch full-text search tool to build a retrieval model in the Chinese military domain. The model is built using the Okapi BM25 algorithm. The model determines the most appropriate response based on the correlation between query Q and database document D, as shown in Equation (4).

$$Score(D,Q) = \sum_{i=1}^{n} IDF(q_i) \frac{f(q_i,D)(K_1+1)}{f(q_i,D) + K_1(1-b+b\frac{|D|}{avgdl})} \quad (4)$$

where *IDF* denotes the inverse document frequency in this equation, and *avgdl* is the average length of all texts.

In the retrieval system, the entities extracted from the user query are used as the keywords to perform full-text retrieval. The top 10 most relevant results were then selected using the Learning-To-Rank Answering (LTRA) apporach, as shown in Algorithm 1. The input to the retrieval model is the searched sentences $S = \{s_1, \ldots, s_n\}$, query intent i_Q, and entities $E = \{e_1, \ldots, e_m\}$. First, intent prediction is performed for each sentence. If a sentence j intends i_j to be the same as i_Q, the candidate sentence j is kept; otherwise, it is discarded. The LTR model ranks the candidate sentences and considers the top k candidate sentences as the most suitable replies.

The LTR model was implemented using the LambdaMART algorithm [68]. LambdaMART is a listwise LTR that combines the LambdaRank and Multiple Additive Regression Tree (MART) algorithms, transforming the search candidate ranking problem into a regression tree problem. To train candidate sentences for ranking, we prepared 10 features, as shown in Table 5. Features 1–5 represent the five NER models for extracting entities from candidate sentences. Features 6 and 7 are used to determine if the candidate sentence matches the military document names and units in the Military Dictionary; the results are displayed as boolean values. Feature 8 represents the longest common subsequence (LCS) between the user query and candidate sentences, as shown in Equation (5). Feature 9 denotes the similarity between the user query and candidate sentence, as shown in Equation (6). Feature 10 denotes the term frequency-inverse document frequency (TFIDF), which is used to calcuate the word importance for user's query and sytem answers based on the corpus, as shown in Equation (7).

Algorithm 1 Learning-To-Rank Answering.

Input: search sentences S, query intent i_Q, entities E
Output: ranked sentences S'
Initialize candidate set C is empty
for sentence $j = 1, \ldots, n$ from S **do**
    if sentence intent $i_j = i_Q$ **then**
        candidate set $C \cup$ sentence j
    end if
end for
while candidate set $C \neq \{\}$ **do**
    Rank sentence $cs \in C$ by the LTR model
end while
Return top-k sentences $S' = \{cs_1, \ldots, cs_k\}$

Table 5. Features of the learning to rank (LTR) model.

No.	Feature	Description	Value
1	Location	Military location or organization entity	yes/no
2	Person	Personnel name, rank and title entity	yes/no
3	Time	Time descriptor	yes/no
4	Event	Military event	yes/no
5	Weapon	Weapon or transport entity	yes/no
6	Document	Military document name	yes/no
7	Unit	Commonly used quantifiers in military affairs	yes/no
8	LCS	Common strings between user's query and system response	[0, 1]
9	Cosine	Similarity between user's query and system response	[0, 1]
10	TFIDF	Word importance for user's query and system response	[0, 1]

$$LCS(i,j) = \begin{cases} 0 \text{ if } i = 0 \text{ or } j = 0 \\ \text{LCS } (i-1, j-1) + 1 \text{ if } i, j > 0 \text{ and } \alpha_i = \beta_j \\ \max\{\text{LCS}(i-1,j), \text{LCS}(i, j-1)\} \text{ if } i, j > 0 \text{ and } \alpha_i \neq \beta_j \end{cases} \quad (5)$$

The above calculates the LCS for input sequences $A = \alpha_1, \alpha_2, \ldots, \alpha_m$ and $B = \beta_1, \beta_2, \ldots, \beta_n$, where $1 \leq i \leq m$ and $1 \leq j \leq n$.

$$Cosine(A,B) = \frac{(A \cdot B)}{(|A| \times |B|)} \quad (6)$$

$$TFIDF = \frac{tf_{i,j}}{\sum_k n_{k,j}} \times log \frac{|D|}{\{j : t_i \in d_j\}} \quad (7)$$

In the above, X and Y are discrete random variables, that is, the correlation between the entity sets of the user query and the candidate sentence.

The question generation module generated new queries based on the template-based representation of the intent–entity relations of queries. The structural composition was viewed as a set of intent–entity relationships within the state space, as shown in Figure 3. The intents of a question included who, where, when, and what, whereas named entities were in six types: person, unit, event, time, weapon, and document (Doc). Candidate sentences could be formed based on the occurrence probabilities of the intent–entity relationships. For example, the intent "who" and entity "event" was related by "be responsible for" in "the commander (who) is responsible for this combat readiness mission (event)." In general, as a response should also have a person as the intent (who) and a combat readiness mission (event) as the entity, the candidate sentences should be ranked according to the presence of the "event" entity and "who" intent in these sentences.

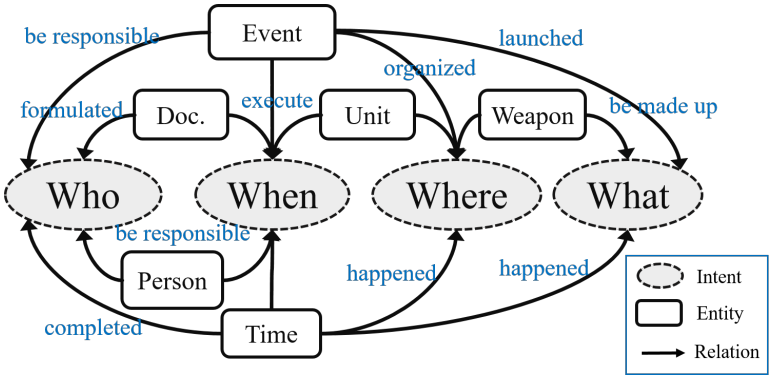

Figure 3. Intent–entity relations.

