

**PETER**

**SZOLOVITS:**

Fortunately, I have a guest today, Dr. Adam Wright, who will be doing an interview-style session and will answer questions for you. This is Adam's bread and butter, exactly how to translate this kind of technology into the clinic. He's currently in the partner system at the Brigham, I guess. But he's about to become a traitor and leave us in Boston and occupy a position at Vanderbilt University, for which we wish him luck. But I'm glad that we caught him before he leaves this summer.

OK, so quite frankly, I wish that I could tell you a much happier story than the one that you're going to hear from me during the prepared part of my talk. And maybe Adam will cheer us up and make us more optimistic, based on his experience. So you may have noticed that AI is hot.

So HIMSS, for example, is the Health Information Management Systems Society. It's a big-- they hold annual meetings and consist of a lot of vendors and a lot of academics. And it's one of these huge trade show kinds of things, with balloons hanging over booths and big open spaces. So for example, they're now talking about a AI-powered health care.

On the other hand, it's important to remember this graph. So this is the sort of technology adoption graph. And it's called the hype cycle. And what you see here is that R&D-- that's us-- produces some wonderful, interesting idea. And then all of a sudden, people get excited about it.

So who are the people that get most excited about it? It's the people who think they're going to make a fortune from it. And these are the so-called vulture capitalists-- venture capitalists. And so the venture capitalists come in and they encourage people like us to go out and found companies-- or if not us, then our students to go found companies. And figure out how to turn this nascent idea into some important moneymaking enterprise.

Now the secret of venture capital is that they know that about 90% of the companies that they fund are going to tank. They're going to do very badly. And so as a result, what they hope for and what they expect-- and what the good ones actually get-- is that one in 10 that becomes successful makes so much money that it makes up for all of the investment that they poured into the nine out of 10 that do badly.

So I actually remember in the 1990s, I was helping a group pitch a company to Kleiner

Perkins, which is the big venture-- one of the big venture capital funds in Silicon Valley. And we walked into their boardroom and they had a copy of the San Jose Mercury News, which is the local newspaper for Silicon Valley, on their table.

And they were just beaming, because there was an article that said that in the past year, the two best and the two worst investments in Silicon Valley had been by their company. But that's pretty good, right? If you get two winners and two really bad losers, you're making tons and tons of money. So they were in a good mood and they funded us. We didn't make them any money.

So what you see on this curve is that there is a kind of set of rising expectations that comes from the development of these technologies. And you have some early adopters. And then you have the newspapers writing about how this is the revolution and everything will be different from here on out. Then you have some additional activity beyond the early adopters. And then people start looking at this and going, well, it really isn't as good as it's cracked up to be.

Then you have the steep decline where there's some consolidation and some failures. And people have to go back to venture capital to try to get more money in order to keep their companies going. And then there's a kind of trough, where people go, oh well, this was another of these failed technological innovations.

Then gradually, you start reaching what this author calls the slope of enlightenment, where people realize that, OK, it's not really as bad as we thought it was when it didn't meet our lofty expectations. And then gradually, if it's successful, then you get multiple generations of the product and it does achieve adoption.

The adoption almost never reaches the peak that it was expected to reach at the time of the top of the hype cycle. But it becomes useful. It becomes profitable. It becomes productive. Now I've been around long enough to see a number of these cycles go by. So in the 1980s, for example, at a time that was now jokingly referred to as AI summer-- where people were building expert systems and these expert systems were going to just revolutionize everything-- I remember going to a conference where the Campbell Soup Company had built an expert system that was based on the expertise of some old timers who were retiring.

And what this expert system did is it told you how to clean the vats of soup-- y know, these

giant million-gallon things where they make soup-- when you're switching from making one kind of soup to another. So you know, if you're making beef consomme and you switch to making beef barley soup, you don't need to clean the vat at all. Whereas if you're switching from something like clam chowder to a consomme, then you need to clean it really well.

So this was exactly the kind of thing that they were doing. And there were literally thousands of these applications being built. At the top of the hype cycle, all kinds of companies, like Campbell's Soup and the airlines and everybody was investing huge amounts of money into this. And then there was a kind of failure of expectations. These didn't turn out to be as good as people thought they were going to be, or as valuable as people thought they were going to be.

And then all of a sudden came AI winter. So AI winter followed AI summer. There was no AI fall, except in a different sense of the word fall. And all of a sudden, funding dried up and the whole thing was declared a failure. But in fact today, if you go out there and you look at-- Microsoft Excel has an expert system-based help system bundled inside it. And there are tons of such applications.

It's just that now they're no longer considered cutting-edge applications of artificial intelligence. They're simply considered routine practice. So they've become incorporated, without the hype, into all kinds of existing products. And they're serving a very useful role. But they didn't make those venture capital firms the tons of money that they had hoped to make.

There was a similar boom and bust cycle in the 2000s around the creation of the worldwide web and e-commerce. OK, so e-commerce. Again, there was this unbelievably inflated set of expectations. Then around the year 2000, there was a big crash, where all of a sudden people realized that the value in these applications was not as high as what they expected it to be.

Nevertheless, you know Amazon is doing just fine. And there are plenty of online e-commerce sites that are in perfectly good operating order today. But it's no longer the same hype about this technology. It's just become an accepted part of the way that you do business in almost everything. Yeah.

**AUDIENCE:** When you speak of expert systems, does that mean rule-based systems?

**PETER SZOLOVITS:** They were either rule-based or pattern matching systems. There were two basic kinds. I think a week from today, I'm going to talk about some of that and how it relates to modern machine

learning. So we'll see some examples. OK, well, a cautionary tale is IBM's Watson Health.

So I assume most of you remember when Watson hit the big time by beating the Jeopardy champions. This was back in the early 2010s or something. I don't remember exactly which year. And they had, in fact, built a really impressive set of technologies that went out and read all kinds of online sources and distilled them into a kind of representation that they could very quickly look up things when they were challenged with a Jeopardy question.

And then it had a sophisticated set of algorithms that would try to find the best answer for some question. And they even had all kinds of bizarre special-purpose things. I remember there was a probabilistic model that figured out where the Daily Double squares were most likely to be on the Jeopardy board. And then they did a utility theoretic calculation to figure out if they did hit the Daily Double, what was the optimum amount of money to bet, based on the machine's performance, in order to optimize.

They decided that humans typically don't bet enough when they have a chance on the Daily Double. So there was a lot of very special-purpose stuff done for this. So this was a huge publicity bonanza. And IBM decided that next they were going to tackle medicine. So they were going to take this technology and apply it to medicine.

They were going to read all of the medical journals and all of the electronic medical records that they could get their hands on. And somehow this technology would again distill the right information, so that they could answer questions like a Jeopardy question, except not stated in its funny backward way. Where you might say, OK, for this patient, what is the optimum therapy? And it would go out and use the same technology to figure that out.

Now that was a perfectly reasonable thing to try. The problem they ran into was this hype cycle, that the people who made this publicly-known were their marketing people and not their technical people. And the marketing people overpromised like crazy. They said surely this is just going to solve all these problems. And we won't need anymore research in this area, because man, we got it.

I'm overstating it, even from the marketing point of view. And so Watson for Oncology used this cloud-based supercomputer to digest massive amounts of data. That data included all kinds of different things. So I'm going to go into a little bit of detail about what some of their problems were. This is from an article in this journal, Statnews, which did an investigative piece on what happened with Watson.

