J Syst Sci Syst Eng (Oct 2018): 27(5): 559-585
DOI: https://doi.org/10.1007/s11518-018-5389-1
ISSN: 1004-3756 (Paper) 1861-9576 (Online)
CN11-2983/N

A FRAMEWORK FOR DOMAIN-SPECIFIC NATURAL LANGUAGE INFORMATION BROKERAGE

Lin Ni   Jiamou Liu

Department of Computer Science, The University of Auckland, New Zealand

lni600@aucklanduni.ac.nz, jiamou.liu@auckland.ac.nz (✉)

Abstract

Service providers – from public institutions to primary care facilities – need to constantly attend to clients' inquiries to provide useful information and directive guidelines. Ensuring high quality service is challenging as it not only demands detailed domain-specific knowledge, but also the ability to quickly understand the clients' issues through their diverse – and often casual – descriptions. This paper aims to provide a framework for the development of an automated information broker agent who performs the task of a helper. The main task of the agent is to interact with the client and direct them to obtain further services that cater their personalized need. To do so, the agent should accomplish a sequence of tasks that include natural language inquiry, knowledge gathering, reasoning, and giving feedback; in this way, it simulates a human helper to engage in interaction with the client. The framework combines a question-answering reasoning mechanism while utilizing domain-specific knowledge base. When the users cannot describe clearly their needs, the system tries to narrow down the possibilities by an iterative question-answering process, until it eventually identifies the target. In realizing our framework, we make a proof-of-concept project, Mandy, a primary care chatbot system created to assist healthcare staffs by automating the patient intake process. We describe in detail the system functionalities and design of the system, and evaluate our proof-of-concept on benchmark case studies.

Keywords: Question and answer system, chatbot, automated information broker, iterative inquiry, language processing, AI and healthcare

1. Introduction

Questioning and answering is a basic form of human communication and has been important to any situations that involve information exchange. Imagine, for example, a student support service at a university campus during a new student orientation. A freshman may encounter a wide range of problems, from course enrolment, paying tuition fees, purchasing textbooks, to looking for classrooms or campus facilities. The student support service may inquiry the student regarding their needs, and guide them with necessary information. Another example involves a help desk in a shopping mall, who would attend to customer's need and make recommendations regarding the shops. Much of the interaction between the help desk and a customer would be the helper inquiries about the intention of the customer aiming to match the specifications with shops in the shopping mall. A third example is the reception in a primary care provider; when a patient arrives at the health facility after experiencing certain symptoms, the reception staff would need to ask a series of questions to the patient in order to obtain an overall picture of the patient's need and deliver this information to suitable medical staffs. In a metaphorical way, the types of interactions above resemble a well-known guessing game, i.e., "Twenty Questions", where participants repeatedly ask questions to a host in the hope of guessing the identity of a person or an object; The host must respond truthfully with "yes" or "no" answers and the game lasts until a participant correctly finds the answer (Bendig 1953).

The scenarios described above – despite having vastly different contexts – embody a similar process of *information brokerage* (Yan et al. 2017, Moskvina and Liu 2016a,b). Here, one may view a organization, i.e., a university, a shopping mall, or a medical center, as a system that provides a range of services (Chen et al. 2016). A client wants to access certain information within the system subject to personalized needs. An agent thus takes the role of an information broker between the client and the system by performing two tasks: Firstly, the broker must identify the personalized needs of the client, and then, the corresponding information in the system (Moskvina and Liu 2016b, Yan et al. 2018).

To ensure a good quality of service, it is crucial to implement effective information management between the client and the system. Generally speaking, three approaches may be used for this task. The first is to rely on well-trained staffs to act as the agent. As the client does not have clear and thorough knowledge about the services, often the client may only describe their needs using vague, varied and casual language. It is the task of the agent to understand this language and interpret the need. The agent should acquire specific domain knowledge in order to find appropriate information. Given these limitations, having good personnel to act as a broker may be costly. The second method is to use a combination of human agent and well-designed procedures such as questionnaires given to the clients. This has been implemented widely in, e.g., hospitals, where patients tick answers in a standard form regarding their symptoms. With this semi-formalized approach, the human agent does not need to have specialist knowledge and hence reduces the personnel cost of the organization as well as risks of human mistakes. However, the rigid format normally reflects limited personalized needs of the client. The third approach is to use the help of a computerized agent. This has become possible in recent years due to the fast advancement of artificial intelligence technology, where chatbots may surpass the Turing test and interact with human in a smooth way (Saygin et al. 2000).

1.1 Contribution

The focus of this paper is on the third approach. We are interested in artificial intelligence agents that mimic the behaviors of a human information broker. In particular, we put forward a framework for such an agent who integrates multiple tasks from natural language understanding, knowledge base query, to reasoning and giving feedbacks. Instead of performing these tasks in series when faced with a client, the agent may – as human agents in a real-world practice – interact with the client through repeated question-answering. By that, we mean that the broker would initiate a series of questions, seeking the client to refine their requests or narrow down the scope. Through an iterative inquiry process, the broker would eventually identify a final solution for the client. Our framework would simulate this process.

To demonstrate the applicability of our framework, we design and implement a prototype of a medical chatbot, Mandy, who serves as an information broker for patients arriving at a primary care facility. Mandy would serve at the reception and collect patient information. The chatbot interacts and understands the symptoms of the patient through natural language communication. A short report is then generated as the outcome of this inquiry process to narrow down the causes of the symptoms, which may in turn help the doctors for further differential diagnosis. The system is meaningful as it may enhance the inquiry experience of end-users. With an aging population, there has been an ever increasing need for new technologies for efficient and reliable healthcare (Caley and Sidhu 2011). This need is especially significant for developing nations who face a fast growing population. An AI system like Mandy is greatly in need. This system thus amounts to a step towards precision-driven healthcare which promotes the application of data science, in particular technologies such as interactive cognitive systems, artificial intelligence and machine learning, to enhance healthcare provision (Dobbie and Ross 2017).

1.2 Paper Organization

The rest of the paper is organized as follows: Section 2 presents related works on chatbots and question-answering systems. Section 3 presents our framework for an interactive automated information broker. In particular, we define our model of iterative inquiry process and describe the three main modules in its implementation: the analysis engine, the hypothesis operator, and the question generator. Section 4 presents a proof-of-concept of Mandy, a medical chatbot developed based on our framework above. We emphasize how our conceptual framework forms the backbone of Mandy and present in detail the realization of each module. We evaluate Mandy using real-world medical test cases and discusses the results. Finally, Section 5 concludes by listing some future works.

2. Related Work

2.1 Question-answering

Question-answering is a well-established research direction in natural language processing. The endeavour for a computerized system that could interact with users through iterative dialogues began since the birth of computer science in the 1960's. It has become a key ability that indicates intelligence. Early systems rely on linguistic approaches where semantic parsing is used to convert natural language input into the database query, and the answer is subsequently extracted from the knowledge base. Examples of such technologies include Baseball (Green Jr et al. 1961) and Lunar (Woods and Kaplan 1977). For Baseball, the knowledge base consists of the month, day, place, teams and scores for each game in the American League within a year and the system's goal is to provide answers to questions regarding teams or games. For LUNAR, on the other hand, the knowledge base contains geological rock samples from the Apollo moon missions. Due to the knowledge engineering bottleneck, early systems have limited reasoning ability to personalized user inputs and are the constraint to very limited domains (Winograd 1971, Bobrow 1964).

Another early question-answering system, Unix Consultant (UC) enables users to learn about the UNIX system (Wilensky et al. 1988). UC adopts a modular approach consisting of separate modules, which perform respectively language analysis, goal recognition, goal planning, and language generation individually. Similar diagnostic tools include the IBM LILOG (Herzog and Rollinger 1991), which is implemented to answer German questions as a tourism help desk, and the central questions in LILOG revolved around problems in text understanding.

Open-domain question-answering starts from the turn of the millennium, with a goal of question-answering in general contexts and have progressed significantly in the last 15 years. The most notable breakthrough is IBM Watson (Ferrucci et al. 2010) being arguably one of the most famous systems. Watson is based on extensive data, statistical and machine learning analysis. It competes on the American quiz show "Jeopardy!" and showed that it could compete at the level of a human champion player. Today, question answering systems is a significant area of research that is continuously evolving and growing.

Speech recognition applications are becoming prevalent, where personalized assistant application such as Siri[1] and Cortana[2] have been integral parts of people's online activities. Chatbot Systems and Spoken Dialogue Systems (SDS) respond with comprehensible sentences and elaborately constructed paragraphs to communicate with the user.

