

# THE DATA ANALYTICS

HANDBOOK

---

BIG DATA EDITION

---

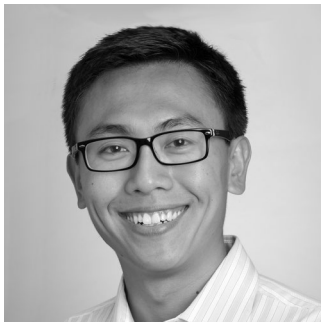
# ABOUT THE AUTHORS



## BRIAN LIOU Content



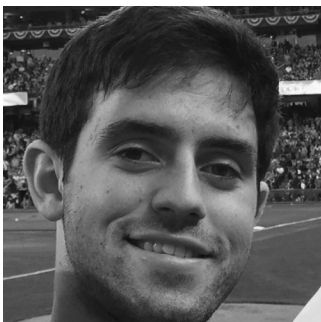
Brian graduated from Cal with simultaneous degrees in Business Administration at the Haas School of Business and Statistics with an emphasis in Computer Science. He previously worked in investment banking before he transitioned into Data Analytics at MightyHive, an advertising technology company backed by Andreessen Horowitz.



## TRISTAN TAO Content



Tristan holds dual degrees in Computer Science and Statistics from UC Berkeley ('14). He first began working as a quantitative technical data analyst at Starmine (Thomson Reuters). From there he worked as a software engineer at Splunk. He has experience working with various Machine Learning models, NLP, Hadoop/Hive, Storm, R, Python and Java.



## DECLAN SHENER Content

Declan is in his third year at UC Berkeley where he studies Computer Science and Economics. He currently interns with San Francisco 49ers as a Data Analyst on their Business Operations team.

# HANDBOOK DESIGN



## ELIZABETH LIN Design



Elizabeth holds a degree in Computer Science with a focus in design from UC Berkeley ('14). Previously, she was a teaching assistant for CS160, User Interface Design and Development. She has interned at Khan Academy as a Product Designer and at LinkedIn as a Web Developer. [www.elizabethlin.com](http://www.elizabethlin.com)

**H**ave you ever wondered what the deal was behind all the hype of “big data”? Well, so did we. In 2014, data science hit peak popularity, and as graduates with degrees in statistics, business, and computer science from UC Berkeley we found ourselves with a unique skillset that was in high demand. We recognized that as recent graduates, our foundational knowledge was purely theoretical; we lacked industry experience; we also realized that we were not alone in this predicament. And so, we sought out those who could supplement our knowledge, interviewing leaders, experts, and professionals – the giants in our industry. What began as a quest for the reality behind the buzzwords of “big data” and “data science,” *The Data Analytics Handbook*, quickly turned into our first educational product. Thirty plus interviews and four editions later, the handbook has been downloaded over 30,000 times by readers from all over the world. The interviews in this edition are ones we have selected as the most insightful based on our own growing experience in the industry. In them, you’ll discover whether “big data” is overblown, what skills your portfolio companies should look for when hiring a data scientist, how leading “big data” and analytics companies interview, and which industries will be most impacted by the disruptive power of data science.

From Tristan, Brian, Declan, & Elizabeth

# TOP 5 TAKEAWAYS

## THE BIG DATA EDITION

### 1. The terms Data Scientist and Data Engineer are not synonymous, but they are not mutually exclusive either.

There exists a hybrid position that requires both a software engineering background as well as mastery of statistical analysis, but that is not the norm. Instead, expertise in one of the two with a good understanding of the other is important. As a data engineer, you will be delivering the data that a data scientist uses, and vice versa. Knowing how your data is being used and understanding your data are crucial to success in either of these positions.

### 2. The fundamentals are important.

Although projects are a great way to showcase your skills, it is more important to have a firm grasp of computer science and statistical concepts. This knowledge is especially important in interviews, where you will likely be asked questions requiring the fundamental knowledge of algorithms and data structures to solve.

### 3. Know your data.

Even though data analysis platforms are rapidly improving, they are far from perfect. Databases are complex and often data is imperfect. Understanding your data and how to manipulate it is the first step to leveraging it effectively.