So you know, they say what I just said. Breathlessly promoting its signature brand, IBM sought to capture the world's imagination and quickly zeroed in on a high-profile target, which was cancer. So this was going to solve the problem of some patient shows up, is diagnosed with cancer, and you want to know how to treat this person. So this would use all of the literature and all of everything that it had gathered from previous treatments of previous patients. And it would give you the optimal solution.

Now it has not been a success. There are a few dozen hospitals that have adopted the system. Very few of them in the United States, more of them abroad. And the foreigners complain that its advice is biased toward American patients and American approaches. To me, the biggest problem is that they haven't actually published anything that validates, in a scientific sense, that this is a good idea. That it's getting the right answers.

My guess is the reason for this is because it's not getting the right answers, a lot of the time. But that doesn't prevent marketing from selling it. The other problem is that they made a deal with Memorial Sloan Kettering-- which is one of the leading cancer hospitals in the country-- to say, we're going to work with you guys and your oncologists in order to figure out what really is the right answer.

So I think they tried to do what their marketing says that they're doing, which is to really derive the right answer from reading all of the literature and looking at past cases. But I don't think that worked well enough. And so what they wound up doing is turning to real oncologists, saying, what would you do under these circumstances? And so what they wound up building is something like a rule-based system that says, if you see the following symptoms and you have the following genetic defects, then this is the right treatment.

So the promise that this was going to be a machine learning system that revolutionized cancer care by finding the optimal treatment really is not what they provided. And as the article says, the system doesn't really create new knowledge. So it's AI only in the sense of providing a search engine that, when it makes a recommendation, can point you to articles that are a reasonable reflection of what it's recommending.

Well, I'm going to stop going through this litany. But you'll see it in the slides, which we'll post. They had a big contract with M.D. Anderson, which is another leading cancer center in the United States. M.D. Anderson spent about \$60 million on this contract, implementing it. And they pulled the plug on it, because they decided that it just wasn't doing the job.

Now by contrast, there was a much more successful attempt years ago, which was less driven by marketing and more driven by medical need. And the idea here was CPOE, stands for Computerized Physician Order Entry. The idea behind CPOE was that if you want to affect the behavior of clinicians in ordering tests or drugs or procedures, what you want to do is to make sure that they are interacting with the computer.

So that when they order, for example, some insanely expensive drug, the system can come back and say, hey, do you realize that there's a drug that costs 1/100 as much, which according to the clinical trials that we have on record is just as effective as the one that you've ordered? And so for example, here at the Beth Israel many years ago, they implemented a system like that. And in the first year, they showed that they saved something like \$16 million in the pharmacy, just by ordering cheaper variants of drugs that could have been very expensive.

And they also found that the doctors who were doing the ordering were perfectly satisfied with that, because they just didn't know how expensive these drugs were. That's not one of the things that they pay attention to. So there are many applications like that that are driven by this. And again, here are some statistics. You can reduce error rates by half. You can reduce severe medication errors by 88%.

You can have a 70% reduction in antibiotic-related adverse drug events. You can reduce length of stay, which is another big goal that people go after. And at least if you're an optimist, you can believe these extrapolations that say, well, we could prevent 3 million adverse drug events at big city hospitals in the United States if everybody used systems like this.

So the benefits are that it prompts with warnings against possible drug interactions, allergies, or overdoses. It can be kept up to date by some sort of mechanism where people read the literature and keep updating the databases this is driven from. And it can do mechanical things like eliminate confusion about drug names that sound similar. Stuff like that.

So the Leapfrog Group, which does a lot of meta analyses and studies of what's effective, really is behind this and pushing it very strongly. Potential future benefits, of course, are that if the kinds of machine learning techniques that we talk about become widely used, then these systems can be updated automatically rather than by manual review. And you can gain the advantages of immediate feedback as new information becomes available.

Now the adoption of CPOE was recommended by the National Academy of Medicine. They wanted every hospital to use this by 1999. And of course, it hasn't happened. So I couldn't find current data, but 2014 data shows that CPOE, for example, for medication orders, is only being used in about 25% of the hospitals. And at that time, people were extrapolating and saying, well, it's not going to reach 80% penetration until the year 2029.

So it's a very slow adoption cycle. Maybe it's gotten better. The other problem-- and one of the reasons for resistance-- is that it puts additional stresses on people. So for example, this is a study of how pharmacists spend their time. So clinical time is useful. That's when they're consulting with doctors, helping them figure out appropriate dosage for patients. Or they're talking to patients, explaining to them how to take their medications, what side effects to watch out for, et cetera.

These distributive tasks-- it's a funny term-- mean the non-clinical part of what they're doing. And what you see is that hospitals that have adopted CPOE, they wind up spending a little bit more time on the distributive tasks and a little bit less time on the clinical tasks. Which is probably not in the right direction, in terms of what pharmacists were hoping for out of systems like this.

Now people have studied the diffusion of new medical technologies. And I think I'll just show you the graph. So this is in England, but this is the adoption for statins. So from the time they were introduced-- statins is the drug that keeps your cholesterol low. From the time they were introduced until they were being used, essentially, at 100% of places was about five and a half, six years. So reasonably fast.

If you look at the adoption of magnetic resonance imaging technology, it took five years for it to have any adoption whatsoever. And that's because it was insanely expensive. So there were all kinds of limitations. You know, even in Massachusetts, you have to get permission from some state committee to buy a new MRI machine. And if another hospital in your town already had one, then they would say, well, you shouldn't buy one because you should be able to use this other hospital's MRI machine.

Same thing happened with CT. But as soon as those limitations were lifted, boom. It went up and then continues to go up. Whereas stents, I actually don't know why they were delayed by that long. But this is for people with blockages in coronary arteries or other arteries. You can put in a little mesh tube that just keeps that artery open. And that adoption was incredibly

quick.

So different things get adopted at different rates. Now the last topic I want to talk about before--  
- yeah.

**AUDIENCE:** So what happens in those years where you just have spikes? What's doing it?

**PETER SZOLOVITS:** So according to those authors, in the case of stents, there were some champions of the idea of stenting who went around and convinced their colleagues that this was the right technology to use. So there was just an explosive growth in it. In the other technologies, in the MRI case, money mattered a lot because they're so expensive. Stents are relatively cheap. And in the case of statins, those are also relatively cheap. Or they've become cheap since they went off patent. Originally, they were much more expensive.

But there are still adoption problems. So for example, there was a recommendation-- I think about 15, maybe even 20 years ago-- that said that anybody who has had a heart attack or coronary artery disease should be taking beta blockers. And I don't remember what the adoption rate is today, but it's only on the order of a half. And so why? This is a dirt cheap drug.

For reasons not quite understood, it reduces the probability of having a second heart attack by about 35%. So it's a really cheap protective way of keeping people healthier. And yet it just hasn't suffused practice as much as people think it should have. All right. So how do we assure the quality of these technologies before we foist them on the world?

This is tricky. So John Ioannidis, a Stanford professor, has made an extremely successful career out of pointing out that most biomedical research is crap. It can't be reproduced. And there are some famous publications that show that people have taken some area of biomedicine, and they've looked at a bunch of well-respected published studies. And they've gone to the lab and they've tried to replicate those studies. Half the time or three-quarters of the time, they fail to do so.

You go, oh my god, this is horrible. It is horrible. Yeah.

