School of DA and AI
HSE University

NoML seminar
13 December 2023

Human Knowledge Models

or When the simplest rules work on par with gradient boostings

Egor Dudyrev
eo.dudyrev@hse.ru
http://egordudyrev.github.io/

About the speaker

- Passed 2 jobs interviews at Glowbyte
- 0.5 years: Intern in Sber Treasury department
- 2.5 years: Data Scientist in DWH department of Sportmaster
- 1.5 years: Junior Research Fellow at ISSA lab, HSE Uni.
- 1 year: Doctorant at LORIA laboratory

Outline

1. **Human Knowledge Models**
   Based on paper: *Dudyrev, E., Semenkov, I., Kuznetsov, S. O., Gusev, G., Sharp, A., and Pianykh, O. S. (2022).* Human knowledge models: Learning applied knowledge from the data. Plos one, 17(10), e0275814.
2. **Fasten up the search of HKM**
   Based on paper: *Dudyrev, E., and Kuznetsov, S. O. (2022). Towards Fast Finding Optimal Short Classifiers. FCA4AI 2022, 23.*
3. **Practical example**
   (The new notebook that even my supervisor has not seen yet)
4. **Conclusions**

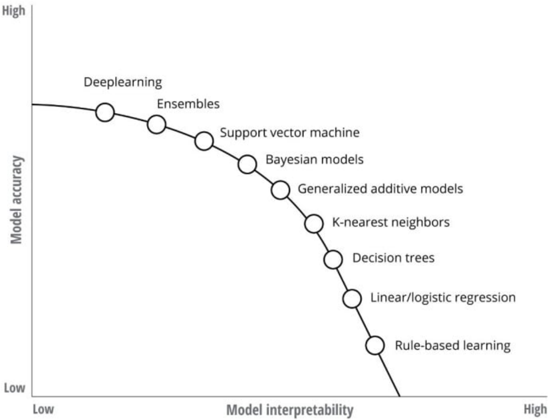
Human Knowledge Models (HKM)

Accuracy - Interpretability Tradeoff

It is assumed that ML model can be either understandable or accurate. And one should always choose how much accuracy one can sacrifice to improve the interpretability.

FIGURE 2

There are inherent trade-offs between model accuracy and model interpretability



Source: DPhi, "Importance of Human Interpretable models & Explainable AI," video featuring Dipanjan (DJ) Sarkar, 29:02, February 13, 2021.

Deloitte Insights | deloitte.com/insights

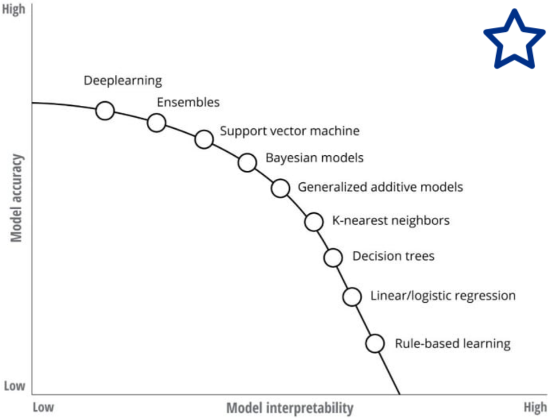
Accuracy - Interpretability Tradeoff

It is assumed that ML model can be either understandable or accurate. And one should always choose how much accuracy one can sacrifice to improve the interpretability.

Though it would nice to have models that are both accurate and understandle

FIGURE 2

There are inherent trade-offs between model accuracy and model interpretability



Source: DPhi, "Importance of Human Interpretable models & Explainable AI," video featuring Dipanjan (DJ) Sarkar, 29:02, February 13, 2021.

