Watson: Beyond Jeopardy!

David Ferrucci *, Anthony Levas, Sugato Bagchi, David Gondek, Erik T. Mueller

IBM T.J. Watson Research Center, United States

A B S T R A C T

This paper presents a vision for applying the Watson technology to health care and describes the steps needed to adapt and improve performance in a new domain. Specifically, it elaborates upon a vision for an evidence-based clinical decision support system, based on the DeepQA technology, that affords exploration of a broad range of hypotheses and their associated evidence, as well as uncovers missing information that can be used in mixed-initiative dialog. It describes the research challenges, the adaptation approach, and finally reports results on the first steps we have taken toward this goal.

© 2012 Elsevier B.V. All rights reserved.

In 2007, IBM Research took on the grand challenge of building a computer system that can perform well enough on open-domain question answering to compete with champions at the game of Jeopardy! In 2011, the open-domain question answering system dubbed *Watson* beat the two highest ranked players in a two-game Jeopardy! match. But, to what degree can the QA technology underlying Watson, called DeepQA, which was tuned for answering Jeopardy! questions, succeed in a dramatically different and extremely specialized domain such as medicine? This paper describes the steps needed to adapt and improve performance in this domain. In addition, whereas Jeopardy! allows only "question in, single answer out" with no explanation, we elaborate upon a vision for an evidence-based clinical decision support system, based on the DeepQA technology, that affords exploration of a broad range of hypotheses and their associated evidence, as well as uncovers missing information that can be used in mixed-initiative dialog.
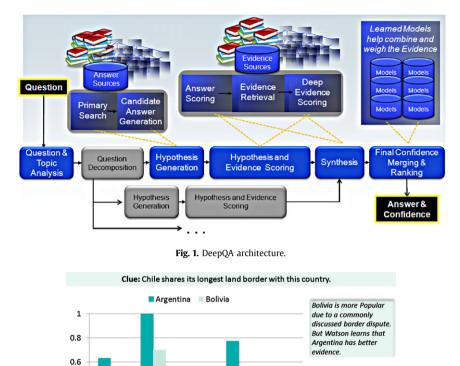
1. DeepQA for Jeopardy!

Jeopardy! is a quiz show that pits three contestants against each other testing their ability to understand and answer rich natural language questions very quickly. These questions often contain complex language, ambiguities, puns, and other opaque references. For any given question, the contestants compete for the first chance to answer via a handheld buzzer.

To be successful at Jeopardy!, players must retain enormous amounts of information, must have strong language skills, must be able to understand precisely what is being asked, and must accurately determine the likelihood they know the right answer. Confidence in the answer is critical, because the first player to buzz in gets the opportunity to answer the question; however, if the player answers incorrectly, the player loses the dollar value associated with the clue. The challenges in the Jeopardy! task are: (1) *Questions come from a broad domain*: Jeopardy! asks questions about hundreds of thousands of things, using rich and varied natural language expressions. (2) *Players must answer questions with high precision and with accurate confidence*: On average, champion players must be able to correctly answer more than 85% of the questions they buzz in for and they must be confident enough to buzz in for at least 70% percent of them. (3) *Answering must be very fast*: Winning players must quickly determine an accurate confidence in a correct answer and buzz in quickly enough to beat their competitors consistently to the buzz.

---

* Corresponding author.
   E-mail address: ferrucci@us.ibm.com (D. Ferrucci).

Fig. 1. DeepQA architecture.

Fig. 2. Dimensions of evidence defined for Jeopardy!

Over a four year period, the team at IBM developed the Watson system that competed on Jeopardy! and the under-lying DeepQA question answering technology [15]. Watson played many games of Jeopardy! against celebrated Jeopardy! champions and, in games televised in February 2011, won against the greatest players of all time, Ken Jennings and Brad Rutter.

But, DeepQA has application well beyond Jeopardy! Contrary to some popular misconceptions, DeepQA does not map the question to a database of questions and simply look up the answer. DeepQA is a software architecture for analyzing natural language content in both questions and knowledge sources. DeepQA discovers and evaluates potential answers and gathers and scores evidence for those answers in both unstructured sources, such as natural language documents, and structured sources, such as relational databases and knowledge bases. Fig. 1 presents a high-level view of DeepQA architecture [15]. DeepQA has a massively parallel, component-based pipeline architecture [16], which uses an extensible set of structured and unstructured content sources as well as broad range of pluggable search and scoring components that allow integration of many different analytic techniques. Machine-learning is used to learn the weights for combining scores from different scorers. Each answer is linked to its supporting evidence.

DeepQA is informed by extensive research in question answering systems, e.g., [7,23,28]. These systems analyze an input question and generate and evaluate candidate answers using a variety of techniques. DeepQA analyzes an input question to determine precisely what it is asking for and generates many possible *candidate answers* through a broad search of large volumes of content. For each of these candidate answers, a *hypothesis* is formed based on considering the candidate in the context of the original question and topic. For each hypothesis, DeepQA spawns an independent thread that attempts to prove it. DeepQA searches its content sources for evidence that supports or refutes each hypothesis. For each evidence–hypothesis pair, DeepQA applies hundreds of algorithms that dissect and analyze the evidence along different *dimensions of evidence* such as type classification, time, geography, popularity, passage support, source reliability, and semantic related-ness. This analysis produces hundreds of features. These features are then combined based on their learned potential for predicting the right answer. The final result of this process is a ranked list of candidate answers, each with a confidence score indicating the degree to which the answer is believed correct, along with links back to the evidence. Fig. 2 shows the dimensions of evidence for the Jeopardy! clue "Chile shares its longest land border with this country."

Each dimension combines the features produced by many algorithms. Each algorithm uses different resources and algorithmic techniques, each with different precision–recall tradeoffs. To form a consumable set of evidence dimensions, features are grouped according to taxonomy of evidence types (e.g., location and popularity as shown in Fig. 2) we defined. The features are combined and weighed according to the trained machine-learning model in order to assess and display the contribution of each evidence type in producing the final confidence score.

Fig. 2 illustrates a comparative evidence profile highlighting some of the dimensions defined for Watson. Evidence profiles were used by developers for debugging, and we imagine they will be useful for end users in many applications to understand and explore evidence associated with a candidate answer.

