Connecting the System to Enhance the Practitioner and Consumer Experience in Healthcare
E. Cummings et al. (Eds.)

doi:10.3233/978-1-61499-890-7-51

Quro: Facilitating User Symptom Check Using a Personalised Chatbot- Oriented Dialogue System

Shameek Ghosh[a,1], Sammi Bhatia[a], Abhi Bhatia[a]
[a] *Medius Health, Sydney, Australia*

Abstract. Automated conversational agents built with medical applications in mind, have the potential to reduce healthcare readmissions and improve accessibility to medical knowledge. In this work, we demonstrate the development and evaluation of an automated chatbot for triage and conditions assessment, based on user inputs in natural language. The implemented bot engages patients in conversation about symptoms experienced and provides a personalized pre-synopsis based on their symptoms and profile. Our chatbot system was able to predict user conditions correctly based on two sets of patient test cases with an average precision of 0.82. Our implementation demonstrates that a medical chatbot can help with automatic triage and pre-assessment of patients with simple symptom analysis and a conversational approach without the use of cumbersome form-based data entry.

Keywords. primary care, conversational chatbot, natural language processing, artificial intelligence.

Introduction

"Listen to your patient; they are telling you the diagnosis" is an oft-quoted aphorism in patient history taking. Ideally, the healthcare journey of a patient involves lengthy history taking, ordering of laboratory tests, and physical examination. However, innumerable users face difficulties in accessing medical information in a highly convoluted healthcare system [1]. Typically, patients tend to seek information about their symptoms or conditions online before visiting a doctor. Such requirements vary significantly while seeking information about diseases to advice on diet. Following from that, online users want to know if their conditions require going to emergency care, visiting the GP or remain at home and rest. This process is also known as medical triage.

In the future, the development of advanced medical triage powered by artificial intelligence algorithms are pertinent for reducing problems related to overcrowding at emergency departments (EDs) in hospitals [2]. Increasing rates of hospital admissions can contribute to significant pressures on national health care systems, funders, and providers of ED services worldwide.

[1] Corresponding Author: Chief Technology Officer, Medius Health, Australia; Email: shameek.ghosh@mediushealth.org.

Conventional symptom checkers like webMD and Australia's health direct fall off on two elements of patient engagement [3]. Firstly, the engagement is missing through the user interface and experience. Secondly, these tools employ simplistic hard-coded rule-based logic that suffer from scalability problems and do not recommend appropriate level of care. Hard-coded rule-based systems fail to tackle complex combinations and sequences of symptoms. Hence, the costs of maintaining and extending these systems rise exponentially with time. Such attempts can be a source of confusion for users since often potentially life-threatening conditions are reported in long lists of likely candidates.

This paper describes the development process of Quro, a personalized healthcare assistant in the form of a smart, chat-based interface as a collaborative effort between medical experts and engineers and is currently deployed as part of our digital health platform. The functioning of the medical chatbot is based on the following structure - using natural language processing to make sense of the user's demands followed by knowledge management to provide an answer. The learning modules in machine learning helps the chatbot improve its response to each interaction. As part of this work, we describe our computational model of triage and how it provides a flexible and efficient basis for modelling triage systems.

1. Related Work

The use of symptom checkers is known to be very popular reporting 15 million users per month to 50 million visits per year [5]. In recent years, Semigram et al. [3] performed a study using 23 symptom checkers and evaluated them against 45 standardized patient vignettes. Results were reported in three categories of triage urgency: emergent care, non-emergent care, and self-care. The study pointed out the limitations of these symptom checkers in their ability to predict conditions requiring reasonable self-care. In a later work, Semigram et al. [5] reported a comparison of these symptom checkers with physician accuracies (based on diagnosis on patient vignettes obtained from the Human Dx project).