In the question generation module, the predicted intent types and the queried entities are combined according to the intent–entity relations shown in Figure 3. Table 6 shows some examples of such parsed sentences. Brackets correspond to slot-fill annotation entities, and bold corresponds to intent types. Templates can be applied in various combinations. To select the best new question from candidate sentences generated by multiple templates, we define Equation (8) to score each query–new-question pair. The higher the score, the stronger the semantic relevance between the newly generated question and the original query.

$$QQ_i = \alpha(LCS_i + Cosine_i) + (1 - \alpha)TFIDF_i \quad (8)$$

In this equation, i refers to the candidate sentence generated by the i-th rule for the same intent; α is a weighting parameter that ranges from 0 to 1; LCS is the longest common subsequence between the user query and generated query; Cosine is the cosine similarity between the user query and generated query; and TFIDF is the importance of words around user's query, which is calculated the product by term frequency and inverse document frequency.

Table 6. Exemplary template-based question generation via intents and corresponding entities.

No.	Intent	Entity	Templates
1	Who	Doc.	Who + is in charge of + [Doc.]?
	誰負責人員調度? Who is in charge of personnel management?		
2	Who	Person	Who + [Person] + reports to?
	作戰科科長需向誰提報旅部作戰計畫? Who does the Chief of operations need to present the brigade combat plan.		
3	Who	Event	Who + is in charge of + [Event]
	野戰照明工作是誰負責? Who is responsible for field lighting?		
4	Who	Unit	Who + are in the +[Location]?
	救災編組有哪些人? Who are in the disaster relief team?		
5	When	Event	When + the + [Event] + will happen
	什麼時候執行綜合演練? When will the joint drill happen?		
6	When	Unit	When + [Location] + will finish + [Event]
	想知道後備部隊在何時要完成動員整備任務? When the reserve forces will complete their mobilization and preparation?		

Table 6. *Cont.*

No.	Intent	Entity	Templates
7	Where	Unit	Where is the + [Location] + located?
	軍團指揮部位於哪裡? Where is the command of army located?		
8	Where	Weapon	Where did the + [Location] + discover the + [Weapon]?
	機旅在哪裡發現敵軍2部戰車? Where did the brigade discover two enemy tanks?		
9	What	Event	What + is the basis for performing this + [Event]?
	執行地面任務是依據什麼準則? What is the criteria for performing this ground mission?		
10	What	Weapon	What + is the range of this + [Weapon]?
	戰車砲攻擊距離有多遠? What is the range of this tank's main gun?		

4. Experiments

This section evaluates the performance of the proposed system, including describing the datasets and metrics used, the experimental evaluation of the intent detection and slot-filling modules, and the response generator's ranking performance evaluation. Finally, the overall performance of the dialogue system is discussed.

4.1. Datasets and Measures

A total of 1307 human-labeled sentences are included in the experimental dataset used for intent classification (who, where, when, and other). Table 7 shows the four types of intent quantitative sentences. As shown in Table 8, the experimental datasets were used to train the five NER models, including people, weapons, places, events, and time. The intent detection and slot-filling tasks in the NLU stage are evaluated using F1-score (Equation (9)) and accuracy (Equation (10)). In the following equations, true positive (TP) and true negative (TN) are the numbers of accurately predicted positives and negatives, respectively. Conversely, false positive (FP) and false negative (FN) are the numbers of wrongly predicted positives and negatives, respectively. Thus, Precision = $|TP|/|TP + FP|$ and Recall = $|TP|/|TP|+|FN|$.

Table 7. The number of datasets used for intent classification.

	Who	Where	When	What
# of sentence	335	320	329	323

Table 8. The number of datasets used for NER training.

	Person	Weapon	Location	Event	Time
# of sentence	1005	1007	1006	1005	1000
# of entity	3693	2465	1843	2399	1820

$$\text{F1-score} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{9}$$

$$\text{Accuracy} = \frac{|TP + TN|}{|TP + FP + TN + FN|} \tag{10}$$

The LTR model was evaluated using the normalized discounted cumulative gain (NDCG), which gives the normalized relevance score of the files retrieved by the search

engine at each rank position. Files closer to the top are given a higher weight (and therefore have a greater degree of influence on NDCG), as shown in Equation (11).

$$NDCG_p = \frac{DCG_p}{IDCG_p} \tag{11}$$

where *IDCG* is ideal discounted cumulative gain, and rel_p represents the list of relevant documents in the corpus up to position p.

$$DCG_p = \sum_{i=1}^{p} \frac{rel_i}{log_2(i+1)} \tag{12}$$

DCG is based on the principle that highly relevant documents appearing lower in a search result list will be penalized by having their relevance grade reduced logarithmically proportional to their position in a search result list. Finally, the query generation module is evaluated by quantifying user satisfaction.

4.2. Performance of the Intent Detection and Slot-Filling Modules

SVM models are used to perform multi-class classification. The dataset is randomly divided into training and test sets in three different proportions. According to the results shown in Figure 4, the 9:1 ratio outperforms the 7:3 ratio in terms of F1 score and accuracy, and achieves 90% accuracy. The performance for predicting the four intents is then performed based on the model trained with a data ratio of 9:1, as shown in Figure 5. The multi-class classifier had the highest F1 score for "where" intent (92%), followed by "when" (91.4%), "who" (91.1%), and finally "what" (88.8%). The average F1-score of the classifier was 90.1%. Due to the limited amount of training data (1307 sentences in four categories), the F1 performance of the trained intent detection model is 88 91%. However, from the perspective of the learning curve (dataset splitting rate), the performance improves as the amount of training data increases. Further comparing the performance of each category, we can see that the "What" category has the most errors, followed by "Who" and "When", and "Where" has the best performance. We analyzed the possible reasons and found two: one is language. For example, in Chinese grammar, the query for "What" is more complex than the other three categories, which may include "Why", "How", "which", etc. Another possible reason is that when the user's query has multiple intents, the classifier will only predict the class with the highest probability, so the number of false negatives for the "what" intent increases.