### 4. Big Data can be used for more than solving business problems.


Don't think that Big Data can only be used to help a business make financial decisions or target consumers better. Big Data will also be used to solve industrial issues such as energy and food shortages. The reality is that there is no industry that Big Data will not touch eventually.

### 5. Check out Spark

Apache Spark is the hottest big data technology developed at UC Berkeley's AMP Lab. It runs programs up to 100X faster than Hadoop MapReduce in memory or 10X faster on disk. It's one of the most contributed open source Big Data technologies and has high level abstractions to enable users to perform large scale data processing in Spark SQL.



## TABLE OF CONTENTS

- 05** MICHAEL JORDAN  
DISTINGUISHED EECS PROFESSOR
  - 09** CHUL LEE  
MYFITNESSPAL
  - 13** JOHN AKRED  
SILICON VALLEY DATA SCIENCE
  - 18** MATT MCMANUS  
DATAMEER
  - 21** JOHN SCHUSTER  
PLATFORA
  - 24** TOM DAVENPORT  
BABSON COLLEGE
- 

## MICHAEL JORDAN

### DISTINGUISHED EECS PROFESSOR AT UC BERKELEY

MICHAEL is the Pehong Chen Distinguished Professor in the Department of Electrical Engineering and Computer Science and the Department of Statistics at the University of California, Berkeley. His research in recent years has focused on Bayesian nonparametric analysis, probabilistic graphical models, spectral methods, kernel machines and applications to problems in signal processing, statistical genetics, computational biology, information retrieval and natural language processing.

## Where do you think the biggest talent fall is within the Big Data Industry?

You have to know something about statistics and you have to know something about computer science. Out in industry, they find it hard to find such people.

## Do you need a graduate level degree in order to be one of those people?

Not necessarily. It depends on what you're doing, but again you do need both a solid background in computer science and statistics. Current undergraduate curricula don't allow you in four years to really get both, unless you're really ambitious and do nothing but that. Unfortunately, classes haven't really been designed to include both computation and statistics. It's not the ideal situation, universities need to change too to blend the education of both. Right now you have to take about half a dozen statistics classes and about half a dozen computation courses of some kind on top of prerequisite courses for those majors. That would be enough to get you a good job, where you could do a good job and have a meaningful career. One way to do that might be to get a masters; either get a bachelor's degree in either computer science or statistics and get master's in the other.

## Undergraduate statistics curriculum tend to be theoretical, at least most universities. How does this affect the way students are preparing to fulfill the roles in Big Data?

Everywhere it is; and they are not very computing centric. I think there is one good class at Berkeley , [Concepts of Computing with Data], which brings together some amount of computing with statistics -- and that's it. Most of the other classes us the computer rarely, if at all. So you get the principals, but you don't really work with algorithms and procedures on real data a lot. Also, you don't realize that a lot of the principals can be implemented in a computationally interesting way without having to do the math. An example is that in industry there is something called the bootstrap, which is a way to get measures of uncertainty on many statistical estimates without using a formula by using a computer to simulate data. That's not taught anywhere in the undergraduate curriculum, which is crazy.

## In many cases data engineers are in charge of creating systems that data analysts use to analyze the data. Do you



## think that these systems will ever be powerful enough to simply do their own analysis?

Not in general. It's really hard to do good inferences. You can't just push buttons. You have to think about: how is the data sampled, how heterogeneous is the data, what do I put together with what; the features to represent the data; what procedures do I use and how. It requires human judgement through many of the stages. Think about databases as a kind of point and case. A real world database requires a large amount of skill to use, and that's just to move data around a query it. Think about the inferential issues about predicting and sampling about how you get the data and what it's being used for. It's just a much broader class of things. If already in classical databases it's hard to automate everything, you have to have humans turning them, it's going to be far harder for inferential problems. There's a narrow class of problems, it may be involved in clustering and classification where it kind of get closer to just pushing a button, but for better or for worse there's going to be a lot of jobs where you're going to have to understand what you're doing in the data analysis and what engineering approaches are feasible in that domain. Of course, data analyst will get help as the interfaces will become better, platforms will be easier to use, systems become more scalable, and more standardization gets put in place. These will eliminate a lot the grunge work that goes on such as collecting data, which is where a lot of work goes in now.

