Bruke Kifle: This is ACM ByteCast, a podcast series from the Association for Computing Machinery, the world's largest education and scientific computing society. We talk to researchers, practitioners, and innovators who are at the intersection of computing research and practice. They share their experiences, the lessons they've learned, and their own visions for the future of computing. I'm your host, Bruke Kifle. The rapid evolution of artificial intelligence and data management has redefined how we access, interact with, and make sense of the vast amount of information available in the digital age. At the heart of this transformation are knowledge graphs, an innovation that connects and organizes disparate data into structured meaningful insights. These powerful systems enable machines to understand complex relationships between data points, opening new frontiers in search, personalization, question answering, and beyond.

From early breakthroughs in data integration to pioneering the creation of knowledge graphs at scale, our next guest, Dr. Xin Luna Dong has been at the forefront of this field for over two decades as a world-renowned expert in knowledge graph and data integration. Luna is a principal scientist at Meta Reality Labs leading the ML efforts in building an intelligent personal assistant, innovating and productionizing techniques on contextual AI, multimodal conversations, search question answering, recommendation and personalization. Prior to joining Meta, she spent nearly a decade working on knowledge graphs at Amazon and Google, and another decade on data integration and cleaning at AT&T Labs and at University of Washington where she received her PhD in computer science. She is the recipient of various awards including the ACM fellow and IEEE fellow, Dr. Xin Luna Dong, welcome to ByteCast.

Xin Luna Dong: Thank you. Thank you very much for the intro.

Bruke Kifle: I'm very excited for this conversation. I'd love to open it up with sort of an open question just to understand what are some of the key inflection points in your personal and professional journey that have inspired you to pursue a career in computing and specifically in knowledge graphs and data integration?

Xin Luna Dong: Nice. That's a very good question. So it's actually a whole bunch of things happening naturally one after another that eventually lead me to do computing and eventually lead me to do knowledge graphs. So let's start with computing. So I was born in China and I grew up there. When I was eight years old, that's the time the country is still poor and I remember we still need tickets to buy rice and we can buy fish twice every month. And so there is one day my mom, she worked for middle school and she told me they got a computer. So that is, I think it's called a COM35 and another one is Apple II. So they got two personal computers and she told me, "Hey, you know what? You can come to play video games." So that's my first interaction with computers. And then I was in my third grade in elementary school and after playing the video games, very simple ones, I started learning coding, programming.

And again, that was very hard for me because at that time I haven't learned English. And I remember I look at the very simple code like computing the sum from one to 100, and I look at the code and I have no idea what it means and why it works. And at that time, I remember when I see I-F, if, I don't know what its meaning of that, but basically I learned, okay, this means there are two branches and the T-H-E-N goes to one branch

and the E-L-S-E goes to another branch and similar for other commands. So that's how I got started. It's just the magic to me. And even though most of the time, as I recall, I just typed in the code letter by letter, number by number, but then seeing the results is fascinating. And then gradually I started understanding why it works and I coded and started participating in those coding competitions.

And I remember the turning point is high school. So when I was at high school, I was starting thinking about what I should do for college and for future career. And there were many suggestions, proposals from my friends, from my parents, but nobody said, computer science or doing coding is a good job. And at that time I was hesitating and my computer teacher, coding teacher handed me some book explaining the A+ algorithm, sorry, A* algorithm. And then I realized, oh, there is some way to give computers some intelligence and it is not just a fast and traverse the whole tree towards all of the solution space.

It can actually do some very smart cutoff and do something smart to find the solution. It's so fast, it is so smart, and it could do something that is really amazing. So I remember that A* algorithm, and that's the point when I started thinking, "Okay, maybe I will just do this for college." And then I got my bachelor degree on computer science, then master's degree, and then PhD on computer science. That's how it naturally goes to computing as my career.

Bruke Kifle: That's-

Xin Luna Dong: Yes. Yeah, that is about computing.

Bruke Kifle: That's an amazing life story and I think it's quite exciting hearing you say that you learned programming before you learned English. And so [inaudible 00:06:32] coding language to actually help you learn the language as well is actually quite such a unique experience. Beyond your entry or journey into the computing profession more broadly, what prompted some of your interest in knowledge graphs and your study?

