Meral Özsoyoğlu Speaks Out

on genealogical data management, searching ontologies, and more

by Marianne Winslett and Vanessa Braganholo



Meral Özsoyoğlu
http://zmo.cwru.edu/

Welcome ACM SIGMOD Record's series of interviews with distinguished members of the database community. I'm Marianne Winslett, and today, we're in Providence, site of the 2009 SIGMOD/PODS conference. I have here with me Meral Özsoyoğlu, who is the Andrew Jennings Professor of Computing in the Department of Electrical Engineering and Computer Science at Case Western Reserve University, where she is also the department head. Meral's research interests lie in databases, bioinformatics, and pedigree data management. She is currently the Editor-in-Chief of ACM Transactions on Database Systems and a trustee of the VLDB Endowment. Her PhD is from the University of Alberta, in Canada. So Meral, welcome!

Thank you!

So first, I have to ask: what is pedigree data management?

Actually, perhaps a better term for it is genealogical data management, rather than pedigree. Pedigree is apparently more commonly used for dogs. But what we mean is family tree information, like parents, grandparents, ancestors, descendants. Pedigree data management is becoming very important in medical informatics. There are many publications and a lot of interest in using family history in clinical decision making. So pedigree, or, genealogical, data, especially for hereditary diseases, is very important, and all clinical institutions are collecting this data. There are not extremely large volumes of pedigree data currently, but it is growing very

fast. It is in the form of an acyclic directed graph, and, when you have large volumes of these acyclic directed graphs, querying them becomes an issue.

I am very much interested in query processing, making it scalable and efficient. So, I thought pedigree data management and querying would be a great topic to work on. Also, there are a lot of opportunities in medical informatics, and Case has a medical school where there are several life science researchers. So, think about a very large directed acyclic graph of genealogical information. Currently, this data is stored in the form of a relational database, and each patient, each person has mother and father information, and, in order to find ancestors having a particular disease, you have to do it by record at a time, which is not scalable for large volumes of data. So, we thought that if we have a special graph structure for pedigree data, and then if we link it to the actual record data, we can identify all the ancestor IDs, and then fetch that data right away as opposed to finding the father, and then finding the father of the father, or parents of the parents, and so on. In fact, we use certain kinds of encoding for this data, and then directly find all ancestors or all, let's say, second degree relatives, and queries like that, or find the inbreeding coefficient…

What's that?

Inbreeding coefficient indicates whether there is inbreeding in the genealogy or the pedigree of an individual. This is computed by finding the common ancestors. There are actually formulas for that. In fact, the formula for calculating the inbreeding coefficient that genealogists use is recursive. But computing it recursively, if it is large, is very time consuming. So we used a path counting formula for the inbreeding coefficient by just counting the paths to common ancestors; but then you have to eliminate duplicate (or overlapping) paths. And, this way, we have a significant improvement in the performance of calculating the inbreeding coefficient.

Another thing is kinship. There are these kinship coefficients. So again, we adopted the same path counting formula for the kinship coefficient, and we published that in the CSB conference last year, and then it was invited to the Journal of Bioinformatics and Computational Biology. So again, since there is not much real pedigree data that is large…

Even from animals?

Animals, yes. But we were collaborating with the Cleveland Clinic Foundation, and they have a hereditary colon cancer, called FAP. Apparently, Cleveland Clinic has the largest pedigree data collection on this disease. It is called Jagelman's registries, and it is the largest in the world. But then, when you look at it from a database researcher's perspective, it is not large at all. So we did our experiments on real data to show how scalable and how better performing our methods are, as compared to the traditional approaches. We weren't satisfied with it since the data was not very large. So we also generated some synthetic pedigree data using a PhD thesis from Iceland on this; where you can adjust parameters like the number of children, life expectancy, and things

like that, so you can get as deep a pedigree as you choose. And, even for very large pedigree data, our methods show significant improvement.

So for example, for that colon cancer database, how many different people do they have in there?How big is big in that domain?

I don't remember the exact numbers now. But I think it was something like hundreds of patients. Because what happens is that one person with the disease comes to the hospital, and, then they start to build the pedigree based on that individual, called progenitor. So they try to get the progenitor's parents, and, then, children and relatives. This is how they grow these family trees. If you think about it, it is hard to grow it with more than a hundred individuals for a particular family. So there are these isolated family trees which are not trees really: most of the time, they have lots of (undirected) cycles. Then, of course, these family trees occasionally merge as well.

We are also using similar techniques in bioinformatics for searching ontologies, for example, Gene Ontology, which we use in PathCase, a metabolic pathways database. Do you know what a metabolic pathway is?

I do, but maybe our readers may not all know.

There are a large number of metabolism-related reactions taking place for organisms, or humans, to live. Reactions have input metabolites, called substrates, output metabolites, called products, catalyzing enzymes, activators, inhibitors and so on. A metabolic network of reactions is viewed not as a graph, but a hypergraph. And the reactions of various metabolisms such as the carbohydrate metabolism or the lipid metabolism form a metabolic network. And, the network is really placed within a compartment hierarchy such as mitochondria inside cytosol, and so on. Genes relate to the metabolic network as gene products function as enzymes, and organism hierarchy also relates to metabolic networks as well. PathCase metabolic pathways database captures metabolic networks of organisms, humans included as graph database, and relationships to genes and organism hierarch as well. Users can zoom in and zoom out, viewing the metabolic network graph at different levels of abstraction. And they can query it, because it is a graph after all! You can search for paths, neighborhoods of a particular metabolite, or a reaction. You can also collapse complete pathways, such as glycolysis, into individual nodes and look at the network of pathways where the nodes themselves are pathways. This was the start of PathCase project, an NSF-supported project.

