e-Thiek - HC 7, 06-01-2021, Henry Prakken (English lecture)

Part 1 Introduction AI & law 2

The fundamental question is what extend legal reasoning and legal problem solving can be automated at all? And how useful can that be for judges, citizens, prosecutors etc. There are two main approaches: the **symbolic/cognitive** and the **statistical, data-centric approach**. The symbolic approach tries to let the computer reason and solve problems in the way humans would do and gives knowledge to the computer so the computer can apply its reasoning to that knowledge. The idea arose of expert systems because general systems were not successful. The expert systems have to solve problems on narrowly define domains. The reasoning and problem-solving mechanisms are more fine-tuned on that domain. The computer can explain how it reached solution to a problem. A drawback is the **knowledge accusation bottleneck**, the knowledge needed for solving problems is entered in the computer in a symbolic form, this is very hard to do in general. Because knowledge consists of common-sense knowledge and human experts are often not fully aware of their knowledge the computer cannot understand fully. The statistical data-centric approach does not have the knowledge accusation bottleneck problem because the computer can do **machine learning**. But a disadvantage of this approach is that it is very hard to understand for humans what the computer means. You need both approaches to get the best results.

Prediction is not decision making

There is a more general problem with AI. Even though these outcome prediction algorithms can be useful in several ways in the law, they are not fit for modelling legal decision making. Because judges don't predict but **decide** and the **justify** their decisions on **legal grounds**. Predictive algorithms don't do this. and how can we know that a predicted decision is correct if it cannot be explained on legal grounds? The knowledge-based algorithms are more fit for modelling or supporting legal decision making. We discussed two kinds of knowledge-based systems: **rule-based systems** and **argumentation systems**. Rule based systems are meant to provide a **unique outcome** and the argumentation systems can give **alternative outcomes** in the case. You could say that ruled based systems are more suitable for easy cases and argumentation systems for the hard cases.

Limitations of legal rule-based systems

Rule-based systems have been useful in the area of **public administration** like social security or tax law. It is very useful in a practical sense, but it is hardly used in supporting judges because these systems don't do legal reasoning like the judges would do. Judges have to determine the facts; the systems are not suitable for determining the facts from evidence. They also cannot handle of exceptions and rule conflicts. They cannot do cases of make arguments.

Legal argumentations systems: the KA bottleneck

To overcome the limitation of rule-based systems there is research on developing realistic models of legal reasoning. So, a kind of knowledge-based systems. They tried to implement **realistic models** of legal reasoning like argumentations with precedents, balancing reasons or values etc. but it is still **hard to apply** in practice because of this KA bottleneck. The required knowledge is hard to manually acquire and code in the computer. Then the question arises why can't we combine these two systems? Is **NLP** the solution? We will discuss this in the rest of this lecture.

Contents

The overview of this week:

- Data-centric approaches
  → Extracting information from texts
  →Predicting outcomes of legal cases
  →Big data/ ML for providing information to legal decision makers or investigators
- Making autonomous systems conform to the law

Part 2 text analytics

NLP for extracting factors from case law texts
Can legally relevant factors be **automatically recognized** in legal texts? Many people in the legal world think this is already possible to a large extent, but it is not yet so advanced. We will discuss how to learn the input required by the HYPO and keto (case-based) systems. The designers of the HYPO system tried to automatically learn the factors from the case-law decisions. First of all, the results were still modest in accuracy. The computer could not recognize the legal factors all by itself, it had to be done by supervised learning were the computer learns its patterns form annotated training data. The human had to annotate the training data in terms of which factors are present in a particular case. Moreover, the task to manually annotate is quite laborious.

Much other work on text analytics (often commercial!)
This does not mean there is no practical application of natural language processing technology in the law. There is a growing body of already commercially available applications of what is called text analytics. In a way these applications are less sophisticated than legal knowledge-based systems because they do not try to solve legal problems. It is more a matter of information **retrieval** or **summary**, which is an easy task for the computer. Case retrieval systems have been around for decades. The quality is better nowadays. Researches started years ago with looking at problems at **extracting** more specific **information**, like who are the persons involved and what was the outcome etc. Automated summarization has also been commercially doable for at least 10 or 15 years now. **E-discovery** is an interesting application where humans have been completely automated. For example, discovering possibly relevant documents in a legal case. More recently there have been successful commercial applications of **contract review**. Lawyers have to check large contacts for their clients, they have to identify clauses. Nowadays there is good AI system that can identify potentially problematic contract clause. Something similar is a recent research project where the algorithm has the task to identify potentially unfair clauses in general terms. This algorithm has an accuracy of 80%. Another useful application is **network analysis**. For instance, you could analyze a body of **cases** to see which cases are cited by other cases, to discover the leading. You can also apply network analysis to regulations for instance to discover the **regulations** that are relevant to the problem of drones. There has even been research solving the knowledge accusation problem for rule-based systems were people have tried to apply natural language processing to automatically extract. This is not possible fully automatically, but you can let the computer do some preprocessing work. So, there is already commercially available AI & law software around.