2. DeepQA for differential diagnosis

DeepQA's approach to Jeopardy! and the success of Watson suggest a powerful new architecture for reasoning over unstructured content. Traditional expert systems use forward reasoning that follows rules from *data* to *conclusions* or backward reasoning that follows rules from *conclusions* to *data*. To build these systems hand-crafted IF–THEN rules for every bit of domain knowledge are manually developed and maintained by skilled engineers or domain experts. An example of a rule taken from the MYCIN system is: **IF:** The stain of the organism is grampos *and* the morphology of the organism is coccus *and* the growth conformation of the organism is chains, **THEN:** There is suggestive evidence (.7) that the identity of the organism is streptococcus.
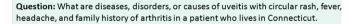
As a result, expert systems are costly and difficult to develop and maintain as new knowledge is discovered. Expert systems are also brittle, because the underlying reasoning engine requires a perfect match between the input data and the existing rule forms. Additionally, not all rule forms can be known in advance for all the forms that input data may take, which further contributes to their brittleness. In contrast to traditional expert systems, DeepQA exploits natural language processing (NLP) and a variety of search techniques to analyze unstructured information to generate likely candidate answers in *hypothesis generation* (analogous to forward chaining). In *evidence collection and scoring* (analogous to backward chaining), DeepQA also uses NLP and search over unstructured information to find evidence for ranking and scoring answers based on natural language content. DeepQA's direct use of readily available knowledge in natural language content makes it more flexible, maintainable, and scalable as well as cost efficient in considering vast amounts of information and staying current with the latest content. What this approach lacks in hand-crafted precision using specific rules, it gains in breadth and flexibility.

In a clinical setting, for example, it can be used to develop a diagnostic support tool that uses the *context* of an input case — a rich set of observations about a patient's medical condition — and generates a ranked list of diagnoses (*differential diagnosis)* with associated confidences based on searching and analyzing evidence from large volumes of content. Physicians and other care providers may evaluate these diagnoses along many different dimensions of evidence that DeepQA has extracted from a patient's electronic medical record (EMR) and other related content sources. For medicine, the dimensions of evidence may include *symptoms, findings, patient history, family history, demographics, current medications*, and many others. Each diagnosis in the differential diagnosis includes links back to the original evidence used by DeepQA to produce its confidence scores and supports the adoption of evidence-based medicine (EBM) "which aims to apply the best available evidence gained from the scientific method to clinical decision making" [14].

When the answers provided by DeepQA are diagnoses of the underlying causes of problems, as in the case of medical diagnosis, then the DeepQA architecture can be thought of as implementing a form of abductive reasoning [25]. As a simple example of abduction, suppose that some piece of structured or unstructured knowledge in the system represents that patients with disease D have symptom S. Then, if the input to the system is that the patient has symptom S, the system will generate the hypothesis that the patient has disease D. The system will then look for evidence to support or refute this hypothesis. For a more complex example of abduction, consider that the system has numerous pieces of knowledge about diseases and their symptoms. Then, given the input that the patient has some set of symptoms, the system's task is to find the best explanation of those symptoms in terms of one or more diseases. The DeepQA architecture does this by generating hypotheses and then, in parallel, evaluating how much evidence supports each hypothesis. In effect, DeepQA is a massive abduction machine.

The use of abduction for medical diagnosis has a long history in the field of artificial intelligence. Pople proposed applying abduction to medical diagnosis and provided algorithms for computing explanations of data (like symptoms) in the context of a collection of axioms (medical knowledge) [27]. Goebel, Furukawa, and Poole presented algorithms for generating diagnoses given medical axioms of the form *disease → symptom* [19]. They further proposed the use of probabilistic logic for preferring one diagnosis over another. They discuss the addition of a probability to *disease → symptom* axioms (sensitivity). Console, Portinale, and Dupré presented an extensive formalization of abductive diagnosis and provided a method for improving the efficiency of solving abduction problems by precompiling knowledge [9].

Fig. 3 shows a proposed set of clinical dimensions of evidence for a patient from Connecticut with a chief complaint of eye pain and inflammation, blurred vision, headache, fever, and circular rash. Each dimension of evidence — findings, demographics, symptoms, and family history — aggregates individual pieces of evidence. A healthcare provider can observe the contribution of each dimension of evidence, as well as drill down into a particular dimension to see the contributing pieces of evidence and provenance information. Accessing this information would help refine their thinking in an evidence-
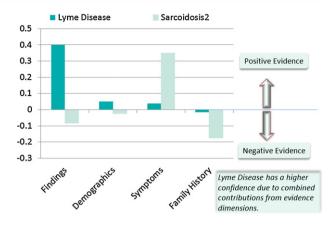
Fig. 3. Evidence dimensions found in differential diagnosis.

based manner. The ability to explore alternative hypothesis (diagnoses), along with the confidence values and associated supporting evidence is a key differentiating feature of DeepQA compared to previous systems.

This general view of DeepQA, as an architecture for building lower cost, more flexible expert-system technology over readily available knowledge, led us to consider applications beyond Jeopardy! and specifically to healthcare. A special edition of the IBM Journal of Research and Development provides a detailed technical description of the Watson system and the underlying DeepQA architecture [10]. In this paper, we motivate the application of DeepQA to healthcare, specifically in clinical decision support. We discuss the first steps we took to adapt DeepQA to the medical domain and how evidence profiles provide a powerful foundation for communicating with healthcare providers.

In the following section, we motivate our work by discussing the problems clinicians face in diagnosis, and review past and current clinical decision support systems along with their strengths and weaknesses. This discussion is followed by our vision of how a system based on DeepQA can become a novel evidence-based decision support tool and, finally we report results on the first steps we have taken toward this goal.

3. Motivation

Improving diagnostic accuracy and speed can directly improve quality of care in patients as well as reduce the overall cost incurred in this process by healthcare systems. Schiff [32] reported diagnostic errors far outnumbering other medical errors by two to four times. Elstein [12] estimated a diagnostic error rate of about 15%, which is in line with findings in a number of autopsy studies [21,33]. Singh and Graber [36] assert that "diagnostic errors are the single largest contributor to ambulatory malpractice claims (40% in some studies) and cost approximately $300,000 per claim on average." Results published from these papers and others highlight the frequency and consequence of diagnostic error in healthcare systems today and motivate the need for approaches that can reduce them.