Xin Luna Dong: Yeah, so this somehow also is related to my childhood. So when I grew up before I went to elementary school, again, the family is so poor that I don't remember I have many books. Maybe a handful of books I could read, but I did not have books of my own. And I also remember when I went to elementary school, at some point, I think that's my third grade, finally we got this library card and this was a huge gift to me. Why do I mention this? Because we don't have books, it's so hard to get to information, to get new information. So whatever questions you have, you don't have much to read to understand it. And I remember we have newspapers, so I remember my mom oftentimes will cut some of the newspaper articles and then paste it to some other old newspapers or magazines or whatever, and that's how we collect information.

So I would say in my childhood, even until mid-high school, it is kind of this crave for information, how to get to understand more information, get more information, and get some information to answer my questions? That's always this craving, how do I get that? And then suddenly, I think it is at the time when I went to graduate school,

suddenly everything changed and we found okay, on the web there is so much information and you can't easily find what you want. And that is actually pre-Google time. And with all of this, there is this idea of I want to get all of the information, I want to organize them in some way that I can easily find things I like. And those things are, I would say subconsciously.

Then other set of coincidences, I got an offer from UW, my advisor is Alon Halevy. He worked on data integration and I was assigned as his temporary student for my first year of PhD. And then gradually learned what he is doing and found it fascinating. And all of this come together that I started working on data integration. And after I work on data integration for almost a decade, so that is a time when there has been this knowledge card launched on Google search. And that is the time I started knowing knowledge graphs and knowledge integration, which I would say is a natural extension of what I have been working on data integration. And then I came to this field.

Bruke Kifle: Wow, that's such a beautiful story. And I think certainly it being grounded in your personal journey and your personal desire from a young age for knowledge, for information, and now being able to pioneer essentially a lot of the work that's enabling knowledge and information discovery for millions of users, billions of users at a global scale, I think is quite a beautiful journey. But with that, I actually want to learn or dive a bit deeper on some of your work on the creation of knowledge graphs at scale. Obviously you've had an impact at Google, at Amazon, at Meta. For our users or for our audience that may not be familiar, could you maybe describe what is a knowledge graph? Why is it relevant? What does it do? What does it help us accomplish? And maybe in the context of some of the everyday products or services that a lot of people are used to using, what are some of the most interesting or impactful use cases in products like Google and Amazon and Meta?

Xin Luna Dong: Sure. This is a fair question. So a knowledge graph, it is a graph, so it has nodes and edges. Each node represents an entity, a real world entity, and each edge represents the relationships between the entities. And a knowledge graph is beautiful for two reasons. First, it is in the graph structure and so it is structured and it is kind of mimic how people understand the world, how human beings understand the world, entity and the relationships between them. And it makes it easier to understand information and to query information. That's number one. Number two, knowledge graphs also have good reputation in terms of the quality, quality in terms of the richness of the knowledge and also the cleanness of the knowledge. It is highly accurate, high coverage. And so this basically is a good store and a giant store of high quality information. So how has it been changing our daily lives?

The first example, and that's also the first success for knowledge graphs is the knowledge panels in search engines for Google for Bing, when you search something like Obama's wife, you will see a knowledge panel on the right give you the information about Barack Obama as an example. And nowadays, because the Google knowledge graph has really grown in the past decades and for a lot of search queries, you will see this knowledge panel which put all of the basic information there in the form that is very easy to understand. And the second example I could give is my work at Amazon, and this is also to build a knowledge graph, but for products, and there are two examples why it

is useful. One is for digital products, because the knowledge graphs helps normalize information, find the relationships between the entities, we are able to, when we build the knowledge graphs, we are able to connect the low resolution and the high resolution songs, for example, music tracks.

And one use case as an example is for the users of Amazon Music and they could sign up to listen to the high resolution songs. And because we understand the relationships of the songs with the different quality, we can make sure we always serve the high quality ones when it is available. And if not, we then serve the medium or lower quality songs. So that comes from the normalization part and the relation part of the knowledge graphs. Another usage is that for all of the products, it's very hard to figure out all of the information. And as we build the knowledge graph, again, we generate the attribute value pairs and show that at the Amazon detail pages.