**AUDIENCE:** You mean like they failed to do so, so they won't reproduce the exact same results? Or what exactly--

**PETER** Worse than that. So it's not that there are slight differences. It's that, for example, a result that



**SZOLOVITS:** was shown to be statistically significant in one study, when they repeat the study, is no longer statistically significant. That's bad, if you base policy on that kind of decision.

So Ioannidis has a suggestion, which would probably help a lot. And that is, basically, make known to everybody all the studies that have failed. So the problem is that if you give me a big data set and I start mining this data set, I'm going to find tons and tons of interesting correlations in this data. And as soon as I get one that has a good p value, my students and I go, fantastic. Time to publish.

Now consider the fact that I'm not the only person in this role. So you know, David's group is doing the same thing. And John Gutttag's and Regina Barzilay's and all of our colleagues at every other major university and hospital in the United States. So there may be hundreds of people who are mining this data. And each of us has slightly different ways of doing it.

We select our cases differently. We preprocess the data differently. We apply different learning algorithms to them. But just by random chance, some of us are going to find interesting results, interesting patterns. And of course, those are the ones that get published. Because if you don't find an interesting result, you're not going to submit it to a journal and say, you know I looked for the following fact phenomenon and I was unable to find it. Because the journal says, well, that's not interesting to anybody.

So Ioannidis is recommending that, basically, every study that anybody undertakes should be registered. And if you don't get a significant result, that should be known. And this would allow us to make at least some reasonable estimate of whether the significant results that were gotten are just the statistical outliers that happened to reach  $p = 0.05$  or whatever your threshold is, or whether it's a real effect because not that many people have been trying this. Yeah.

**AUDIENCE:** [INAUDIBLE] why do you think this is? Is it because of the size of some core patients? Or bias in the assay? Or just purely randomness in the study?

**PETER**  
**SZOLOVITS:** It could be any of those. It could be that your hospital has some biased data collection. And so you find an effect. My hospital doesn't, and so I don't find it. It could be that we just randomly sub-sampled a different sample of the population. So it's very interesting. Last year I was invited to a meeting by Jeff Drazen, who's the executive editor of the New England Journal. And he's thinking about-- has not decided-- but he's thinking about a policy for the New England Journal, which is like the top medical journal, that says that he will not publish any

result unless it's been replicated on two independent data sets.

So that's interesting. And that's an attempt to fight back against this problem. It's a different solution than what Ioannidis is recommending. So this was a study by Enrico Carrara. And he's talking about what it means to replicate. And again, I'm not going to go through all this. But there's the notion that replication might mean exact replication, i.e. You do exactly the same thing on exactly the same kind of data, but in a different data set.

And then partial replication, conceptual replication, which says, you follow the same procedures but in a different environment. And then quasi replication-- either partial or conceptual. And these have various characteristics that you can look at. It's an interesting framework. So this is not a new idea. The first edition of this book, *Evaluation Methods in Biomedical Informatics*, was called *Evaluation Methods in Medical Informatics* by the same authors and was published a long time ago. I can't remember.

This one is relatively recent. And so they do a multi-hundred page, very detailed evaluation of exactly how one should evaluate clinical systems like this. And it's very careful and very cautious, but it's also very conservative. So for example, one of the things that they recommend is that the people doing the evaluation should not be the people who developed the technique, because there's innate bias. You know, I want my technique to succeed.

And so they say, hand it off to somebody else who doesn't have that same vested interest. And then you're going to get a more careful evaluation. So Steve Pauker and I wrote a response to one of their early papers recommending this that said, well, that's so conservative that it sort of throws the baby out with the bathwater. Because if you make it so difficult to do an evaluation, you'll never get anything past it.

So we proposed instead a kind of staged evaluation that says, first of all, you should do regression testing so that every time you use these agile development methods, you should have the set of cases that your program has worked on before. You should automatically rerun them and see which ones you've made better and which ones you've made worse. And that will give you some insight into whether what you're doing is reasonable.

Then you might also build tools that look at automating ways of looking for inconsistencies in the models that you're building. Then you have retrospective review, judged by clinicians. So you run a program that you like over a whole bunch of existing data, like what you're doing with Mimic or with Market Scan. And then you do it prospectively, but without actually affecting

patients.

So you do it in real time as the data is coming in, but you don't tell anybody what the program results in. You just ask them to evaluate in retrospect to see whether it was right. And you might say, well, what's the difference between collecting the data in real time and collecting the data retrospectively?

Historically, the answer is there is a difference. So circumstances differ. The mechanisms that you have for collecting the data differ. So this turns out to be an important issue. And then you can run a prospective controlled trial where you're interested in evaluating both the answer that you get from the program, and ultimately the effect on health outcomes.

So if I have a decision support system, the ultimate proof of the pudding is if I run that decision support system. I give advice to clinicians, the clinicians change their behavior sometimes, and the patients get a better outcome. Then I'm convinced that this is really useful. But you have to get there slowly, because you don't want to give them worse outcomes. That's unethical and probably illegal.

And you want to compare this to the performance of unaided doctors. So the Food and Drug Administration has been dealing with this issue for many, many years. I remember talking to them in about 1976, when they were reading about the very first expert system programs for diagnosis and therapy selection. And they said, well, how should we regulate these? And my response at the time was, God help us. Keep your hands off.

Because if you regulate it, then you're going to slow down progress. And in any case, none of these programs are being used. These programs are being developed as experimental programs in experimental settings. They're not coming anywhere close to being used on real patients. And so there is not a regulatory issue.

And about every five years, FDA has revisited that question. And they have continued to make essentially the same decision, based on the rationale that, for example, they don't regulate books. If I write a textbook that explains something about medicine, the FDA is not going to see whether it's correct or not. And the reason is because the expectation is that the textbook is making recommendations, so to speak, to clinical practitioners who are responsible experts themselves.

So the ultimate responsibility for how they behave rests with them and not with the textbook.

And they said, we're going to treat these computer programs as if they were dynamic textbooks, rather than colleagues who are acting independently and giving advice. Now as soon as you try to give that advice, not to a professional, but to a patient, then you are immediately under the regulatory auspices of FDA. Because now there is no professional intermediate that can evaluate the quality of that advice.

So what FDA has done, just in the past year, is they've said that we're going to treat these AI-based quote-unquote devices as medical devices. And we're going to apply the same regulatory requirements that we have for these devices, except we don't really know how to do this. So there's a kind of experiment going on right now where they're saying, OK, submit applications for review of these devices to us. We will review them.

And we will use these criteria-- product quality, patient safety, clinical responsibility, cybersecurity responsibility, and a so-called proactive culture in the organization that's developing them-- in order to make a judgment of whether or not to let you proceed with marketing one of these things. So if you look, there are in fact about 10 devices, quote-unquote-- these are all software-- that have been approved so far by FDA. And almost all of them are imaging devices.

They're things that do convolutional networks over one thing or another. And so here are just a few examples. Imagen has OsteoDetect, which analyzes two-dimensional X-ray images for signs of distal radius fracture. So if you break your wrist, then this system will look at the X-ray and decide whether or not you've done that.

Here's one from IDx, which looks at the photographs of your retina and decides whether you have diabetic retinopathy. And actually, they've published a lot of papers that show that they can also identify heart disease and stroke risk and various other things from those same photographs. So FDA has granted them approval to market this thing.