IBM's Watson and follow-ups
In the remaining slides we will discuss some developments that according to some that it may be possible that the computer fully automatically generates interesting legal arguments from natural language sources without the requirement that first explicit symbolic knowledge is entered in the computer. This is the so-called **Watson project of IBM** (the big IT company). Watson was playing an American tv quiz with questions about politics, sports, geography etc. IBM Watson system had access to huge amounts of unstructured information like Wikipedia. It turned out to beat the best human players in this quiz. This was a big breakthrough in AI, because it looks like a solution to the KA bottleneck. This system could find its knowledge without explicit symbolic knowledge. There is also a legal spin off called ROSS, that was funded by IBM.

ROSS
The ROSS system is used by quite some law firms around the world. It is not easy to find information about a system, its design is not fully public. So, researches cannot validate the quality of these commercial systems. Anyway, one opponent of ROSS is the **AI legal search engine**. This tool accepts questions in plain English and returns answers based on legislation case law and other sources. The system will give references and probability of correctness. A second main component is the **Brief Analyzer (EVA).** This tool is an integration of the text analytics tools that we talked about. It automatically generates briefs and creates hyperlinks to every case cited in a brief, checks the subsequent history of cited cases, and finds cases having similar language.

Debater

The final development is IBM's next challenge '**the debater'**. The challenge was to have a computer program to have a debate with a human. It has access to all kind of natural language sources, so it automatically mines these **arguments** and then tries to re-use these arguments with humans. In June 2018 IBM demonstrated that a system debated with a human. The debate was about whether government support for space travel is desirable and the second whether tele medicine is desirable. System debated with a human (Israeli debate champion). The public found the arguments of the human better but not much better. The arguments that were generated by the system were quite realistic. IBM (commercial) fully controlled the experiment, so we don't really know how good it is. But the question for us is could maybe this kind of technology automatically generate interesting legal arguments without the need to represent legal knowledge in the computer? We will see maybe in 5, 10 or 15 years we will see this kind of system in our law firm.

Part 3 outcome prediction

Supervised machine learning

We return to outcome predicting algorithm that predict outcomes of legal cases. The algorithm is trained on a collection of **training data** which is labelled, the labels are the **outcomes** of the case. Then, after the algorithms has learned **statistical** connections between input and output it is tested on **test data.** The outcome is hidden for the model, so we can see how well the algorithm **predicts** the outcome. The most often report is the **accuracy:** percentage of test data for which the outcome is correctly predicted. Then, you hope to apply the model in practice.

Three kinds of case outcome prediction algorithms

There are three kinds of case outcome prediction algorithms. The first was applied to the **raw text** of a case, the second was applied to **external data** and the third type was applied to **legally relevant** features.

Approaches to predicting outcomes of cases

To refresh our memory of the two kinds of systems we saw in week 6, first of all we saw that outcome predictors based on the **unstructured text** of past decisions have a number of problems. First, they cannot explain the reasons for their decisions, they don't understand it, they just find frequencies of word combinations. Second, it is not really prediction because large parts of the text of the new decision already has to be available. In short, it's hard to imagine legally useful applications of this kind of technology. Another type of an outcome predictor is the machine learning with data partly **unrelated to the merits of the case** (court, judge, jurisdiction, parties, attorneys). This is also been applied commercially in practice to decisions of other courts. All these factors are unrelated to the merits of the case, but you can still predict in new cases. As long as they predict over 50% it can be useful for example for attorneys whether they have to settle or sue. It cannot give legal reasons for a decision, but it can be useful.

Predicting misuse of trade secrets on the basis of legal factors

We saw that the approaches to case outcome prediction on the basis of natural language text of a case has a fundamental problem, namely it's not able to explain the outcome in legal terms. We now discuss some attempts to tackle that problem. Namely, to let the machine work with as input legally relevant information. For instance, the legal factors of the HYPO and KETO systems. This has been done by several decades. Legal researches would list the relevant factors in a problem domain and then they would apply convential/statistical techniques to make outcome predictions. At least you can explain the outcomes in terms of legally relevant factors. You can go even a step further, that you don't do statistical analyses, but you use a knowledge-based system to generate a solution to a legal problem and to regard that solution as a prediction. This has been done by Ashley and Bruninghaus. They did it with human and coded precedents in terms of pro and con. The HYPO and KETO systems have a notion of **relevance** of a precedent to the new case. For example, how many factors are shared between the case and the precedent. Then, they applied several voting criteria to make an outcome **prediction**. Outcomes can be explained in terms of **precedents**. The accuracy of this system was **80%** but the relevant features must be known in advance and coding is **laborious** and can be **subjective**.