And finally, coming to our work at Meta. So here we are building smart assistants and one assistant as an example, is on the wearable devices. It's called Ray-Ban Meta, and it's some glasses you can wear and you can ask questions to the glasses. And when you questions, this is basically question answering and it needs to pull information from different sources to answer the questions. And we found using the knowledge graphs, we can reduce the latency of QA, question, answering by one second, and we can also improve the quality of this answer generation when we use large language models. So here are the several examples.

Bruke Kifle: I think that's, you really described the importance of this technology, but also just how widespread it is in the day-to-day products that we use, whether it be music streaming, product shopping, or even things like search. I'm quite a familiar with the search space having worked on the Microsoft Bing product. And so knowledge graphs were a very integral part of the experience that you described with the knowledge panel. So I think it's quite exciting to see how much of a foundational core technology this is for discovery and information access. One thing that came to mind as you were describing obviously a big part or a core foundation of knowledge models or knowledge graphs is clean, high-quality, high-fidelity data. And in this digital age, there's a lot of data. The ability to extract, label, and actually build these knowledge graphs I presume is very challenging. And so how have some of the challenges associated with data extraction, cleaning, labeling, and integration sort of evolved, especially with the age or the rise of machine learning and AI techniques? Have you found it to improve? Is it a process? Has it facilitated sort of the knowledge graph process?

Xin Luna Dong: That's a very good question. So let's first see there have been different generations of methods in terms of extracting, integrating, and cleaning information. The first one, I would call it runtime data integration. So in a sense, web search is runtime. You ask a question, a query, search query, and then you see 10 blue links and then you look at them and figure out your answer. And in parallel to that, the database community comes up with this data integration ideas where you get one query, this is a database query, and that is translated into the queries that could be understood by the underlying data sources. And their answers are retrieved, sent back to the middle point and answers are unioned and returned to the users. So that's two decades ago, and that is kind of this runtime data integration.

A knowledge graph provides this offline integration. When we build the knowledge graphs, in a sense we are assembling, integrating all of the information oftentimes in heterogeneous forms, putting them together, normalize it, and then serve it at runtime. This makes a lot of hard work done at the offline time. So I would call that the second generation. So as we have all of the new AI technologies and machine learning methods, we kind of get the tool to improve each step. So in addition, we get one new generation of data integration. I would call that a data internalization or knowledge internalization into the large language models. And when we train those large language models, they get a lot of data from the web and try to internalize the popular knowledge which occur often on the web into the large language models.

So this is kind of a different way of internalize, sorry, integrating the information. So that's kind of the third generation of data integration. So to recap, two ways that the machine learning models and the large language models are sort of evolving, data extraction, data cleaning. The first one is basically to give new tools to generate better extraction cleaning results. And the second one is to provide a whole new generation of methods for data integration.

Bruke Kifle: That's very exciting. I think it's quite exciting to see how the rise of machine learning and AI techniques are driving improvements in how we do extraction, cleaning, labeling, and integration. But I think there's also the overarching question of at present, we're seeing a lot of impressive results with large language models that are revolutionizing natural language processing, a lot of the core tests. And I think you touched on it with some of the work at Meta with the Ray-Ban glasses. So in your view, how do you see knowledge graphs fitting into the future of the tech landscape? Do you see them as complimentary to LLMs? Obviously you described the use case of LLMs or tools to help generate data and cleaning and labeling, but as it pertains to the actual uses in some of the core technologies, whether it be search, whether it be personal assistance, do you see [inaudible 00:20:41] and knowledge graphs as complimentary? Do they serve distinct roles? Where do you see these two coming together?

Xin Luna Dong: Very good question. So let's say what people are expecting early last year. So that's the time suddenly everyone is aware of gen AI, aware of large language models and hoping large language models can do everything and providing smooth conversations in QA. So at that time, the hope is whatever questions we ask, large language models will answer it. To achieve that goal, basically it requires large language models to have all of the knowledge, word knowledge, and of course it does not happen. And the last year we did some study and we found large language models have very low quality in answering questions with dynamic information changing every second, every day, or even every year for those questions. Large language models have very low quality in terms of question answering. Large language models also are not good at answering questions regarding torso to tail entities. And surprisingly, I would say oftentimes less than 1% of the entities are tail entities, sorry, are head entities.