## Do you think the demand for these positions is overhyped?

No the demand is not overhyped. I think the demand for certain methodologies is overhyped of what you can and can't achieve. Some of the business plans are little crazy and overhyped, but the demand from my perspective is infinite relative to the supply. Every company is looking for a data analyst or data scientist for the foreseeable future. I don't see that going away because I don't think that problems are going to be solved that quickly. Also, this is kind of scalable. Every company has data and every company can use data in their plans somehow. Any company who is going to use data at all seriously needs people who are working with the platform and who understand what data analysis is, and probably not just one person. I think as company's mature, I think company's are going to realize they're going to need a bigger staff than they find to do the data side of things. A lot of companies are willing to put a lot of money and effort into hiring programmers and not much into the data

analysis, and they're going to realize that it's a mistake.

### What Big Data technologies do you see becoming very popular within the next five years?

I don't like to say that there's a specific technology, I think that there are pipelines that you would build that have pieces to them. How do you process the data, how do you represent it, how do you store it, what inferential problem are you trying to solve. There's a whole toolbox or ecosystem that you have to understand if you are going to be working in the field.

### Do students who want to prepare for these kinds of roles need to find resources outside of the university if they are trying to get into quickly?

A lot of the resources outside are not giving you the foundational fundamentals you need that are mathematical, statistical, and computational. They're teaching you the toolbox without the understanding of the toolbox. I think supplementing coursework by working with real world datasets is productive. If you've got a solid set of fundamentals, working on projects is a great way to improve your skills.



Does your data engineering team recruit people without much work experience in the industry (i.e. undergraduates, etc.)? If so, what is the biggest downfall that you see in these types of applicants?

We don't expect fresh graduates to have a complete or thorough grasp of data science projects or skills. What we do care the most is their potential. We expect them to have a very strong preparation in the fundamentals of computer science and math. We care more about that than projects that they have completed.

Employers always say projects are important, but what types of projects specifically? What are the best resources for starting and doing these projects?

The best resource for starting or doing projects is doing school projects or research projects. I would rather see people who have worked on research projects than projects they did in their own spare time that don't really have any depth. One of the problems that I've seen quite a bit is that people have claimed to have worked on their own data science pet projects, for example, a Twitter sentiment analysis project but there is no true depth associated with any of these projects. Thus, it is very difficult for us to clearly assess whether or not these projects were truly meeting some requirements that are normally associated with large scale practical data science problems. In addition, I believe that internships are also very important because they expose you to real world problems and they give you a good understanding of what kinds of technology stacks allow you to excel in this type of environment.

How much of a statistical background does a typical data engineer at MyFitnessPal need to have? Do the titles "Data Scientist" and "Data Engineer" tend to be mutually exclusive or do employees tend to have a lot of crossover skills?

Data engineering at MyFitnessPal does not require a strong background in statistics. However, Data Scientists and Data engineers are not completely mutually exclusive because both require some basic knowledge of system architectures. Data Engineers have to be smart enough to understand what Data Scientists are talking about, but it's not necessary to have a heavy statistics background.

At MyFitnessPal, Data Scientist roles and Data Engineering roles are well defined since the emphasis for each role is somewhat different. However, the overall trend in Silicon Valley is that everyone is part of the engineering organization. Thus, as a data scientist, you need to have a basic understanding of computer science, specifically data structures and algorithms. This is especially true when you have to deal with large scale data since the computation aspect of it becomes very critical. You can always come up with a fancy algorithm, but when you actually try to apply that in practice, making sure that your algorithm scales is very important. You need to have a basic understanding of the system or algorithm aspect of the problem. Even though you may not end up implementing that by yourself, you still need to be able communicate with other engineers. Similarly, when you're a data engineer, you need to have a basic understanding of statistics, so when you're talking to a data scientist you can understand what they are getting across to you. This is for the sake of communication, and working with a team that consists of people with different background as opposed to a skillset that is absolutely necessary to get a data science or data engineering job at MFP.