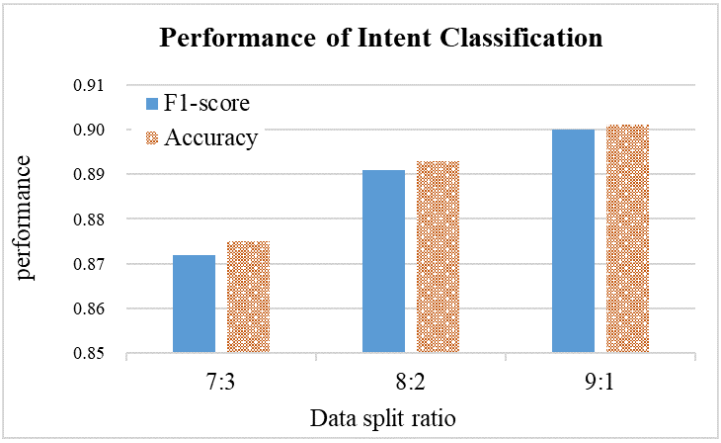

Figure 4. Performance of intent classification.

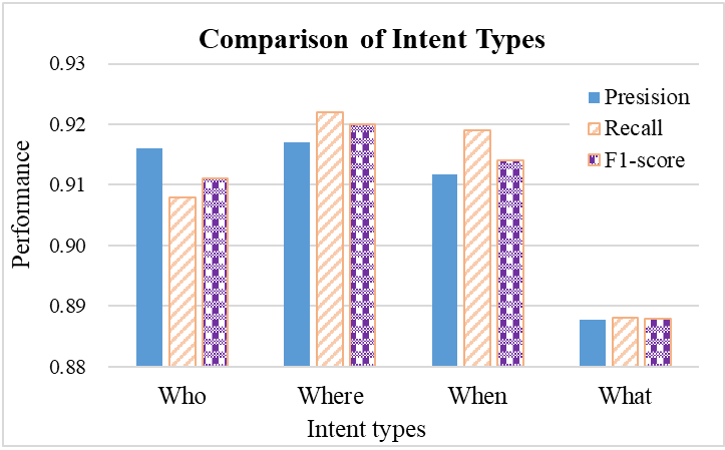

Figure 5. Performance of four types of intent classification.

The performance of the NER model is evaluated by performing five-fold cross-validation on the dataset. Figure 6 shows the performance of NER in recognizing military names, weapons, military locations, military events, and time entities. In terms of F1-score, the five models have the highest accuracy (0.943) for person names, followed by time. Conversely, it performed slightly worse at identifying military event, at 0.848.

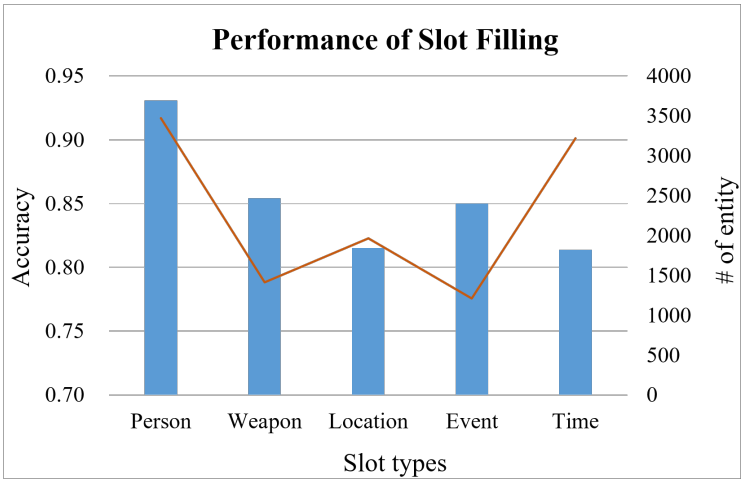

Figure 6. Performance of five types of named-entity recognition (NER).

4.3. Performance in Response Generation

Next, it evaluates the performance of retrieval-based answers. First, a question-and-answer dataset of 180 military joint tactical action lists is manually collected as standard answers. Three sentences are then randomly selected for each query, resulting in four candidates. Responses in the training dataset are ranked from 1 to 4, where 4 represents the highest score for the questions–answer pairing. We use the NDCG indicator as to the evaluation indicator for the LTR model. The training and testing datasets are divided into three different scales, and the experimental results for each scale are shown in Figure 7. Since the 9:1 ratio gives the highest NDCG score, this model is used as the LTR model for our conversational system.

Finally, eight military scenarios are randomly selected from 40 question-and-answer test data for performance evaluation. The Table 9 shows questions and answers for eight military scenarios. The intent is first identified for each question, and then responses are generated based on the extracted intents and entities and intent–entity relationships. Each intent may generate multiple sets of candidate sentences, and the first set of responses is the candidate sentence with the highest relevance to the query. The results of the dialogue were qualitatively assessed by 47 military-related personnel. The proportion of responses

that meet user needs is shown in the Figure 8. If the user is not satisfied with the first response, the dialogue system generates a new query and then provides the second answer. Experimental results show that, on average, 66% or more of users are satisfied after the second round of conversations. Therefore, new queries generated by the system in the second round of dialogue (when the first answer is not satisfactory) significantly improved user satisfaction.

Table 9. Eight scenarios of question answering.