We are now working on the next phase of this project, called PathCase for Systems Biology. Original Pathcase assumed a static network. But then there is also the dynamics of it. Systems biologists model the dynamics of individual pathways, subnetworks of pathways, or just reactions using various kinetic models. These models are difficult to produce and verify. Nevertheless, their numbers are in hundreds to thousands in different data sources. Our new project maps systems biology models to/from the full metabolic network, and allows systems biologists to search, query, and visualize model and network information together.

So I see, so the main CS research issues that you took out of this biological problem had to do with the search and indexing aspect?

Exactly, and also the querying component.

Your encoding that you mentioned, is that like a Dewey encoding, or what kind of encoding is that?

(Yes. It is a prefix based encoding like Dewey encoding.) We call it nodecodes. In fact, a long time ago, when working in multimedia databases, we were dealing with searching acyclic directed graphs; so we used some sort of encoding of paths in such a way that, given two nodecodes for two nodes, you can find the paths between two nodes directly by looking at the nodecodes, not by traversing the graph. We used this idea for pedigree searching, with some differences. In terms of pedigrees each node has a biological mother and father, so you know that the in-degree of each node is definitely two. There are many optimization issues as well. For example, for pedigree, sometimes we collapse each family into a single node, that has a significant improvement on the performance.

How can a young computer science professor advance the state of the art in biology, while still building a strong reputation in CS?

This is a very good question! Currently I see many computer science researchers learning biology; at least many of my PhD students also take biochemistry; they also take bioinformatics, systems biology and computational biology courses. This requires significant time and effort. And, you need a really good collaborator. But again, there are a lot of funding opportunities and research challenges in the intersection. So I think collaborative research is very time consuming, but, at the same time, it is very rewarding.

What happened to all that work that people did on nested relations?

My answer might be biased, but I think all the work that was done on nested relations actually formed a foundation for XML, XML querying, and XML storage and so. In fact, today I just saw a demo in SIGMOD demonstrations that was utilizing relation nesting. So I think that is the nature of research: you work on certain topics and it becomes so popular that people keep producing results without truly knowing what will come of these results. And then, after some time, those results become useful and timely in another context, and people start using them. If you think about it, we now see datalog or rule-based systems research coming back; they were very active about 10-15 years ago; there was this quiet period; and now it is back.

You did your undergraduate and master's degrees in Turkey. Is it true that as many women as men get degrees in computer science, in Turkey?

I'm not sure if it is as many, but my sense is that definitely a larger percentage of women are going to engineering and sciences in Turkey than in the US. I did my sabbatical in 1991-1992 in

Turkey at Bilkent University, teaching the same courses I teach at Case. About 35-40% of students in Turkey were women, and, at Case, that was a much lower percentage.

So why do you think that is?

I think it is most probably because in Turkey there is a university entrance exam; everybody has to take this university entrance exam. And students are placed to colleges and disciplines based on their scores. Highest score areas are engineering and medical school (computer science is also very high), then comes sciences. So once a girl, a female, gets a high score, nobody questions whether that area is right for her or not. In here, one of my friends told me that "you know, I really liked math very much, but I didn't think going to computer science or sciences would be the right thing for a female". I never felt that way when I was growing up in Turkey. Here they always have the question: is this right for me? I never asked that question to myself.

Well, what about after you came to the US, did the different attitudes make you ask that question?

It didn't make me ask that question; it surprised me. I was the first women faculty member in my department, and the second women faculty member in the history of Case Engineering School at the time--we are talking about 29 years ago. Several people thought I was a secretary, and were asking me where the professor was, and I kept explaining them "I am the professor". But the attitudes surprised me. I didn't think whether this is right for me or not.

Have you been any other firsts?

I was the first female faculty member in our department. Then, when I became a department chair two years ago, I was the first female department chair in our department, and also the first female department chair in the history of the engineering school. I am also the first female editor-in-chief of ACM TODS.

Why are there so many Turkish database researchers?

I didn't think that there are too many.

Not too many, but a lot!

Still, more than what it used to be. I remember a PODS conference in 1983, I think in Atlanta. I remember some of my Greek colleagues teased me that there are so many Greek database researchers, but not many Turkish database researchers. So I am happy that there are now more Turkish database researchers! As far as I remember, the first Turkish database researcher who published internationally was Esen Ozkarahan. He did his PhD in Toronto on database machines. This was late '70s, I think. He is the first that I know. And then, I think, me, Tekin (my husband), and Tamer Ozsu. And there are now many younger Turkish researchers: Ugur Cetintemel, Selcuk Candan, Fatma Ozcan, Nesime Tatbul, and many others. That's very good.

Do you have any words of advice for fledgling or midcareer database researchers?

This is a good question. I will say this: databases is a great field because data is everywhere. So it is a never-ending… You can always find challenging problems. I remember a time where, in database conferences, we were thinking "oh, this is the end of databases; all the problems are solved". But it is not. So I would say, just keep on looking for challenging problems, and the ones that you like--that is very important, you want to enjoy what you do. There are many challenging problems to be solved.

Among all your past research, do you have a favorite piece of work?

From my research?

Yes. Something you produced, and if you don't have a favorite, that's fine too.

I think my favorite piece of work was determining tree query membership and query optimization using semi-join reductions. That was my very first paper. I was a PhD student, and I published it in COMPSAC conference in 79'.

If you could change one thing about yourself as a computer science researcher, what would it be?

As a researcher?

Well, I could say, as a computer scientist.

I think I would like to be better at coding. I wish I was a better programmer.

Thank you very much for talking with me today.

Thank you!