Can algorithmic outcome predictors suggest decisions?
To what extend are these systems useful for judges? I think it's not such a good idea to use these predictors for decisions. There are several reasons, first of all we have seen that the performance of these systems is still modest, they often **misinterpret**, and they **cannot explain** the predictions which is needed to justify legal decisions. The fundamental problem is that machine learning based outcome predictors are completely **based on the past**, but the future can change. It can be that opinions change and if you rely on machine learning that is based on the past that can become a problem. Another problem is that the **accuracy** figures cannot be regarded as the probability that the judge will take the predicted decision. Accuracy is not how often the judge follows the algorithm but how often the algorithm follows the judge. So, the idea that an algorithmic prediction is the 'normal' outcome, which can only be ignored if there are special circumstances, is wrong. The bottom-line is that predicting outcomes of decisions is not making decisions, these are two fundamental different things.

Medical analogy
Outcome predicters don't have to be perfect as soon as they perform better than humans than you should use them. In medical domain, human **doctors** should use **diagnostic algorithms** if these perform better than human doctors. So human **judges** must use **algorithmic outcome predictors** if these perform better than human judges.

Medical analogy breaks down
Why does this medical analogy break down? The main reason is, that unlike in the medical case were the doctor and the diagnostic algorithm perform **the same task**, the human judge and the outcome predictor perform **different tasks**. The human judge is not trying to predict his decision, but the algorithm is. If the algorithm correctly predicts legally incorrect decisions, then it counts as a success for the outcome predictor. It's irrelevant that the outcome may be legally incorrect. Outcome predictors finds **statistical** correlations and the judge must find **legal** reasons.

Part 4 algorithmic experts

Examples of algorithmic experts
We have been quite skeptical about the use of outcome predictors for judges. Only knowledge-based outcome predictors have a place in court, if they are used the judge should only look at the legal explanation. This doesn't mean that outcome predictors have no place at all in the law, we already mentioned they could be useful for solicitors or citizen who want to have an indication for a chance of success in court. Outcome predictors based on extraneous factors could be useful for legal researches who want to know to what extend judges are influenced by non-legal factors, for example bias. For now, we discuss a different kind of data centric machine learning based algorithm that can be of use for judges. This is what I call algorithmic experts. Quite often judges have to make factual estimates, then it may be that a data centric algorithmic can give information about a specific issue. A first example is **predictive policing**, where the police wants to decide where they have to put their policeman on the street. Algorithms can predict where the most crimes are about to happen. Another example **is checking for social security fraud**. Algorithms can also be applied to **predicting the probability** of recidivism, dropping out from school or having financial debts. The final example is when farmers have to ask for environmental permit for activities that involve nitrogen emission. A computer program can predict the **environmental impact** of specific farming activities.

Recidivism prediction
One quality issue is that there is a danger of bias in the data. One example is a system in the US called COMPAS which predicts recidivism of criminals. This system seems to be biased against black people. But this claim is not justified, in the next slide I will explain that. The main message is that these accusations of bias are **not always correct**. Researchers discovered that black people had a higher **false positive** rate, so higher probability that the system would falsely predict that they would commit recidivism. The opposite was also true, they also had a higher **true negative** rate, so higher rate of

correct predictions. So, the **accuracy** was the same for all groups. It can be mathematically shown that both having the same false positive rate and the same accuracy is impossible if the 'base rates' differ.

Fairness criteria
If the base rate differs for the groups, you cannot have both same false-positive rate and the same overall accuracy for all groups. So, what do you want, the same false-positive rate or the same overall accuracy for all groups? Or do you maybe want to optimize on other measure, in the literature several fairness criteria have been proposed in the algorithm fairness. You have to remember that the choice between which fairness criteria you want is not technical but **legal**!