2.2 Healthcare Service Systems

A healthcare system is a type of complex service platform that integrates people, processes and products about primary care. A healthcare system aims to facilitate the efficient use of information technology in primary care service providers, and help to improve the productivity and patient experience (Tien and Goldschmidt-Clermont 2009). The development of healthcare service systems needs to be adaptive and has been a focus in the system science and engineering community. As argued in (Tien and Berg 2003), the resulting system should be information- driven, customer-centric, e-oriented, and productivity-focused. As advancements of information and Web technologies enable easy access to healthcare services by a much larger and diverse patient population, there is an increasing need to design healthcare service systems that are aware of the individual differences and attend to personal needs (Edgren 2006). A patient who has the need for help would no longer act as passive receivers of service or treatment, but rather, they are active information seekers and demands more involvement in the process. Therefore, there is

[1] Siri https://www.imore.com/siri
[2] Cortana https://www.microsoft.com/en-us/cortana

an increasing need to utilize artificial intelligence to bring the patients personalized demands more accurately and easily to the service providers. Existing work in the system science community along this line includes decision support system to prioritize goals during a pandemic (Araz 2013), automated assessment of drug-side effects from electronic medical records (Dang and Ho 2017), smart capacity allocation scheme for better management of outpatient capacities (Jiang et al. 2017), and dynamic resource allocation for patient scheduling (Bakker and Tsui 2017).

The earliest medical natural language conversational program, ELIZA (Weizenbaum 1966) is designed to respond roughly as psychotherapists. In the last 20 years, chatbot systems are widely used in healthcare for both commercial products and academic researches. Florence Bot is a taking pill reminder [3]. Your.MD [4] and HealthTap [5] are miniature doctors. Some studies verified SDS could help intervening human habits, in terms of smoking (Ramelson et al. 1999), dietary behaviour (Delichatsios et al. 2001) and physical activity (Farzanfar et al. 2005). Similarly, some others also used SDS for chronic illness monitor systems, like for hypertensive diabetic (Black et al. 2005). Medical counseling and education is another area which often requires the delivery of SDS (Bickmore and Giorgino 2006; Hubal and Day 2006, Bickmore et al. 2010).

Clinical decision support systems (CDSS) have also been intensively studied. As an example, MYCIN (Victor et al. 1979) is widely recognized as one of the very first rule-based expert systems that were used for diagnosing infectious diseases. The system specializes in bacterial infections and it has been adapted as NEOMYCIN – a teaching and learning platform (Clancey and Letsinger 1982). Other systems such as INTERNIST-I (Miller et al. 1982) uses a much larger collection of medical knowledge, obtained from hospital case records to assist medical personnel in diagnosing internal conditions the patient may have. The system learns the patient's medical history to deliver more accurate results. In the system's evaluation, the output was reviewed by a panel of medical experts. CDSS systems are becoming increasingly adopted in primary care. A study identifies 192 commercially available applications at the time of writing (Martínez-Pérez et al. 2014). One of the better-known achievements in this area is from IBM's Watson Health (High 2012). The system seamlessly combines natural language processing, dynamic learning and hypothesis generation and evaluation to provide useful systems in many key areas such as oncology, genomics, and medical imaging. While most of the CDSS systems are designed to be used by the specialists but not the patients themselves due to the NLP and ethic issues.

Word embedding has shown its power in many research and models of medical domain. De Vine et. al. (De Vine et al., 2014) adopted a variation of word embedding approach and built a neural language model with two medical corpora for measuring semantic similarity between medical concepts. Comparing with the other 6 state-of-the-art benchmarks, empirical findings demonstrate that their approach

[3] Florence Bot, https://florence.chat/
[4] Your.MD, https://www.your.md/
[5] HealthTap, https://www.healthtap.com/

correlated more strongly to medical professionals' judgement. ADRMine (Nikfarjam et al. 2015) is a machine learning-based concept extraction system to extract mentions of adverse drug reactions from the highly informal text in social media. This model of word semantic similarities has significantly improved the extraction performance from informal, user-generated content. Besides, the high accuracy also made word embedding prevalent in clinical abbreviation disambiguation (Xu et al., 2015, Liu et al. 2015). Pechsiri and Sukharomana studies the use of word embedding to link symptoms and potential diseases which is similar to the approach we adopt in our prototype system (Pechsiri and Sukharomana 2017).

3. A Framework of Interaction

3.1 A Model of Iterative Inquiry Process

We now formulate the problem under investigation. Imagine a client who would like to access one or several services that are provided by a service provider. Any service would meet a number of specific needs of the client. As an example, a new exchange student may inquire about course enrolment in a specific discipline, and the corresponding service to this inquiry would be, say, the course coordinator in the corresponding department. The course coordinator's role is defined by a combination of attributes, e.g., the discipline of the department, the level of the courses that she is coordinating, etc. Formally, we gather all specific needs into a set A of attributes.

Definition 1 *Let* A *be a universal set of attributes and let* S *be a set of services. A service provider* P *specifies a relation from the set* A *to* S*, namely,* $\mathrm{P}\colon \mathrm{A}\times\mathrm{S}\to\mathcal{R}$ *such that* $\mathrm{P(a,s)}$ *represents the level to which the service s meets the attribute a for all* $\mathrm{a}\in\mathrm{A}$ *and* $\mathrm{s}\in\mathrm{S}$.

The client, who has no information about P, would need to find the service $y\in\mathrm{S}$ that best fits a given set of attributes $X \subseteq A$. For example, we say that the service provider P is binary if $\mathrm{P(a, s)} \in\{0, 1\}$, and we say that $y\in\mathrm{S}$ best fits $X \subseteq \mathrm{A}$ if for all $a\in\mathrm{X}$, $\mathrm{P(a, y)}=1$, and for all $a \notin \mathrm{X}$, $\mathrm{P(a, y)} = 0$. In this case, given $X \subseteq \mathrm{A}$, the goal of an information broker is to find $y \in S$ that best fits X if it exists.

The challenge lies in that the set of attributes X of the client is often unknown to the information broker, and is expressed through a natural language description, making the identification of the right service less obvious. Imagine the scenario when a patient arriving at a hospital with a complex combination of symptoms (e.g. different forms of pains at different parts of the body). It is normally difficult for the patient to give a specific, precise and accurate description of the situation, and a (non-professionally trained) receptionist would need to inquiry the patient with iterations of questions to make clear of the type of help the patient needs.

We capture this process through a process of iterative inquiry. Informally, the system would simulate what a human agent would do: After the client puts forward a natural language description, the system analyzes the description and generates a hypothesis regarding what may be the right solution. When the hypothesis is not conclusive, more evidence is sought from the client demanding further clarification of certain facts. These then would feed into the system for further refinement of the hypothesis. The process terminates after a number of rounds of such question-answering between the agent and the client.

Formally, by a hypothesis, we mean a list H of services in the set S with a corresponding likelihood, i.e., a sequence

$$H = (\mathrm{y}_1{:}\ \mathrm{p}_1,\ \mathrm{y}_2{:}\ \mathrm{p}_2, \ldots, \mathrm{y}_\mathrm{k}{:}\ \mathrm{p}_\mathrm{k})$$

where $y_1, y_2, \ldots, y_k \in S$ and each $p_1, p_2, \ldots, p_k$ belongs to the interval [0, 1]. A hypothesis update refers to the process that, starting from a given hypothesis H and a new piece of evidence E, generates an updated hypothesis H'. An interview corresponds to repeated hypothesis update that is performed through question-answering, which we formally define below.

Definition 2 *A question-answering pair is* $\langle \mathrm{Q}, \mathrm{A}\rangle$ *where* Q *is a question and* A *is an answer to* Q*, both of which are in natural language.*

An iterative inquiry is a sequence

$$(\langle \mathrm{A}_0, \mathrm{H}_0\rangle\langle \mathrm{Q}_1, \mathrm{A}_1, \mathrm{H}_1\rangle\langle \mathrm{Q}_2, \mathrm{A}_2, \mathrm{H}_2\rangle \ldots \langle \mathrm{Q}_\mathrm{l}, \mathrm{A}_\mathrm{l}, \mathrm{H}_\mathrm{l}\rangle)$$

where

- A_0 is an initial description,
- H_i is an initial hypothesis given A_i,
- each $\langle \mathrm{Q}_\mathrm{i}, \mathrm{A}_\mathrm{i}\rangle$ is a question-answering pair for $i \geq 1$, and H_i is the updated hypothesis from $\mathrm{H}_{\mathrm{i}-1}$ given $\langle \mathrm{Q}_\mathrm{i}, \mathrm{A}_\mathrm{i}\rangle$.

The goal of the automated information broker is to generate an iterative inquiry with the client so that the eventual hypothesis H_l truthfully reflects the client's personalized needs. Next, we present a system structure that accomplishes this goal.

3.2 A Methodic Framework

Our framework specifies the architecture of a system that generates iterative inquiries. The framework iteratively runs three modules; Figure 1 illustrates the general flow of the process.