Deloitte Insights | deloitte.com/insights

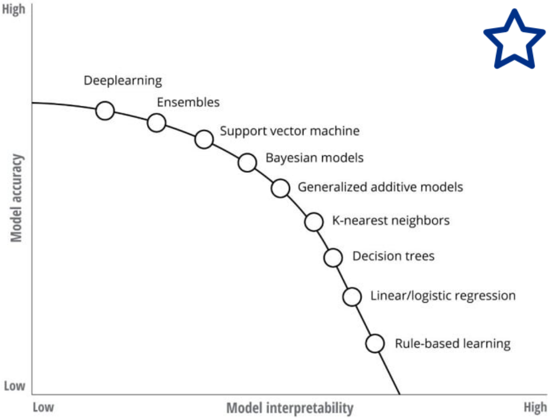
Accuracy - Interpretability Tradeoff

It is assumed that ML model can be either understandable or accurate. And one should always choose how much accuracy one can sacrifice to improve the interpretability.

Though it would nice to have models that are **both accurate and understandle and fast**

FIGURE 2

There are inherent trade-offs between model accuracy and model interpretability



Source: DPhi, "Importance of Human Interpretable models & Explainable AI," video featuring Dipanjan (DJ) Sarkar, 29:02, February 13, 2021.

Deloitte Insights | deloitte.com/insights

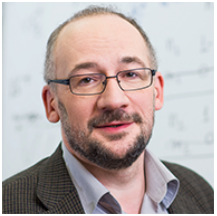
The case of COVID

Oleg Pianykh

Investigator
Radiology, Mass General Research Institute

Assistant Professor of Radiology
Harvard Medical School

He was looking for simple rules that could describe the COVID mortality reasons.



Sergei O. Kuznetsov

School Head
School of Data Analysis and Artificial Intelligence, HSE

Laboratory Head
International Laboratory for Intelligent Systems and Structural Analysis

He was managing the simple rules verification from the data science perspective

Example of a HKM

A patient has a high risk of complications IF:

- "Low blood oxygen" AND
- ("Age > 79" OR "Fluoroscopy score < 2")

A model should have **no more than 4 binary conditions**, each found only once in the rule.

The reasons for the number 4 are provided in the paper:

Cowan N. The magical number 4 in short-term memory: A reconsideration of mental storage capacity. Behavioral and Brain Sciences. 2001;24(1):87-114.

Experiment design

4 datasets:

- H COVID-19 diagnosis (shape 1925x229)
- E Water potability (shape 3276x9)
- F Company bankruptcy (shape 5910x64)
- S NIM game (shape 2000x10)

12 model architectures, divided by classes:

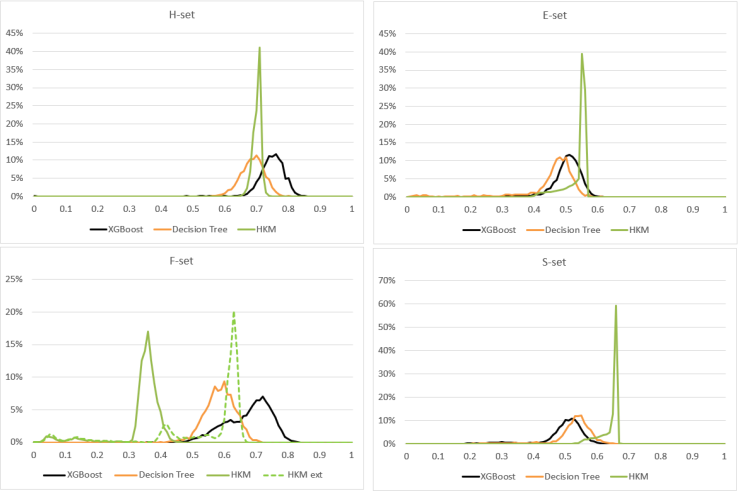
- Simulatable AI (e.g. small decision trees)
- Interpretable AI (e.g. big trees, small ensembles)
- Black Box (big ensembles of decision trees)

For every model architecture, we have built from 1000 to 5000 models with cross-validation, hyperparameter optimisation, etc.

Prediction quality

F1 score distribution on test data depending on the model.