A recent study by Graber [20] reviews literature related to the causes of diagnostic error and discusses results obtained in a study of 100 "error cases." They report that 65% of these cases had system-related causes and 75% had cognitive-related causes. System errors were "most often related to policies and procedures, inefficient processes, and difficulty with teamwork and communication, especially communication of test results."

Graber [20] reported that cognitive errors were primarily due to "faulty synthesis or flawed processing of the available information." The predominant cause of cognitive error was *premature closure*, defined as "the failure to continue considering reasonable alternatives after an initial diagnosis was reached." Graber additionally identified four more major contributors to the cognitive errors: faulty context generation, misjudging the salience of a finding, faulty detection or perception, and failed use of heuristics.

Graber [20] concluded that the cognitive errors "overwhelmingly reflect inappropriate cognitive processing and/or poor skills in monitoring one's own cognitive processes (metacognition)" and suggested (1) "compiling a complete differential diagnosis to combat the tendency to premature closure," (2) using the "crystal ball experience: The clinician would be told to assume that his or her working diagnosis is incorrect, and asked, What alternatives should be considered?" and (3) augmenting "a clinician's inherent metacognitive skills by using expert systems." Singh and Graber [36] also noted that "clinicians continue to miss diagnostic information … one likely contributing factor is the overwhelming volume of alerts, reminders, and other diagnostic information in the Electronic Health Record (EHR). Better techniques to summarize and present data are needed to enable clinicians to find the proverbial 'needle in the haystack' in the midst of voluminous data."

To compound these problems, published medical information is growing and changing extremely quickly, making the information difficult for the healthcare professional to read, process, and remember. Many emergency medical or critical

situations require very rapid assessment and correct and timely action. These challenges require mentally weighing many variables and exploring alternatives rapidly, which contributes to the cognitive overload inherent in many aspects of this practice.

Our vision for DeepQA is motivated by the problems and suggested solution outlined above. Our approach is to provide a decision support tool that will help the physician overcome the cognitive challenges described above by providing (1) the automatic extraction and presentation of relevant information from the EMR, (2) an extensive differential diagnosis with associated confidences and evidence profiles, and tooling to explore supporting evidence, and (3) a mixed initiative dialog to suggest exploration of missing information and inform decisions based on evidence gathered from vast amounts of structured and unstructured information such as medical texts, encyclopedias, journals, and guidelines. We review some past and current medical diagnostic systems below as a backdrop for a discussion of system related issues and a comparison with our approach.

4. Review of medical diagnostic systems

Diagnosis systems can be classified into systems that use structured knowledge, systems that use unstructured knowledge, and systems that use clinical decision formulas, rules, trees, or algorithms.

4.1. Diagnosis systems using structured knowledge

The earliest diagnosis systems used structured knowledge or classical, manually constructed knowledge bases. The Internist-I system developed in the 1970s used disease-finding relations and disease-disease relations, with associated numbers such as sensitivity — the fraction of patients with a disease who have a finding [24]. The MYCIN system for diagnosing infectious diseases, also developed in the 1970s, used structured knowledge in the form of production rules stating that, if certain facts are true, then one can conclude certain other facts with a given certainty factor [4]. DXplain, developed starting in the 1980s, used structured knowledge similar to that of Internist-I, but added a hierarchical lexicon of findings [2]. The Iliad system developed in the 1990s added more sophisticated probabilistic reasoning. Each disease has an associated a priori probability of the disease (in the population for which Iliad was designed) and list of findings along with the fraction of patients with the disease who have the finding (sensitivity) and the fraction of patients without the disease who have the finding (1 − specificity) [40].

4.2. Diagnosis systems using unstructured knowledge

In 2000, diagnosis systems using unstructured knowledge started to appear. These systems use some structuring of knowledge as well. For example, entities such as findings and disorders may be tagged in documents to facilitate retrieval.

ISABEL uses Autonomy information retrieval software and a database of medical textbooks to retrieve appropriate diagnoses given input findings [31]. Autonomy Auminence uses the Autonomy technology to retrieve diagnoses given findings and organizes the diagnoses by body system [1]. First CONSULT allows one to search a large collection of medical books, journals, and guidelines by chief complaints and age group to arrive at possible diagnoses [17]. PEPID DDX is a diagnosis generator based on PEPID's independent clinical content [26].

4.3. Diagnosis systems using clinical rules

Clinical decision rules have been developed for a number of disorders, and computer systems have been developed to help practitioners and patients apply these rules. The CaseWalker system uses a four-item questionnaire to diagnose major depressive disorder [5]. The PKC Advisor provides guidance on 98 patient problems such as abdominal pain and vomiting [29].

4.4. Strengths and limitations of current systems

The strengths of current diagnosis systems are that they can improve clinicians' diagnostic hypotheses [18] and can help clinicians avoid missing important diagnoses [30]. But, current diagnosis systems aren't widely used [3] for the following reasons: (1) They aren't integrated into the day-to-day operations of health organizations [8,34]. A patient may be seen by many different healthcare workers, and patient data may be scattered across many different computer systems in both structured and unstructured form. (2) They are difficult to interact with [3,34]. Entry of patient data is difficult, the list of diagnostic suggestions may be too long, and the reasoning behind diagnostic suggestions is not always transparent. (3) They aren't focused enough on next actions; they don't help the clinician figure out what to do to help the patient [34]. They are unable to ask the practitioner for missing information that would increase confidence in a diagnosis. (4) They aren't always based on the latest, high-quality medical evidence and are difficult to keep up-to-date [35].

5. DeepQA in healthcare

Our goal with DeepQA is to address some of the weaknesses of prior approaches and to help healthcare professionals overcome the cognitive challenges they face in differential diagnosis, treatment, and other aspects of patient care outlined above.

A key differentiating characteristic of DeepQA is its strength in using search and NLP techniques to process knowledge present in natural language content. These techniques can be used to extract relevant information from EMRs to provide the context for solving individual cases. The same techniques used by DeepQA for Jeopardy! may be adapted to generate diagnoses and treatment options and then collect evidence from huge volumes of information to support or refute those diagnoses and treatments.