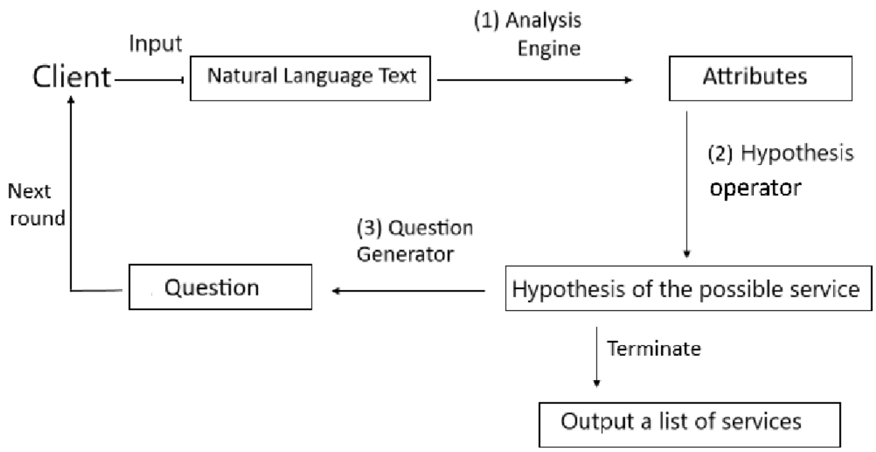

Figure 1 Main work flow in the interactive system

Module I: **Analysis Engine.** The analysis engine performs the task of knowledge extraction from human input (Wang and Tang 2016). Upon reading the client's natural language input, this module extracts a set of attributes $X \subseteq A$ that are representative of the meaning of the input. In this way, this module understands the casual description. In other words, the module performs:

INPUT A text document containing the client's input.

OUTPUT Attribute set $X \subseteq A$.

In an implementation of the analysis engine, a natural method is to derive a word embedding, a model that transforms a word into a vector (Bengio et al. 2003). The technique gives an efficient and robust semantic model in a very general context. There are two steps that word embedding plays a key role: Firstly, when the clients present their description in a lay language, the analysis engine picks up keywords and constructs bags of words. The algorithm analyzes the most likely services by computing the similarity of the client's attributes and all attributes in A. Secondly, when the input keywords do not appear in the attribute set A, the analysis engine computes words similarity, which is pre-trained on a large dataset of medical documents. The words similarity will allow the system to find the attribute in the set A that best align with the input description.

Module II: Hypothesis Operator. The second module processes the output set $X \subseteq A$ of the analysis engine and computes a hypothesis H; if the interview process has repeated for at least one round, then the hypothesis operator would take the current hypothesis H and incorporate the new question-answering pair into H and obtained an updated hypothesis H'. In other words, the module performs:

INPUT Attributes $X \subseteq A$, hypothesis H (empty if this is the initial iteration).

OUTPUT Hypotheses H'.

This module would need to realize the links between the attributes and the services. This can be achieved in many ways. An easy implementation would be to encode such links into a knowledge base.

Module III: Question Generator. Module III takes the list of hypothesized services $y_1, y_2, \dots, y_l \subseteq S$ that have a high likelihood, and generates a new question with a most likely attribute for the client to confirm. Unless the system has obtained enough information to confirm a service as output, it will continue to pose new questions to the client. The input list of hypotheses comes from the output of the second module; elements in the list are ordered by their likelihood. The output is a service in the final hypothesis that has the highest likelihood. The system will select from a built-in knowledge base the most likely unconfirmed attribute the client has.

The development of an interactive inquiry system for the purpose of information brokerage involves realizing each of the modules above. In this way, the framework hides the technical implementation details and can be adapted to a wide range of application scenarios.

4. Patient Intake with Mandy – A Proof-of-concept

4.1 Introduction to the Application Scenario

To demonstrate the feasibility of the framework above, we present Mandy, a medicare chatbot that welcomes incoming patients at a primary care facility. The chatbot interacts with a patient by carrying out an iterative inquiry process, understanding their chief complaints in natural language, and submitting reports to the doctors for further analysis. The system provides a mobile-app front end for the patients, a diagnostic unit, and a doctor's interface for accessing patient records.

The "overcrowding" issue or long waiting time at emergency units of hospitals and other primary care services has been a worldwide challenge (Bernstein et al. 2009; Richardson, 2006; Di Somma et al. 2015). To cope with the increasing population and an ever increasing demands of patients, a number of countries have implemented targets for reducing waiting time at the healthcare providers, e.g., New Zealand has implemented a "6-hours target" for the waiting time of patients at the emergency department since 2009 (Jones et al. 2012).

Despite vast technological advancement, present-day clinics still very much rely on healthcare staff to handle patient intake and carry out initial interviews in a manual way (Lipkin et al. 1984). On the other hand, it is widely viewed that data mining and AI may offer unprecedented opportunities and broad prospects in health (Khoury and Ioannidis 2014). Existing patient interview support applications often take the form of expert systems. A common challenge faced by all these applications is the ambiguity and diversity of patient answers. As a result, traditional expert systems usually fail to deliver effective decision support and lacks the flexibility that suits individual needs (Hunt et al. 1998). An example of AI-driven intake interview assistance system is provided by Warren in (Warren 1998). The system sets up complicated rules based on clinical experts' experience and medical knowledge. However, it does not demonstrate capabilities on personalizing the questions to patients and is not able to learn about the individual nature of patients. To apply the system, a clinic needs to provide necessary staffs with sufficient medical background to operate the system. The complicated interface of the system also requires considerable training time, which all adds extra costs to the health provider.

Efforts have been made to deploy humanoid robots (e.g., Pepper in Belgian hospitals[6]) in hospitals. However, a robot is expensive (e.g. Pepper comes with a price tag of £28000) and would not be able to efficiently cope with a large amount of people. Many industry giants are increasingly investing in AI-enhanced medical diagnosis tools. Notable products include Google DeepMind Health[7], IBM Watson Health[8] and Baidu's Melody[9]. The ambitious goal of these platforms is to allow AI to access and process a vast amount of lab test results and genomic data for precision-driven medical diagnosis and predictions. These systems differ from Mandy in their scope and purpose. Mandy is not directed to give precise diagnosis and prediction, but rather, it simply serves as an information broker between the incoming patient and the primary care service. The system aims to free up the time of healthcare staffs for more meaningful interactions with patients, and help to enable physicians to operate more efficiently.

Mandy is an integrated system that provides a range of functionalities:

1. Mandy provides a patient-end mobile application that pro-actively collects patient narratives of illness and register background

[6] http://www.bbc.com/news/technology-36528253
[7] https://deepmind.com/applied/deepmind-health/
[8] https://www.ibm.com/watson/health
[9]http://research.baidu.com/baidus-melody-ai-powered-conversational-bot-doctors-patients/

information; this may take place at an arbitrary time before the doctor's appointment and at an arbitrary location.

2. Mandy is equipped with natural language processing (NLP) modules that understand patients' lay language, process the patient symptoms, and generate interview questions.

3. Based on interactions during the interview, Mandy will generate a report for the doctor regarding the patient's symptoms and likely causes.

4. Mandy also provides a doctor-end desk-top application for the doctors to check their patients' records and interview reports.

The potential benefits of Mandy are many-fold. Firstly, the system aims to reduce the workload of medical staffs by automating the patient intake process, and providing initial reporting to doctors. Secondly, Mandy provides personalized intake service to the patients by understanding their symptom descriptions and generating corresponding questions during the intake interview. Thirdly, by interacting with a chatbot, the patient avoids the need to express his health concerns out loud to people other than the doctor. This also reduces the likelihood of patients not seeking medical help due to shyness or cultural boundaries (Taber et al. 2015). Furthermore, many studies have shown that patients tend to be more honest when facing a robot rather than a human health staff (Ahmad et al. 2009). So, Mandy is likely to collect truthful information about the patients.

4.2 System Design and Implementation

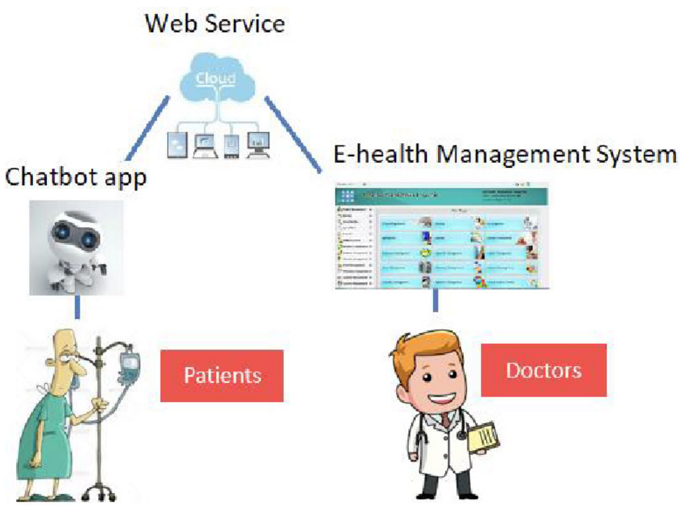

Figure 2 An illustration of the application scenario and system architecture of Mandy

Fig. 2 illustrates the architecture of Mandy. The patient interacts with Mandy through a mobile chatbot. All algorithms are executed and all data are processed in a web service (cloud). This means that all sentences to and from the patients are generated and analyzed in the cloud, respectively. After the intake interview, Mandy scores the patient's record and generate a report regarding the patient's conditions. The doctor can then login into the e-health information management system to access the personalized reports generated for the patient.