**In many cases, data engineers are in charge of creating systems data analysts can use. How important is a statistical analysis background for creating these systems?**

The definition of Data Engineering at MyFitnessPal is the development of data products for users because we are a customer facing app. Again, data engineering emphasizes the engineering aspect of data product development as opposed to data science.

**What specific technologies does your data engineering team use? Do you expect job applicants to already have a strong grasp of these technologies or is there a fair amount of learning by doing?**

For offline big data technology stacks, we use Spark and Hadoop. For the actual service development, we use Scala. We don't expect every new employee to already have a strong grasp of these technologies before joining MyFitnessPal. What we care the most about is whether the candidate is smart enough and has a strong preparation in the foundations of computer science. If the candidate is smart and motivated, then he/she can pick up these new technologies very quickly. Note that tools change all the time, but fundamentals rarely change

over time.

Describe to me what the typical interview process is like for a Data Engineer. Additionally, what makes an applicant really stand out during this process?

Our typical interview process is very similar to that of other big companies in Silicon Valley. Our first interview is a technical phone screen consisting of coding and basic algorithm questions. If you pass that, the next interview is onsite. Our on-site interview consists of both technical and behavioral questions. For the technical part, we ask coding, algorithm, and math questions. For the behavioral part, we ask about past work experiences, personal and career goals. Once again, it is not much different than applying to be a software engineer in other companies.



With the emergence of Spark, Do you think there will still be a continued application of the traditional old-school batch-style MapReduce? Or will Spark simply be the new industry standard?

I'd disagree with you on calling batch-style Hadoop old school (laughs), but I definitely think that Spark won't completely replace the traditional batch-style disk based approach. There are situations where you don't quite have a use case for Spark. For example, a major retailer might want to know the difference between having Christmas on a Tuesday versus a Wednesday, but doesn't want to keep that deep historical data involved in memory all the time. These certain situations might not warrant the need to migrate all data to in-memory, and instead you might leverage Hadoop YARN. Moreover the spark cluster you need might be used for other critical real-time operations, which will compete with your historical analysis for resources. I think there might be enough cases where batch will stick around. I think it'll evolve into a more YARN oriented world where resources are managed and shared.

In your opinion, what is something that everyone is doing wrong?

I just came back from spending a lot of time with some large enterprises, and I'd say the biggest mistake is treating Big Data as a vendor selection process. Some company hears about Big Data and competitive business hype around it. Then they go "Ok, I need to go buy more Big Data widgets for my IT department and then I'll be doing Big Data". By contrast, where people are successful with big data is when they have a clear agenda and an business opportunity to leverage Big Data. Now, if you don't have any familiarity with the technology, you might need to explore and experiment on where to apply the technology. But there is a lot of noise coming from the vendor community who want to sell their products (which is understandable), and they have an agenda to position a product as "solving a certain problem". But in reality, in order to get to that point where you realize value, there is a lot to be done beyond the vendor products. Therefore the correct approach is to treat Big Data as a strategic investment in capabilities that are now viable, because these architectures allow you to do different things. What I see folks doing wrong is that they treat it as yet another vendor selection process in the IT department, but fail to relate that decision back to business objectives and imperatives.



## So if you're just trying to buy "Big Data" without knowing what to do with it, you're basically wasting money?

Well, of course you can learn things from experimentation. But if you don't learn where to point it, you'll be stuck with only some intangible value. People seem to get into a state of analysis paralysis where they've spent some time playing with Big Data, but still don't understand where to point it. We're doing a fine business called Data Strategy and Architecture Advisory, which helps you answer the question of "what should my organization be doing with my data and what does that mean in terms of required technology and capability investment?" This offering stemmed from our observation of the current state of affairs in the industry.