The ability to effectively process unstructured content found in medical resources and EMRs allows the practitioner to work with the most current knowledge available and reduces the burden associated with reading and synthesizing vast amounts of data stored in a patient record. It also helps ensure that the evidence provided in support of a set of possible solutions is readable and consumable by human users because the content is typically created by other experts in natural language rather than by knowledge engineers in formal rules.

We started to envision how DeepQA can be used in interaction with healthcare professionals. Physicians we have interviewed all stress the need for ease of use in medical decision support systems, especially those that are used during a patient encounter. System input must be minimal and efficient, and information provided must be unobtrusive and relevant. Our goal is to minimize the input required, by automating the extraction of EMR information relevant to the current situation and providing information at a glance as new suggestions are obtained. Standing queries for diagnosis or treatment will be running as a background process, further minimizing the input required. A history of the practitioner's interaction with the system on a particular case will provide a context for future interactions. This interaction will support system-generated suggestions as well as give practitioners the opportunity to ask directed natural language medical questions to obtain additional information they seek and will help them overcome many of the cognitive challenges discussed above, such as premature closure and faulty context generation.

Fig. 4 illustrates a design for a user interface for clinical decision support. The left side presents information (labeled Factors) related to the patient's chief complaint, which would be automatically extracted from an EMR. Each factor is shown under its respective dimension of evidence, i.e., symptoms, findings, family history, and demographics. The top left side of Fig. 4 shows the current differential diagnosis (only the top 3 of a long list are shown in this example) and associated confidence values for each candidate. A practitioner can select a particular candidate diagnosis — in this case Uveitis is selected — and explore the contribution of each dimension of evidence. A particular dimension (e.g. Symptoms) is selected revealing the contributing pieces of evidence as well as where they came from in the *Sources* tab area. The complete text is accessible via links.

In addition, the Factors tab on the bottom right may be selected to explore factors that are *present* as well as *absent* from the current case as illustrated in Fig. 5. The ability to explore alternative hypotheses (diagnoses), along with the confidence values and associated evidence is another key differentiating feature of DeepQA. This ability to gather evidence surrounding a hypothesis can also be used to discover information that is missing from the current clinical context and can drive mixed-initiative dialog that help clinicians gather additional information and refine their thinking in an evidence based manner. We are beginning to explore this kind of information and interaction, and we believe it will be an essential feature of our clinical decision support system.
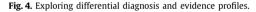
6. Research challenges

Several challenges need to be addressed to apply DeepQA to clinical decision support. We divide them into the challenge of embedding the DeepQA capability into a clinical decision support system and the challenge of adapting the internal components of DeepQA to the medical domain.

The decision support system must be able to extract relevant clinical information from EMR systems. We expect that certain portions of the clinical information such as admission notes, consults, clinical assessments, and discharge summaries will continue to be best expressed and communicated in natural language. Our challenge is to apply natural language and reasoning techniques to extract, relate, and structure this information along a timeline of medical encounters.

Questions in the medical domain bring along a broader context that is described in the patient's medical history. Patient history comprises (1) a description of the chief complaint, (2) a history of the present illness, (3) a review of the major body systems, and (4) social and family history [13]. Using named entity and relation annotators, we need to extract key clinical concepts that form the context for decision support. These include signs, symptoms, findings, active and past diseases, current medications, allergies, demographics, family history and many others. The concepts need to be broad enough to capture the descriptive intent of the clinician. For example, rather than just extracting "heart murmur" as a finding, we should also extract the related modifiers as well: "heart murmur is harsh, systolic, diamond-shaped and increases in intensity with Valsalva." Relations, for example, that indicate a specific family member had a particular disease, or that a symptom is mentioned in negation, need to be accurately captured from the language parse results. Laboratory test results need to be interpreted and evaluated for clinical significance. The extraction of this information from the patient's EMR provides context necessary for hypothesis generation and evaluation performed by DeepQA.

Fig. 4. Exploring differential diagnosis and evidence profiles.

Fig. 5. Factors related to Uveitis diagnosis.

Significant challenges also exist in developing the manner in which the output of DeepQA is presented to the healthcare professionals. A clinical decision support system must help a practitioner overcome cognitive errors discussed above by explaining how a certain hypothesis was generated and what factors were considered in its evaluation. As described in our approach, we must be able to decompose the confidence in a hypothesis into its constituent dimensions of evidence and compare them across multiple competing hypotheses so that practitioners can arrive at their own conclusions.

A useful capability to improve the quality of decision making is to identify the missing information about the patient that detracts from the confidence for a hypothesis, as a whole or along a specific dimension. This missing information offers a set of questions back to the healthcare professional to explore and answer. Significant opportunities for research remain in evaluating the potential *informativeness* of such missing information from the large amounts of information that is not recorded in a patient's EMR. When comparing across competing hypotheses, the missing information can also be evaluated and ranked according to its *discriminitiveness* among these hypotheses. This feedback could focus additional data gathering

activities, such as diagnostic tests, to those that are more likely to confirm or reject hypotheses, sharpening the differential diagnosis in the process.

Significant areas of research remain within the natural language processing capability of DeepQA itself. We are beginning to address these challenges, starting from those within DeepQA's hypothesis generation and verification components. The following section describes our efforts to date and their impact on medical question answering performance.

7. Medical domain adaptation

For our first phase of adaptation, we obtained 5000 medical questions from the American College of Physicians (ACP). They come from a Jeopardy!-like competition, called *Doctor's Dilemma*, that medical interns, residents, and fellows participate in once a year. This set includes questions about diseases, treatments, lab tests, and general facts. Some examples of these questions and associated answers are:

- *The syndrome characterized by joint pain, abdominal pain, palpable purpura, and a nephritic sediment.* Answer: Henoch–Schonlein purpura.
- *Familial adenomatous polyposis is caused by mutations of this gene.* Answer: APC gene.
- *The syndrome characterized by narrowing of the extra-hepatic bile duct from mechanical compression by a gallstone impacted in the cystic duct.* Answer: Mirizzi's syndrome.

We are currently focusing on evaluating performance on medical diagnosis questions from this set. Diagnosis questions generally describe symptoms, findings, and other contextual medical information and require a diagnosis as an answer — thus a good first step towards differential diagnosis. Below, we report our progress in adaptation and the results of our first few experiments.