Session	Examples of Question Answering
S1	Q:通信單位用什麼裝備開設? What equipment will the communications group use to set up?
	A: 通信單位用多波道系統開設。 The communications group uses a multi-channel system to set up.
S2	Q:誰與火力協調機構協調建立安全空域? Who coordinates with the Fire Coordination Agency to establish safe airspace?
	A: 由聯絡官/防砲官協調火力協調機構。 Army liaison and anti-gunnery officers coordinate with the Fire Coordination Agency.
S3	Q: 誰決定勤務支援的優先順序? Who decides the prioritization of service support?
	A:指揮官決定勤務支援的優先順序。 The commander decides the prioritization of service support.
S4	Q:聯合空降將部隊投入哪裡? Where does Joint Airborne put troops?
	A:將作戰部隊及後勤支援部署到目標區域。 Deploy combat forces and their logistical support into target areas.
S5	Q:災害應變中心開設分幾級？ What are the levels of disaster response centers?
	A:共分兩級。情況輕者以二級開設，情況嚴重者一級開設。 There are two levels. Milder cases are opened at Level 2, and severe cases are offered at Level 1.
S6	Q:從駐地演習到動員需要多長時間? How long will it take from station maneuvers to mobilization?
	A:駐地機動至動員需2小時。 Station maneuvers to mobilization takes 2 h.
S7	Q: 火力支援組位於哪裡? Where is the fire support team?
	A:火力支援組編成位於作戰地區，建立聯合軍民防空情報傳遞系統。 The fire support team has been deployed in the combat zone to establish a joint military-civilian air-defense information transmission system.
S8	Q: 戰備整備是什麼? What is combat readiness?
	A:戰備整備依據地面防衛作戰構想，預想作戰推移與需求。 Combat readiness is based on the concept of ground defense operations, misunderstood combat progress, and needs.

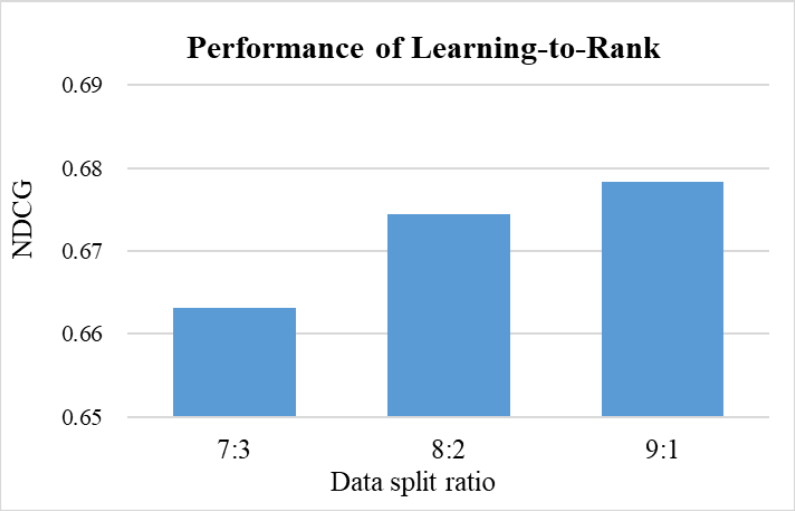

Figure 7. Performance of the learning-to-rank model.

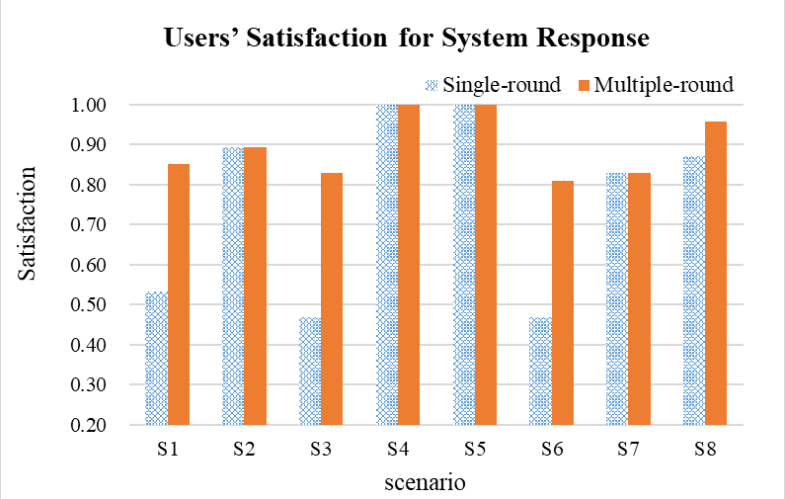

Figure 8. Users' satisfaction for system response.

4.4. Discussion

Here, we conduct an error analysis of the module performance and discuss the challenges of research and limitation. For the intent classification module, we found that the "what" category has the highest error rate (12%) because there are some queries with different interpretations, such as "why", "how much", "how to do", etc. For example, "how many days can each cavalry company unit fight independently?" "How many exchange centers and medium-sized communication centers can a communication force establish?" These false-negative examples of the "what" leads to lower recognition performance than other classes. It will be possible to further segment the user's intent in the future to guarantee that every feature of that intent is clearly defined.

For the slot-filling module, the accuracy for five types of entities is identified by the military personal name as the highest (0.917), followed by time (0.901), location (0.823), weapon (0.788), and military event (0.776) as the lowest. There are two main reasons for the poor attribution of military events: (1) Event names are longer than other entities, making it more challenging to identify the limits of the entity. Still, the system identifies part of the names of military activities. (2) The event name contains time or location, which is misjudged as another entity.

For the retrieval-based answering module, we use the learning-to-rank method to achieve results. From the learning curve point of view, with the increase in data, the efficiency of the system response improves (NDCG = 0.678). Candidate sentences are

added only when the intent of the sentence is the same as the intent of the user's query. Then, we adopt these entities in the sentence as sorting features, and finally, sort them based on the LambdaMART algorithm. As we analyze why the correct sentence is not ranked first, we discover that pronouns may represent entities in sentences or omit them; thus, some candidate sentences do not identify entities related to the query, resulting in a low ranking score.

Based on the methods comparison in related literature, Sullivan [46] compared CNN and SVM, two ML algorithms with good performance records in the current NLP literature. However, the CNN model is not necessarily better than the SVM model based on a detailed statistical analysis of the experimental results. Under these experimental conditions, the SVM model using the radial basis function kernel produced statistically better results. However, SVM has its limitations. SVM is not suitable for large datasets because the complexity of algorithm training depends on the size of the dataset [69]; SVM is not ideal for training imbalanced datasets, which causes the hyperplane to be biased towards the minority class [70]. In terms of performance, choosing an "appropriate" kernel function is crucial. For example, using a linear kernel when the data are not linearly separable can lead to poor algorithm performance.