Mandy logic flow simulates a well-established clinical reasoning process for differential diagnosis, which consists of a series of well-defined steps (McFillen et al. 2013, Realdi et al. 2008, Stern et al. 2014). These steps are guidelines for medical inquiries by a practitioner:

1. Data acquisition: Collect patient's history and symptoms, which forms the basis for the initial diagnostic reasoning.

2. Problem representation: Summarize the chief complaints of the patient.

3. Developing differential diagnosis: Come up with the hypotheses list base on the data acquired.

4. Prioritizing differential diagnosis: Decide which should be the leading one among the hypotheses list.

5. Testing hypothesis: If additional data is required to confirm the hypotheses, order lab tests to take place.

6. Review and re-prioritize differential diagnosis: Rule out some diseases and then try to determine the cause of the symptoms. If a diagnosis cannot be drawn, go back to step 3.

7. Test new hypotheses: Repeat the process until a diagnosis is produced.

Our framework, as presented in Fig. 1, aligns well with Steps 1-4 of the clinical reasoning process. Mandy starts by asking the patient's chief complaint. After the patient inputs a text in natural language, the analysis engine extracts the symptoms in a standard corpus from the patient description text. In this way, the system gets an accurate problem representation. Then, the hypothesis operator module comes up with a list of hypothetic diseases based on the symptoms provided by the patient's complaint. The system ranks the possibility of the hypothetic diseases. If there is enough information for proposing the final hypothesis list, the procedure will terminate; Otherwise, the question generator will produce another question for the patient and repeats the procedure back to the analysis engine.

We next describe the key data structures and realizations of each module. The internal algorithms of Mandy rely on the following sets:

a. A symptom is a subjective, observable condition that is abnormal and reflects the existence of certain diseases. For ease of terminology, we abuse the notion including also signs, which are states objectively measured by others. A patient feature is a fact reflecting the patients, age, gender, geographical and demographical information and life styles (e.g. smoking, alcoholic). Mandy uses a set A of words representing standard symptoms and patient features that are extracted from an external knowledge base. This corresponds to the attribute set in our framework.

b. A disease is a medical condition that is associated with a set of symptoms. Mandy also uses a set S of standard diseases. This corresponds to the service set in our framework. The connection between A and S is captured by a matching function $f: S \to 2^A$ where each disease $d \in S$ is associated with a subset $f(d) \subseteq A$. In our implementation, we will construct the matching function f from explicit medical knowledge. A more elaborated way to construct such an f is to use data-driven techniques that extract the correlation between A and S using machine learning. This would be a potential future work of our study.

Example 1 *For the diseases "allergies" and "asthma", we have:*

f(allergies) = {sneezing, runny nose, stuffy nose, cough, postnasal drip, itchy nose, itchy eyes, itchy throat, watery eyes, dry skin, scaly skin, wheezing, shortness of breath, chest tightness}

f(asthma) = {cough, difficulty breathing, chest tightness, shortness of breath, wheezing, whistling sound when exhaling, frequent colds, difficulty speaking, breathless}

4.2.1 Module I: Analysis Engine

The analysis engine understands user's natural language input and extracts a set of symptoms and features from the set A. The word embedding algorithm we apply is Google's Word2Vec that maps words into their semantic vector representation (Bengio et al. 2003, Mikolov et al. 2013a,b, Zhang et al. 2016). To develop a model that is suitable for the specific medical domain, we collect a large corpus of natural language disease and symptom descriptions from a variety of sources, and train these data sets using the Word2Vec algorithm[10].

Based on the model, any word can be mapped to an N-dimension vector. Comparing the similarity of words is thus reduced to the problem of comparing the distance of different vectors. In cases that the patient describes his symptoms in nonstandard words (lay language), if the system can find out standard symptoms which are closed to what the patient input, the system will generate a new question with the standard word to the patient to confirm if the patient has the symptom[11]. The work flow of this module is illustrated in Figure 3.

Example 2 *We give two symptoms with their top-10 similar words:*

a. rash: blisters, itchy, scabies, bumps, hives, ringworm, scaly, bite, flaky, planus

b. nausea and vomiting: dizziness, abdominal pain, nausea, drowsiness, lightheadedness, cramps, sleepiness, vertigo, weakness, bloating

4.2.2 Module II: Hypothesis Operator

This module aims to map the set $X \subseteq A$ of patient's symptoms with a set of hypothesized diseases in S and evaluate their corresponding likelihoods. We propose an algorithm, named Positive-Negative Matching Feature Count (P − N)MFC, to compare the similarity between A and $f(d)$ for all $d \in S$. The algorithm runs the following steps: Suppose that we have a set A_+ of positive symptoms of the patient and a set A_- of negative symptoms. Suppose also that the set of diseases S is $\{d_1, d_2, \ldots\}$ and let $A_{d_i} = f(d_i)$ be the set of symptoms corresponding to d_i. The algorithm is described in Procedure 1.

4.2.3 Module III: Question Generator

This module takes a list of hypothesized diseases $C \subseteq S$ as input, and generates a new question with a most likely symptom for the patient to confirm. Unless Mandy has obtained enough information to derive a diagnosis, the system will continue to pose new questions to the patient. Diseases in the input hypothesis are ordered by the likelihood according to the current patient info. The output is a symptom that Mandy selects from the knowledge base which

[10] https://en.wikipedia.org/wiki/Word2vec
[11] http://mccormickml.com/2016/04/19/word2vec-tutorial-the-skip-gram-model/

represents the most likely symptom the patient has. Mandy will form a question that asks the patient to confirm or reject this symptom. The detailed steps of the algorithm are as follows:

1. Update A_+ and A_- according to the patient input.

2. If A_+ has a new element, perform the (P − N)MFC algorithm to get the most likely disease $d \in S$. If $f(d) \backslash A_+ \neq \emptyset$, randomly choose one such symptom in $f(d)$ but not in A_+ and ask about it in the next question.

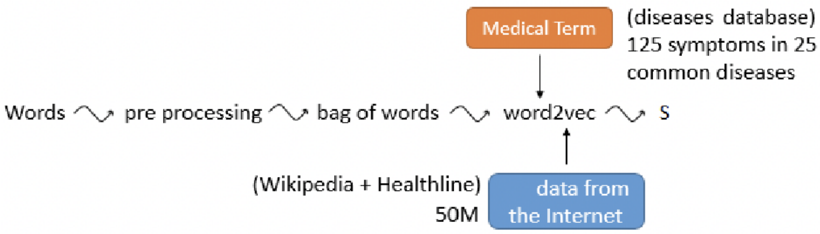

Figure 3 The algorithmic process of the analysis engine in Mandy

---

Procedure 1 (P − N)MFC

Input: positive symptom set A_+ and negative symptom set A_-.
Output: Hypothesis H
1: for every $d_i \in S$ do

2: Set $n_i^+ \leftarrow |A_+ \cap A_{d_i}|$.

3: Set $n_i^- \leftarrow |A_- \cap A_{d_i}|$.

4: Set $\sigma_i \leftarrow (n_i^+ - n_i^-)$.

5: $\rhd$ σ_i is the similarity value of the patient's symptoms with d_i.

6: end for
7: Output the set of diseases $d_i \in S$ that has the top-k highest σ_i value with their corresponding σ_i.

---

3. If $f(d)$ does not contain any symptom not in A_+, the system will analyze patient's input, then choose the most similar symptom in our standard corpus, and use it in the next question.

4. Once the system has got enough information from the patient, it will generate a diagnosis result, list top-most possible diseases which are related to the patient's symptoms.

4.3 Prototype Development

We deploy a prototype of Mandy on an

Amazon Web Services Cloud[12]. It provides services for both the mobile app (see Fig. 4) and the website app.

Knowledge about symptoms and diseases is constructed based on external sources[13]. In this proof-of-concept, we select 25 common diseases. The dataset for Word2Vec to train a word embedding consists of crawled entries from the Internet. Firstly, on Wikipedia[14], the crawler dredges data from the main page of "disease" and visit each medical terminology using hyperlinks. To collect more colloquial sentences, we also crawled data from Healthline[15]. The collected dataset contains approximately 20,000 web pages on Wikipedia and about 10,000 web pages on Healthline with a size of ≈50 MB. Most diseases and symptoms have synonyms or equivalent terminologies. When choosing words to be included in our corpus, we base our decisions on Google Books Ngram Viewer[16] to identify the most frequently used terms in the last 50 years.

4.4 Performance Evaluation

We extracted case studies from a standard medical textbook which contains numerous real-life patient complaints with suggested diagnosis (Stern et al. 2014). Each case involves a patient and a natural language description of their medical conditions. The case is also provided a medical hypothesis as the result of differential diagnosis, which can be viewed as ground truth results. We evaluate the performance of our proof-of-concept on four randomly selected disease categories: Chest Pain, Respiratory Infections, Headache and Dizziness. From these categories, we investigate the result of our system on 11 case studies.

