MIT CSAIL Alliances | Tim Kraska podcast

Welcome to MIT's Computer Science and Artificial Intelligence Labs Alliance's podcast series. My name is Steve Lewis. I'm the Assistant Director of Global Strategic Alliances for CSAIL at MIT. In this podcast series, I will interview principal researchers at CSAIL to discover what they're working on, and how it will impact society.

Today, on our podcast, I'll be speaking to Professor Tim Kraska. Tim is an Associate Professor of Electrical Engineering and Computer Science in MIT's Computer Science and Artificial Intelligence Lab, the co-founding Director of Data Systems and AI Lab, known as DSAIL at MIT, and the co-founder of Einblick Analytics, Inc., which we'll talk about later on in the podcast.

Welcome, Tim. Can you tell the listeners a little bit about the focus of your research, and some of your bold aspirations?

Of course. Thanks, Steve, first of all, for having me here. So my research area is either in applying machine learning for systems. So we are looking, currently, into how we can improve systems, but we could make them instance optimize through machine learning.

And the other big area is how we built systems to make machine learning or data science, in general, easier to use. Like how we can be broader that more people can take advantage of their data, and enable more people to do data-driven decision-making.

I see. And toward that end, you had a project called Northstar. Can you tell us a little bit about the goal of Northstar?

For sure. So Northstar was a research project that started roughly, seven years ago. And the original motivation was that we wanted to build a system, which works on this interactive whiteboards like the Microsoft Office Hub, the Google Jamboard, and so on.

And then over time, it evolved into a project to make data science more generally accessible to a broader range of users, particularly, people, which now what termed citizen data scientists. So this is a term by Gartner. And because Gartner realized there is simply not enough data scientists to go around, just like at the same time, everybody wants to have them. They are like more and more data appearing, everybody wants to do data-driven decision-making.

So the big question is like how do we enable all these people to have the right domain expertise, who know the problem really well, but maybe, are not like computer scientists or mathematicians, and have the necessary Python skills to do data science, but still enable them so that they can analyze the data. And this was the motivation behind Northstar. And over time, just became a company for Einblick.

Actually, Einblick is completely based on Northstar. So it's an official MIT and Brown's spinoff. So the Northstar project ventures several phases. Like when we started out, it was like, yeah, we targeted just like intereactive whiteboards, like the Microsoft Office Hub, the Google Jamboard.

You can think of them as like large, attachable TVs you put on the wall. And we saw that people started to use them for video conferencing, as well as like having a shared byproduct experience in different remote locations. But we always thought, hey, that could be so much more. So why can we not create a data environment that people can work together, like in a minority type of fashion.

So we developed the first prototype of that. And then deployed at a bunch of companies. And the main feedback was like, oh, yeah, that's great, but we are not in a meeting room all the time. And so based on that, we created the second version of the software. Now, we also got funded by DARPA because DARPA had this grand vision of putting data scientist in every single operational unit.

We got four more companies that they wanted to do something similar. So we created the second version of the software as a prototype. We've developed a complete new back end because we also found that the system is not sufficient for supporting these types of interactions the users were making.

We developed new AutoML tools, and we deployed the software, again, at a bunch of companies. And this time, the feedback was very different. Instead of saying, there's something fundamentally missing that we cannot do, it's more like, oh, feature request--like, we want to have a form of dashboard, we need to have exports of different formats, and other things.

And that was the time when we said, yeah, we really want to see the software live, and that people use it. At the same time, it feels wrong to use PhD students to do the work for it. And so, we decided to do a spinoff, which is called Einblick to support another software in a commercial setting, as well as also for governments.

Yeah, if the listeners out there want to go to einblick.ai, there's a video on the main page, which gives you a good feel for the UI and the simplicity of it. But yeah, I think it gets to be very, very powerful. And that's very exciting. So switching gears a little bit, can you tell me what's new and exciting in the domain of database design and architecture?

Of course. Like one topic we are particularly interested in right now is just like how to use machine learning to improve systems. So if you think about traditional systems, normally, if you design a system, you target a whole range of use cases.