## From your personal experience, what are you most excited about? This could be about technology, trend, or people.

What I've always thought was so exciting was the opportunity to bring data and insight to decisions that have traditionally been made based on gut instinct and personal experience. I think when we take Big Data to the larger industrial and economic space, there will be opportunities for systematic improvements such as preventative and proactive scheduled maintenance. This applies everywhere from smart farming, to smart power distribution grids, to smart supply chains. This will have a profound affect on the economy in a perhaps less "sexy" way when compared to the consumer technology innovations (such as personalization etc). This is not to take away from the consumer technology. But when I think about the industrial enterprise context, I get really excited. Big Data gives us a new visibility to the parts of the world that have the most impact on people's lives.

## In your opinion, what is the most fascinating use of data science that your team has accomplished at SVDS?

I'll give you a couple of general example and then a few specific ones to paint you a clear picture.

We've spent a lot of time advising customers starting with the imperatives of their businesses; what can we do to take those objectives, map them to the technology and capabilities they need, and help them understand how to apply them. For example, we're working with a pharmaceutical client who is starting to take some Big Data approaches to their R&D department. It's

an exciting opportunity to introduce these approaches into an industry that has yet to adopt them widely. Another example is when we did some work with Edmunds.com (who kindly let us talk about it). We helped them build a pipeline of unstructured data that describes the features and options on new vehicles. So when BMW releases a new 4-series, it'll send over various extensive 400-pages PDF documents that describe all possible configurations for the car. Edmunds had a team of folks who focused on manually configuring data models to support the searches for relevant information (so the customer can run searches like "German coupe with a hefty engine"). The process used to take weeks, now it takes 1-2 days.

Specifically, the capability here is implementing Idibon (a NLP startup) into a data pipeline and building applications that would allow the customer to successfully exploit that capability. This is a combination of working with Big Data capabilities to bring new technology, coupling it with a robust processing pipeline that ingest lots of unstructured data, and chunking it up and feeding it to an classifier / entity resolver. The result is a process that changes the way in which the client does business. To my earlier comments, if you treat this as a vendor selection process, you might get a component of it, but you you'll struggle to come up with the specific working solution.

But when you treat this as an end problem and understand it as the road map to capability, it becomes profound. So now Edmunds has the capability to point at customers' reviews and start to get an understanding for things such as sentiment analysis. Now when you see a user like a BMW X5, you understand whether if it's because the user likes the styling, the handling, or the speed. So now the understanding of preference for style and features becomes another game changer enabled by the first innovation. This in turn can lead to other different useful results for the organization.

Given that SVDS is a niche Data Consulting firm, can you speak to how different it is to work there as a consulting data scientist versus working as a full time hired data scientist at a larger organization such as Google?

One is not better than the other, but there are important differences. There are people who are suited for one and not suited for the other. The benefit of working as a consultant is that we're able to go on a project (using a real example here) where they're building the capability to get a unified picture of inventory

across warehouses and retail stores. This involves heavy engineering, but also data science since people put stuff in their shopping carts and inventory is never where you think it is. As a result you need to understand the probabilistic aspect of the problem and things such as time to fulfill order based on the location of its destination etc.

So with us you might work on that for 3 months, but the next project might be optimizing interventions for a healthcare company [what are the best interventions to get a diabetic to adhere to the treatment regimen and regularly test the blood sugar level etc.] These are very different problem spaces. Of course they might leverage similar technology, but if you're curious about exploring different problems (and want to be at the forefront of that exploration), you get to see that variety with us. On the other hand, other people might instead want to get on the LinkedIn's famous "People You May Know" team. Certain aspects of that problem were solved a long time ago, and I imagine that the approach has remained relatively consistent. So of course you'll be working on the long tail of small incremental improvements. Some people love getting very, very deep into a problem like that.

We provide premium services to companies, and that means we have to lead the market in terms of the technologies we work with and how we implement them. For example, we currently work a lot with Spark. We've got Spark in production for a major US retailer over the holidays [Fall/Winter of 2014] and that's a pretty novel thing. Since we demand a premium in the market place due to our team and capabilities, learning these new technologies and capabilities is existential for us. We dedicate a lot of time to training our team and exploring that space.