Example 3 *The following provides an example. "Mr. W is a 56-year-old man who comes to your office with chest pain. Mr. W comes in regularly for management of hypertension and diabetes, both of which are under good control. He has been having symptoms since just after his last visit 4 months ago. He feels squeezing, substernal pressure while climbing stairs to the elevated train he rides to work. The pressure resolves after about minutes of rest. He also occasionally feels the sensation during stressful periods at work. It is occasionally associated with mild nausea and jaw pain."*

In this case study, the patient Mr. W complained that he felt chest pain with squeezing, sub-sternal pressure while climbing stairs. The only symptom recognized is chest pain. The diagnostic hypotheses including stable angina, GERD, and Musculoskeletal disorders from the book (Stern et al., 2014) are shown in Table 1.

a. **Evaluating the Analysis Engine**. Word embedding allows the analysis engine to extract symptoms from the patients' natural language input, even when they do not appear in the input text verbatim. We first evaluate the effectiveness of the Word2Vec algorithm in facilitating this task. Recall that the output of the analysis engine is a set $X \subseteq A$ of possible symptoms of the patient. In our test, we take the set X computed from the natural language

---
[12] https://aws.amazon.com/
[13] online databases such as http://www.diseasesdatabase.com
[14] https://en.wikipedia.org/
[15] http://www.healthline.com/
[16] https://books.google.com/ngrams

description of each case study. We then compare X with the set of symptoms $X' \subseteq A$ that appear in the input text verbatim.

Example 4 *For Mr. W described in Example 3, the set X' contains only the standard symptom "chest pain". With the help of Word2Vec, however, the analysis engine is able to identify a number of other possible symptoms by looking up relevant words that appear in the input description. Below are the top-matching*

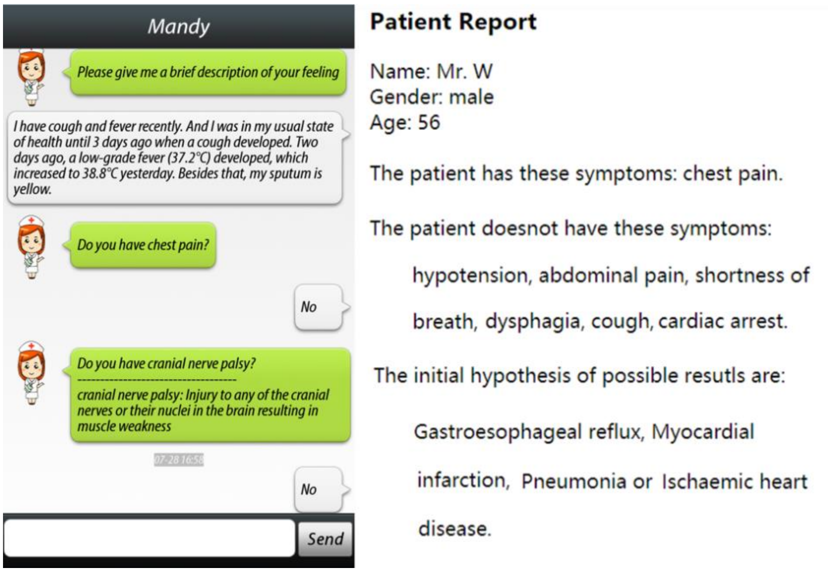

Figure 4: Left: The app user interface; Whenever users encounter obscure medical terms, the relevant from Merriam-Webster Dictionary can be viewed by clicking the dialog box. Right: The generated initial interview outcome report

Table 1 Diagnostic hypotheses for Mr. W

Diagnostic Hypotheses	Clinical Clues	Important Tests
Leading Hypothesis		
Stable angina	Substernal chest pressure with exertion	Exercise tolerance test Angiogram
Active Alternative— Most Common		
GERD*	Symptoms of heartburn, chronic nature	EGD* Esophageal pH monitoring
Active Alternative		
Musculoskeletal disorders	History of injury or specific musculoskeletal chest pain syndrome	Physical exam Response to treatment

** EGD, esophagogastroduodenoscopy; *GERD, gastroesophageal reflux disease.*

standard symptoms with the words "chest pain" and "nausea" from the input text.

chest pain: *abdominal pain, muscle weakness*

nausea: *abdominal pain, constipation, nausea and vomiting*

In this way, the analysis engine is able to come up with five symptoms in X from Mr. W's description: "chest pain, abdominal pain, muscle weakness, constipation, nausea and vomiting". This helps the hypothesis operator (Module II) to narrow down potential hypotheses for the patient.

***Example 5** Another case study includes the following description: "Mr. D. is a 29-year-old white man who complains of dizziness. Detailed questioning reveals that he has had a constant spinning sensation for the last several weeks. Although head movement exacerbates the symptom, it is persistent even when he is still. He has a prior history of migraines for several years. Vertigo has never preceded or accompanied the headache. HEENT exam reveals horizontal nystagmus on leftward and rightward gaze that lasts 1-2 minutes. The nystagmus does not fatigue with repetition of the maneuver."*

For this patient, no standard symptom is found from the description above. Never the less, with Word2Vec the analysis engine is able to pick up three standard symptoms that are relevant to the input text:

dizziness: *abdominal pain, nausea and vomiting*

migraines: *headache*

The symptoms identified by Word2Vec are "abdominal pain", "nausea and vomiting" and "headache" which help the hypothesis operator to derive potential hypotheses.

In all 11 case studies, the Word2Vec-based analysis engine is able to extract significantly more possible standard symptoms from the input text as compare to the methods of matching the words exactly. Table 2 lists the number of symptoms extracted using both methods. The column X' lists the number of symptoms that appear in the patient description (which can be obtained through exact word matching); The column X is the number of symptoms recognized by the analysis engine. Notice that without the word embedding, in all cases there are at most 2 words in the description that match standard symptoms verbatim. Without the use of word embedding such as Word2Vec, the analysis engine would not produce a meaningful result.

Table 2 Comparison between the numbers of standard symptoms found by word matching ($|\boldsymbol{X}'|$) and by the analysis engine with Word2Vec ($|\boldsymbol{X}|$) in 11 case studies

	Mr.D	Mr.H	Mr.J	Mr.J2	Mr.M	Mr.P	Mr.W	Mrs.G	Mrs.L	Mrs.P	Ms.L
$\lvert X' \rvert$	0	2	1	2	1	1	1	1	1	2	1
$\lvert X \rvert$	3	4	5	8	4	4	5	5	4	5	3

b. Evaluating the Generated Questions. Mandy is intended to communicate with the patients just like a real healthcare staff. An ideal intake interviewer should pose a list of personalized questions that truthfully reflect the medical conditions of the patient and lead to meaningful information for their treatment. Thus, the questions generated by Mandy during an interview amounts to a crucial criterion for its effectiveness.

After the analysis engine extracts standard symptoms, we input only the first symptom to the system and check if the system can generate high-quality questions. We regard the questions which covered the other symptoms as "high-quality" if they are sufficient and important for the doctors to come up with the hypothesis list.

Example 6 *One case study includes the following patient description: "Mrs. G is a 68-year-old woman with a history of hypertension who arrives at the emergency department by ambulance complaining of chest pain that has lasted 6 hours. Two hours after eating, moderate (5/10) chest discomfort developed. She describes it as a burning sensation beginning in her mid chest and radiating to her back. She initially attributed the pain to heartburn and used antacids. Despite multiple doses over 3 hours, there was no relief. Over the last hour, the pain became very severe (10/10) with radiation to her back and arms. The pain is associated with diaphoresis and shortness of breath. The patient takes enalapril for hypertension. She lives alone, is fairly sedentary, and smokes 1 pack of Cigarettes each day. She has an 80 pack year smoking history." The symptoms extracted from the text are* ***severe chest pain*** *associated with* ***diaphoresis*** *and* ***shortness of breath****. To evaluate the generated questions, we only provide "severe chest pain", and see if "diaphoresis" and "shortness of breath" will be asked by* Mandy.

After we input ***"severe chest pain****" as the answer to the first question,* Mandy *generated the following interaction. The answers to the questions were obtained from understanding the text description above:*

Mandy: Do you have dysphagia? -No
Mandy: Do you have hypotension? -No
Mandy: Do you have cardiac arrest? -No
Mandy: Do you have hyperhidrosis? -Yes
Mandy: Do you have fever? -No
Mandy: Do you have abdominal pain? -No
Mandy: Do you have shortness of breath? -Yes
Mandy: Do you have nausea and vomiting? -No
Mandy: Do you have productive cough? -No
Mandy: Do you have any other symptoms? -No

Among the 9 questions symptoms, two of them match exactly as our expected questions. Thus, we evaluate the accuracy as: **Question** $\boldsymbol{Accuracy} = matched\ questions \div expected\ questions = 2/2 = 100\%$

Using the same approach, we calculate question accuracy for six test cases (the other five test cases are all single symptom cases, so they cannot be used to evaluate the question accuracy). See Table 4. Among the six cases, two are from each of Chest Pain and Respiratory Issues, and a single case is from each of Dizziness and Headache. Besides the case for Dizziness which only asks 2 high-quality questions out of the expected 3 ones, the question accuracies for the other cases are all 100%.

c. The Performance of the Diagnosis Module. Another natural evaluation criterion is the diagnosis capability of Mandy. For this, we input the entire paragraph of patient description into the system as the answer to the first question. We then answer subsequent questions manually based on the understanding of the case description. When the system has no more questions for the patient, we check if the output hypothesis list from the system matches with the ground truth hypotheses from the book.

