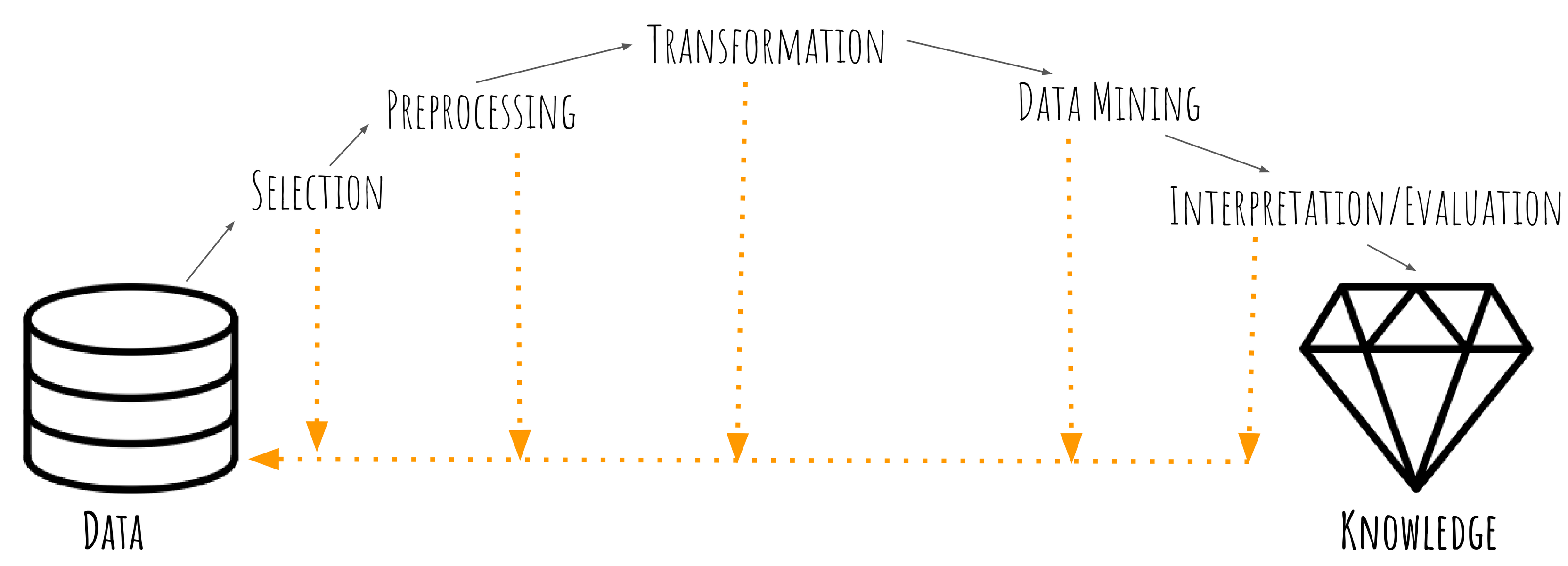
Incorporating Domain Knowledge in Data Science Processes and Tools

Abstract Using Data Mining tools data scientists can discover and extract valuable knowledge from data. Whether this knowledge is valuable in terms of quality, novelty and usefulness can only be judged using appropriate domain knowledge. In a news story on a database of german doctors and their affiliations to to pharmaceutical companies the value of clustering them is determined by the journalist who aims to write up the story about them. If in another project the power consumption of a machine is reduced by 0.01% by applying a data-based prediction model, the relevance of the result can only be determined by experts on the matter. Overall the data science process in its explorative and creative form always requires two skills: Expertise in the data science toolbox and expertise in the domain. The first skill is needed to map out possible paths of exploration and execute them and the second to evaluate paths taken and determine which expected results from new paths could be useful. Given this scenario different tools and architectures for data processing are appropriate.

1 Data Engineering Process

Usually data science projects are set up to analyse data from domains the data scientist has little knowledge about. The data scientist is native to the methods of analysis, but in order to develop new hypotheses the exploratory processes requires intermediate results to be interpreted to estimate how satisfying the answer is and to generate new questions. Existing processes for data science, e.g. KDD (Knowledge Discovery in Databases)[1] or CRISP-DM [2], affirm the importance of domain specific knowledge by proposing "Understanding the domain" as the first step, sometimes combined with an understanding of the business and data itself. On this foundation other steps like preprocessing, model selection and pattern discovery are executed. In the end, the results are visualized to communicate the gained knowledge effectively. According to the exploratory nature, these steps are retaken iteratively. Some exemplary questions are suggested in Figure 0.