In a large or more mature product organization, you might be 1 of the 20 people working on the Hadoop cluster (which you'll learn a lot), versus with us you'll be 1 of the 4 people who bring Hadoop to a new use case that has never done before. Of course, people learn differently and some people might do better in the first scenario. The fun thing is that we get to interact with the people doing similar things on those teams at product companies a lot, and we have a lot mutual respect for each other. It's just different strokes for different folks, as they say.



Does Datameer recruit people without much work experience, i.e. recent college graduates? What is a downfall that you tend to see in these types of applicants?

Yes we do. We have a major engineering office in Germany, so we do a lot of recruiting in that area. There are some special programs in Germany where you can get a masters working at a company. One of the major downfalls is when you work closely with students, you get a lot of the academic problems which is nice, but the biggest gap we're missing with those folks are software engineering skills. Actually being able to build products, think from a user perspective, direct your ability to do data analysis and implement great features with the focus on actually generating a product. We find we have to spend a lot of time with these people coming up to speed on general software engineering practices, usability, and delivering real product features. We have a pretty varied level of experience when it comes to working with these people. It's a different thing implement something in a side project than implementing a new feature on a full on enterprise product. Some people come on and pick this up fast and others don't. The biggest challenge is finding folks that have thought from an application and product perspective.

What's the best way to gain these skills?

Just doing it. It's interesting to see work on a project that is incorporated into a product, a user-facing interface, or a data visualization. When you're thinking a little bit above the algorithmic level, like how can this help a user or product. I think an internship where you're working on products and working with the software engineering part of it. There are a lot of people that come out of school that can do the math and the stats, but being in the line of fire how do you write code that can be maintained, worked on with a lot people, and delivered as a feature.

What type of tools do you use at Datameer?

We are one of the first native Hadoop analytics platforms. We recently abstracted ourselves a layer above the hadoop so now we can integrate with any type of execution framework, like Spark. We have a product that sits on top of a hadoop environment. We integrate pretty nicely with the entire Hadoop ecosystem.

Do you expect applicants to know Hadoop when they are applying to a position at Datameer?

Not necessarily. You have to show basic CS level competency, so distributed algorithms. In general, we find Hadoop is kind of easy, and as far as an API goes MapReduce is not that complex. If you can prove that you understand the distributed and algorithmic concepts, and have experience on a project that is user facing application that's good enough. We can teach you the Hadoop API if we need to.

Are the terms data scientist and data engineer mutually exclusive or do expect people to have a background in both statistics and software engineering?

I think some of our ideal candidates have both. What I'm seeing is that data literacy is very important. We're not looking for Machine Learning researchers, but having some background in statistics and data science is very important for our product. We look for a sweet spot where you have a statistics background and have experience in software engineering.

In terms of your product, do you think it's important for data analyst to pay attention to upcoming technologies like Spark when your product is supposed to abstract away these issues?

If you're looking to get hired by us on the engineering side you definitely have to pay attention to these new technologies. If you want to get a job with one of our customers doing the analysis, there is going to be a proliferation of tools coming to you, to put you into that position. We sell into people who want to do the analysis. We sell into armies of data scientists that don't want to write code, they want to think about the data and they need to tools to do what they do.

What is a trend you see with the company's adopting datameer? Who is the most hungry to analyze their data?

We have a wide variety across regions. On the east coast, we have a big interest in finance. All the big banks are building out massive hadoop clusters. They are doing lots of customer analytics, fraud detection, and these are viewed as traditional big data issues. On the west coast we have emerging companies in gaming, internet, mobile that are using newer technologies to create data products. So on the east coast, there is a more traditional use of this information whereas on the west coast we're seeing new companies use data in new ways.



## What is the typical interview process for a data engineering position at Platfora? What are you looking for in a candidate?

At Platfora, we have a pretty defined interview process. Usually, we meet candidates two or three times. Sometimes just the hiring manager will meet with candidate at first and decide if it makes sense to proceed to a first round interview and then a second round interview. Typically, the interview team sizes are about 6 to 8 people total, in some cases a little bit more for senior positions. For entry level positions, it would probably be around 6. There are actually a few different profiles. There is a software engineer who is programming and building products that process data in some way. The skillset required is a computer science background. Systems and parallel programming are big pluses. Some of the modern technologies like Hadoop as well.