There are two main goals that were associated with the development of our system. To provide a useful service to the user it is crucial that the triage system is not too risk averse. This is a common occurrence for most automated symptom checkers where users are asked to seek emergency care, sometimes regardless of the type of reported symptoms. Secondly, capturing all relevant potential symptoms in as little time as possible during the questioning processes is highly essential to facilitate a proper assessment.

2. Methodology

Our chatbot user dialogue is enabled by a suite of data mining and natural language processing algorithms, while following a linear design approach. The following sections discuss the overall architecture, followed by the symptom extraction process, facilitating user dialogue, and finally leading to a condition assessment.

2.1. Architecture of a Goal-oriented Healthcare Chatbot

The proposed chatbot solution employs a finite state model for facilitating a linear dialogue with the user. For efficiently facilitating user conversation towards the final goal of a condition assessment, state transition logic was created for a finite set of dialogue states. Natural language generation to user responses are based on pre-defined templates, and system initiative to prompt easily interpretable responses from the user.

The conversational agent begins with a user greeting state, and quickly moves on to three main conversational phases. These are - (1) acquisition of basic information, (2) symptom extraction, (3) sequential question answering, and (4) condition assessment. As part of basic user information acquisition, the agent initially begins with querying about the user's gender, age, smoking history, history of heart problems. In the next state, the user is prompted to input his or her symptom or medical problem, using natural language. Methods in symptom extraction process are described in section 2.3. The process of symptom extraction from the user is repeated twice. In the third phase, a sequential search algorithm is initiated using the user supplied symptoms. The primary goal of our multi objective search optimization algorithm is to maximize the relevance of each question asked, while facilitating convergence towards a relevant selection of conditions with each step. After a predefined stopping criterion, the user dialogue is concluded and a pre-synopsis of conditions relevant to the user's symptoms are reported. The flow of control in our solution is shown in Figure 1.

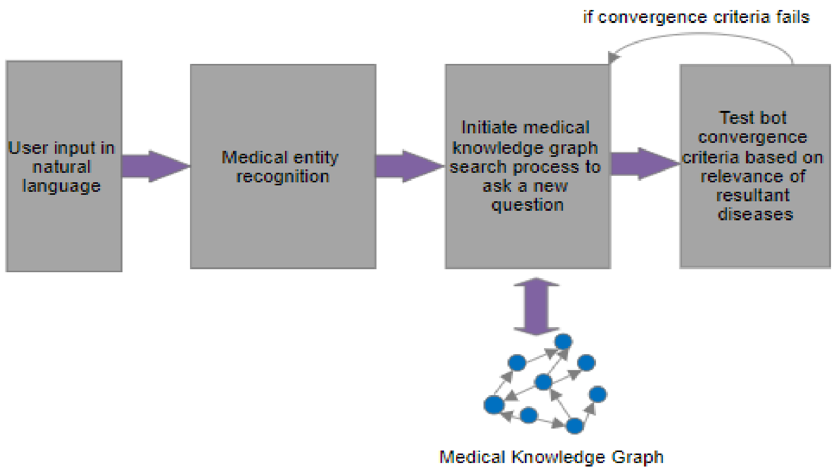

Figure 1. Quro Medical Chatbot Control Flow.

2.2. Slot-filling of Demographic Information to Guide Dialogue

Slot filling for demographic information, age and gender, is conducted with relatively naive algorithms due to the simplicity of the tasks. There is generally a set number of terms one uses to describe biological sex, and age is generally only described numerically or with the words spelled out. We chose not to include implicit and explicit confirmation for these steps, since users generally felt confident in bot's ability to accurately understand their inputs. To do so would disrupt the flow of conversation.

2.3. Recognition and Relation Extraction of Medical Entities Using User Inputs

Symptom extraction from user input involves employing algorithms used to recognize potential medical entity substrings in natural language text. A "medical entity" can refer to an instance of a medical concept such as Sign, Symptom, Disease, Drug and many more. Typically, medical entity recognition consists in: (i) identifying medical entities in the free text, and (ii) determining their categories. This is followed by detecting potential semantic relationships between the extracted medical entities. For example, in the following sentence " I am having tummy or bowel problems and had persistent diarrhea 6 weeks ago.”, the medical entity "bowel problems" should be identified as a finding, and "diarrhea" as a disease. An important obstacle to identifying medical entities in natural text is related to the high variation of terminologies in the healthcare domain (e.g. tummy problems = bowel problems).

