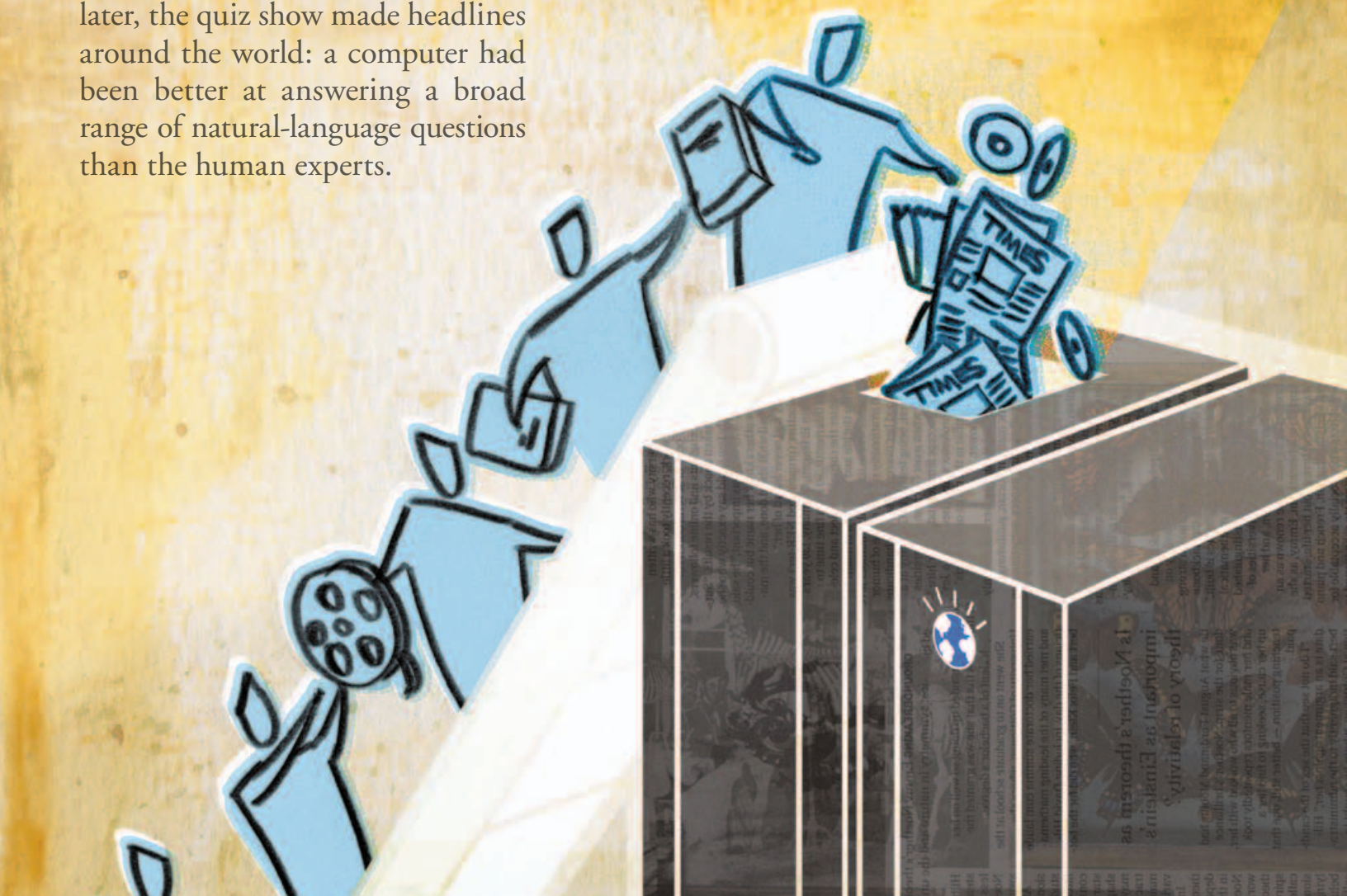
BUILDING THE Watson team

BY DAVID FERRUCCI

On January 14, 2011, I was in the audience at IBM's Watson Research Lab in Yorktown, New York, along with company executives, major clients, and my project team when our Watson computer soundly defeated two human champions in the third round of their *Jeopardy!* competition. Publicly aired a month later, the quiz show made headlines around the world: a computer had been better at answering a broad range of natural-language questions than the human experts.