A different profile is a Data Scientist who uses the product to analyze and understand the data. It's a totally different skillset like mathematics, statistics, and machine learning.

## Do you expect an engineering to have a statistical background? Afterall, they are building a product that data scientists will use.

It's definitely a plus. It depends on what part of the product that the engineer is working on. For example, at Platfora, there is a distributed query engine. The people who are working on the distributed query engine know a lot about parallel programming, MPP databases, and data processing in general. There are other engineers who are working on other parts of our system who might not be considered data engineers, but rather engineers working on a data related product.

## Can you give us an example of an industry product that a data engineer at Platfora would work on? Can you describe the process like who they report to and what tools they use?

The way that I have the engineering team organized, is that there is a server team which handles that master worker architecture and all the software that runs in that cluster. I also have a front end team which is handling our web application which is our interface to how you administer, configure, and use the product analyze the data. Our third team is the Automation, QA, and testing team. Those three teams together work in conjunction to build our product. Some of the tools we use GIRA for bug tracking for . A lot of engineers use intelliJ



as their development environment. We use stash and git for source control. We have a whole bunch of internally written scripts and different pieces of automation that build and test the product in a continuous integration sense.

Employers are always saying that they're looking for side projects that a candidate has done that relates to the product they are trying to build. Do you agree with this notion?

Personally, I am not very focused on projects. I am focused on the interview process for entry level positions. I believe in hiring really strong software engineers. It would be unusual that someone without much work experience to have done the exact right project that I'm looking to hire someone for. However, what I can say, there are certain projects that are nice to see on resumes. If someone has done something with hadoop or hadoop ecosystem technologies, such as MapReduce, Spark or Hive. Anyone of those would be an indicator that the person is interested in data or likes data processing. Again, it's not a requirement. Another example that has nothing to do with Hadoop or Apache, would just be general database experience and distributed systems experience.

Once data platforms gets powerful enough, will data related issues be abstracted away from the user?

That doesn't feel right to me. The more knowledge that an analyst has about the data the more effective he or she is going to be. I don't really believe that any products today are anywhere near where you don't need to understand the actual data, the structure of the data, and the problems with the data.

I believe the growth in big data and specifically big data analytics will be in the top of the stack, or the application layer. We've spent the last 5 to 8 years building the infrastructure to process big data. There are lots of great tools and technologies available on the market today. Typically when that happens, all that stuff in the infrastructure layer gets commoditized and people start building applications on top of it. I think that's the phase that we are currently in and that's where I see the biggest growth being in big data in the next 5 years.

Any advice you would like to add for someone interested in getting involved in the industry?

I've gotten all my best opportunities through professors and former bosses and colleagues. So building those relationships, and doing exceptionally well to get noticed is the main piece of advice I have.



It seems like in your book, you are very optimistic about Big Data and the advancements it will have in the world. There is an interesting point that Michael Jordan, a professor at UC Berkeley, brought up. He thinks that Big Data will return a lot false positives in the future. He parallels it to having a billion Monkey's typing at once, one of them will write Shakespeare. How serious is this concern for the future?

It's certainly true that if you are only looking at measures of statistical significance, working with Big Data is going to generate a lot of false positives because by definition a certain percentage are going to be significant. Some of those are going to make sense and some of them aren't. If all you're using is Machine Learning to generate statistically significant relationships among variables, that statement is true. I think it's one of the reasons why you need to use a little judgement about whether or not that finding makes sense and is there anything we can do with it. I think given the vast amount of data, we have to use Machine Learning, but I generally advocate that it's still accompanied by some human analyst who can help make sense of the outcomes.

In your book, you describe futuristic scenarios. Do you think it's as realistic to foresee major examples of disaster from the use of Big Data?