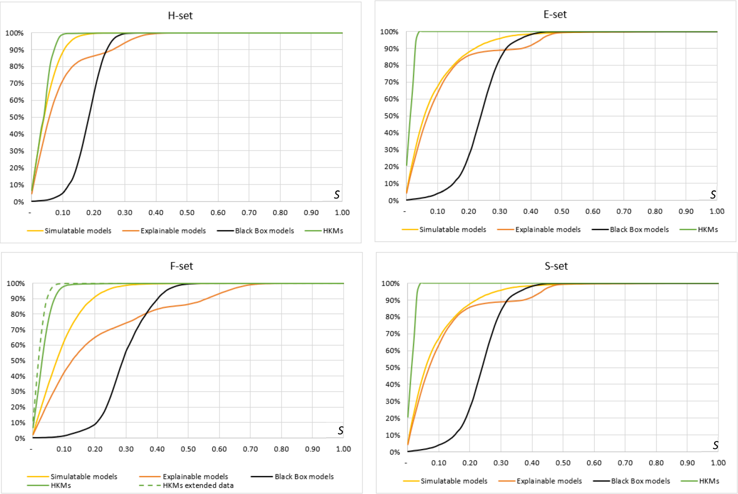
For the H-set, HKMs work almost on par with XGBoost, and for the E-set and S-set they even work better than the boostings.
For the F-set, HKMs work much worse than XGBoost. However, feature engineering helps improve their predictions.



Model overfitting

Reading the plot: "Overfitting of y per cents of models is not higher than x". Overfitting of a model = $\frac{|\text{train}-\text{test}|}{\text{train}}$.
I.e. the more the line is on the left, the smaller is the overfitting.

For all the datasets, HKMs almost never overfit.



Fasten up the search of HKMs

Problems with running the models from the paper

We found that Human Knowledge Models work (sometimes). However, the original code for the models is written with Matlab and works for 2 days on a dedicated server. This is OK for the medecin (acc. to O. Pianykh). But it is very weird for the ML world.

So, we should find out how to fasten up the process of finding efficient Human Knowledge Models.

Problem formulation

$$\text{Best rule} = \arg\max_{\text{short rules}} \text{F1}(\text{rule})$$

The rising questions:

1. What is a “rule”?
2. What are the “short rules”?
3. How to optimise the F1 score?

Formal Context

We use the notation of Formal Concept Analysis (FCA).

	needs water to live	lives in water	lives on land	needs chlorophyll	di-coty-ledon	mono-coty-ledon	can move	has limbs	breast feeds
fish leech	X	X					X		
bream	X	X					X	X	
frog	X	X	X				X	X	
dog	X		X				X	X	X
water weeds	X	X		X		X			
reed	X	X	X	X		X			
bean	X		X	X	X				
corn	X		X	X		X			

Objects $G = \{\text{fish leech}, \text{bream}, \text{frog}, \dots\}$,
Attributes $M = \{\text{needs water to live}, \text{lives in water}, \dots\}$.

// G and M come from German: object = der Gegenstand, attribute = das Merkmal.
Also say that G_+ are the "positive" objects, and $G_- = G \setminus G_+$ are the "negative" objects.

Set of premises

Let us setup the set of premises $\mathbb{P}$:

$$M \subseteq \mathbb{P}, \quad \forall p, q \in \mathbb{P},\ p \wedge q, p \vee q \in \mathbb{P}$$

Or, depending on the premise size:

$$\begin{aligned}
\mathbb{P}_1 &= M \\
\mathbb{P}_2 &= \{p \wedge q, p \vee q \mid p, q \in \mathbb{P}_1\} \\
\mathbb{P}_3 &= \{p \wedge q, p \vee q \mid p \in \mathbb{P}_1, q \in \mathbb{P}_2\}
\end{aligned}$$

We will search the optimal rules in the space $\mathbb{P}_1 \cup \mathbb{P}_2 \cup \mathbb{P}_3$.

For every premise $p \in \mathbb{P}$ we say that $p' \subseteq G$ are the objects, described by p.