Another one is Viz, which automatically analyzes CT scans for ER patients and is looking for blockages and major brain blood vessels. So this can obviously lead to a stroke. And this is an automated technique that does that. Here's another one. Arterys measures and tracks tumors or potential cancers in radiology images. So these are the ones that have been approved.

And then I just wanted to remind you that there's actually plenty of literature about this kind of stuff. So the book on the left actually comes out next week. I got to read a pre-print of it, by Eric Topol, who's one of these doctors who writes a lot about the future of medicine. And he

actually goes through tons and tons of examples of not only the systems that have been approved by FDA, but also things that are in the works that he's very optimistic that these will again revolutionize the practice of medicine.

Bob Wachter, who wrote the book on the left a couple of years ago, is a little bit more cautious because he's chief of medicine at UC San Francisco. And he wrote this book in response to them almost killing a kid by giving him a 39x overdose of a medication. They didn't quite succeed in killing the kid. So it turned out OK.

But he was really concerned about how this wonderful technology led to such a disastrous outcome. And so he spent a year studying how these systems were being used, and writes a more cautionary tale. So let me turn to Adam, who as I said, is a professor at the Brigham and Harvard Medical School. Please come and join me, and we can have a conversation.

**ADAM WRIGHT:** So my name is Adam Wright. I'm an associate professor of medicine at Harvard Medical School. In that role, I lead a research program and I teach the introduction to biomedical informatics courses at the medical school. So if you're interested in the topics that Pete was talking about today, you should definitely consider cross-registering in VMI 701 or 702. The medical school certainly always could use a few more enthusiastic and technically-minded machine learning experts in our course.

And then I have a operational job at Partners. Partners is the health system that includes Mass General Hospital and the Brigham and some community hospitals. And I work on Partners eCare, which is our kind of cool brand name for Epic. So Epic is the EHR that we use at Partners. And I help oversee the clinical decision support there.

So we have a decision support team. I'm the clinical lead for monitoring and evaluation. And so I help make sure that our decision support systems of the type that Pete's talking about work correctly. So that's my job at the Brigham and at Partners.

**PETER** Cool. And I appreciate it very much.

**SZOLOVITS:**

**ADAM WRIGHT:** Thanks. I appreciate the invitation. It's fun to be here.

**PETER** So Adam, the first obvious question is what kind of decision support systems have you guys  
**SZOLOVITS:** actually put in place?

**ADAM WRIGHT:** Absolutely. So we've had a long history at the Brigham and Partners of using decision support. Historically, we developed our own electronic health record, which was a little bit unusual. About three years ago, we switched from our self-developed system to Epic, which is a very widely-used commercial electronic health record.

And to the point that you gave, we really started with a lot of medication-related decision support. So that's things like drug interaction, alerting. So you prescribe two drugs that might interact with each other. And we use a table-- no machine learning or anything too complicated-- that says, we think this drug might interact with this.

We raise an alert to the doctor, to the pharmacist. And they make a decision, using their expertise as the learned intermediary, that they're going to continue with that prescription. Let's have some dosing support, allergy checking, and things like that. So our first set of decision support really was around medications.

And then we turned to a broader set of things like preventative care reminders, so identifying patients that are overdue for a mammogram or a pap smear or that might benefit from a statin or something like that. Or a beta blocker, in the case of acute myocardial infarction. And we make suggestions to the doctor or to other members of the care team to do those things.

Again, those historically have largely been rule-based. So some experts sat down and wrote Boolean if-then rules, using variables that are in a patient's chart. We have increasingly, though, started trying to use some predictive models for things like readmission or whether a patient is at risk of falling down in the hospital. A big problem that patients often encounter is they're in the hospital, they're kind of delirious. The hospital is a weird place. It's dark. They get up to go to the bathroom. They trip on their IV tubing, and then they fall and are injured.

So we would like to prevent that from happening. Because that's obviously kind of a bad thing to happen to you once you're in the hospital. So we have some machine learning-based tools for predicting patients that are at risk for falls. And then there is a set of interventions like putting the bed rails up or putting an alarm that buzzes when if they get out of bed. Or in more extreme cases, having a sitter, like a person who actually sits in the room with them and tries to keep them from getting up or assists them to the bathroom. Or calls someone who can assist them to the bathroom.

So we have increasingly started using those machine learning tools. Some of which we get

from third parties, like from our electronic health record vendor, and some of which we sort of train ourselves on our own data. That's a newer pursuit for us, is this machine learning.

**PETER** So when you have something like a risk model, how do you decide where to set the threshold?

**SZOLOVITS:** You know, if I'm at 53% risk of falling, should you get a sitter to sit by my bedside?

**ADAM WRIGHT:** It's complicated, right? I mean, I would like to say that what we do is a full kind of utility analysis, where we say, we pay a sitter this much per hour. And the risk of falling is this much. And the cost of a fall-- most patients who fall aren't hurt. But some are. And so you would calculate the cost-benefit of each of those things and figure out where on the ROC curve you want to place yourself.

In practice, I think we often just play it by ear, in part because a lot of our things are intended to be suggestions. So our threshold for saying to the doctor, hey, this patient is at elevated risk for fall, consider doing something, is pretty low. If the system were, say, automatically ordering a sitter, we might set it higher. I would say that's an area of research.

I would also say that one challenge we have is we often set and forget these kinds of systems. And so there is kind of feature drift and patients change over time. We probably should do a better job of then looking back to see how well they're actually working and making tweaks to the thresholds. Really good question.

**PETER** But these are, of course, very complicated decisions. I remember 50 years ago talking to  
**SZOLOVITS:** some people in the Air Force about how much should they invest in safety measures. And they had a utility theoretic model that said, OK, how much does it cost to replace a pilot if you kill them?

**ADAM WRIGHT:** Yikes. Yeah.

**PETER** And this was not publicized a lot.

**SZOLOVITS:**

**ADAM WRIGHT:** I mean, we do calculate things like quality-adjusted life-years and disability-adjusted life-years. So there is-- in all of medicine as people deploy resources, this calculus. And I think we tend to assign a really high weight to patient harm, because patient harm is-- if you think about the oath the doctors swear, first do no harm. The worst thing we can do is harm you in the hospital.

So I think we have a pretty strong aversion to do that. But it's very hard to weigh these things. I think one of the challenges we often run into is that different doctors would make different decisions. So if you put the same patient in front of 10 doctors and said, does this patient need a sitter? Maybe half would say yes and half would say no. So it's especially hard to know what to do with a decision support system if the humans can't agree on what you should do in that situation.

**PETER  
SZOLOVITS:**

So the other thing we talked about on the phone yesterday is I was concerned-- a few years ago, I was visiting one of these august Boston-area hospitals and asked to see an example of somebody interacting with this Computerized Physician Order Entry system. And the senior resident who was taking me around went up to the computer and said, well, I think I remember how to use this.

And I said, wait a minute. This is something you're expected to use daily. But in reality, what happens is that it's not the senior doctors or even the medium senior doctors. It's the interns and the junior residents who actually use the systems.

**ADAM WRIGHT:** This is true.

**PETER  
SZOLOVITS:**

And the concern I had was that it takes a junior resident with a lot of guts to go up to the chief of your service and say, doctor x, even though you asked me to order this drug for this patient, the computer is arguing back that you should use this other one instead.