I think there are different levels of negative outcomes. There is the level of producing marketing offers that are not accurate and well-targeted. We all experience that everyday. It's no great tragedy and it wastes our time to get all this crap. So that's the benign but negative outcome. Then there's the more troublesome negative outcome around data; security and privacy breaches. We know that happens a fair amount, and it might happen even more, unless we take some dramatic steps to address it. The thing that I am most worried about is that various types of intelligent machines are going to replace fairly expert and well-educated knowledge workers in doing their jobs. I find that to be a scary prospect. These knowledge workers can augment the work of intelligent machines as opposed to be out of a job because of them. I'm not personally worried about AI taking over taking over humanity, but there are some quite smart people who are. I don't lose much sleep over it, but I do worry about loss of jobs.

In the case of Machine Learning tools replacing knowledge

workers. How should one educate him or herself on Big Data to avoid something like this occurring, or to avoid the mistakes that could occur from the use of Big Data and Machine Learning techniques?

I think if you are a highly quantitative person, then there is one course of action and that means understanding how these algorithms and meta-algorithms work. Knowing their limitations, understanding what the assumptions are behind them, knowing when the models need to be reexamined, and so on. If you are not a quantitative person and it's not a viable prospect to engage with the machine in a serious way, then I think you have to find something that a machine is not very good at. There are still a fair number of those, so you need to focus on that kind of thing. Maybe specialize very narrowly in something that nobody would ever write and automated system for. Focus on something that machines aren't good at like persuading humans to change their behavior as opposed to coming up with the right answer. I think, in many cases, computers are going to be better at coming up with the right answer. However, persuading people to act on that answer is a matter of clinical psychology. You could focus on persuading people to change their behavior. Even sales, you could call an example of that. You either have to either engage with the machines, or find something that they can't do very well, and make that your focus.

In terms of Big Data education for professionals, what area is lacking the most: implementation or management?

The number one area that companies complain about in terms of the people they hire is the poor ability to communicate about analytics. If we are going to be successful getting organizations to change the way they make decisions, you not only have to do the analytics, but you have to persuade someone to adopt a new way of thinking based on the data and analysis. I think the area that education is most lacking is on the communication side.

What is a good way to build up the ability to communicate technical concepts?

I think you have to think a lot about the time-honored practice of storytelling: what makes for a good story and what kind of story you're trying to tell. I wrote about this in *Keeping Up With the Quants* and classified different kinds of analytical stories. You have to make sure the story is one that appeals to your audience. If it's a highly technical audience then you can tell a highly

technical story, but chances are good it's going to be a non-technical audience. If this is the case, you have to use language that resonates with that specific audience. If you're in business, then it's mostly going to be a language of ROI, conversion, lift, and things that people are familiar with in that context. You have to spend a lot of time thinking of a clever way of communicating your idea. Use all the tools of storytelling, like metaphors and analogies, and provide as many examples of possible. Unfortunately, I don't think there is a lot of content out there about how to communicate effectively about analytics. I heard this week about an organization, a large academic medical center, whose Chief Analytics Officer hired former journalists to do the communication. So you could imagine we have some division of labor so the people who are really good at communicating, like a journalist, could take some of the load off the analysts themselves. Let's face it, it's hard to find people who are both a talented analyst and a talented communicator.

**In the process of creating the Big Data @ Work, was there a Big Data anecdote or story you learned that was the most exciting thing about what Big Data can revolutionize?**

There are a number of cases where Big Data is beginning to transform industries. GE is transforming industrial equipment, Schneider transforming trucking, UPS transforming shipping, Monsanto transforming agriculture. It's a really exciting prospect that we have these companies that are making expensive bets that have potential to dramatically transform how the industry operates and doing good for the world in the process. GE is very focused on how we can lower energy consumption for gas turbines, jet engines, etc. Monsanto is trying to figure out how we're going to feed nine billion people. UPS has already cut their energy uses and carbon creation. They have saved half a billion this year with their Big Data and analytics approaches for driver routing. It's all incredibly exciting. There is a huge amount of potential, but also a lot of work to realize.

**LEARN INSIGHT IN DATA**

<http://www.teamleada.com/>