In other words, for 99% of the entities, they fall in the bucket of torso to tail, less popular or not popular at all. And large language models do not have rich knowledge about them and often hallucinate when answer questions about them. And the third thing is in some specific areas, for example, biology, medicine and large language

models do not necessarily have all of the information. Even for basic things like the taxonomy of the concepts, large language models are not good at them. So even though large language models are very good at generating the answers, understand the texts, it does not have all of this information. And so in future, I would guess, hypothesize that first large language models will continue to be a very good interface to interact with users, answer questions, understand the user's needs, and in addition, it will continue to have better and better reasoning capabilities, and so can answer complex questions.

Third, it will have more and more knowledge, but it may not get all of the word knowledge, especially the factual information and even the taxonomy information. It may not get all of that internalized in the model itself and it will then resort to knowledge graphs and maybe some other data sources for such information. I would use an analogy. So just like human beings, we have some knowledge in our head and we could reason and we can, for example, when we write an article, we can do pretty good job. However, there are often information, numbers, dates that we cannot remember and then we need to refer to some external data sources. And knowledge graph will serve as one of such important data sources.

Bruke Kifle: ACM Bytecast is available on Apple Podcasts, Google Podcasts, Podbean, Spotify, Stitcher, and TuneIn. If you're enjoying this episode, please subscribe and leave us a review on your favorite platform. I think you capture a lot of the challenges at present and certainly looking into the future as well with LLMs as sort of a modality or sort of entry towards general intelligence, but also identifying some of legitimate shortcomings as it pertains to having a knowledge of entities, having a knowledge built in that is fully comprehensive of the world. And so really calling the need or value for knowledge graphs I think is a very core argument here. I'm curious, as you look to the future, what emerging, of course, alongside some of the developments with LLM performance, what emerging technologies or trends you're most excited about in terms of the value of knowledge graphs, but also improving knowledge graph creation, whether it be multimodal AI or progress that's being made in that direction? So what are some emerging technologies that you think will have some critical value in the application, but also the creation of knowledge graphs?

Xin Luna Dong: Sure, sure. So I view as my mission to help people access information and I call it provide the right information at the right time. And this basically requires a few things. The first one is we really need to provide relevant and accurate information. So that's the first part. And the second part is we need to be able to provide information in various modalities and we need to understand the stuff in various modalities, like viral information and the context information. And the third part is when we provide such information, we also want to provide in a way that is personalized to address the user's needs. So related to this, I would say there are a few things that I find it very fascinating. I'm super interested and I'm also hoping to contribute to it. And the first one, of course, this is RAG. It's basically retrieval-augmented generation and how to have the large language models retrieve information from valuable data sources, including knowledge graphs and then generate answers, recommendations, et cetera to the user, to answer user questions.

So I personally have been working in this field in the past, I would say nearly 18 months. And it's a lot of work to do. It sounds very natural and simple at the first glance, but if we use the obvious methods, it just does not give us the best results. And this year we came up with this benchmark called the CRAG, Comprehensive RAG Benchmark, and we use it to host the KDD Cup competition. And we also used it to evaluate all of the state-of-the-art RAG systems from Google, from OpenAI, from Bing, and from ourselves as well. And the quality, we see the improvement from baseline RAG solutions to the competition, the KDD Cup competition solutions and to state-of-the-art results. But if we give a score, the score is 0.5 out of one. So we are only halfway there. So there are still a lot to do to improve everything.

So that's one area I'm very interested in. And I can see once we make solid progress on that, we can change people's experiences in terms of this getting information and addressing their information needs. So that's the first one. And the second one I'm super interested is how to really build such information to allow effortless access to proprietary data. Let's say I have some enterprise data or I'm a small business owner and I have a small catalog or I'm for this field and I have information about this particular small field and how can I easily serve the data to external people? And oftentimes those people are not technical savvy and it will take them tremendous amount of efforts to build their own QA system, but how will we be able to have something that is generalizable and can sort of access the information they have in their own storage and access it and use it to answer questions?