Let's assume you develop a new database, you built the data warehouse for all types of retailers, manufacturers, like pick your favorite industry, and the same type of system will deploy that. The reason why we built systems at base-- building a system from scratch takes a long, long time. So all I'm saying is just like it takes seven years, for example, to develop a more or less stable database.

The downside is because of this general behavior of the system that attack with so many workloads, we probably don't get the best possible performance out of the database for particular use case. So let's assume like you would develop a system only for hallmark, only for this modern dashboard, and you know it runs only on a Monday, probably, you would design the whole system very, very differently. And you wouldn't make the compromises you have to make in designing a general purpose system.

And so the question we are asking ourselves is like how can we leverage machine learning to build something which is instance optimized. So a system which self-adjusts based on the workload, as well as the data.

That's interesting. So do you use machine learning to enable that more efficient design of that system?

Yes. Machine learning is like one big component of it. So in the end, we use whatever works, and it doesn't have to be machine learning. It can often be just like traditional optimization techniques. But machine learning, obviously, plays an important role because it gives us a tool to navigate this really large subspace of potential configurations and options.

In other cases, it goes further than that. Because sometimes, it's possible to replace traditional components of a system through a model. So for example, we did some work, where we show that machine learning models can replace traditional B3 indexes. Or we have another line of work, where we show that a machine learning model can replace a query optimizer.

And now traditional, more headquarter designs are entirely replaced by a model. But what we use, essentially, depends always on the problem we are trying to solve.

I see. So I assume that that model is more efficient and faster than running down the whole B3. I mean, what types of speed improvements or optimizations are you seeing using a model versus a traditional approach?

There are often several advantages to get from it. Like in the case for indexing, for example, the index lookup is a little bit faster in certain cases. But particularly for index, the big difference comes from the size of the index. So model, like how it works is we use the model to represent the data through continuous functions. And in some form, this does a higher form of entropy compression, which can be much, much more compact.

And now, if the index is more compact, you can fit more data into my memory, and that has like significant performance advantages. So there's a recent paper, for example, by Google, which shows like if you integrate just non-index factors into big table, they got up to 50% better throughput numbers, mainly, because they are so much smaller.

But we want to keep it further. So sometimes, the size, the lookup performance gets better. But the other important aspect is also you get to like self-adjustments. For example, the query optimizer-- the big advantage there is that the query optimizer starts to learn from mistakes, and self-adjust based on that. And that's something traditional techniques normally don't do.

I see. And is available commercially? Is this open source? Or is it still in the lab?

So this is roughly, still in the lab. We are currently building a new system, which we call SageDB, where we are trying to integrate everything. And we're also planning, potentially, to make an open source version of it. There are parts of the techniques we developed, so far, which are all open source.

So if you go to our DSAIL website, that's dsail.csail.mit.edu, we actually have a list of all the code we published so far. So you can try out, for example, our learn query optimizer for post-grads. Just download it, and test it.

That's great. Let's talk about an important issue these days-- privacy. And is it possible to design a privacy preserving database?

It's a very tough question. I think it is possible. It's definitely possible. The question is how useful would that database be. We made it some work with like a project called SchengenDB in that area. But this is not really about privacy. It's more like that we are compliant as GDPR.

It's just slightly different in the sense of that GDPR, for example, has a requirement if a user comes in, and asks for "please delete all the records related to me from the database", you have to comply to that. And even that simpler problem, how to make sure that you can delete all the records for a given user is not trivial.

For example, if you have a model of the user information as part of it, do you also need to delete the model because the model, arguably, contains some information about the user?

Yes. And what about log files, and things like that? I can imagine that the hole goes deep as far as that is concerned. So can you talk a little bit more, in detail, about SchengenDB, and how it may be used to, at least, comply with GDPR or protect privacy?

Yes. So SchengenDB is just like what's the design concept because we thought like, OK, what needs to change in order for database to be fully GDPR compliant. And it turns out, it's actually more complicated than you might think.