Applying DeepQA to any new domain requires adaptation in three areas:

1. *Content adaptation* involves organizing the domain content for hypothesis and evidence generation, modeling the context in which questions will be generated.
2. *Training adaptation* involves adding data in the form of sample training questions and correct answers from the target domain so that the system can learn appropriate weights for its components when estimating answer confidence.
3. *Functional adaptation* involves adding new domain-specific question analysis, candidate generation, hypothesis scoring and other components.

7.1. Content adaptation

Content for the medical domain ranges from textbooks, dictionaries, clinical guidelines, and research articles, to public information on the web. There is often a tradeoff between *reliability* and *recency* of information available from these content sources. By using training questions, the machine-learning models in DeepQA can learn what weight to attach to them. Alternatively, the decision maker may choose to do so manually, adjusting the confidence in a hypothesis based on its sources.

The content adaptation process navigates through the chapter and section header hierarchy of textbooks and organizes the information according to the objectives of the system. Given our focus on diagnostic support, we scan the header hierarchy for disease names and keyword variants for their causes, symptoms, diagnostic tests, and treatments. The text content in these sections is then converted into an XML format that information retrieval engines take as input for indexing. The text is further analyzed for identification of medical concepts and their semantic types according to the Unified Medical Language System terminology [37]. This extra information provides for a structured query-based lookup to complement text-based information-retrieval approaches.

We can supplement medical content from standard sources such as textbooks and published guidelines with knowledge available from a variety of online sources. This process of corpus expansion was developed for DeepQA. This uses existing knowledge about a concept, such as a description of symptoms for a given disease, and searches the web for similar passages. This query results in the generation of "pseudo-"documents that contain a broad range of passages that discuss various aspects of the target concept.

We have so far incorporated a modest set of medical content sources focused on internal medicine. These are *ACP Medicine*, *Merck Manual of Diagnosis and Therapy*, *PIER* (a collection of guidelines and evidence summaries), and *MKSAP* (a study guide from ACP). We complement these sources with online medical content.

The content is adapted for four purposes. Information about each disease found in these sources is extracted into "pseudo-documents." First, these are retrieved during document search and the corresponding disease is proposed as a candidate answer. Second, in passage search the entire source content is searched to find relevant passages that match the question. The passages are returned for use by candidate answer generation. Third, during the evidence scoring phase, the content is searched to see if there is textual support for a given candidate answer. Fourth, we have analyzed the content to extract associations between diseases and their symptoms, findings, and tests. This structured knowledge base is used in primary search when we encounter questions looking for a diagnosis.

7.2. Training adaptation

DeepQA relies on machine-learning methods to determine how to weigh the contribution of the various search and scoring components in the question answering pipeline. They use a training set of questions with known correct answers. We randomly selected 1322 Doctor's Dilemma questions for training. Note that we included both diagnosis and non-diagnosis questions in training, which showed better performance on our development set than training on the much smaller set of diagnosis questions alone.

7.3. Functional adaptation

DeepQA defines a general set of processing steps needed in a hypothesis evidencing system as shown in Fig. 1. Conceptually, this pipeline includes analyzing and interpreting a question, searching, generating candidate hypotheses, retrieving supporting evidence, and finally scoring and ranking answers. New analytic components can be easily integrated into each of these steps to meet the requirements of a particular domain. Many of the existing components developed for the Watson core system are domain-independent and therefore reusable. New domains, however, enable new domain-specific resources such as taxonomies, collections of text for capturing knowledge, as well as domain-specific question formulations and reasoning axioms, all of which fit naturally into specific functional areas of DeepQA. We refer to this process as *functional adaptation*. In the following sections we outline some of the main functional areas of DeepQA where members from the Watson research team have begun to perform functional adaptation for the medical domain.

7.4. Adapting to domain-specific taxonomies and reasoning

A valuable type of resource in the medical domain is medical ontologies such as UMLS [37], which contain taxonomies MeSH and SNOMED. Medical taxonomies encode variant phrasings for the same concept (e.g. "age-related hearing loss" is equivalent to "presbycusis") as well as hyponymy relations (e.g. "pyoderma gangrenosum" is a type of "skin disease"). If the system can accurately recognize concepts, these relations may be reasoned over to better evidence hypotheses. The first task discussed below is concept detection, in which the system must accurately map from text as expressed in questions and evidence passages into the taxonomy using entity disambiguation techniques. Afterward, we discuss how, once detected, reasoning techniques may be applied over concepts to better score candidate answers.

7.5. Concept detection

Named entity disambiguation: Accurate named entity detectors exist for the medical domain such as UMLS MetaMap [38]. Although, as many expect, the precise terminology of medical language aids in disambiguation, it turns out there are special challenges in segmentation and disambiguation. This is evident for acronyms (e.g., Liu et al. [22] found that 54% of three letter acronyms were ambiguous considering expansions in UMLS alone) but also for terms like "hypertension" which could be interpreted as "Hypertensive disease" but also as a finding, "Hypertensive adverse event," in the UMLS taxonomy. Furthermore, proper segmentation must be used to identify the appropriate level of specificity (e.g., "carcinoma," "pancreatic carcinoma," or "non-resectable pancreatic carcinoma").

Measurement recognition and interpretation: Lab findings and other numeric measurements are critical in the medical domain. Use of these demands recognition capabilities incorporating context, for instance to identify that "22 y.o." maps to the concept, "Young Adult," or that "320 mg/dL blood glucose" maps to "Hyperglycemia." While in some cases this information may be associated with health records in structured (coded) form, that is not always the case. Furthermore the unstructured medical knowledge sources from textbooks used to generate and score answers are not structured and represent this information only in text or tabular form. We have built a rule-based annotator that identifies measurements and test results as expressed in text. Based on existing guidelines, measurements are interpreted to be normal, high, or low, and mapped using general tables to the corresponding UMLS concept.

Unary relations: Normal, high, and low values may also be expressed lexically (e.g. "elevated T4") and we have trained statistical classifiers [39] and built rule based detectors to identify cases of this. Additionally, we have collected a set of mapping rules to map to specific concepts in UMLS when they exist (e.g. mapping from "blood pressure is elevated" to the "Hypertension" concept). Negation may be considered a unary relation and we have adapted NegEx [6] to work with the DeepQA parser to identify concepts which are negated.