So this is the second topic I'm super interested in, and this is again related to RAG. The third thing I'm interested in is personalization. So in a sense, I don't know if you have heard of this like Memex vision, this was from 1945 as far as I remember. And the idea is someone wear a camera on their forehead and record whatever they see for their lifetime and this digitalize their life and they can sort of ask any question about their past. And in the sense we have made a lot of progress on this, but on the other hand, we are not able to do this yet, but we are getting closer and closer to this. And will it happen that eventually we have an assistant that would view the world from our own perspectives, look at what has been happening to us and then use it to answer our utility question as well as provide personalized answers.

As an example, we recently launched this feature, RBM Smart Glasses. Basically you could, at the time you park, you could say, "Okay, remember this parking lot number?" And then later on when you come back to find your car, you can ask for, "Oh, where did I park?" And there are many, many such cases where you could remember your past and how can we make it even more intelligent? This whole personalized information system, management system, recommendation system to basically build the second brain of people, that will be fascinating as well. And finally, what I want to mention is this contextualization and how do we contextualize everything, QA recommendation, understand the user's context and contextualize the service, provide proactive services. So that will be very interesting as well. That's a long answer.

Bruke Kifle: No, no, I think that's quite exciting. As you're describing some of the work or your future vision for personalization with the ability to effectively have an assistant or a second brain, I think I was just really fascinated by what a future like that might look like, but I

think a lot of great, great visions for what the future might look like in this space, whether it be the personalization, whether it be how we build systems that are more contextualized or stateful to users, and thinking about RAG as sort of a key enabler for that and recognizing that certainly there's a lot of progress to be made with retrieval-augmented generation, but also thinking about how we can bring that to proprietary data that may not be publicly available or on the web.

And so I think those are all critical unlocks for improving the way humans interact with technology and providing additional value and then ensuring that this is accessible in multiple use cases as well. So I think you described a lot of interesting things. I'm curious, are there any upcoming projects or interesting areas of research at Meta Reality Labs, or I guess you touched on some of them here, but any developments within maybe the broader AI community that you're particularly excited about, perhaps within the knowledge graph space, data integration space, but also just more broadly, any interesting directions that keep you up at night and are quite exciting for you?

Xin Luna Dong: Sure. So I have been mentioning about all of this, like factuality, personalization, contextualization, multimodality, all of these are interesting projects I have been working on. I'm super excited about. I do want to come back to mention one more thing. So we have been talking about data integration. This is a problem that different communities have been working on for decades and it's not a solved problem. And nowadays, when we have data from different data sources with heterogeneity on the schema, on the form of the data, we still have difficulties to seamlessly integrate them. But I'm hoping eventually in the next decade, maybe not long, in the next decade, we will be able to provide some seamless fusion and integration of data. And it is not just the data itself, it is the seamless fusion of data and models. And yes, and here models are gen AI models, large language models.

So some of the data will be internalized into the large language models, gen AI models, some of the data will stay at their regional form. And we don't necessarily need to do a lot of data manipulation, data massaging, and some of the data will be put together into something like knowledge graphs. So I kind of feel this is a field that is so hard and we haven't found a solution yet, but with knowledge graph and large language models are coming in space, and we might be able to get there in the next decade to really provide this seamless, I call it do neural knowledge. Basically we have knowledge in symbolic forms, in knowledge graphs, and also in neural forms, in large language models. And then people can seamlessly access them through the large language models. I'm fascinated by that vision, and I hope that could happen soon.

Bruke Kifle: I think certainly with people like you driving the future of this area, I have no doubt that that will be possible. But I think you touch on a very exciting future for the role of data in driving a lot of the progress that we hope to continue to see in AI. As we near the end of this episode, we have a lot of audience members or audiences around the world who are interested in taking inspiration from the journey of amazing researchers, practitioners such as yourself. And so what advice would you give to young professionals, young researchers who are interested in computer science, maybe even knowledge graphs or information systems as they look to embark on a career or sort of a profession in this space?