Two factors for the superior performance of the state-of-the-art are rich training datasets and high-speed hardware such as GPUs. We choose the CRF-based method, mainly considering the amount of data and training cost. Since obtaining a large amount of military training data in Chinese is a challenge, this study implements a dialogue AI system applied to military training scenarios using a limited military dialogue dataset. Using a CRF-based model is indeed a baseline approach. Nonetheless, this is an initial and fruitful result for the agency. For future work, we consider applying transfer learning (meta-learning) to extend multiple military domains with small datasets to improve the scalability of dialogue systems.

Another challenge in preparing military corpora is that for the Out-Of-Vocabulary (OOV) problem, the vocabulary of the user's question may not be included in the Mission list or Military Dictionary, so the retrieval system may not have a corresponding sentence for the entity. As a result, the question-generation module has to generate new queries to confirm the user's intent or generate further questions.

5. Conclusions

Conversational AI has found commercial applications in entertainment, food, and medicine. However, relatively little research has been conducted on AI applied to military dialogue. One of the challenges this study faces is that a large number of Chinese training datasets are not easy to obtain, and the existing research mainly uses English public social training datasets. Another challenge is to consider the practice of the whole system, which comprises several modules. In contrast, many studies have focused on improving several specific modules (NLU or NLG). The main contribution of this work is to combine multiple research topics into one framework, including intent detection, slot filling, and response generation. We applied various machine-learning techniques, including filling slots with NER models, intent detection with classifiers, answering with retrieval and learned-to-rank (LTR) models, and template-based methods to generate new queries. We design a task-oriented conversational system according to the actual needs of military missions. Since its module functions and datasets are independent of each other, this architecture can accelerate the training of problem-specific conversational systems in other service domains since only the datasets need to be replaced. Each method module can also be further considered to be replaced by methods with higher performance or efficiency in the future for comparison. From the evaluation results of the experiment, it is feasible to realize the application of dialogue AI in military scenarios based on intent detection and response generation technology. The experimental results show that the query satisfaction in eight scenarios is greater than 80% after two rounds of dialogue based on retrieval-based response generation. We integrated technologies such as natural language processing, information retrieval, and natural language generation, and used the limited military

corpus to achieve the expected preliminary results of the plan. Through dialogue AI, we can help military trainers conduct multi-round question-and-answer sessions.

In future work, this research could improve in two directions: (1) Considering the amount of data and the feasibility of integrating multiple system modules, we choose the CRF-based method and SVM as the baseline for NLU tasks. This study has used the limited military conversational dataset to implement a conversational AI system for military training scenarios. In spite of this, using the CRF and SVM models are indeed preliminary approaches to implementation. During future research, we would like to apply transfer learning to expand multiple military domains with small datasets and apply few-shot learning to enhance performance. (2) Applying deep-learning architectures to replace template-based question-generation methods improves their accuracy and language expressiveness. Although current template-based queries have no obvious semantic problems, the generated sentence patterns are limited. Functional expansion based on the above two directions enables the system to take the proposal as a whole. In addition, in this military training scenario, we plan to use distant supervision to automatically label our data to expand the number of datasets and improve model training accuracy. Finally, we plan to study the feasibility of integrating Semantic Web technologies. Knowledge representation and reasoning and the construction of sentence generation using knowledge graphs architectures based on these techniques are refined to improve the NLG process of conversational AI systems.

Author Contributions: H.-M.C. conducted conceptualization, methodology, investigation, writing-original draft preparation, review & editing, supervision. D.-W.C. conducted data curation, software. All authors have read and agreed to the published version of the manuscript.

Funding: This research is sponsored by the Ministry of Science and Technology, Taiwan, under grant MOST 108-2221-E-606-013-MY2.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The part of data that supports the findings of this study is available on request from the corresponding author. The data are not publicly available due to privacy and military restrictions.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Khari, J. Facebook Messenger Passes 300,000 bots. *VentureBeat*, 1 May 2018. Available online: https://venturebeat.com/2018/05/01/facebook-messenger-passes-300000-bots/ (accessed on 4 October 2021).
2. Leah. What Do Your Customers Actually Think About Chatbots? *Userlike*, 12 July 2021. Available online: https://userlike.com/en/blog/consumer-chatbot-perceptions (accessed on 4 October 2021).
3. Helena P. What Does the Future of Military Comms Look Like? STEM Awards 2020. Available online: https://www.telegraph.co.uk/education/stem-awards/defence-technology/military-communication-on-the-battlefield/ (accessed on 4 October 2021).
4. Shafquat, H.; Sianaki, O.A.; Ababneh, N. A Survey on Conversational Agents/Chatbots Classification and Design Techniques. In Proceedings of the Workshops of the 33rd International Conference on Advanced Information Networking and Applications (WAINA-2019), AINA Workshops, Matsue, Japan, 27–29 March 2019.
5. Singh, S.; Beniwal, H. A survey on near-human conversational agents. *J. King Saud Univ.-Comput. Inf. Sci.* 2021, *in press*. [CrossRef]
6. Goel, P.; Ganatra, A. A Survey on Chatbot: Futuristic Conversational Agent for User Interaction. In Proceedings of the 2021 3rd International Conference on Signal Processing and Communication (ICPSC), Coimbatore, India, 13–14 May 2021; pp. 736–740. [CrossRef]
7. Ramesh, K.; Ravishankaran, S.; Joshi, A.; Chandrasekaran, K. A Survey of Design Techniques for Conversational Agents. In Proceedings of the Second International Conference, ICICCT 2017, New Delhi, India, 13 May 2017.
8. Trieu, H.; Iida, H.; Bao, N.P.H.; Nguyen, L.M. Towards Developing Dialogue Systems with Entertaining Conversations. In Proceedings of the 9th International Conference on Agents and Artificial Intelligence (ICAART 2017), Porto, Portugal, 24–26 February 2017.
9. Altinok, D. An Ontology-Based Dialogue Management System for Banking and Finance Dialogue Systems. *arXiv* **2018**, arXiv:1804.04838.