The event was the culmination of almost four years of intensive work by my team of artificial-intelligence researchers and engineers. My own qualifications as project leader and principal investigator included a background in artificial intelligence, automated reasoning, and UIMA (unstructured information management architecture), and the experience of working on an open-domain question-and-answer (Q&A) project. But there's also the fact that I was the only one willing to take on the very public challenge of pushing Q&A technology well beyond the current state of the art and putting the result to the test on national television. The risk–reward trade-off was daunting. The project had been shopped around the company by senior executives for several years. Most people didn't think it was possible.

The Challenge

Their skepticism wasn't unreasonable; the difficulties were enormous. Answering *Jeopardy!* questions is not like playing chess, which computers have done at the highest level for years. Unlike that game, with its strict, unambiguous rules and numerous but finite potential moves, *Jeopardy!* questions have no formal logic; many of them are quirky and playful in ways that many humans understand but machines don't. There is no way to map those clues to axioms; they are not regular enough. So we would have to teach Watson to "reason" from unstructured content—that is, large volumes of naturally occurring text.

The computer would also have to acquire knowledge from unstructured sources (for instance, encyclopedias, dictionaries, thesauruses, the works of Shakespeare, the Bible) in a form the system could evaluate and use to answer the questions.

And it would have to do it quickly. Over two years into the project, after we had made a lot of progress developing algorithms that could parse the clues and arrive at reasonable responses, it still took the system about two hours to answer a single question. We had to get that down to a few seconds.

Nevertheless, the feasibility study my group carried out at the end of 2006 concluded that it should be possible to reach the goal in three to five years. We based that assessment on the facts that mechanical reasoning had advanced in recent years, that sufficient computer power for inductive reasoning was now available, and that lots of semi-structured reasoning data existed.

Senior IBM executives gave their approval, but the technical people, who understood the magnitude of the problems, still had doubts. Many of those computer scientists were more comfortable working on their own small projects and publishing papers that announced their modest contributions to the field than devoting a chunk of their careers to a big, risky project.

Building the Team

There is no formula for winning people over and melding them into a team. It was all about personal relationships: meeting with people—often one on one—and repeating the argument that I believed so strongly in my heart. I talked to them about the potential sense of accomplishment of being part of one of the biggest achievements in the history of computer science. I reminded the scientists that they could write and publish five years' worth of papers without answering or even grappling with any of the big questions. The Watson project would give us a chance to be real scientists, to achieve greater certainty about whether this goal could be reached. We would be able to say, "Here's how we did it," or, "Here's why the current state of the art cannot accomplish this task." Even failure would advance the field significantly.

My evangelizing attracted about a dozen scientists; the team eventually grew to twenty-five. Over time, the group developed a strong culture of cooperation and a shared determination to get something done. That practicality was partly inspired by the

engineers on the team. They were I-want-to-make-stuff-work people; they didn't care about publishing papers. They won the respect of the scientists, who adopted some of their can-do attitude.

One of the things that helped make the team cohesive and collaborative was putting the multidisciplinary algorithm group in a single room and doing away with the physical and psychological barriers to working together that exist in traditional offices. We did not even have cubicles in the room, just tables with monitors on them. People were in each other's faces; everyone knew what everyone else was doing. At the beginning, some people had trouble focusing and preferred a private office, but most of them adjusted to the more public space. It became a lively hub of activity. Being away from the lab was a disadvantage—you could quickly lose track of what was happening.

The lab also provided one-stop shopping for other subteams. For example, the infrastructure team was focused on scale-out and reducing latency. They were not involved in algorithm development and were ordinarily not in the lab. But when they found a bottleneck and needed to discuss which algorithms were contributing to the slowdown, they would come to the lab. Everyone who needed to be involved in the discussion was there. The lab made for very efficient one-stop shopping.

The Watson team environment was a big change for scientists who were accustomed to working mainly alone, searching for that one formula or algorithm that can better the last publication. A few people who could not adjust left the team. That is usually part of how new cultures develop: most people adopt the new behaviors and those who can't move on to something else.