Problem formulation (v2)

$$p_{\text{best}} = \arg \max_{p \in \mathbb{P}_1 \cup \mathbb{P}_2 \cup \mathbb{P}_3} \text{F1}(p)$$

The rising questions:

1. ~~What is a "rule"~~?
2. ~~What are the "short rules"~~?
3. How to optimise the F1 score?

F1 score

F1 score is the harmonic mean of precision and recall:

$$\text{F1} = 2\frac{\text{precision} * \text{recall}}{\text{precision} + \text{recall}},$$

$$\text{precision} = \frac{\text{tp}}{\text{predicted\_positive}}, \qquad \text{recall} = \frac{\text{tp}}{\text{real\_positive}}.$$

So, how to optimise it?

F1 score can be express in terms of basic measures tp, fp, fn:

$$\text{F1} = \frac{2\text{tp}}{2\text{tp} + \text{fp} + \text{fn}}$$

And still, how to optimise it?

Jaccard Score

It turned out that F1 score is comonotonic to Jaccard score:

$$\mathrm{F1}(p) \sim J(p)$$

That is, for every $p, q \in \mathbb{P}$, $\mathrm{F1}(p) \leq \mathrm{F1}(q) \iff J(p) \leq J(q)$.

Where Jaccard score is the Jaccard coefficient between the positive objects G_+ and the objects that are predicted positive p':

$$J(p) = \frac{|G_+ \cap p'|}{|G_+ \cup p'|}$$

$$J(p) = \frac{\mathrm{tp}}{|G| - \mathrm{tn}} = \frac{\mathrm{tp}}{|G_+| + \mathrm{fp}}$$

Problem formulation (v3)

$$p_{\text{best}} = \arg\max_{p \in \mathbb{P}_1 \cup \mathbb{P}_2 \cup \mathbb{P}_3} J(p)$$

The rising questions:

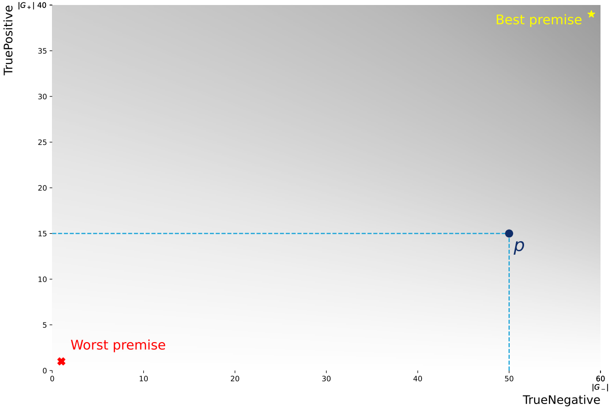
1. ~~What is a "rule"~~?
2. ~~What are the "short rules"~~?
3. How to optimise the ~~F1~~ Jaccard score?

TP TN space

Every premise p (so as any binary classifier) can be placed on TP, TN plane.

The axes:

- x: amount of true-negative predictions $tn_p = 0, 1, \ldots, |G_-|$,
- y: amount of true-positive predictions $tp_p = 0, 1, \ldots, |G_+|$.



Conjunction and disjunction

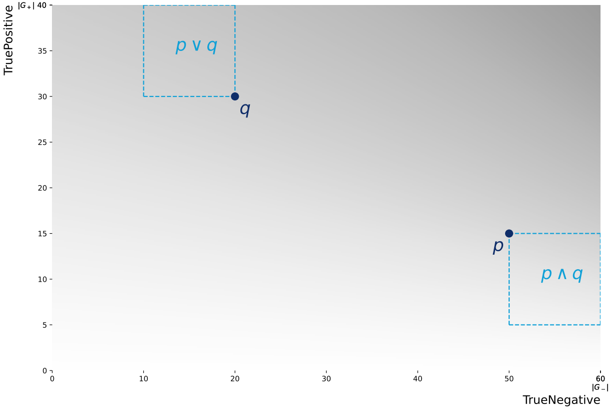
Given the coordinates of premises p, q, one can estimate the coordinates of their conjunction and disjunction:

Conjunction:

$$tp_{p \wedge q} = |TP_{p \wedge q}| = |TP_p \cap TP_q|$$
$$tn_{p \wedge q} = |TN_{p \wedge q}| = |TN_p \cup TN_q|$$

Disjunction:

$$tp_{p \vee q} = |TP_{p \vee q}| = |TP_p \cup TP_q|$$
$$tn_{p \vee q} = |TN_{p \vee q}| = |TN_p \cap TN_q|$$



Comparing with a threshold

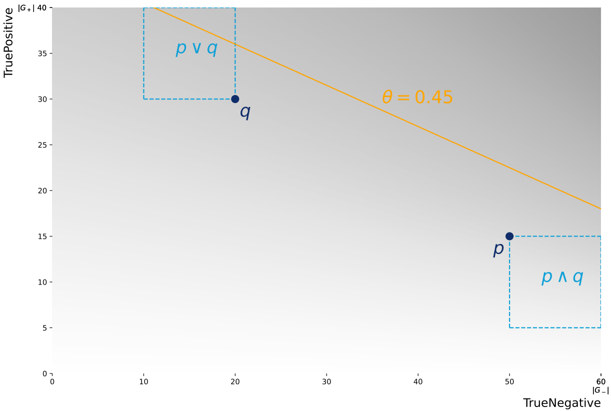
Given two premises p, q and a prediction quality threshold $\theta \in [0, 1]$, one can compare the quality of $p \wedge q, p \vee q$ with the threshold θ:

Conjunction:

$$tp_p \leq |G_+|\theta \implies J(p \wedge q) \leq \theta$$

Disjunction:

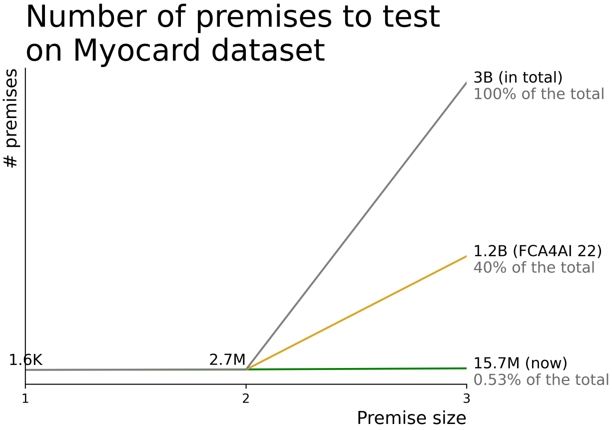
$$tn_p \leq |G| - \frac{|G_+|}{\theta} \implies J(p \vee q) \leq \theta$$



Conclusions regarding the fast search

The paper from FCA4AI'22 describes the algorithm that passes through every premise of 1 and 2 attributes, and through the filtered set of premises of 3 attributes. The filtering of premises is based on comparing the conjunction and disjunction borders to the threshold.

A new (unpublished) algorithm uses the idea of logically equivalent premises and makes the search space even smaller.



Practical example

Google Colab with the example

The link: https://github.com/EgorDudyrev/PeaViner/blob/main/Myocard%20case%20with%20PeaViner.ipynb

The Python package: https://github.com/EgorDudyrev/PeaViner

Conclusions

Conclusion

During the talk we:

- described the Human Knowledge Models (HKM);
- showed that sometimes even the simplies HKMs work with SOTA accuracy;
- and given all this, HKMs do not overfit;
- discussed how to fasten up the search for the short rules;
- found out that F1 score is almost the same thing as Jaccard score; and
- built HKM models on Myocard data.

Contacts

Thank you for your attention!

Contacts:

- eo.dudyrev@hse.ru
- https://egordudyrev.github.io (There are links for personal , ,)
- https://github.com/EgorDudyrev/PeaViner
- https://github.com/EgorDudyrev/PeaViner/blob/main/Myocard%20case%20with%20PeaViner.ipynb

Appendix

The relation between F1 and Jaccard scores

$$F1(p) = \frac{2}{1 + \frac{1}{J(p)}}$$