Module 1 - Computational Story Discovery

Hello and welcome back to week one.

In this video I want to talk to you about how data mining techniques can enable something called: computational story discovery, which is one of the more promising application areas of algorithms in journalism.

I'll give you some specific examples of computational story discovery tools and approaches, and I'll talk about the benefits and also the strategic considerations of using some of these tools.

So let's start out with an example to show you how this works. Now I'm not going to show this this video now, but you can see the link is there: doctors.ajc.com, and I would recommend that you check it out.

The story is about how the Atlanta Journal Constitution newspaper found thousands of doctors across the United States who had been disciplined for sexual misconduct in their practice, but who were still seeing patients and who still had their licenses.

Now to get the story reporters scraped more than 100,000 documents online and then used a machine learned to classifier in order to identify a subset of those documents where the doctors that there were most interesting to report on for their story.

OK! So let's look at how they actually got this story. They started because a reporter had noticed a pattern based on his previous reporting. He found that there were about 70 doctors just in the state of Georgia who had been sanctioned for sexual misconduct, but who were still practicing.

And so they knew there was an important story here, but what they didn't know was how widespread this story was. Is it a national story and in order to answer that question they wanted to scale up and increase the scope and the comprehensiveness of their investigation and look at the problem nationally.

So to do this they train a classifier, and how that's done is they took a subsample of documents, they took hundreds of documents and they read through them manually and tag them as either interesting or uninteresting in terms of what they were looking for in their investigation.

Then they trained a classifier based on the interesting and uninteresting documents they had labeled. This ended up being a standard model called logistic regression model, and what they basically did was they trained this model to look for patterns of words associated with the interesting cases and patterns of words associated with the uninteresting cases, and then once the classifier had learned those patterns of words they could put a new document through the classifier and the classifier would label it as interesting or uninteresting for them.

And so how they then use this is they took those hundred thousand documents that they had scraped and they put them through the classifier in order to have the classifier label those documents as interesting or uninteresting.

And of course they were tens of thousands of documents that were labeled as uninteresting, but there were also about 6,000 documents that were labeled as interesting, and of course those were the 6,000 documents that they then assigned to journalists on the team to read through and really dig into and understand the context of those cases.

So, here we see a classifier helped take a 100,000 documents down to 6,000. That's still a lot of documents, but it's within the realm of reason where if you have a team of investigative journalists working over the course of many months, you could get through that many documents, and of course it was important to do additional reporting on those documents in order to finish the investigation. Let's look at another example of computational story discovery. This tool is called "newsworthy" and you can actually play with this and sign up for it online at: newsworthy.fc.

What news really does is it monitors open data sets. Things like crime, unemployment, real estate statistics and then determines if there is some kind of anomaly or outlier or trend in that data. And If it finds something that it thinks might be interesting or newsworthy then it generates a lead like the one that you see in this slide here, where there's a headline, there's some text that kind of explains a little bit what's going on in the data, there's a graph that shows the trend in the data, and then there's some links there to download the chart or actually get to the original data set that's underlying this lead.

Journalists can subscribe to these alerts on their website and receive the alerts whenever something interesting pops up in the data that's being monitored. Of course additional reporting work often still needs to be done before these types of leads are ready to publish, but at the very least a tool like this can help orient journalists attention to what could potentially be an interesting story.

Yet another example is this application of data mining to the task of computational fact spotting or identifying claims in media that are fact checkable and which might be worthwhile to fact check.

So this system is called "claim buster" and it uses a machine learned classifier to rate statements on a scale from zero to one based on how worthy the statement is. So you can see in the example in the slide there are some statements that have been raided by the tool that come from the 2016 3rd presidential debate and you can see the scores color coded in blue on the left hand side of each statement.

Now where this gets interesting is actually using these scores to help inform journalists of where to pay attention or where to look for potentially fact checkable claims, and that's exactly what the Tech & Check Initiative is doing at Duke University.

They use these scores to monitor for fact checkable claims that have been said on CNN every day so every day they use the claim buster algorithm to rate and rank all of the statements made on CNN from the previous day, and then they send a subset of those statements on to different newsrooms in a format

like you're seeing here sort of just a ranked list, so that the journalists in those newsrooms can decide on their own whether or not they actually want to do the fact check. So again this is just about kind of drawing attention to potential leads for fact checkers, potential things for them to work on.

The last example I'll give here today is an example of story discovery in the realm of sort of social media monitoring tools. This tool is called Reuters Tracer. It monitors millions of tweets every day and it clusters them into event clusters and then ranks those event clusters based on the newsworthiness of those events and also ranks them and rates them based on the estimated veracity of the event.

So journalists can use this tool to sort of identify credible breaking news events that may be going on in the world. Now one of the main advantages of this tool is speed, and in the course of Development Reuters has actually done a number of evaluations on the tool, and they found for instance that for the Brussels airport bombing in 2016, Tracer was able to detect that event two minutes before local media and ten minutes before the BBC had reported that event.

And looking across a larger subset of events they found that an 84 percent of cases Reuters would have helped them be faster in sending their own alerts out. So let's step back for a minute and look at some of the benefits of these examples that I've shown you.

So in the AJC example we saw that you know the reporters were able to take what might have ended up being a state or regional investigation and make it national. So computational story discovery can help investigations become more comprehensive, it can increase their scope and help you look at a larger set of data for that investigation.

Computational Story Discovery can also help with monitoring data and documents and media continuously and helping to orient attention to areas of potential interest. So we see this you know in the example of newsworthy for instance, right? You're continuously monitoring, the tools continuously monitoring, these data sets that are published, and it's just sending alerts as needed to orient journalists attention.

This is also the same premise for the Tech & Check Initiative. They're monitoring CNN for potentially check worthy statements and then drawing attention to those statements for fact checkers who might want to then dig into these things. And then finally there are some speed advantages here. Certainly in breaking news scenarios there may be some advantages to having computational techniques which help discover events going on.

So in terms of your own strategic considerations of how to deploy these tools I would just say one thing to be aware of is whether these types of algorithms might be nudging or biasing your coverage in some way. So you know, you want to make sure that you're covering what you're covering because you want to be covering it, not just because some tool alerted you to something going on over here.

The worst case scenario was that would be that you get distracted from covering what you actually want to cover because the computational tool sort of diverted or distracted your attention somewhere else.

Another thing to be wary of is that you want to make sure you don't get overwhelmed from these algorithmic news leads.

It's certainly possible that these techniques can generate many dozens or hundreds of alerts or leads, and you want to make sure that you think about how that is going to integrate into your overall workflow and ensure that you don't become overwhelmed yourself with these leads.

And then finally I just want to underscore this point of, you know these approaches are not about replacing people. You still very much need reporters involved in these workflows in order to do reporting, in order to understand the context of the information that's getting alerted through the system.

So these techniques are useful for kind of directing attention, but we still need trained reporters to dig in and do the reporting on the lead once it is sent to them. So that's what I wanted to cover for today. In the next video I'm going to talk about computational thinking as a way to help you figure out how to better integrate these types of tools and approaches into your workflows. Thanks and I'll see you there.