Progress

A key was getting an extensible architecture in place that reflected our scientific hypothesis for how a Q&A system effective enough to win at *Jeopardy!* should work. It was a hypothesis based on the decades of collective experience represented by the team, synthesized and laid down in software designs and a framework implementation we called DeepQA. An essential assumption underlying this architecture was that there would be no silver bullet, no single algorithm, no one hero that would solve Q&A; rather, we would have to assemble and combine many algorithms that analyzed the content from different perspectives.

These algorithms could be developed independently by different researchers; the architecture itself would allow for the automatic combination of their results based on statistical machine learning. This approach allowed us to move more rapidly, reducing the requirement to anticipate a perfectly integrated solution and wait for a completely specified top-down design. We had different members of the team work on competing algorithms—that is, algorithms that took different approaches to understanding and answering questions by, for instance, focusing on different parts of speech in the *Jeopardy!* clues. The approaches would simultaneously generate their own reliability scores, and those scores would be weighted statistically to get an overall confidence level. Developing these competing algorithms gave us a big jump in capability.

As we progressed, the size of the incremental improvements got smaller and smaller because the problems remaining were the hardest ones. Consider this clue: "On hearing of the discovery of George Mallory's body, he told reporters he still thinks he was first." This is an example of what we called "missing link" questions. Watson had to discover some unmentioned entity that could lead to the right answer. In this case the missing link was Mount Everest. The answer is "Edmund Hillary."

We did a great deal of rigorous testing using questions randomly selected from past *Jeopardy!* games. (The randomness was important to avoid the unconscious bias we might have

introduced by selecting the kinds of questions we believed Watson could answer.) Some of Watson's early answers were ludicrously wrong. Consider this clue: "*NY Times* Headlines: An exclamation point was warranted for the 'end of' this! in 1918."

An early version of Watson answered with "a sentence." The correct answer was "World War I." You might assume Watson had all the headlines of every *New York Times* paper and knew to just look it up. But it was rarely so easy. Among many improvements that resulted from analyzing this error, one was focused on analyzing and weighing temporal information in a clue. A key here was "1918." Watson had to understand that while an exclamation point may end a sentence, the relevant information here had to be unique to a particular point in time.

When one of our scientists got the idea of ordering the test questions chronologically, we discovered that Watson's performance decreased on questions written post 2002. That was when the game was changed to make the categories and questions more entertaining for viewers—that is, perhaps a bit quirkier and funnier. The game was a moving target and we had to keep working to make Watson more flexible.

That we succeeded became clear in early 2011, when Watson became a *Jeopardy!* champion. The system offered a very few ludicrously wrong answers and many astonishingly correct ones and beat its human competitors.

After *Jeopardy!*

Since then, I've worked hard to keep the team together in the face of pressure to chop it up by sending members to other project teams. *Jeopardy!* was only a step on the road to improving analytics and natural-language processing. There is still a lot of work to do.

We are currently working on a machine dedicated to medical diagnosis and treatment evaluation, one that can help make more reliable decisions and explain how it arrives at its recommendations based on analyzing the most current and reliable sources of information. Unlike Watson on *Jeopardy!*, this system will be interactive, cooperating with its human partners. So, for instance, it will be able to say, "I've come up with five possible answers based on the patient data and categorized the evidence based on symptoms, drug interactions, patient history, and demographics. If you can confirm the following, my confidence in the top-most recommendation will increase." The system will also show the documents most relevant to its recommendation to humans who can evaluate their reliability based on their own knowledge and experience and ultimately make a decision.

After that, who knows? We are still at the very beginning of developing capabilities that will astound and serve us in the future. ●

DAVID FERRUCCI is a research staff member and leader of the Semantic Analysis and Integration Department at IBM's T.J. Watson Research Center. His team of twenty-five researchers focuses on developing technologies for discovering knowledge in natural language and leveraging those technologies in a variety of intelligent search, data analytics, and knowledge management solutions.

A FEW PEOPLE WHO COULD NOT ADJUST LEFT THE TEAM. THAT IS USUALLY PART OF HOW NEW CULTURES DEVELOP: MOST PEOPLE ADOPT THE NEW BEHAVIORS AND THOSE WHO CAN'T MOVE ON TO SOMETHING ELSE.