Example *7 For the case of Mr. W described in Example 3, the hypotheses report from our system shows that one out of the four hypotheses is matched with the guide book (GERD). Another hypothesis "Myocardial infarction" (MI) from our system shares the same disease category with "stable angina" from the guide book. We regard MI as correct because it is close enough and "stable angina" does not exist in our current disease corpus.*

Therefore, we conclude that the final accuracy of our system for this case is:

$$\boldsymbol{Prediction\ Accuracy} = matched\ hypotheses\ from\ our\ system \div diagnostic\ hypotheses\ in\ guide\ book = 2/3 = 67\%$$

To further evaluate our proof-of-concept, we input Mr. W's case on two well-known existing medical chatbots Your.MD and HealthTap from the Facebook Messenger Bots Platform. The conversations are shown in Fig. 5 and Fig. 6. Even including "chest pain" in the description, the results provided by HealthTap were not convincing. Similarly, after Your.MD checked 30 symptoms, the two conclusions were far from the correct one. On this test case, Mandy *clearly outperforms these existing chatbots as the questions are related to the symptoms and the hypotheses list also make sense.*

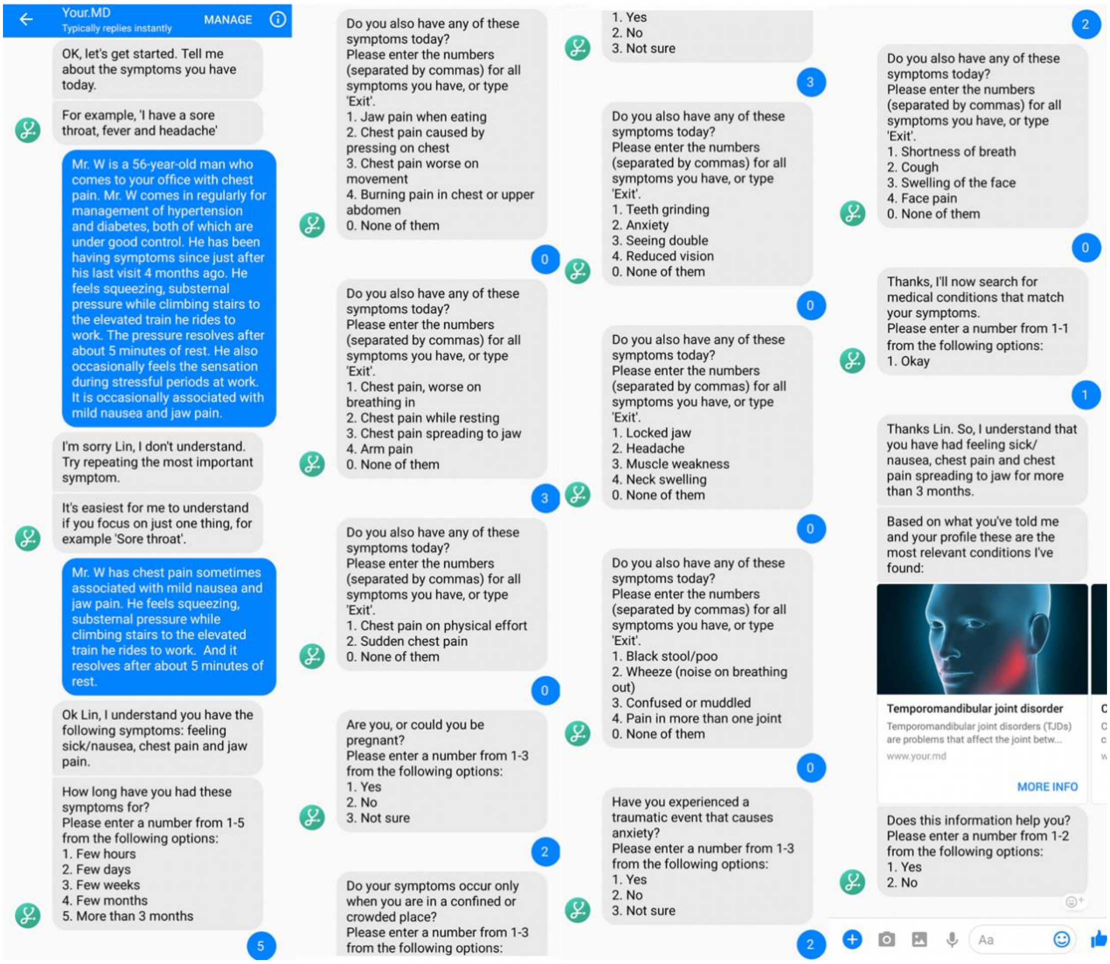

Figure 5 Mr. W's case on Your.MD

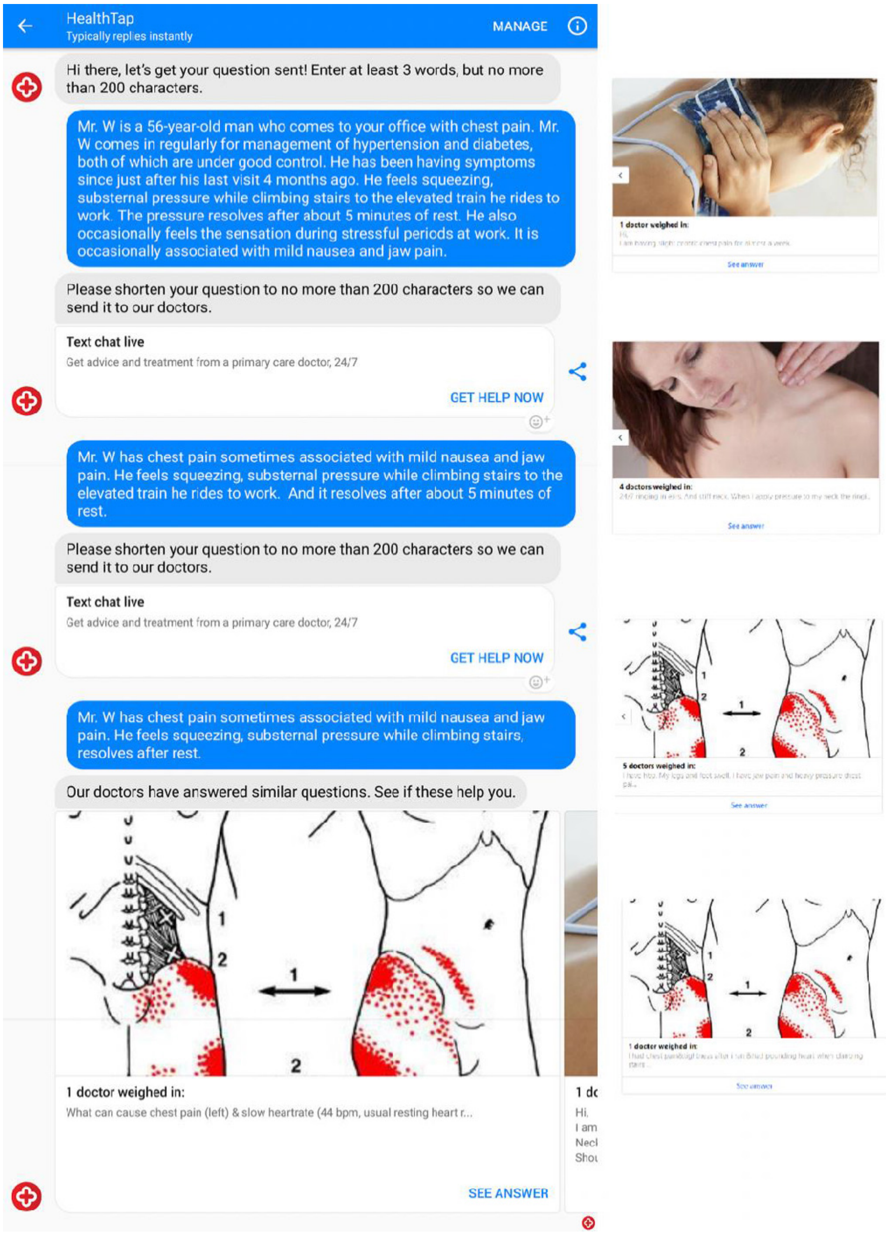

Figure 6 Mr. W's case on HealthTap

Example 8 *Another case study includes the following patient description: "Mr. H is a 31-year-old man, previously in excellent health who arrives at the emergency department complaining of chest pain. He reports that the pain began 10 days earlier. It was initially mild but has become more severe. The pain is accompanied by mild cough and shortness of breath. Five days earlier, he had come to the emergency department and musculoskeletal chest* *pain was diagnosed; he was given nonsteroidal anti-inflammatory drugs (NSAIDs) and discharged. Since the pain has become more severe, it has become pleuritic. He says it is located over the right lateral lower chest wall. His dyspnea is still only mild. He also has noted low-grade fevers with temperatures running about 38 ◦C."*

The generated questions given the above description are:

Mandy: Do you have hypotension? -No
M: Do you have bronchospasm? -No
M: Do you have productive cough? -No
M: Do you have hypotension? -No
M: Do you have hepatomegaly? -No
M: Do you have dysphagia? -No
M: Do you have cardiac arrest? -No
M: Do you have dizzy? -No
M: Do you have bronchial breathing? -No
M: Do you have earache? -No
M: Do you have cachexia? -No
M: Do you have episcleritis? -No
M: Do you have suppurative otitis media? -No
M: Do you have any other symptoms? -No

The symptoms extracted from the text are shortness of breath, chest pain, fever, and cough. The generated hypotheses are "COPD, Gastroesophageal reflux, Myocardial infarction, Acosta syndrome, Pneumonia, Upper respiratory tract infection, AIDS, Tuberculosis, Rheumatoid arthritis, Influenza, Asthma, Measles or Ischaemic heart disease". These results match well with those given in Stern et al., 2014); See Table 3.