10. Zeng, G.; Yang, W.; Ju, Z.; Yang, Y.; Wang, S.; Zhang, R.; Zhou, M.; Zeng, J.; Dong, X.; Zhang, R.; et al. MedDialog: Large-scale Medical Dialogue Datasets. In Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), Online, 16–20 November 2020; pp. 9241–9250. [CrossRef]
11. Liu, W.; Tang, J.; Qin, J.; Xu, L.; Li, Z.; Liang, X. MedDG: A Large-scale Medical Consultation Dataset for Building Medical Dialogue System. *arXiv* **2020**, arXiv:2010.07497.
12. Sharma, M.; Russell-Rose, T.; Barakat, L.; Matsuo, A. Building a Legal Dialogue System: Development Process, Challenges and Opportunities. *arXiv* **2021**, arXiv:2109.00381.
13. Wang, C.; Chen, D.; Hu, Y.; Ceng, Y.; Chen, J.; Li, H. Automatic Dialogue System of Marriage Law Based on the Parallel C4.5 Decision Tree. *IEEE Access* **2020**, *8*, 36061–36069. [CrossRef]
14. Huang, C. The Intelligent Agent NLP-Based Customer Service System. In Proceedings of the 2021 2nd International Conference on Artificial Intelligence in Electronics Engineering, Phuket, Thailand, 15–17 January 2021; Association for Computing Machinery: New York, NY, USA, 2021; pp. 41–50. [CrossRef]
15. Heller, C.H. The Future Navy—Near-Term Applications of Artificial Intelligence. *Nav. War Coll. Rev.* **2019**, *72*, 7.
16. Chui, M.; Manyika, J.; Miremadi, M. *Where Machines Could Replace Humans—And Where They Can't (Yet)*; McKinsey & Company: Chicago, USA, 2016.
17. Kim, S.; Salter, D.; DeLuccia, L.; Tamrakar, A. Study on Text-Based and Voice-Based Dialogue Interfaces for Human-Computer Interactions in a Blocks World. In Proceedings of the 8th International Conference on Human-Agent Interaction, HAI'20, Virtual Event, 10–13 November 2020; Association for Computing Machinery: New York, NY, USA, 2020; pp. 227–229. [CrossRef]
18. Anwer, S.; Waris, A.; Sultan, H.; Butt, S.I.; Zafar, M.H.; Sarwar, M.; Niazi, I.K.; Shafique, M.; Pujari, A.N. Eye and Voice-Controlled Human Machine Interface System for Wheelchairs Using Image Gradient Approach. *Sensors* **2020**, *20*, 5510. [CrossRef]
19. Merdivan, E.; Singh, D.; Hanke, S.; Holzinger, A. Dialogue Systems for Intelligent Human Computer Interactions. *Electron. Notes Theor. Comput. Sci.* **2019**, *343*, 57–71. [CrossRef]
20. Gervits, F.; Leuski, A.; Bonial, C.; Gordon, C.; Traum, D., A Classification-Based Approach to Automating Human-Robot Dialogue. In *Increasing Naturalness and Flexibility in Spoken Dialogue Interaction, Proceedings of the 10th International Workshop on Spoken Dialogue Systems, Siracusa, Italy, 24–26 April 2019*; Marchi, E., Siniscalchi, S.M., Cumani, S., Salerno, V.M., Li, H., Eds.; Springer: Singapore, 2021; pp. 115–127. [CrossRef]
21. Robb, D.A.; Chiyah Garcia, F.J.; Laskov, A.; Liu, X.; Patron, P.; Hastie, H. Keep Me in the Loop: Increasing Operator Situation Awareness through a Conversational Multimodal Interface. In *ICMI '18, Proceedings of the 20th ACM International Conference on Multimodal Interaction, Boulder, CO, USA, 16–20 October 2018*; Association for Computing Machinery: New York, NY, USA, 2018; pp. 384–392. [CrossRef]
22. Allouch, M.; Azaria, A.; Azoulay, R. Conversational Agents: Goals, Technologies, Vision and Challenges. *Sensors* **2021**, *21*, 8448. [CrossRef]
23. He, T.; Xu, X.; Wu, Y.; Wang, H.; Chen, J. Multitask Learning with Knowledge Base for Joint Intent Detection and Slot Filling. *Appl. Sci.* **2021**, *11*, 4887. [CrossRef]
24. Cortes, C.; Vapnik, V. Support-Vector Networks. *Mach. Learn.* **1995**, *20*, 273–297. [CrossRef]
25. Manning, C.D.; Raghavan, P.; Schütze, H. *Introduction to Information Retrieval*; Cambridge University Press: Cambridge, MA, USA, 2008.
26. Liu, T.Y. Learning to Rank for Information Retrieval. *Found. Trends® Inf. Retr.* **2009**. *3*, 225–331. [CrossRef]
27. Hongshen Chen.; Liu, X.; Yin, D.; Tang, J. A Survey on Dialogue Systems: Recent Advances and New Frontiers. *arXiv* **2017**, arXiv:1711.01731.
28. Adebayo, K.J.; Caro, L.D.; Robaldo, L.; Boella, G. Legalbot: A Deep Learning-Based Conversational Agent in the Legal Domain. In Proceedings of the 22nd International Conference on Applications of Natural Language to Information Systems, NLDB 2017, Liège, Belgium, 21–23 June 2017.
29. Weizenbaum, J. ELIZA—A computer program for the study of natural language communication between man and machine. *Commun. ACM* **1966**, *9*, 36–45. [CrossRef]
30. Fitzpatrick, K.K.; Darcy, A.M.; Vierhile, M. Delivering Cognitive Behavior Therapy to Young Adults with Symptoms of Depression and Anxiety Using a Fully Automated Conversational Agent (Woebot): A Randomized Controlled Trial. *JMIR Ment. Health* **2017**, *4*, e7785. [CrossRef] [PubMed]
31. Zhang, Y.; Chen, X.; Ai, Q.; Yang, L.; Croft, W.B. Towards Conversational Search and Recommendation: System Ask, User Respond. In Proceedings of the 27th ACM International Conference on Information and Knowledge Management, CIKM'18, Torino, Italy, 22–26 October 2018; Association for Computing Machinery: New York, NY, USA, 2018; pp. 177–186. [CrossRef]
32. Zhao, T.; Eskénazi, M. Towards End-to-End Learning for Dialog State Tracking and Management using Deep Reinforcement Learning. *arXiv* **2016**, arXiv:1606.02560.
33. Goh, O.S.; Jaya Kumar, Y.; Sam, Y.H.; Leong, P. The Evaluation of User Experience Testing for Retrieval-based Model and Deep Learning Conversational Agent. *Int. J. Adv. Comput. Sci. Appl.* **2021**, *12*, 2021. [CrossRef]
34. Zhang, L.; Li, W.; Bai, Q.; Lai, E. Graph-Based Self-Adaptive Conversational Agent. In *AAMAS '21, Proceedings of the 20th International Conference on Autonomous Agents and MultiAgent Systems, Online, 3–7 May 2021*; International Foundation for Autonomous Agents and Multiagent Systems: Richland, SC, USA, 2021; pp. 1791–1793.
35. Roque, A.; Leuski, A.; Sridhar, V.K.R.; Robinson, S.; Vaswani, A.; Narayanan, S.S.; Traum, D.R. Radiobot-CFF: A spoken dialogue system for military training. In Proceedings of the INTERSPEECH, Pittsburgh, PA, USA, 17–21 September 2006.