**ADAM WRIGHT:** Yeah, it does. And in fact, I actually thought of this a little more after we chatted about it. We've heard from residents that people have said to them, if you dare page me with an Epic suggestion in the middle of the night, I'll never talk to you again. So just override all of those alerts.

I think that one of the challenges is-- and some culpability on our part-- is that a lot of these alerts we give have a PPV of like, 10 or 20%. They are usually wrong. We think it's really important, so we really raise these alerts a lot. But people experience this kind of alert fatigue, or what people call alarm fatigue. You see this in cockpits, too.

But people get too many alerts, and they start ignoring the alerts. They assume that they're wrong. They tell the resident not to page them in the middle of the night, no matter what the computer says. So I do think that we have some responsibility to improve the accuracy of these alerts. I do think machine learning could help us.



We're actually just having a meeting about a pneumococcal vaccination alert. This is something that helps people remember to prescribe this vaccination to help you not get pneumonia. And it takes four or five variables into account. We started looking at the cases where people would override the alert. And they were mostly appropriate.

So the patient is in a really extreme state right now. Or conversely, the patient is close to the end of life. And they're not going to benefit from this vaccination. If the patient has a phobia of needles, if the patient has an insurance problem. And we think there's probably more like 30 or 40 variables that you would need to take into account to make that really accurate.

So the question is, when you have that many variables, can a human develop and maintain that logic? Or would we be better off trying to use a machine learning system to do that? And would that really work or not?

**PETER**

So how far are we from being able to use a machine learning system to do that?

**SZOLOVITS:**

**ADAM WRIGHT:**

I think that the biggest challenge, honestly, relates to the availability and accuracy of the data in our systems. So Epic, which is the EHR that we're using-- and Cerner and Allscripts and most of the major systems-- have various ways to run even sophisticated machine learning models, either inside of the system or bolted onto the system and then feeding model inferences back into the system.

When I was giving that example of the pneumococcal vaccination, one of the major problems is that there's not always a really good structured way in the system that we indicate that a patient is at the end of life and receiving comfort measures only. Or that the patient is in a really extreme state, that we're in the middle of a code blue and that we need to pause for a second and stop giving these kind of friendly preventive care suggestions.

So I would actually say that the biggest barrier to really good machine-learning-based decision support is just the lack of good, reliably documented, coded usable features. I think that the second challenge, obviously, is workflow. You said-- it's sometimes hard to know in the hospital who a patient's doctor is. The patient is admitted. And on the care team is an intern, a junior resident, and a fellow, an attending, several specialists, a couple of nurses. Who should get that message or who should get that page?

I think workflow is second. This is where I think you may have said, I have some optimism. I

actually think that the technical ability of our EHR software to run these models is better than it was three or five years ago. And it's, actually, usually not the barrier in the studies that we've done.

**PETER**

**SZOLOVITS:**

So there were attempts-- again, 20 years ago-- to create formal rules about who gets notified under what circumstances. I remember one of the doctors I worked with at Tufts Medical Center was going crazy, because when they implemented a new lab information system, it would alert on every abnormal lab. And this was crazy.

But there were other hospitals that said, well, let's be a little more sophisticated about when it's necessary to alert. And then if somebody doesn't respond to an alert within a very short period of time, then we escalate it to somebody higher up or somebody else on the care team. And that seemed like a reasonable idea to me. But are there things like that in place now?

**ADAM WRIGHT:**

There are. It works very differently in the inpatient and the outpatient setting. At the inpatient setting, we're writing very acute care to a patient. And so we have processes where people sign in and out of the care team. In fact, these prevalence of these automated messages is an incentive to do that well. If I go home, I better sign myself out of that patient, otherwise I'm going to get all these pages all night about them.

And the system will always make sure that somebody is the responding provider. It becomes a little thornier in the outpatient setting, because a lot of the academic doctors at the Brigham only have clinic half a day a week. And so the question is, if an abnormal result comes back, should I send it to that doctor? Should I send it to the person that's on call in that clinic? Should I send it to the head of the clinic?

There are also these edge cases that mess us up a lot. So a classic one is a patient is in the hospital. I've ordered some lab tests. They're looking well, so I discharge the patient. The test is still pending at the time the patient is discharged. And now, who does that go to? Should it go to the patient's primary care doctor? Do they have a primary care doctor? Should it go to the person that ordered the test? That person may be on vacation now, if it's a test that takes a few weeks to come back.

So we still struggle with-- we call those TPADs-- tests pending at discharge. We still struggle with some of those edge cases. But I think in the core, we're pretty good at it.

**PETER**

So one of the things we talked about is an experience I've had and you've probably had that--

**SZOLOVITS:** for example, a few years ago I was working with the people who run the clinical labs at Mass General. And they run some ancient laboratory information systems that, as you said, can add and subtract but not multiply or divide.

**ADAM WRIGHT:** They can add and multiply, but not subtract or divide. Yes. And it doesn't support negative numbers. Only unsigned integers.

**PETER**  
**SZOLOVITS:** So there are these wonderful legacy systems around that really create horrendous problems, because if you try to build anything-- I mean, even a risk prediction calculator-- it really helps to be able to divide as well as multiply. So we've struggled in that project. And I'm sure you've had similar experiences with how do we incorporate a decision support system into some of this squeaky old technology that just doesn't support it? So what's the right approach to that?

**ADAM WRIGHT:** There are a lot of architectures and they all have pros and cons. I'm not sure if any one of them is the right approach. I think we often do favor using these creaky old technology or the new technology. So Epic has a built in rule engine. That laboratory you talked about has a basic calculation engine with some significant limitations to it.

So where we can, we often will try to build rules internally using these systems. Those tend to have real-time availability of data, the best ability to sort of push alerts to the person right in their workflow and make those alerts actionable. In cases where we can't do that-- like for example, a model that's too complex to execute in the system-- one thing that we've often done is run that model against our data warehouse.

So we have a data warehouse that extracts the data from the electronic health record every night at midnight. So if we don't need real-time data, it's possible to run-- extract the data, run a model, and then actually write a risk score or a flag back into the patient's record that can then be shown to the clinician, or used to drive an alert or something like that.

That works really well, except that a lot of things that happen-- particularly in an inpatient setting, like predicting sepsis-- depend on real-time data. Data that we need right away. And so we run into the challenge where that particular approach only works on a 24-hour kind of retrospective basis. We have also developed systems that depend on messages.

So there's this-- HL7 is a standard format for exchanging data with an electronic health record. There's various versions and profiles of HL7. But you can set up an infrastructure that sits outside of the EHR and gets messages in real time from the EHR. It makes inferences and

sends messages back into the EHR.

Increasingly, EHRs also do support kind of web service approaches. So that you can register a hook and say, call my hook whenever this thing happens. Or you can pull the EHR to get data out and use another web service to write data back in. That's worked really well for us. You can also ask the EHR to embed an app that you develop.

So people here may have heard-- or should hear at some point-- about SMART on FHIR, which is a open kind of API that allows you to develop an application and embed that application into an electronic health record. We've increasingly been building some of those applications. The downside right now of the smart apps is that they're really good for reading data out of the record and sort of visualizing or displaying it. But they don't always have a lot of capability to write data back into the record or take actions.

Most of the EHR vendors also have a proprietary approach, like an app store. So Epic calls theirs the App Orchard. And most of the EHRs have something similar, where you can join a developer program and build an application. And those are often more full-featured. They tend to be proprietary.