And you just mentioned the right point here-- it's just like the logs. So for example, if you have a transactional database, normally, everything is updated in the database, but the ground truth-- the one that just full tolerant and reliable is the log.

So everything is stored in the log, and you always use the log in order to, for example, if something goes run recover. But that also means that the log contains, for example, all the user information because the update something is written to the log.

So now, if you want to be really fully GDPR compliant and the legal definitions are not 100% clear there, it's just like-- for example, one of the questions is do you need to modify the log, which is used to be a total no go. Because if you change something of the log, and you do it wrong, you might not be able to use the database afterwards anymore. You cannot recover.

And this is where the complication comes in. Is this really needed-- legal advice? Yeah, actually, we talked to experts in the field, and even they don't know. Because the definitions talk about reasonable efforts. But if the ruling becomes like, yeah, logs are a problem too, then we need to redesign all the systems and have to actually be GDPR compliant.

Yeah, I could see that would be a really big challenge for sure. But interesting work that is going on in that field. Where could our listeners find out more about SchengenDB?

So we have a paper, which was published at the workshop. That's probably the best starting point. Or simply, reach out to me any time. It's kraska@mit.edu. And I'm more than happy to answer any questions about SchengenDB or any of the other projects.

Very good. So what is the future of the database design, of database architecture systems?

I think performance is still definitely a strong research area. You might think about this database is not already fast enough. And my answer to that is always that data, right now, is increasing at an unprecedented pace by essentially Moore's law ending.

And so we need to come up with new methods to deal with the increase in data, and still be able to efficiently analyze everything we get, because nobody wants to delete data ever. Just like normally, you don't want to do that.

And so on one hand, like an exciting area, definitely, it's like instance optimized systems like applying machine learning to improve systems and tailor them for a given workload and data distribution. I'm personally, very excited about that area.

But there are other interesting topics I can think of. For example, how to make analytics more accessible to a broader range of users. So Einblick is one example of it. But there are many new approaches as well to choose, like natural language processing. So that you can just ask your question in normal language, and something should come back.

There are some commercial systems right now, but they are very, very limited. For example, you can ask a question like, show me the sales in California, and it would show you a shot. But then the follow up question about, oh, why did my sales drop-- none of those systems, right now, could answer. And so that's another very exciting area, I think.

And then of course, the expansion to new types of data. For example, video, in particular, is very interesting, or images-- how do you build systems to efficiently search and analyze video data. There's so many cameras nowadays, and let alone in the Boston area. And maybe, you want to answer query like, how many cars every day go over red light.

And that's very, very hard to sync those techniques. So it requires some mix of machine learning to understand the video data, but then also like more traditional processing techniques to answer just analytical question to get them to enter quantitative measure, hopefully, with some error bounds.

That would be exciting for sure. Can you talk about anything you've done with our member companies through the CSAIL Alliance program?

Yes, for sure. Particularly, the Northstar project with the CSAIL Alliance Program has been great. Probably, all our initial customers came either through the CSAIL Alliance Program or one conference we did. So they're super, super helpful.

And we are always looking for partners in industry to tryout our software. Everybody has benefits from it because the industry part, I can try out just new type of software, and see if it actually works for them, get all the access to it. We also learn a ton by seeing how people would use it, and what the real problems are.

The thing that just comes to my mind is like CIBC, the bank. We are just starting again a small talk series with them to just show them on how Northstar and now, Einblick works. And this came, definitely, out of the CSAIL Alliance Program. So it has been great so far. And we would love to even get more involved than we used to far.

That's excellent. Well, Tim, it was great to talk to you. Thank you for your time today.

Thanks, Steve.

If you're interested in learning more about the CSAIL Alliance program and the latest research at CSAIL, please visit our website at cap.csail.mit.edu. And listen to our podcast series on Spotify, Apple Music, or wherever you listen to your podcasts. Tune in next month for a brand new edition of the CSAIL Alliance's podcast and stay ahead of the curve.