Deleting protected attributes does not promote fairness
There is some good news, there is a new research area on fairness in algorithmic decision making. Researches now more about fairness and how to design systems, how to collect data so that **bias discrimination** is reduced. A concept that is often used is a protected attribute, so some property of a person for instance like gender or ethnicity. There are some techniques to discover possible bias and to prevent that bias. However, there is also bad news because the GDPR (AVG) generally **forbids** processing **sensitive** data. This is a problem for discovering bias because its well-known that simply omitting the protected attribute from your data is not a good method to reduce/avoid bias. So simply not record a person's ethnicity is not a good way to prevent discriminating on the basis of ethnicity. Because this so-called protected attribute like ethnicity can statistically correlate with other factors that are in the data like average income, your postal code area, level of education etc. if nothing else is done, could still be discriminate, but there would be no way to discover it because the ethnicity is not in the data. This problem is created by a legal solution that was meant to prevent discrimination.

Quality aspects of machine learning
We are a bit skeptical about the use of outcome predictors, but we are less skeptical about algorithmic experts. But it's not an easy task that they are of sufficient quality. These kinds of applications are performing the same task as humans, so it makes sense to compare them with humans. This is a list of quality aspects of machine learning:

- Correlation → statistical correlation is not the same as causation. This is a genuine problem in practice.
- Data can be **incorrect** → it often happens that the data that the government collects is simply incorrect.
- Data can be **sparse**
- Data selection can be **biased** → many face recognitions perform better on white people because the training data is based on white people, so this is also discrimination. Employee hiring: if in the past more men were hired, then male characteristics may be overvalued.
- Learned model is **overfitted** on the data
- Relevant **knowledge** can be ignored → the future is not the same as the past, the types of cases that arise differ. Outcome predictors are often based on the past that's a problem. Cathy O'Neill: it is hard to argue against a probabilistic/statistical prediction on concrete grounds. You are suspected not because of what you have done, but because of general characteristics.
- Acting on predictions can change the data → predictive policing, once the police starts surveilling more in a particular area the police will always find something and that will be added to the data, and next time the algorithm will predict and even higher crime rate.

So, there a quite some quality issues with these systems. But a system does not have to be perfect, as long as it performs better than humans. (Holds both for judges/decision makers and e.g. crime investigators).

Examples of strange correlations
Correlation does not imply causation. It also illustrates a possible use of algorithm case outcome predictors for legal researches who want to know whether judges are influenced by non-legal factors by their decisions, maybe even biased. Some examples of strange correlations that was found in research:

- Judges are more severe on bail decisions just before lunch → maybe because they were tired, or their blood sugar decreased. But you don't want this of course.
- Judges punish more severely after their favorite American Football team lost, in particular stricter against young black Americans.
- Judicial dissent increases just before US presidential elections
- The chance of a negative asylum decision increased after a sequence of positive decisions
- Prison sentences are lower on defendant's birthdays

Conclusion
Algorithmic **outcome predictors** only have a place in court if they are **knowledge-based** because then they can explain their predictions in **legal** terms. And if this kind of system is used by judges than the judge should not look at this numerical quality measures like accuracy but only at the legal quality of the explanation that is given by the system. About the use of algorithmic **experts and investigators** I am more optimistic. They can be useful in court and law enforcement, but they have some issues with **quality** assurance and **explain ability**. There is a growing subfield in AI that studies how algorithmic decision-making application can be explained. The best solution would be to work together (data and AI scientists, lawyers, society) in making this work where it can be beneficial.

State of the art in AI support for judiciary
I've said that these simple **rule-based** systems can at best model routine decisions but only if there are no problems with data verification like government data bases. **Argumentation** systems which are more realistic in terms of legal argumentation, a lot of research as so-called proof of concept research has so far been turned out to be very out to skill them up to practical use because of the KA bottleneck. Attempts to apply machine learning of the natural language to automatically extract the legal knowledge like factors and values from the natural language sources had not have much success yet. Although there are big developments like IBM debater. More modest applications so-called **text analytics** are already possible and commercially available. Finally, **outcome predictors** can be useful for lawyers and academics, but not for judges. Judges can at best benefit firm **algorithmic experts**.

Do we even want to fully automate judges? No (unrealistic): we want to support them, so that the **combination** of human judge and computer performs better than either human or computer alone.

Oefenvragen
1. Vraag 2 van week 6 was: Welke soorten kennis zou een AI-systeem nodig hebben om in 'hard cases' te beslissen? Bespreek van elke soort kennis die je noemde in welke mate een algoritme die kennis nu automatisch kan leren uit data.
2. Vind je de website jurisays.com nuttig voor juristen?
3. Waarom is de medische analogie wel toepasbaar op algoritmische deskundigen?
4. Noem een voordeel en een nadeel van een kennisgebaseerde aanpak bij het zodanig ontwerpen van autonome systemen dat ze het recht respecteren.