So if you build one Epic app, you have to then build a Cerner app and an Allscripts app and an eClinicalWorks app separately. There are often heavy fees for joining those programs, although the EHR vendors-- Epic in particular-- have lowered their prices a lot. The federal government, the Office of the National Coordinator of Health IT, just about a week and a half ago released some new regulations which really limit the rate at which vendors can charge application developers for API access basically to almost nothing, except for incremental computation costs or special support. So I think that may change everything now that that regulation's been promulgated. So we'll see.

**PETER  
SZOLOVITS:**

So contrary to my pessimistic beginning, this actually is the thing that makes me most optimistic. That even five years ago, if you looked at many of these systems, they essentially locked you out. I remember in the early 2000s, I was at the University of Pittsburgh, where they had one of the first centers that was doing heart-lung transplants.

So their people had built a special application for supporting heart-lung transplant patients, in their own homemade electronic medical records system. And then UPMC went to Cerner at the time. And I remember I was at some meeting where the doctors who ran this heart-lung transplant unit were talking to the Cerner people and saying, how could we get something to

support our special needs for our patients?

And Cerner's answer was, well, commercially it doesn't make sense for us to do this. Because at the time there were like four hospitals in the country that did this. And so it's not a big money maker. So their offer was, well, you pay us an extra \$3 million and within three years we will develop the appropriate software for you. So that's just crazy, right?

I mean, that's a totally untenable way of going about things. And now that there are systematic ways for you either to embed your own code into one of these systems, or at least to have a well-documented, reasonable way of feeding data out and then feeding results back into the system, that makes it possible to do special-purpose applications like this. Or experimental applications or all kinds of novel things. So that's great.

**ADAM WRIGHT:** That's what we're optimistic about. And I think it's worth adding that there's two barriers you have to get through right. One is Epic has to sort of let you into their App Orchard, which is the barrier that is increasingly lower. And then you need to find a hospital or a health care provider that wants to use your app, right.

So you have to clear both of those, but I think it's increasingly possible. You've got smart people here at MIT, or at the hospitals that we have in Boston always wanting to build these apps. And I would say five years ago we would've told people, sorry, it's not possible. And today we're able, usually, to tell people that if there's clinical interest, the technical part will fall into place. So that's exciting for us.

**PETER** Yeah

**SZOLOVITS:**

**ADAM WRIGHT:** Yeah

**AUDIENCE:** Question about that.

**ADAM WRIGHT:** Absolutely

**AUDIENCE:** Some of the applications that you guys develop in house, do you also put those on the Epic Orchard, or do you just sort of implement it one time within your own system?

**ADAM WRIGHT:** Yeah, there's a lot of different ways that we share these applications, right. So a lot of us are researchers. So we will release an open source version of the application or write a paper and

say, this is available. And we'll share it with you. The App Orchard is particularly focused on applications that you want to sell.

So our hospital hasn't decided that we wanted to sell any applications. We've given a lot of applications away. Epic also has something called the Community Library, which is like the AppOrchard, but it's free instead of costing money. And so we released a ton of stuff through the Community Library.

To the point that I was poking out before, one of the challenges is that if we build a Smart on FHIR app, we're able to sort of share that publicly. And we can post that on the web or put it on GitHub. And anybody can use it. Epic has a position that their APIs are proprietary. And they represent Epic's valuable intellectual property or trade secrets. And so we're only allowed to share those apps through the Epic ecosystem.

And so, we often now, when we get a grant-- most of my work is through grants-- we'll have an Epic site. And we'll share that through the Community Library. And we'll have a Cerner site. And we'll share it through Cerner's equivalent. But I think until the capability of the open APIs, like Smart on FHIR, reaches the same level as the proprietary APIs, we're still somewhat locked into having to build different versions and distribute three-- each EHR under separate channels. Really, really good question.

**PETER** And so what's lacking in things like Smart on FHIR--

**SZOLOVITS:**

**ADAM WRIGHT:** Yeah.

**PETER** --that you get from the native interfaces?

**SZOLOVITS:**

**ADAM WRIGHT:** So it's very situational, right. So, for example, in some EHR implementations, the Smart on FHIR will give you a list of the patient's current medications but may not give you historical medications. Or it will tell you that the medicine is ordered, but it won't tell you whether it's been administered. So one half of the battle is less complete data. The other one is that most EHRs are not implementing, at this point, the sort of write back capabilities, or the actionable capabilities, that Smart on FHIR is sort of working on. And it's really some standards for us.

So if we want to build an application that shows how a patient fits on a growth curve, that's fine. If we went to build an application that suggests ordering medicines, that can be really

challenging. Whereas the internal APIs that the vendors provide typically have both read and write capabilities. So that's the other challenge.

**PETER SZOLOVITS:** And do the vendors worry about, I guess two related things, one is sort of cognitive overload. Because if you build 1,000 Smart on FHIR apps, and they all start firing for these inpatients, you're going to be back in the same situation of over-alerting.

And the other question is, are they worried about liability? Since if you were using their system to display recommendations, and those recommendations turn out to be wrong and harm some patient, then somebody will reach out to them legally because they have a lot of money.

**ADAM WRIGHT:** Absolutely. They're worried about both of those. Related particularly to the second one, they're also worried about just sort of corruption or integrity of the data, right. So somehow if I can write a medication order directly to the database, and it may bypass certain checks that would be done normally. And I could potentially enter a wrong or dangerous order.

The other thing that we're increasingly hearing is concerns about protection of data, sort of Cambridge Analytica style worries, right. So if I, as an Epic patient, authorize the Words With Friends app to see my medical record, and then they post that on the web, or monetize it in some sort of a tricky way, what liability, if any, does my health care provider organization, or my-- the EHR vendor, have for that?

And the new regulations are extremely strict, right. They say that if a patient asks you to, and authorizes an app to access their record, you may not block that access, even if you consider that app to be a bad actor. So that's I think an area of liability that is just beginning to be sorted out. And it is, I think, some cause for concern. But at the same time, you could imagine a universe where, I think, there are conservative health organizations that would choose to never authorize any application to avoid risk. So how you balance that is not yet solved.

**PETER SZOLOVITS:** Well-- and to avoid leakage.

**ADAM WRIGHT:** Absolutely.

**PETER SZOLOVITS:** So I remember years ago there was a lot of reluctance, even among Boston area hospitals, to share data, because they were worried that another hospital could cherry pick their most lucrative patients by figuring out something about them. So I'm sure that that hasn't gone away

as a concern.

**ADAM WRIGHT:** Absolutely, yeah.

**PETER** OK, we're going to try to remember to repeat the questions you're asking--

**SZOLOVITS:**

**ADAM WRIGHT:** Oh great, OK.

**PETER** --because of the recording.

**SZOLOVITS:**

**ADAM WRIGHT:** Happy to.

**PETER** Yeah.

**SZOLOVITS:**

**AUDIENCE:** So how does a third party vendor deploy a machine learning model on your system? So is that done through Epic? Obviously, there's the App Orchard kind of thing, but is there ways to go around that and go directly into partners and whatnot? And how does that work?

**ADAM WRIGHT:** Yeah. So the question is how does a third party vendor deploy an application or a machine learning model or something like that? And so with Epic, there's always a relationship between the vendor of the application and the health care provider organization. And so we could work together directly. So if you had an app that the Brigham wanted to use, you could share that app with us in a number of ways.