To address the above issues, the following approach was adopted.

- Natural language text is split into sentences, and noun phrases are extracted.
- After the initial step of text processing, medical entities are detected using multiple ontologies UMLS (Unified Medical Language System) [9], Disease Ontology (DO) [10], Symptom Ontology (SO) [10], MeSH [6], ICD10 [7], and SNOMED [8].
- Later, we filter the obtained medical entities using the standardized UMLS code for each entity that was identified.

Following the extraction of medical entities, we focus on extracting relations between these medical entities. Typically, our approach is based on the use of linguistic patterns. For every couple of medical entities, we collect the possible relations between their semantic types in the UMLS Semantic Network. For this, we construct patterns for each relation type and match them with the sentences to identify the correct relation. The relation extraction process relies on two criteria: (i) a degree of specialization associated to each pattern and (ii) an empirically-fixed order associated to each relation type which allows to order the patterns to be matched. We target relation types: “causes”, “complicates”, and “sign or symptom of”. For each entity pair detected in a sentence, we construct a dependency graph based on the sequence of tokens extracted after initial pre-processing of text. Generally, the parsed dependency graph of a sentence is used to analyze the sentence structure. Later, the shortest path connecting each entity in the dependency graph based on a dictionary of relevant similar terms such as "causes", "symptom of", "leads to", and an associated verb phrase is used to extract a relation between two entities.

2.4. Using a Medical Knowledge Graph to Enable Sequential Search

Traditionally, a triage has corresponded to a sequence of questions and answers. In most implementations, a tree like model is used to handle whenever a symptom is triggered. However, such systems are not scalable and are difficult to maintain. To handle complex question answering tasks, we employed a linked medical knowledge graph (developed internally) to explore the associations between all the potential medical entities recognized by the user input. The selected entities are then ranked by their strength of associations with user selected entities. The ranking of entities is also facilitated by a frequent occurrence of entities attached to the original extracted entities. Finally, this mechanism is used to select the top ranked symptom that is send back using a natural language template response to the user. Additionally, the strength of symptom-disease associations are also used to develop a iterative ranking of diseases. Based on empirical

Testing, the conversation ends after 10 to 15 questions are asked. Currently our system is available for further testing and usage at www.quro.ai for public usage.

3. Results and Discussion

The initial evaluation of our chatbot was carried using a total of 30 clinical scenarios. These patient vignettes were divided into three categories of triage urgency: emergent care, GP, and self-care. Each category consisted of 10 clinical scenarios. Each scenario consisted of a maximum of 5 and a minimum of 2 symptoms associated with the a given condition.

For each vignette, we considered evaluations based on two criteria. In the first criteria, we compared if at least one of the top 3 reported conditions was a correct expected assessment. In the second criteria, we evaluated if 2 out of 3 reported conditions were expected conditions by our in-house specialists. Confusion matrices for our evaluations for both the criteria are shown in Table 1 and Table 2.

Table 1. Confusion matrix: At least 1 out of 3 predicted conditions is expected.

Category	Predicted: Yes	Predicted: No
Self-care	8	2
GP	7	3
Emergent care	10	0

Table 2. Confusion matrix: At least 2 out 3 predicted conditions are expected.

