EXperimental Learning

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

Big Data and Social Analytics certificate course

MODULE 3 UNIT 1
Video 1 Transcript

MIT BDA Module 3 Unit 1 Video 1 Transcript

Speaker key

AS: Arek Stopczynski

HY: Hapyak

AS: Hello again. I hope you're having a blast with the course so far. Right now we are going to talk about data and actually looking at what does it mean and how do you start handling the data, especially this high-resolution behavioral data? Because what is amazing about using personal sensors, they can generate a lot, a lot of data. But it's not only amazing. This is actually a burden that you need to handle.

Using behavioral data that you collect off personal sensors may feel a little bit like trying to drink off a fire hose, especially in the beginning. In the Copenhagen Network Study where we handed out 1,000 phones to our students, we started by doing some calculations and we saw, oh, the phones generate around one megabyte per person per hour and we thought, that's great, that's easy.

00:01:04

And then we realized we need to multiply it by 1,000 people that we have in the study. Then suddenly it was one gigabyte per hour that was hitting our servers, 24 gigabytes per day, which is not crazy but suddenly it was something that we actually had to design for and architecture for: multiple databases, sharding and so on.

So the huge volume of the data also means that it might be easy to get lost and not be able to answer a very, very important question. Are you getting the data that you need? Is the data quality good enough, and think, how do you actually ask what is my target data quality?

Oftentimes people may forget, especially when they just start doing this type of personal data analysis and sensing, to take a step back and ask, would I be happy if I only received 50% of the data? Are certain types of the analyses I'm trying to run more sensitive to the missing data? Will I be doing stuff or will I maybe want to do stuff in the future that will require 90% of data quality? Because every time we need high data quality it is costly.

00:02:21

It is costly to iron out bugs in the sensing software, in the backend, to actually have uptime for the servers, to actually have people who will take care when something goes wrong and to actually have the monitoring system in place. So at any point in time you can check very easily, within few seconds, and understand is my data quality good enough or am I missing something?

When we think about data quality it's very important to ask the question, what signal which we'll use to actually understand what is the data quality that we are seeing? Because we don't want something

that is confounded by the human behavior. So, for example, let's imagine that we are sensing a certain set of behavioral data points and we are doing geofencing so we would only sense on campus. If we are seeing that we are missing 60% of the data, is it because the actual software is failing or the server is dropping the data or is it simply because someone hasn't been really attending the university that much and they are not on campus 60% of the time?

00:03:26

So it's very important when you are building the data pipeline and thinking about the data pipeline, to ask what will be this indicator that we know exactly how much of that we expect. It may be something that is artificially created such as actual ping that the phone will generate every few minutes or it may be something where you know or have a good feeling that it should be extremely stable and independent of the behavior of an individual.

For example what we started doing is when the phones scanned for Bluetooth devices around, we started including the empty scans. So even if the scan wouldn't return any results there would still be a data point that we could see because based on that we can distinguish between missing data and, oh, there was simply no one around. There is complexity in measuring data quality. The failure can happen in multiple places so it's very important to be able to actually measure and answer the question, where did the problem occur? So you are seeing that, oh, you know what signal you are measuring, you know what is your target quality, you are seeing that you are actually missing things.

How do you go about answering, where is it happening? And to help with that it's very important to start verbose. Start with logging absolutely everything and even duplicating the raw data as much as you can. For example, in the Copenhagen Network Studies what we started doing is we were copying every single file that was arriving to our server, immediately before any other type of processing. It was not only a backup but also for us to understand whether we are losing any data in the processing part or whether we are just losing that before, if we are seeing any data loss.

HY: Which of the following statements are true?

a. **Errors in data are best prevented by logging and duplicating raw data as early and as often as possible.**
b. Errors in data are best prevented by logging and duplicating raw data in the final stages of data processing.

AS: Since the process of handling the data is so complex, the data originates on the device, it needs to be stored, it needs to be safely uploaded, then it needs to be unpacked and decrypted and analyzed and populated to the database and so on and so forth. There are multiple places where the process can fail and it's very important to think about this process, that you want to contain the errors. So, in other words, you want to place the data in a safe state as quickly as possible.

00:05:49

For example, thinking about the transmission of the data. So phones they have the data packages, the databases and they are uploading them to the server. What we ended up doing was save the data files as quickly as possible, terminate the transmission and worry about decrypting and unpacking and

populating to the database and analyzing only after that. So the phone hands over the data and it's done. It minimizes the times when the data is in flux and something bad can happen to it.

Something extremely important to remember is you will always lose some data. There is no such thing as perfect dataset. Reasons for that will be multiple. Your software will crash. Your users will do crazy things with the phone that you never thought were possible but, yes, they will actually do them. For example, we started seeing users installing battery-optimizing software on their phones that ended up killing our sensing software after a while and we had to discover it and we had to handle it, basically asking our users not to do that.

00:06:58

You will always lose some data. Just be careful, are you within the limits of what you want to accept and can you actually respond to the problems if there are any? But remember, the rule is don't lose data. If you see that you are losing the data. This is P Zero, this is Priority Zero bug that will, and should, take your entire team off the tasks they are doing right now to go and fix it. Every time you are losing the data you are losing the value. So even though it's bound to happen, it will always happen, minimize that. Optimize for that. Don't lose data.

After you collect the data, spotting the problems is a real detective work and it takes time but it's definitely worth taking this time and we'll talk about this process in more depth in the upcoming video. But it's a never-ending process of actually getting intimate and understanding your data better and better. With the Copenhagen network study we are still spotting new things in the data – problems, errors, particularities of the data – after two years since we started the deployment. It's definitely worth the time to actually get to know that.

And a trick is some problems might be obvious and some problems may not be obvious at all. So obvious problems with the data is, for example, GPS reading showing up at zero, zero, and they are clear that people do not travel that often to this very particular point on Earth. This is just bad readings from the GPS. It's easy to filter them out. However, in certain cases, it's really what the data means that requires a lot of thought.

So, for example, you might be seeing two users that are seeing each other, through the phones and Bluetooth scanning, all the time. The question is, is it really true that those people are spending so much time? Or maybe they are just living in a dorm right next to each other and actually are sensing each other through the walls. And you need to be able to answer those questions and continuously keep asking them, what the data means, is it correct and does it translate to what we human beings actually think about the data?

00:09:20

What would be the ultimate solution to data quality problem from your researcher or deployment manager perspective? This would be probably to make the users themselves care about the data quality. So create applications for them. Make something that reflects their life back at them because what happens then is if those users spot the problems in the data, they will come to you and they will say this is not right. Now, my stats are totally off. Please fix it.

And this is the ultimate goal. It's not easy to achieve but this is what puts you in the position where you have all your users actually caring about the data they are producing.

HY: Which of the following statements are true?

a. If you follow all of the necessary steps you will have a perfect data set.
b. There will always be unforeseen issues that will cause data loss; the best one can do is to minimize loss.

Which of the following statements is correct?

a. By feeding data back to users in an intuitive and interesting way, users can help you point out and solve problems in your data set.
b. When solving problems in your data set it is recommended that you don't involve users in the process, as this would complicate the process.

AS: So if you think about the deployment and if you can spend time or resources to actually develop a solution that reflects data back at users, this is usually a win for everyone involved.