So Epic supports this thing called Predictive Modeling Markup Language, or PMML. So if you train a model, you can export a PMML model. And I can import it into Epic and run it natively. Or you can produce a web service that I call out to and gives me an answer. We could work together directly. However, there are some limitations in what I'm allowed to tell you or share with you about Epic's data model and what Epic perceives to be their intellectual property.

And it is facilitated by you joining this program. Because if you join this program, you get access to documentation that you would otherwise not have access to. You may get access to a test harness or a test system that lets you sort of validate your work. However, people who join the program often think that means that I can then just run my app at every customer, right. But with Epic, in particular, you have to then make a deal with me to use it at the



Brigham and make a deal with my colleague to use at Stanford.

Other EHR vendors have developed a more sort of centralized model where you can actually release it and sell it, and I can pay for it directly through the app store and integrate it. I think that last mile piece hasn't really been standardized yet.

**AUDIENCE:** I guess one of my questions there is, what happens in the case that I don't want to talk to Epic at all? And just I looked at your data and just like Brigham and Women's stuff. And I build a really good model. You saw how it works, and we just want to deploy it.

**ADAM WRIGHT:** Epic would not stop us from doing that. The only real restriction is that Epic would limit my ability to tell you stuff about Epic's guts. And so you would need a relatively sophisticated health care provider organization who could map between some kind of platonic data, clinical data, model and Epic's internal data model. But if you had that, you could.

And at the Brigham, we have this iHub Innovation Program. And we're probably working with 50 to 100 startups doing work like that, some of whom are members of the Epic App Orchard and some who choose not to be members of the Epic App Orchard. It's worth saying that joining the App Orchard or these programs entails revenue sharing with Epic and some complexity. That may go way down with these new regulations. But right now, some organizations have chosen not to partner with the vendors and work directly with the health care provider organizations.

**PETER SZOLOVITS:** So on the quality side of that question, if you do develop an application and field it at the Brigham, will Stanford be interested in taking it? Or are they going to be concerned about the fact that somehow you've fit it to the patient population in Boston, and it won't be appropriate to their data?

**ADAM WRIGHT:** Yeah, I think that's a fundamental question, right, is to what extent do these models generalize, right? Can you train a model at one place and transfer it to another place? We've generally seen that many of them transfer pretty well, right. So if they really have more to do with kind of core human physiology, that can be pretty similar between organizations. If they're really bound up in a particular workflow, right, they assume that you're doing this task, this task, this task in this order, they tend to transfer really, really poorly.

So I would say that our general approach has been to take a model that somebody has, run it retrospectively on our data warehouse, and see if it's accurate. And if it is, we might go

forward with it. If it's not, we would try to retrain it on our data, and then see how much improvement we get by retraining it.

**PETER**

And so have you in fact imported such models from other places?

**SZOLOVITS:**

**ADAM WRIGHT:**

We have, yeah. Epic provides five or six models. And we've just started using some of them at the Brigham or just kind of signed the license to begin using them. And I think Epic's guidance and our experience is that they work pretty well out of the box.

**PETER**

Great.

**SZOLOVITS:**

**AUDIENCE:**

So could you say a little bit more about these rescores that are being deployed, maybe they work. Maybe they don't. How can you really tell whether they're working, even just beyond patient shift over time, just like how people react to the scores. Like I know a lot of the bias in fairness works is like people, if a score agrees with their intuition, they'll trust it. And if it doesn't, they ignore the score. So like how-- what does the process look like before you deploy the score thing and then see whether it's working or not?

**ADAM WRIGHT:**

Yeah, absolutely. So the question is, we get a risk score, or we deploy a new risk score that says, patient has a risk of falling, or patient has a risk of having sepsis or something like that. We tend to do several levels of evaluation, right. So the first level is, when we show the score, what do people do, right? If we-- typically we don't just show a score, we make a recommendation. We say, based on the score we think you should order a lactate to see if the patient is at risk of having sepsis.

First we look to see if people do what we say, right. So we think it's a good sign if people follow the suggestions. But ultimately, we view ourselves as sort of clinical trialists, right. So we deploy this model with an intent to move something, to reduce the rate of sepsis, or to reduce the rate of mortality in sepsis. And so we would try to sort of measure, if nothing else, do a before and after study, right, measure the rates before, implement this intervention, and measure the rates after.

In cases where we're less sure, or where we really care about the results, we'll even do a randomized trial, right. So we'll give half of the units will get the alert, half the units won't get the alert. And we'll compare the effect on a clinical outcome and see what the difference is. In

our opinion, unless we can show an effect on these clinical measures, we shouldn't be bothering people, right. Pete made this point that what's the purpose of having-- if we have 1,000 alerts, everyone will be overwhelmed. So we should only keep alerts on if we can show that they're making a real clinical difference.

**AUDIENCE:** And are those sort of like just internal checks, are there papers of some of these deployments?

**ADAM WRIGHT:** It's our-- it's our intent to publish everything, right. I mean, I think we're behind. But I'd say, we publish everything. We have some things that we've finished that we haven't published yet. They're sort of the next thing to sort of come out. Yeah.

**AUDIENCE:** I guess so earlier we were talking about how the models are just used to give recommendations to doctors. Do you have any metric, in terms of how often the model recommendation matches with the doctor's decision?

**ADAM WRIGHT:** Yeah, absolutely.

**AUDIENCE:** Can you repeat the question?

**ADAM WRIGHT:** Oh yeah. Thanks, David. So the question is, do we ever check to see how often the model recommendation matches what the doctor does? And so there's sort of two ways we do that. We'll often retrospectively test the model back. I think Pete shared a paper from Cerner where they looked at these sort of suggestions that they made to order lactates or to do other sort of sepsis work. And they looked to see whether the recommendations that they made matched what the doctors had actually done.

And they showed that they, in many cases, did. So that'll be the first thing that we do is, before we even turn the model on, we'll run it in silent mode and see if the doctor does what we suggest. Now the doctor is not a perfect supervision, right, because the doctor may neglect to do something that would be good to do. So then when we turn it on, we actually look to see whether the doctor takes the action that we suggested.

And if we're doing it in this randomized mode, we would then look to see whether the doctor takes the action we suggested more often in the case where we show the alert, than where we generate the alert but just logged it and don't-- don't show it. Yeah. Yes, sir?

**AUDIENCE:** So you'd mentioned how there's kind of related to fatigue--

**ADAM WRIGHT:** Yeah.

**AUDIENCE:** --if it's a code blue, these alarms will--

**ADAM WRIGHT:** Right.

**AUDIENCE:** And you said that cockpits have-- pilots now--

**ADAM WRIGHT:** Yeah.

**AUDIENCE:** --that have similar problems. My very limited understanding of aviation is that if you're flying, say, below 10,000 feet, then almost all of the--

**ADAM WRIGHT:** Yeah.

**AUDIENCE:** --alarms get turned off, and--

**ADAM WRIGHT:** Yeah.

**AUDIENCE:** --I don't know if there seems to be an airlock for that, for--

**ADAM WRIGHT:** Yeah.

**AUDIENCE:** --hospitals yet. And is that just because the technology workflow is not mature enough yet, only 10 years old?