Table 3 Diagnostic hypotheses for Mr. H

Diagnostic Hypotheses	Clinical Clues	Important Tests
Leading Hypothesis		
Pleural effusion or pneumonia	Cough and shortness of breath with pleural effusion	Chest radiograph Thoracentesis for associated physical exam findings
Active Alternative - Most Common		
Pericarditis	Pain relieved by leaning forward Friction rub, ECG changes	ECG Echocardiogram
Active Alternative - Must Not Miss		
Pulmonary embolism	Risk factors Tachycardia	Ventilation-perfusion scan Helical CT Pulmonary angiogram
Other Alternative		
Subdiaphragmatic abscess	intra-abdominal process Fevers	Abdominal ultrasound CT

Following the same approach, we calculate all the prediction accuracy for the 11 test cases and the result is shown in Table 4. The low prediction accuracies for Dizziness and Headache are mainly caused by the lack of training data and knowledge in brain diseases in our system. This can be improved in a future update of the proof-of-concept.

Table 4 Question and Prediction accuracy of Mandy over the case studies

Disease category	Question accuracy	Prediction accuracy
Respiratory issues	100%	100%
Chest Pain	100%	64%
Headache	100%	25%
Dizziness	66.7%	14%

5. Conclusion and Future Work

In this paper, we describe a conceptual framework to design and implement a model of iterative inquiry for an automated information broker that bridges clients with a service provider. The developed automated agent should understand natural language input, reasons to generate hypothesis and refine the hypothesis with repeated rounds of question-answering. Emphasis has been put on describing a medicare chatbot system Mandy based on our framework. Mandy handles patient intake in a primary care facility and provide assistance to both patients and doctors. We use Word2Vec to analyze natural language input in this particular domain which works well according to our evaluation experiments.

The framework can be used to realize a range of similar applications where information brokerage is of concern. For example, when developing a helpdesk agent to handle incoming students inquiries at a university, the set of attributes may be keywords/key-phrases that are relevant to the issues of the students, and the set of the services would be contacts and locations of different departments and of various student services. When developing a shopping mall information desk agent to serve consumer demands, the set of attributes may be features that represent the customers' intensions, i.e., descriptive terms of different products, while the set of services include the shops. A natural future work is to deploy systems in the other contexts where the framework could be of use.

The current prototype of Mandy is still very elementary in that it can handle only limited medical-related conditions due to a very small knowledge base. To develop it to become an industry-level product, one would need to enrich its knowledge base. This includes not only an expanded set of diseases but also an integrating model to incorporate the domain knowledge of biomedical terminologies and ontologies, such as Unified Medical Language System (UMLS) (Bodenreider, 2004), LexGrid (Pathak et al., 2009), and the International Classification of Diseases (ICD)[17], that are able to represent the semantics of a wider range of natural language inputs.

One may also notice a limitation with the current system where Mandy can only ask the patient yes/no questions regarding some symptoms. This is due to the rather simple design

17 WHO — International Classification of Diseases. http://www.who.int/classifications/icd/en/

of the question generator module. A more complex design would allow it to generate more realistic and varies forms of questions so that more information can be obtained from the patient throughout the conversation.

Our framework is general in that it has room to further improve the functionality. For example, one may add a fourth module, a profile learner, which creates a profile for each client. This module will evaluate the output of an iterative inquiry process and use this to build a personalized model for a specific client. In this way, the system would become adaptive to individual clients which may provide better end-user experience.

6. Acknowledgments

The first author is partially funded by a scholarship offered by Precision Driven Health in New Zealand, a public-private research partnership aimed at improving health outcomes through data science. Initial progress of the research was reported in the PDH & Orion HealthBlog https://orionhealth.com/global/knowledge-hub/blogs/meet-mandy-an-intelligent-and-interactive-medicare-system/ (Ni et al., 2017)

The authors would like to thank the anonymous referees for the constructive feedback to improve the quality of the paper.

References

[1] Ahmad, F., Hogg-Johnson, S., Stewart, D. E., Skinner, H. A., Glazier, R. H., and Levinson, W. (2009). Computer-assisted screening for intimate partner violence and controla randomized trial. Annals of internal medicine, 151(2):93–102.

[2] Araz, O. M. (2013). Integrating complex system dynamics of pandemic influenza with a multi-criteria decision making model for evaluating public health strategies. Journal of Systems Science and Systems Engineering, 22(3):319–339.

[3] Bakker, M. and Tsui, K.-L. (2017). Dynamic resource allocation for efficient patient scheduling: A data-driven approach. Journal of Systems Science and Systems Engineering, 26(4):448–462.

[4] Bendig, A. W. (1953). Twenty questions: an information analysis. Journal of Experimental Psychology, 46(5):345.

[5] Bengio, Y., Ducharme, R., Vincent, P., and Jauvin, C. (2003). A neural probabilistic language model. Journal of machine learning research, 3(Feb):1137–1155.

[6] Bernstein, S. L., Aronsky, D., Duseja, R., Epstein, S., Handel, D., Hwang, U., McCarthy, M., John McConnell, K., Pines, J. M., Rathlev, N., et al. (2009). The effect of emergency department crowding on clinically oriented outcomes. Academic Emergency Medicine, 16(1):1–10.

[7] Bickmore, T. and Giorgino, T. (2006). Health dialog systems for patients and consumers. Journal of biomedical informatics, 39(5):556–571.

[8] Bickmore, T. W., Pfeifer, L. M., Byron, D., Forsythe, S., Henault, L. E., Jack, B. W., Silliman, R., and Paasche-Orlow, M. K. (2010). Usability of conversational agents by

patients with inadequate health literacy: evidence from two clinical trials. Journal of health communication, 15(S2):197–210.

[9] Black, L.-A., McTear, M., Black, N., Harper, R., and Lemon, M. (2005). Appraisal of a conversational artefact and its utility in remote patient monitoring. In Computer-Based Medical Systems, 2005. Proceedings. 18th IEEE Symposium on, pages 506–508. IEEE.

[10] Bobrow, D. G. (1964). A question-answering system for high school algebra word problems. pages 591–614. ACM.

[11] Bodenreider, O. (2004). The unified medical language system (umls): integrating biomedical terminology. Nucleic acids research, 32(suppl 1):D267–D270.

[12] Caley, M. and Sidhu, K. (2011). Estimating the future healthcare costs of an aging population in the uk: expansion of morbidity and the need for preventative care. Journal of Public Health, 33(1):117–122.

[13] Chen, P., Cheung, Y., Lee, V. C., and Hart, A. (2016). Knowledge sharing via informal communities in a globally distributed organization. In International Symposium on Knowledge and Systems Sciences, pages 30–43. Springer.

[14] Clancey, W. J. and Letsinger, R. (1982). NEOMYCIN: Reconfiguring a rule-based expert system for application to teaching. Department of Computer Science, Stanford University.

[15] Dang, T.-T. and Ho, T.-B. (2017). Sequence-based measure for assessing drug-side effect causal relation from electronic medical records. In International Symposium on Knowledge and Systems Sciences, pages 53–65. Springer.

[16] De Vine, L., Zuccon, G., Koopman, B., Sitbon, L., and Bruza, P. (2014). Medical semantic similarity with a neural language model. In Proceedings of the 23rd ACM international conference on conference on information and knowledge management, pages 1819–1822. ACM.

[17] Delichatsios, H. K., Friedman, R. H., Glanz, K., Tennstedt, S., Smigelski, C., Pinto, B. M., Kelley, H., and Gillman, M. W. (2001). Randomized trial of a “talking computer” to improve adults’ eating habits. American Journal of Health Promotion, 15(4):215–224.

[18] Di Somma, S., Paladino, L., Vaughan, L., Lalle, I., Magrini, L., and Magnanti, M. (2015). Overcrowding in emergency department: an international issue. Internal and emergency medicine, 10(2):171–175.

[19] Dobbie, G. and Ross, K. (2017). Precision driven health: A new zealand research partnership. International Journal of Integrated Care, 17(3).

[20] Edgren, L. (2006). Health consumer diversity and its implications. Journal of Systems Science and Systems Engineering, 15(1):34–47.