7.6. Reasoning over concepts using taxonomic resources

Domain-specific taxonomic reasoning can be used to evidence correct hypothesis via: concept matching between question and evidence passages, type coercion of answers given the desired answer type, identifying specificity of answer, and equivalent answer merging.

Concept term matching: The synonymy and hyponymy encoded in taxonomies may be directly used to enhance term matching within DeepQA. Term matching is used by the DeepQA passage scorers, which attempt to justify hypotheses using unstructured content. DeepQA uses an ensemble of passage scorers with different precision/recall tradeoffs, ranging from

bag-of-words and subsequence matching to techniques that align predicate argument structures between supporting text passage and question. Each passage scorer contributes a score for each hypothesis–passage pair. The passage scoring framework allows the easy integration of different term matchers, and so it was straightforward to incorporate UMLS taxonomy matching.

Type coercion: DeepQA scores how easily a candidate answer may be "coerced" to the desired lexical answer type of the question. Typing information is available in domain taxonomies [37] as well as extractions from domain text content. Entity disambiguation is used to map candidate answers from text into the medical taxonomies. Lexical answer types (LATs) expressed in the question (e.g. "skin condition," "cause") must also be mapped through predicate disambiguation to types in the taxonomy. While "skin condition" maps directly to concepts in MeSH and SNOMED, LATs like "cause" may map to multiple concepts via a set of predicate mapping rules we have collected. Once both the candidate answer and type have been mapped to concepts in the taxonomy, specialized techniques can produce scores based on ancestry and other metrics over the hyponymy tree to identify if the candidate answer is of the right type.

Answer specificity: Candidate answers may range in generality or specificity. A diagnosis to a high-level disease may not be very useful to a practitioner whereas a diagnosis to a specific disease variant will have a lower probability of being correct. Consider a diagnosis of "bicuspid aortic valve" versus "heart defect." Although either may help lead the user to a useful answer, the level of specificity desired may vary for presentation to specialists or general practitioners. DeepQA includes support for identifying generic classes versus instances, and in medical adaptation we have further added scores using the medical taxonomies to identify the level of specificity of a candidate.

Answer merging: DeepQA uses an ensemble of candidate answer generators that generate candidate answers from passages. These candidate answers may be variants referring to the same concept. By adding an answer merger that uses taxonomies to identify variant forms, the system can merge the evidence for equivalent answers.

7.7. Adapting to domain-specific text collections

As discussed in Section 7.1, the medical domain offers large amounts of domain-specific text. In functional adaptation, this text may be used to build new resources to be used by the system as well as to provide evaluation data for developers to diagnose new refinements necessary for the domain. We discuss two such resources we have constructed thus far, a Symptom KB and an LSA resource, and how they are used in the system. Then we review some of the refinements developed to address particular challenges that arise in medical text.

7.8. Resources mined over medical text

Latent semantic analysis [11] is an unsupervised technique which we used to produce a latent semantic index over our medical corpus. This index loosely captures "topics" as they occur in the corpus. Then, at answer scoring time, an LSA similarity is computed between the terms in the clue and the terms associated with the candidate answer in the LSA index.

Structured symptom matching: While existing medical resources representing sensitivities and specificities can produce a precise probability of a diagnosis given extracted symptoms, these can be difficult to obtain in machine-readable form and keep current and consistent. As a step in this direction, we built an unsupervised resource over our unstructured medical content, where the association between symptoms and diseases was mined to produce a corpus-derived mutual-information-based structured resource representing the informativeness of a symptom for a given condition. This resource is used within DeepQA by looking up conditions associated with extracted symptoms and adding a score for that condition based on the informativeness of the associated symptoms.

7.9. Refinements to handle medical text

Multidimensional passage scoring: A medical question typically represents multiple *factors* describing correct hypotheses. If a heart murmur is described as "harsh, systolic, diamond-shaped and increases in intensity with Valsalva," each of these modifiers ("harsh," "systolic," etc.) may be considered a separate factor and its relationship to the hypothesis could be discovered in a different text passage. We added a rule-based component that segments a question into factors. Then the suite of DeepQA passage scorers is run on passages for each factor, and scores are aggregated over factors via an ensemble of rollup functions (e.g. max, average, etc.), where the functions are selected using feature selection.

Supporting passage discourse chunking: In *supporting passage retrieval*, the DeepQA system performs a passage search for relevant passages containing a candidate answer by using terms from the question and the candidate answer. Then the passages retrieved are scored for the candidate answer using passage scorers. The assumption is that the passage text retrieved is associated with the candidate answer. However, in the medical domain, this assumption is frequently violated. Frequently passages discuss the differentiation of similar presenting conditions, e.g. a passage retrieved for collagenous colitis:

– **Collagenous colitis** and **lymphocytic colitis** are distinguished by the presence or absence of a thickened subepithelial collagen layer. The cause of **microscopic colitis** syndrome is uncertain.
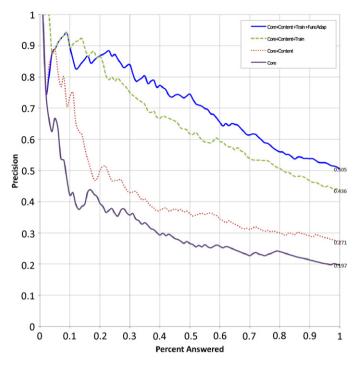
Fig. 6. Precision on ACP Doctor's Dilemma questions.

This passage mentions three distinct forms of colitis. While the association of symptoms with each condition can be understood from the syntactic structure, recall-based passage scorers such as bag-of-words similarity would associate the same score with any of the three colitis mentions. An immediate improvement we implemented was to perform very simple discourse chunking based on which sentences contain the candidate. This produces a discourse-focused supporting passage for scoring alongside the full passage.[1] In the example above, this would avoid the confusion with microscopic colitis syndrome. Of course there remains potential confusion with respect to lymphocytic colitis, which illustrates the need for syntactic scorers and better interpretation of such structures.

By specializing to the forms of evidence available in the medical domain, the domain adaptations discussed above help to realize the vision shown in Fig. 4, wherein evidence is analyzed along medically meaningful dimensions, and where evidence passages relevant to those dimensions are used in support or refutation of hypotheses.