Xin Luna Dong: Sure. That's a great question, and I think I have two suggestions. So the first one is always keep open-minded. So I started working on data integration in the year of 2002. It's a little bit over 20 years. And the technologies have evolved so much, improved so much, and the tools we used at that time was extremely different from the tools we used 10 years ago and is very different from the tools we use now. And there have been so many changes. And to stay on top of it, to sort of always make progress and to contribute to the renovations, it's very important to always learn. And there are always a lot of things to learn and how to manage that. I would say my method is to first go deep, so I find an area that is relevant and then I go quite deep.

And after that, this go deep, meaning I read a bunch of papers. It also means I do some of my own research. So I have fairly deep understanding of this small area or maybe a reasonably big area. And after that, broaden it and go deep again. So this kind of brought me from data integration to data quality, meaning integration plus cleaning, to knowledge integration, and then to knowledge graph construction, all of the cleaning, integration, extraction world. And then to all of this knowledge graph construction, knowledge graph application, and smart assistance. I feel like going deep, broaden, going deep, broaden, this allows me to learn a lot of new stuff to gradually achieve the goal I had from the very beginning. So that's one thing about keep open-minded. Another thing about keep open-minded is as we enter a field, we often learn something and then form some hypothesis.

And for example, I grew up from the database community and I started with thinking that structured data is the best way or is the way that people use to store their data, to access their data. And with those hypothesis, it might limit what I could do. And related to this, keep open-minded, meaning oftentimes jump out of the box and re-examine all of the hypothesis. So for example, honestly, last time when I changed my job, when I moved from Amazon to Meta, I chose a field that is not necessarily directly related to knowledge graphs. Knowledge graphs is a part of it, but a small part. And I just want to see to serve end users, are knowledge graphs absolutely needed and in which way, and what are other sources, information sources or methods that are critical? I don't want to just limit myself thinking knowledge graphs are the only way to do it.

So I think I really benefited from that trial. It is not always easy, but this allows me to broaden my scope to, it kind of opened a new door for me. So that's my first pieces of advice. And the second one is, interestingly, it's almost the opposite. Focus, focus, focus, focus. And I personally have been active in multiple different research fields like database, data mining, and recently NLP and adding multimodal as well. And also, I have been working as an scientist in industry, so I do research, write papers, and the meanwhile, I work on productionizing technologies, building features. And I did go through the different steps like building up prototypes and then develop technologies and then pushing the last mile to get things out. And it's a big diversity of the stuff. But on the other hand, I feel it's both learning and the lessons.

The learning is for all of the stuff I have been doing, there is one theme into it, how to help people access information easily. And because of that, although it could be things from different communities, from different industry versus research, but it all come under the same theme. So there is a focus there, and it is still much easier for me to

grasp information from neighboring communities, neighboring fields, and to enrich my tool set. The second one I would say is learning. Sometimes I got ambitious and I want to do everything and gradually I realize, okay, here is my passion, here is my strength and I have limited time. Life is short. And really, really drill down to what excited me and also what I'm good at.

Bruke Kifle: That's such an amazing set of pieces of advice accumulated over such a rich career as a researcher, as a practitioner, as a developer of products used by millions. And I'll just quickly synthesize them, it's a balance of both having a focus, so in your particular case, the central theme of knowledge, discovery, and access to information. But within that, keeping an open mind, whether it be out of the box thinking to examine sort of the work that you're doing and the problem that you're solving, but also in having sort of a lifelong learning mentality. And so exploring depth, but also equally exploring breadth. And so I think that's a great set of advice for those looking to explore a career in computing or more broadly just discover their life passion and their life career as well. Dr. Luna, we just want to say thank you for joining us on ByteCast. This has been an amazing discussion and we certainly look forward to the future impact that you will continue to drive in your line of work.

Xin Luna Dong: Thank you very much.

Bruke Kifle: ACM ByteCast is a production of the Association for Computing Machinery's Practitioner Board. To learn more about ACM and its activities, visit ACM.org. For more information about this and other episodes, please visit our website at learning.acm.0-R-G/B-Y-T-E-C-A-S-T. That's learning.acm.org/bytecast.