36. Gandhe, S.; Whitman, N.; Traum, D.; Artstein, R. An Integrated Authoring Tool for Tactical Questioning Dialogue Systems. In Proceedings of the 6th IJCAI Workshop on Knowledge and Reasoning in Practical Dialogue Systems, Pasadena, CA, USA, 12 July 2009.
37. Malik, N.; Sharan, A.; Biswas, P. Domain knowledge enriched framework for restricted domain question answering system. In Proceedings of the 2013 IEEE International Conference on Computational Intelligence and Computing Research, Enathi, India, 26–28 December 2013; pp. 1–7.
38. Moldovan, D.; Paşca, M.; Harabagiu, S.; Surdeanu, M. Performance Issues and Error Analysis in an Open-Domain Question Answering System. In *ACL '02, Proceedings of the 40th Annual Meeting on Association for Computational Linguistics, Philadelphia, PA, USA, 6–12 July 2002*; Association for Computational Linguistics: Philadelphia, PA, USA, 2002; pp. 33–40. [CrossRef]
39. Setyawan, M.Y.H.; Awangga, R.M.; Efendi, S.R. Comparison Of Multinomial Naive Bayes Algorithm And Logistic Regression For Intent Classification In Chatbot. In Proceedings of the 2018 International Conference on Applied Engineering (ICAE), Batam, Indonesia, 3–4 October 2018; pp. 1–5.
40. Wang, S.; Manning, C. Baselines and Bigrams: Simple, Good Sentiment and Topic Classification. In Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), Jeju Island, Korea, 8–14 July 2012; Association for Computational Linguistics: Jeju Island, Korea, 2012; pp. 90–94.
41. Amber, N.; Sahare, P.; Pandya, K. Intent Detection and Slots Prompt in a Closed-Domain Chatbot. *arXiv* **2018**, arXiv:1812.10628.
42. Chen, L.; Zhang, D.; Mark, L. Understanding User Intent in Community Question Answering. In *WWW '12 Companion, Proceedings of the 21st International Conference on World Wide Web, Lyon, France, 16–20 April 2012*; Association for Computing Machinery: New York, NY, USA, 2012; pp. 823–828. [CrossRef]
43. Bhargava, A.; Celikyilmaz, A.; Hakkani-Tür, D.; Sarikaya, R. Easy contextual intent prediction and slot detection. In Proceedings of the 2013 IEEE International Conference on Acoustics, Speech and Signal Processing, Vancouver, BC, Canada, 26–31 May 2013; pp. 8337–8341. [CrossRef]
44. Draskovic, D.; Gencel, V.; Zitnik, S.; Bajec, M.; Nikolić, B. A software agent for social networks using natural language processing techniques. In Proceedings of the 2016 24th Telecommunications Forum (TELFOR), Belgrade, Serbia, 22–23 November 2016; pp. 1–4.
45. Gaikwad, G.; Joshi, D.J. Multiclass mood classification on Twitter using lexicon dictionary and machine learning algorithms. In Proceedings of the 2016 International Conference on Inventive Computation Technologies (ICICT), Coimbatore, India, 26–27 August 2016; Volume 1, pp. 1–6. [CrossRef]
46. Sullivan, K.O. Comparing the Effectiveness of Support Vector Machines and Convolutional Neural Networks for Determining User Intent in Conversational Agents. Master's Thesis, Technological University Dublin, Dublin, Ireland, 2018.
47. Troussas, C.; Krouska, A.; Sgouropoulou, C.; Voyiatzis, I. Ensemble Learning Using Fuzzy Weights to Improve Learning Style Identification for Adapted Instructional Routines. *Entropy* **2020**, *22*, 735. [CrossRef]
48. Rustamov, S.; Bayramova, A.; Alasgarov, E. Development of Dialogue Management System for Banking Services. *Appl. Sci.* **2021**, *11*, 10995. [CrossRef]
49. Liu, Y.; Qiu, M.; Qu, C.; Chen, C.; Guo, J.; Zhang, Y.; Croft, W.B.; Chen, H. IART: Intent-aware Response Ranking with Transformers in Information-seeking Conversation Systems. *arXiv* **2020**, arXiv:2002.00571.
50. Weld, H.; Huang, X.; Long, S.; Poon, J.; Han, S.C. A survey of joint intent detection and slot-filling models in natural language understanding. *arXiv* **2021**, arXiv:2101.08091.
51. Hemphill, C.T.; Godfrey, J.J.; Doddington, G.R. The ATIS Spoken Language Systems Pilot Corpus. In *Speech and Natural Language, Proceedings of the Workshop Held at Hidden Valley, Proceedings of the Workshop Held at Hidden Valley, PA, USA, 24–27 June 1990*; Texas Instruments Inc.: Dallas, TX, USA, 1990.
52. Mrksic, N.; Séaghdha, D.Ó.; Wen, T.; Thomson, B.; Young, S.J. Neural Belief Tracker: Data-Driven Dialogue State Tracking. *arXiv* **2016**, arXiv:1606.03777.
53. Shah, P.; Hakkani-Tür, D.; Tür, G.; Rastogi, A.; Bapna, A.; Nayak, N.; Heck, L.P. Building a Conversational Agent Overnight with Dialogue Self-Play. *arXiv* **2018**, arXiv:1801.04871.
54. Sneiders, E. Automated Question Answering: Template-Based Approach. Doctor's Thesis, Royal Institute of Technology and Stockholm University, Stockholm, Sweden, 2002.
55. Stapley, B.; Benoit, G. Biobibliometrics: Information retrieval and visualization from co-occurrences of gene names in MEDLINE abstracts. In Proceedings of the Pacific Symposium on Biocomputing, Honolulu, HI, USA, 5–9 January 2000; Volume 2000, pp. 529–540. [CrossRef]
56. Fiszman, M.; Rindflesch, T.; Kilicoglu, H. Integrating a Hypernymic Proposition Interpreter into a Semantic Processor for Biomedical Texts. In Proceedings of the AMIA...Annual Symposium Proceedings/AMIA Symposium, Washington, DC, USA, 8–12 November 2003; Volume 2003, pp. 239–243.
57. Bhoir, V.; Potey, M.A. Question answering system: A heuristic approach. In Proceedings of the Fifth International Conference on the Applications of Digital Information and Web Technologies (ICADIWT 2014), Bangalore, India, 17–19 February 2014; pp. 165–170.
58. Liu, Z.-X.; Chang, C.-H, Chatlog Disentanglement based on Similarity Evaluation Via Reply Message Pairs Prediction Task. *Int. J. Comput. Linguist. Chin. Lang. Process.* **2019**, *24*, 63–77.
59. Bartl, A.; Spanakis, G. A retrieval-based dialogue system utilizing utterance and context embeddings. *arXiv* **2017**, arXiv:1710.05780.