Category	Predicted: Yes	Predicted: No
Self-care	4	6
GP	6	4
Emergent care	10	0

As shown in the confusion matrices, our bot achieved an accurate outcome (in criteria) in 25 out of 30 cases (83.3%) and in 20 out of 30 cases (66.6%). Interestingly, the chatbot demonstrated a high recall of 100% for emergent care. An interesting aspect of our system was even though we populate our database with red-flag symptoms (for emergent care), our system does not rely on these red flag rules to infer an emergent care condition. There were 25 true positives (TPs) and 3 false positives (FPs) in criteria 1, and 20 TPs and 6 FPs in criteria 2. Accordingly, our chatbot showed an overall average precision of 0.82. Rather, the reasoning algorithm used for all cases is consistent, and is strongly dependent on the process of optimization. Although the current evaluation of Quro was performed at a smaller scale, we plan to perform continuous evaluations with more complex patient vignettes to further test our chatbot's capability. Generally, consumer trust towards a chatbot is limited due numerous factors viz. (1) limited end-user engagement using improved design principles that help understand a consumer journey towards accessing health information, and (2) extremely limited use of artificial intelligence and machine learning techniques for generating responses to the user, and towards making a probabilistic disease prediction. As part of this work, we have consistently focused on engaging users, and benchmarking our disease prediction models using multiple sets of clinical vignettes. Such works have also been reported in recent years internationally [1-2].

4. Conclusions

An automated medical conversational platform powered by learning algorithms that provides personalized assessments based on symptoms is described. The bot's symptom recognition and condition assessment performance could be greatly improved by adding support for more medical features, such as location, adverse events, and medical entities. The conversational platform is extended on a regular basis to improve the detail and quality of the conditions assessment. Currently, we enable light progressive prompting, but in the immediate future, we would implement more structured progressive prompting mechanisms when the bot is unable to understand user input. Thus, automated medical agents will quickly assess patients, recommend appropriate triage, and connect to the right medical care to reduce costs and increase convenience for patients as well as reduce the burden on overworked doctors. Numerous studies in the past have reported better performances using machine learning algorithms in comparison traditional risk calculators for estimating risk. Our system is not a diagnostic tool and rather provides a pre-assessment of probable conditions using learning algorithms across 7 million medical entities and patterns over a large-scale knowledge graph. Our solution used multiple sets of clinical vignettes following similar study design in the past, for benchmarking of symptom checkers. Further, we use the same mechanisms of flagging messages to a user as triage solutions like HealthDirect. Feedback regarding results are obtained and reviewed by clinicians involved in the development of the system and will be a feature rolled out to users too.

References

[1] M.A. Clarke, et al., Health information needs, sources, and barriers of primary care patients to achieve patient-centered care: A literature review. *Health informatics journal*, **22** (2016), pp.992-1016.

[2] B. Graham, R Bond, M. Quinn, M. Mulvenna, Using Data Mining to Predict Hospital Admissions from the Emergency Department. *IEEE Access* (2018).

[3] H. L. Semigran, A. Jeffrey, G. Courtney, A. Mehrotra. Evaluation of symptom checkers for self-diagnosis and triage: audit study. BMJ 351 (2015): h3480.

[4] B. Gann, Giving patients choice and control: health informatics on the patient journey, Yearbook of medical informatics, vol. 7.1, 2011.

[5] H. L. Semigran, Comparison of physician and computer diagnostic accuracy. JAMA internal medicine 176, no. 12 (2016): 1860-1861.

[6] E. C. Lipscomb, Medicalsubject headings (MeSH). Bulletin of the Medical Library Association 88, no. 3 (2000): 265.

[7] International Classification of Diseases 10th Revision. World Health Organization. 2010. Retrieved February 26, 2010.

[8] R. Cornet, Nicolette de Keizer. Forty years ofSNOMED: a literature review. BMC medical informatics and decision making, vol. 8, no. 1, p. S2. BioMed Central, 2008.

[9] A. McCray, The UMLS Semantic Network. Symposium on Computer Applications in Medical Care, pp. 503-507. American Medical Informatics Association, 1989.

[10] A. Kibbe et al, Disease Ontology 2015 update: an expanded and updated database of human diseases for linking biomedical knowledge through disease data. Nucleic acids research 43, no. D1 (2014): D1071-D1078.