[21] Farzanfar, R., Frishkopf, S., Migneault, J., and Friedman, R. (2005). Telephone-linked care for physical activity: a qualitative evaluation of the use patterns of an information technology program for patients. Journal of Biomedical Informatics, 38(3):220–228.

[22] Ferrucci, D., Brown, E., Chu-Carroll, J., Fan, J., Gondek, D., Kalyanpur, A. A., Lally, A., Murdock, J. W., Nyberg, E., Prager, J., et al. (2010). Building watson: An overview of the

deepqa project. AI magazine, 31(3):59–79.

[23]Green Jr, B. F., Wolf, A. K., Chomsky, C., and Laughery, K. (1961). Baseball: an automatic question-answerer. pages 219–224. ACM.

[24]Herzog, O. and Rollinger, C.-R. (1991). Text understanding in LILOG: integrating computational linguistics and artificial intelligence: final report on the IBM Germany LILOG-Project. Springer.

[25]High, R. (2012). The era of cognitive systems: An inside look at ibm watson and how it works. IBM Corporation, Redbooks.

[26]Hubal, R. C. and Day, R. S. (2006). Informed consent procedures: An experimental test using a virtual character in a dialog systems training application. Journal of Biomedical Informatics, 39(5):532– 540.

[27]Hunt, D. L., Haynes, R. B., Hanna, S. E., and Smith, K. (1998). Effects of computer-based clinical decision support systems on physician performance and patient outcomes: a systematic review. Jama, 280(15):1339–1346.

[28]Jiang, B., Tang, J., and Yan, C. (2017). Outpatient capacity allocation considering adding capacity to match high patient demand. Journal of Systems Science and Systems Engineering, 26(4):487–516.

[29]Jones, P., Chalmers, L., Wells, S., Ameratunga, S., Carswell, P., Ashton, T., Curtis, E., Reid, P., Stewart, J., Harper, A., et al. (2012). Implementing performance improvement in new zealand emergency departments: the six hour time target policy national research project protocol. BMC health services research, 12(1):45.

[30]Khoury, M. J. and Ioannidis, J. P. (2014). Big data meets public health. Science, 346(6213):1054– 1055.

[31]Lipkin, M., Quill, T. E., and Napodano, R. J. (1984). The medical interview: a core curriculum for residencies in internal medicine. Annals of Internal Medicine, 100(2):277–284.

[32]Liu, Y., Ge, T., Mathews, K., Ji, H., and McGuinness, D. (2015). Exploiting task-oriented resources to learn word embeddings for clinical abbreviation expansion. Proceedings of BioNLP 15, pages 92–97.

[33]Martínez-Pérez, B., de la Torre-Díez, I., López-Coronado, M., Sainz-De-Abajo, B., Robles, M., & García-Gómez, J. M. (2014). Mobile clinical decision support systems and applications: a literature and commercial review. Journal of medical systems, 38(1), 4.

[34]McFillen, J. M., O'Neil, D. A., Balzer, W. K., and Varney, G. H. (2013). Organizational diagnosis: An evidence-based approach. Journal of Change Management, 13(2):223–246.

[35]Mikolov, T., Chen, K., Corrado, G., and Dean, J. (2013a). Efficient estimation of word representations in vector space. arXiv preprint arXiv:1301.3781.

[36]Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., and Dean, J. (2013b). Distributed representations of words and phrases and their compositionality. In Advances in neural information processing systems, pages 3111–3119.

[37]Miller, R. A., Pople Jr, H. E., and Myers, J. D. (1982). Internist-i, an experimental computer-based diagnostic consultant for general internal medicine. New England

Journal of Medicine, 307(8):468–476.

[38]Moskvina, A. and Liu, J. (2016a). How to build your network? a structural analysis. In Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence, pages 2597–2603. AAAI Press.

[39]Moskvina, A. and Liu, J. (2016b). Integrating networks of equipotent nodes. In International Conference on Computational Social Networks, pages 39–50. Springer.

[40]Ni, L., Lu, C., Liu, N., and Liu, J. (2017). Mandy: Towards a smart primary care chatbot application. In International Symposium on Knowledge and Systems Sciences, pages 38–52. Springer.

[41]Nikfarjam, A., Sarker, A., O'Connor, K., Ginn, R., and Gonzalez, G. (2015). Pharmacovigilance from social media: mining adverse drug reaction mentions using sequence labeling with word embedding cluster features. Journal of the American Medical Informatics Association, 22(3):671–681.

[42]Pathak, J., Solbrig, H. R., Buntrock, J. D., Johnson, T. M., and Chute, C. G. (2009). Lexgrid: a framework for representing, storing, and querying biomedical terminologies from simple to sublime. Journal of the American Medical Informatics Association, 16(3):305–315.

[43]Pechsiri, C. and Sukharomana, R. (2017). Comparative study of using word co-occurrence to extract disease symptoms from web documents. In International Symposium on Knowledge and Systems Sciences, pages 91–100. Springer.

[44]Ramelson, H. Z., Friedman, R. H., and Ockene, J. K. (1999). An automated telephone-based smoking cessation education and counseling system. Patient Education and Counseling, 36(2):131–144.

[45]Realdi, G., Previato, L., and Vitturi, N. (2008). Selection of diagnostic tests for clinical decision making and translation to a problem oriented medical record. Clinica Chimica Acta, 393(1):37–43.

[46]Richardson, D. B. (2006). Increase in patient mortality at 10 days associated with emergency department overcrowding. The Medical Journal of Australia, 184(5):213–216.

[47]Saygin, A. P., Cicekli, I., and Akman, V. (2000). Turing test: 50 years later. Minds and Machines, 10(4):463–518.

[48]Stern, S., Cifu, A., and Altkorn, D. (2014). Symptom to Diagnosis An Evidence Based Guide. McGraw Hill Professional.

[49]Taber, J. M., Leyva, B., and Persoskie, A. (2015). Why do people avoid medical care? a qualitative study using national data. Journal of general internal medicine, 30(3):290–297.

[50]Tien, J. M. and Berg, D. (2003). A case for service systems engineering. Journal of systems science and systems engineering, 12(1):13–38.

[51]Tien, J. M. and Goldschmidt-Clermont, P. J. (2009). Healthcare: A complex service system. Journal of Systems Science and Systems Engineering, 18(3):257–282.

[52]Victor, L. Y., Buchanan, B. G., Shortliffe, E. H., Wraith, S. M., Davis, R., Scott, A. C., and Cohen, S. N. (1979). Evaluating the performance of a computer-based consultant. Computer programs in biomedicine, 9(1):95–102.

[53]Wang, C. and Tang, X. (2016). The online

debate networks analysis: A case study of debates at tianya forum. In International Symposium on Knowledge and Systems Sciences, pages 140–150. Springer.

[54]Warren, J. R. (1998). Better, more cost-effective intake interviews. IEEE Intelligent Systems and their Applications, 13(1):40–48.

[55]Weizenbaum, J. (1966). Eliza—a computer program for the study of natural language communication between man and machine. Communications of the ACM, 9(1):36–45.

[56]Wilensky, R., Chin, D. N., Luria, M., Martin, J., Mayfield, J., and Wu, D. (1988). The berkeley unix consultant project. Computational Linguistics, 14(4):35–84.

[57]Winograd, T. (1971). Procedures as a representation for data in a computer program for understanding natural language. Technical report.

[58]Woods, W. A. and Kaplan, R. (1977). Lunar rocks in natural english: Explorations in natural language question answering. Linguistic structures processing, 5:521–569.

[59]Xu, J., Zhang, Y., Xu, H., et al. (2015). Clinical abbreviation disambiguation using neural word embeddings. Proceedings of BioNLP 15, pages 171–176.

[60]Yan, B., Chen, Y., and Liu, J. (2017). Dynamic relationship building: exploitation versus exploration on a social network. In International Conference on Web Information Systems Engineering, pages 75–90. Springer.

[61]Yan, B., Liu, Y., Liu, J., Cai, Y., Su, H., and Zheng, H. (2018). From the periphery to the center: Information brokerage in an evolving network. In Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI 2018, July 13-19, 2018, Stockholm, Sweden., pages 3912–3918.

[62]Zhang, W., Jiang, Y., and Yoshida, T. (2016). Deep context identification of deceptive reviews using word vectors. In International Symposium on Knowledge and Systems Sciences, pages 213–224. Springer.

Lin Ni is a Chatbot Developer at the National Institute for Health Innovation (NIHI), the University of Auckland. She received a B.E. degree from Northeastern University (China) in 2010, and a Postgraduate Diploma in Computer Science from the University of Auckland in 2017. Her working experience and research interests include Car Navigation systems, Recommendation systems, Multi-agent systems, and Chatbots.

Jiamou Liu is a lecturer in the Department of Computer Science at the University of Auckland. He was a senior lecturer at the Auckland University of Technology between 2011 and 2015 and a researcher at the Department of Computer Science of Leipzig University from 2009 to 2010. He received a PhD from The University of Auckland in 2010. His research interests include Social Network Analysis, Mutliagent Systems and Artificial Inteligence.