60. Juraska, J.; Karagiannis, P.; Bowden, K.; Walker, M. A Deep Ensemble Model with Slot Alignment for Sequence-to-Sequence Natural Language Generation. In Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers), New Orleans, LA, USA, 1–6 June 2018; Association for Computational Linguistics: New Orleans, LA, USA, 2018; pp. 152–162. [CrossRef]
61. Song, K.; Tan, X.; Qin, T.; Lu, J.; Liu, T. MASS: Masked Sequence to Sequence Pre-training for Language Generation. *arXiv* **2019**, arXiv:1905.02450.
62. Wang, J.; Liu, J.; Bi, W.; Liu, X.; He, K.; Xu, R.; Yang, M. Improving Knowledge-aware Dialogue Generation via Knowledge Base Question Answering. *arXiv* **2019**, arXiv:1912.07491.
63. Moon, S.; Shah, P.; Kumar, A.; Subba, R. OpenDialKG: Explainable Conversational Reasoning with Attention-based Walks over Knowledge Graphs. In Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, Florence, Italy, 28 July–2 August 2019; Association for Computational Linguistics: Florence, Italy, 2019; pp. 845–854. [CrossRef]
64. Bockhorst, J.; Conathan, D.; Fung, G.M. Knowledge Graph-Driven Conversational Agents. In Proceedings of the 33rd Conference on Neural Information Processing Systems (NeurIPS 2019), Vancouver, Canada, 8–14 December 2019.
65. Chih Wei, H.; Lin, C.J. A comparison of methods for multiclass support vector machines. *IEEE Trans. Neural Netw.* **2002**, *13*, 415–425. [CrossRef]
66. Yuan, W.; Ling-yu, Z.; Ya-xuan, Z.; Lu, H.; Ding-yi, F. Combining Support Vector Machines, Border Revised Rules and Transformation-based Error-driven Learning for Chinese Chunking. In Proceedings of the 2010 International Conference on Artificial Intelligence and Computational Intelligence, Sanya, China, 23–24 October 2010; Volume 1, pp. 383–387. [CrossRef]
67. Hamada, A.; Dafoulas, G.; Ismail, M. Intent Classification for a Management Conversational Assistant. In Proceedings of the 2020 15th International Conference on Computer Engineering and Systems (ICCES), Cairo, Egypt, 15–16 December 2020; pp. 1–6. [CrossRef]
68. Burges, C.J.C.; Svore, K.M.; Wu, Q.; Gao, J. *Ranking, Boosting, and Model Adaptation*; Technical Report MSR-TR-2008-109; Microsoft Research: Redmond, WA, USA, 2008.
69. Boser, B.E.; Guyon, I.M.; Vapnik, V.N. A Training Algorithm for Optimal Margin Classifiers. In *COLT '92, Proceedings of the Fifth Annual Workshop on Computational Learning Theory, Pittsburgh, PA, USA, 27–29 July 1992*; Association for Computing Machinery: New York, NY, USA, 1992; pp. 144–152. [CrossRef]
70. He, H.; Ma, Y. Class Imbalance Learning Methods for Support Vector Machines. In *Imbalanced Learning: Foundations, Algorithms, and Applications*; The Institute of Electrical and Electronics Engineers, Inc.: Piscataway, NJ, USA, 2013; pp. 83–99. [CrossRef]