**ADAM WRIGHT:** Yeah.

**AUDIENCE:** Or is that kind of the team's question about the incentives between if you build the tool and it doesn't flag this thing--

**ADAM WRIGHT:** Yeah.

**AUDIENCE:** --the patient dies, then they could sued. And so they're just very--

**ADAM WRIGHT:** Yeah, no, we try, right? So since we often don't know about the situations in a structured way at the EHR. And so most of our alerts are suppressed in the operating room, right? So during an-- when a patient is on anesthesia, their physiology is being sort of manually controlled by a doctor. And so we often suppress the alerts in those situations.

I guess I didn't say the question, but the question was, do we try to take situations into account

or how much can we? We didn't used to know that a code blue was going on, because we used to do most of our code blue documentation on paper. We now use this code narrator, right? So we can tell when a code blue starts and when a code blue ends. A code blue is a cardiac arrest and resuscitation of a patient. And so we actually do increasingly turn a lot of alerting off during a code blue.

I get an email or a page whenever a doctor overrides an alert and writes a cranky message. And they'll often say something like, this patient is dying of a myocardial infarction right now, and your bothering me about this influenza vaccination. And then what I'll do is I'll go back-- no, seriously, I had that yesterday.

And so what I'll do is I'll go back and look in the record and say, what signs did I have this patient sort of in extremis? And in that particular case, it was a patient who came into the ED and very little documentation had been started, and so there actually were very few signs that the patient was in the acute state. I think this, someday, could be sorted by integrating monitor data and device data to figure that out. But at that point, we didn't have a good, structured data at that moment, in the chart, that said this patient is so ill that it's offensive to suggest an influenza vaccination right now.

**PETER**  
**SZOLOVITS:** Now, there are hospitals that have started experimenting with things like acquiring data from the ambulance as the patient is coming in so that the ED is already primed with preliminary data.

**ADAM WRIGHT:** Yeah.

**PETER**  
**SZOLOVITS:** And in that circumstance, you could tell.

**ADAM WRIGHT:** So this is the interoperability challenge, right? So we actually get the run sheet, all of the ambulance data, to us. It comes in as a PDF that's transmitted from the ambulance emergency management system to our EHR. And so it's not coming in in a way that we can read it well.

But to your point, exactly, if we were better at interoperability-- I've also talked to hospitals who use things like video cameras and people's badges, and if there's 50 people hovering around a patient, that's a sign that something bad is happening. And so we might be able to use something like that. But yeah, we'd like to be better at that.

**PETER** So why did HL7 version 3 not solve all of these problems?

**SZOLOVITS:**

**ADAM WRIGHT:** This is a good philosophical question. Come to BMI 701 and 702 and we'll talk about the standards. HL7 version-- to his question-- version 2 was a very practical standard. Version 3 was a very deeply philosophical standard--

**PETER** Aspirational.

**SZOLOVITS:**

**ADAM WRIGHT:** --aspirational, that never quite caught on. And it did in pieces. I mean, FHIR is a simplification of that.

**PETER** Yeah.

**SZOLOVITS:**

**ADAM WRIGHT:** Yes, sir?

**AUDIENCE:** So I think usually, the machine learning models evaluates the difficult [INAUDIBLE].

**ADAM WRIGHT:** Yes, sir.

**AUDIENCE:** When it comes to a particular patient, is there a way to know how reliable the model is?

**ADAM WRIGHT:** Yeah, I mean, there's calibration, right? So we can say this model works particularly well in these patients, or not as well in these patients. There are some very simple equations or models that we use, for example, where we use a different model in African-American patients versus non-African-American patients, because there's some data that says this model is better calibrated in this subgroup of patients versus another.

I do think, though, to your point, that there's a suggestion, an inference from a model-- this patient is at risk of a fall. And then there's this whole set of value judgments and beliefs and knowledge and understanding of a patient's circumstances that are very human.

And I think that that's largely why we deliver these suggestions to a doctor or to a nurse. And then that human uses that information plus their expertise and their relationship and their experience to make a suggestion, rather than just having the computer adjust the knob on the ventilator itself.

A question that people always ask me, and that you should ask me, is, will we eventually not need that human? And I think I'm more optimistic than some people that there are cases where the computer is good enough, or the human is poor enough, that it would be safe to have a close to closed loop. However, I think those cases are not the norm. I think that there'll be more cases where human doctors are still very much needed.

**PETER SZOLOVITS:** So just to add that there are tasks where patients are fungible, in the words that I used a few lectures ago. So for example, a lot of hospitals are developing models that predict whether a patient will show up for their optional surgery, because then they can do a better job of over-scheduling the operating room in the same way that the airlines over over-sell seats.

Because, statistically, you could win doing that. Those are very safe predictions, because the worst thing that happens is you get delayed. But it's not going to have a harmful outcome on an individual patient.

**ADAM WRIGHT:** Yeah, and conversely, there are people that are working on machine learning systems for dosing insulin or adjusting people's ventilator settings, and those are high--

**PETER SZOLOVITS:** Those are the high risk.

**ADAM WRIGHT:** --risk jobs.

**PETER SZOLOVITS:** Yep. All right, last question because we have to wrap up.

**AUDIENCE:** You had alluded to some of the [INAUDIBLE] problems--

**ADAM WRIGHT:** Yes.

**AUDIENCE:** --of some of these models. I'm, one, curious how long [INAUDIBLE].

**ADAM WRIGHT:** Yeah.

**AUDIENCE:** And I guess, two, once it's been determined that actually a significant issue has occurred, what are some of the decisions that you made regarding tradeoffs of using the out-of-date model that looks at [INAUDIBLE] signal versus the cost of retraining?

**ADAM WRIGHT:** Retraining? Yeah. Yeah, absolutely. So the question is the set-and-forget, right? We build the

model. The model may become stale. Should we update the model? And how do we decide to do that? I mean, we're using-- it depends on what you define as a model. We're using tables and rules that we've developed since the 1970s. I think we have a pretty high desire to empirically revisit those.

There's a problem in the practice called knowledge management or knowledge engineering, right? How do we remember which of our knowledge bases need to be checked again or updated? And we'll often, just as a standard, retrain a model or re-evaluate a knowledge base every six months or every year because it's both harmful to patients if this stuff is out-of-date, and it also makes us look stupid, right?

So if there's a new paper that comes out and says, beta blockers are terrible poison, and we keep suggesting them, then people no longer believe the suggestions that we make, that said, we still make mistakes, right? I mean, things happen all of the time. A lot of my work has focused on malfunctions in these systems.

And so, as an example, empirically, the pharmacy might change the code or ID number for a medicine, or a new medicine might come on the market, and we have to make sure to continually update the knowledge base so that we're not suggesting an old medicine or overlooking the fact that the patient has already been prescribed a new medicine. And so we tried to do that prospectively or proactively. But then we also tried to listen to feedback from users and fix things as we go. Cool.

**PETER  
SZOLOVITS:**

And just one more comment on that. So some things are done in real time. There was a system, many years ago, at the Intermountain Health in Salt Lake City, where they were looking at what bugs were growing out of microbiology samples in the laboratory. And of course, that can change on an hour-by-hour or day-to-day basis. And so they were updating those systems that warned you about the possibility of that kind of infection in real time by taking feeds directly from the laboratory.

**ADAM WRIGHT:** That's true.

**PETER  
SZOLOVITS:**

All right, thank you very much.

**ADAM WRIGHT:** No, thank you, guys.

[APPLAUSE]