8. Experimental results

Fig. 6 and Fig. 7 show the performance of subsequent stages of domain adaptation for the system evaluated on 188 unseen Doctor's Dilemma diagnosis questions. Fig. 6 evaluates precision and confidence estimation only in terms of the top answer for each question. "Precision" measures the percentage of questions the system gets right for its top answer out of those it chooses to answer. "Percent answered" is the percentage of questions the system is required to answer, which it selects according to its highest estimated confidence score on the top answer. The precision at 100% answered is the "accuracy" of the system. Fig. 7 shows the "Recall@10" which is the percentage of questions for which the correct answer appears in the top 10 ranked answers. We believe this metric to be a more useful target for consideration within decision support settings (e.g., differential diagnosis), where a user may further and interactively evaluate top answers from the system.

We display performance after each stage of domain adaptation. *Core* demonstrates the baseline performance of applying the core DeepQA system, with general-purpose content and models trained on Jeopardy! questions, directly to the Doctor's Dilemma set. With an accuracy of 19% and Recall@10 of 49%, the core system shows a reasonable capability to apply to new domains, especially considering that a wide range of specialized medical content published in textbooks, journals and many other sources was not present. *Core + Content* shows the baseline system with medical content adaptation but Jeopardy!-trained models, which results in a 7% increase in accuracy and a 5% improvement in Recall@10. The largest improvement was obtained by training adaptation, using 1322 Doctor's Dilemma questions for training in *Core + Content + Train*, which

---

[1] We retained full passage scores alongside focused scores because frequently information spans multiple sentences via co-reference, and in some cases larger contexts accommodate our imperfect co-reference detection and discourse chunking.
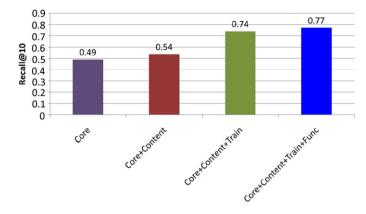
Fig. 7. Recall@10 on ACP Doctor's Dilemma questions.

shows an additional 16% jump in accuracy and a 20% improvement in Recall@10. Finally, *Core + Content + Train + Function*, which includes functional adaptation, shows a 7% improvement in accuracy and a 3% improvement in Recall@10.

Although the largest improvement thus far was due to introducing domain-specific training, we have run experiments that show that the training appears to be saturating and the system will not likely show large gains from the addition of more training data. Instead future improvements will depend largely on functional adaptation which brings its own challenges. Firstly, the general-purpose NLP components included in the core system perform at a respectable level, so that the domain-specific adaptations must improve performance for those aspects that existing components do not currently handle. Second, functional adaptation is a more intensive and difficult process requiring improvements in domain-specific NLP and leveraging medical resources.

It is important to note that a Watson-based clinical decision support system will have very different requirements than the Watson system that competed in Jeopardy! Watson's task in Jeopardy! was to generate a single correct answer in response to a question and to buzz in with that answer if the answer's confidence estimate exceeded a dynamically computed threshold. Watson did this by generating a set of candidate answers (hypotheses) and then collecting and scoring evidence for each answer. The hypothesis with the most compelling evidence was selected as the best answer. In effect, the hypotheses competed within the evidence space. Watson will continue to use this paradigm in clinical decision support. But, in clinical decision support, Watson's task will be to assist healthcare practitioners in evaluating a set of hypotheses. The focus will shift from getting the right answer in the top position to producing a set of likely hypotheses backed by high-quality evidence. The system will help caregivers overcome the cognitive challenges they face by enabling them to interact with comparative evidence profiles and with the evidence to secure more informed decisions. The ability to easily explore evidence, and the quality of the evidence provided, are critically important as well. Our work using the Doctor's Dilemma question set is just a first step in adapting Watson to the medical domain. Ultimately, Watson's success will be based on its ability to integrate effectively into clinical workflow, to improve quality of care, and to reduce costs.

9. Conclusion

Improving diagnostic and treatment accuracy can directly impact the quality of care in patients as well as reduce the overall cost incurred by our healthcare systems. DeepQA defines a powerful new architecture for structuring and reasoning over unstructured natural language content and provides a foundation for developing decision support systems that can address many of the cognitive challenges clinicians face, as well as address some of the weaknesses of prior approaches. We discuss our vision for applying it to extract, structure, and reason over natural language content found in medical textbooks, encyclopedias, guidelines, electronic medical records, and many other sources. We suggest that this technology provides the basis for a novel clinical decision support tool affording valuable assistance in differential diagnosis, exploration of evidence that can justify or refute diagnoses, and mixed-initiative dialog to help clinicians employ evidence-based practice in their daily workflow.

Acknowledgements

References

[1] Autonomy Auminence, http://www.autonomyhealth.com.
[2] G.O. Barnett, J.J. Cimino, J.A. Hupp, E.P. Hoffer, DXplain: An evolving diagnostic decision-support system, JAMA 258 (1) (1987) 67–74.
[3] E.S. Berner, Diagnostic decision support systems: Why aren't they used more and what can we do about it?, in: AMIA Annu. Symp. Proc., 2006, pp. 1167–1168.
[4] B.G. Buchanan, E.H. Shortliffe (Eds.), Rule-Based Expert Systems: The MYCIN Experiments of the Stanford Heuristic Programming Project, Addison–Wesley, Reading, MA, 1984.
[5] D.S. Cannon, S.N. Allen, A comparison of the effects of computer and manual reminders on compliance with a mental health clinical practice guideline, J. Am. Med. Inform. Assoc. 7 (2) (2000) 196–203.
[6] W. Chapman, W. Bridewell, P. Hanbury, G. Cooper, B. Buchanan, A simple algorithm for identifying negated findings and diseases in discharge summaries, J. Biomed. Inform. 34 (5) (2001) 301–310.
[7] C. Clarke, G. Cormack, T. Lynam, Exploiting redundancy in question answering, in: Proceedings of SIGIR, 2001, pp. 358–365.
[8] E. Coiera, Guide to Health Informatics, second edition, Hodder Arnold, 2003.
[9] L. Console, L. Portinale, D.T. Dupré, Using compiled knowledge to guide and focus abductive diagnosis, IEEE Trans. Knowl. Data Eng. 8 (5) (1996) 690–706.
[10] Deep Q&A: This is Watson, IBM J. Res. Develop. 56 (3&4) (May/July 2012).
[11] S. Deerwester, S.T. Dumais, G.W. Furnas, T.K. Landauer, R. Harshman, Indexing by latent semantic analysis, J. Am. Soc. Inform. Sci. 41 (1990) 391–407.
[12] A.S. Elstein, Clinical reasoning in medicine, in: J. Higgs, M.A. Jones (Eds.), Clinical Reasoning in the Health Professions, Butterworth–Heinemann, Woburn, MA, 1995, pp. 49–59.
[13] Evaluation and Management Services Guide, Department of Health and Human Services Centers for Medicare & Medicaid Services, December 2010, ICN: 006764.
[14] Evidence-Based Medicine, http://en.wikipedia.org/wiki/Evidence-based_medicine.
[15] D. Ferrucci, E. Brown, J. Chu-Carroll, J. Fan, D. Gondek, A.A. Kalyanpur, A. Lally, J.W. Murdock, E. Nyberg, J. Prager, N. Schlaefer, C. Welty, Building Watson: An overview of the DeepQA project, AI Magazine (Fall 2010) 59–79.
[16] D. Ferrucci, A. Lally, UIMA: An architectural approach to unstructured information processing in the corporate research environment, Nat. Lang. Eng. 10 (3–4) (2004) 327–348.
[17] First CONSULT, http://www.firstconsult.com.
[18] C.P. Friedman, A.S. Elstein, F.M. Wolf, G.C. Murphy, T.M. Franz, P.S. Heckerling, P.L. Fine, T.M. Miller, V. Abraham, Enhancement of clinicians' diagnostic reasoning by computer-based consultation: A multisite study of 2 systems, JAMA 282 (19) (1999) 1851–1856.
[19] R. Goebel, K. Furukawa, D. Poole, Using definite clauses and integrity constraints as the basis for a theory formation approach to diagnostic reasoning, in: Proceedings of the Third International Conference on Logic Programming, 1986, pp. 211–222.
[20] M. Graber, N. Franklin, R. Gordon, Diagnostic error in internal medicine, Arch. Intern. Med. 165 (2005) 1493–1499.
[21] W. Kirch, C. Schafii, Misdiagnosis at a university hospital in 4 medical eras, Medicine (Baltimore) 75 (1996) 29–40.
[22] H. Liu, A. Lussier, C. Friedman, A study of abbreviations in the UMLS, in: Proceedings of the American Medical Informatics Association Symposium, 2001, pp. 393–397.
[23] D. Moldovan, S. Harabagiu, M. Pasca, R. Mihalcea, R. Girju, R. Goodrum, V. Rus, The structure and performance of an open-domain question answering system, in: Proc. of the 38th Meeting of the Association for Computational Linguistics, 2000, pp. 563–570.
[24] J.D. Myers, The background of INTERNIST-I and QMR, in: Proceedings of ACM Conference on History of Medical Informatics, 1987, pp. 195–197.
[25] C.S. Peirce, Abduction and induction, in: J. Buchler (Ed.), Philosophical Writings of Peirce, Dover, Mineola, NY, 1901.
[26] PEPID, http://www.pepid.com/products/ddx/.
[27] H.E. Pople, On the mechanization of abductive logic, in: Proceedings of the Third International Joint Conference on Artificial Intelligence, 1972, pp. 147–152.
[28] J. Prager, E. Brown, A. Coden, D. Radev, Question answering by predictive annotation, in: Proceedings of the ACM SIGIR Conference on Research and Development in Information Retrieval, 2000, pp. 184–191.
[29] PKC Advisor, http://www.pkc.com/software/advisor/index.aspx.
[30] P. Ramnarayan, G.C. Roberts, M. Coren, V. Nanduri, A. Tomlinson, P.M. Taylor, J.C. Wyatt, J.F. Britto, Assessment of the potential impact of a reminder system on the reduction of diagnostic errors: A quasi-experimental study, BMC Med. Inform. Decis. Mak. 6 (2006) 22.
[31] P. Ramnarayan, A. Tomlinson, A. Rao, M. Coren, A. Winrow, J. Britto, ISABEL: A web-based differential diagnostic aid for paediatrics: Results from an initial performance evaluation, Arch. Dis. Child. 88 (5) (2003) 408–413.
[32] G.D. Schiff, Diagnosing Diagnosis Errors: Lessons from a Multi-Institutional Collaborative Project, Advances in Patient Safety, vol. 2, 2005, pp. 255–278.
[33] K.G. Shojania, E.C. Burton, K.M. McDonald, L. Goldman, Changes in rates of autopsy detected diagnostic errors over time, JAMA 289 (2003) 2849–2856.
[34] T. Shortliffe, Medical thinking: What should we do?, in: Proceedings of Medical Thinking: What Do We Know? A Review Meeting, 2006, http://www.openclinical.org/medicalThinking2006Summary2.html.
[35] I. Sim, P. Gorman, R.A. Greenes, R.B. Haynes, B. Kaplan, H. Lehmann, P.C. Tang, Clinical decision support systems for the practice of evidence-based medicine, J. Am. Med. Inform. Assoc. 8 (6) (2001) 527–534.
[36] H. Singh, M. Graber, Reducing diagnostic error through medical home based primary care reform, JAMA 304 (4) (2010) 463–464, http://dx.doi.org/10.1001/jama.2010.1035.
[37] UMLS, http://www.ncbi.nlm.nih.gov/books/NBK9676/, version 2011AA.
[38] UMLS MetaMap, http://www.nlm.nih.gov/research/umls/implementation_resources/metamap.html, MetaMap version 2010.
[39] C. Wang, J. Fan, A. Kalyanpur, D. Gondek, Relation extraction with relation topics, in: Conf. on Emp. Methods in Natural Language Processing, 2011, pp. 9:1–9:12.
[40] H.R. Warner, P. Haug, O. Bouhaddou, M. Lincoln, H. Warner, D. Sorenson, J.W. Williamson, C. Fan, ILIAD as an expert consultant to teach differential diagnosis, in: Proc. Annu. Symp. Comput. Appl. Med. Care, 1988, pp. 371–376.