How much noise is in the data, is it expected?	Which problem should be pursued first?	Should an apparent data bias be carried over to a prediction?	How can the result be explained?
What data is available at which times?	What structure is inherent to the data?	Model introspection: Are the intermediate results expected in the context?	Is the knowledge found of value, interest or novel?

Figure 0 The KDD Process and important domain-centered questions during different steps

Depending on how the domain knowledge is distributed among persons, different tools are appropriate to collaborate

Applications specifically developed for experts, supplying a range of highly task-dependent analytical tools (Fig. 1.3)	Frameworks for general analytics building on preprocessed data or plugins from data science experts (e.g. Tableau)	General-purpose frameworks for advanced analytics used by experts but with integrated communication to domain experts (Fig. 1.2)	General-purpose frameworks for data science giving the possibility for customisation and interaction with algorithms (Fig. 1.1)

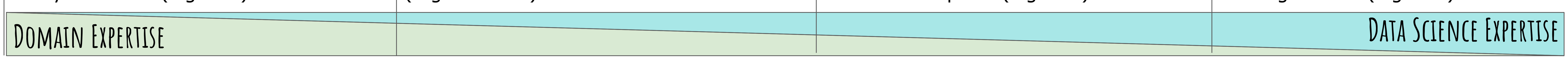


Figure 1.1 R used to analyse data to produce a report with figures

Figure 1.2 An iPython notebook, a question by a journalist, answered in code below by the data scientist [4]

Figure 1.3 An exploratory visualisation, a product for measuring heat in housing by enersis [5]

2 Collaboration Tools

Questions and hypotheses are drivers of exploratory processes. In traditional software development a common language for communication about the domain is sufficient. Exploratory data science projects require not only communication but even more collaboration. The data scientist can only interpret a result correctly within the domain and the domain expert can only interpret it with an intuition of the model. Only where their skills overlap they can effectively navigate the exploratory analysis. The load of communication required to teach each other about their respective field depends on the distribution of skills between persons and locations. A developer of the XING job recommender system knows what data is relevant to the system and has a grasp of the job search because he is in the same situation. He has data science and domain skills. In contrast, a data scientist exploring the Panama Papers has to collaborate very closely with a journalist to identify promising stories. Both scenarios require different amounts of collaboration and tools. Exemplary solutions are shown in Figure 1, ranging from tools mainly used by data scientists to expert tools created specifically for one use case by an domain expert. A clear communication of questions and results is especially explicit in the ipython notebook in Figure 1.2.

The goal of a data science project can usually be in one of two groups: Those aimed at developing a human understanding, e.g. supporting human decision making on machine performance and those aiming at 'machine understanding', e.g. understanding topics of movies. Collaboration with experts comes more naturally to the first type. In this case usually models with low capacity but understandable mechanisms are used, e.g. simple regression models , decision trees and intuitive engineered features. The solutions presented in Figure 1 were of this kind. When the results are interpretable along the way feedback from domain experts is easily incorporated. Projects aiming for machine understanding also profit from human understanding. Most deep learning models are more effective when known properties of the data, e.g. 2d structure of pixels are incorporated via convolutional neural nets. The quality of these intermediate results can only be estimated when the scientists translate them back, e.g. by showing exemplary words for topics of a hidden representation.

During time requirements for tools by people and different projects change, so a flexible architecture is helpful

3 Architectural Considerations

During time an exploratory process is usually narrowed down to a specific discovery or problem that can be solved by the data. This means different amounts of collaboration are in place during time. Consequently different tools and different amounts of data are used during time. The flexibility needed for these different stages of maturity is mirrored in most software stacks we have seen in the lecture. Figure 2 depicts an abstract view. Different data streams are piped into a common data store or kept there static. The store is maintained as clean and consistent as possible. Usually all data that can possibly be useful is stored, regardless of whether it is in use right know to give rise to future applications and exploration, since storage is cheap. On top of that many different technologies are applied for different purposes, from initial exploration of sampled data to a launch of a tool with analytical views fed with real-time measurements.

The tools for collaboration with domain experts are integrated or on top of the analytical tools. Their flexibility depends on how much they abstract from arbitrary manipulation of the data to restricted understandable use.

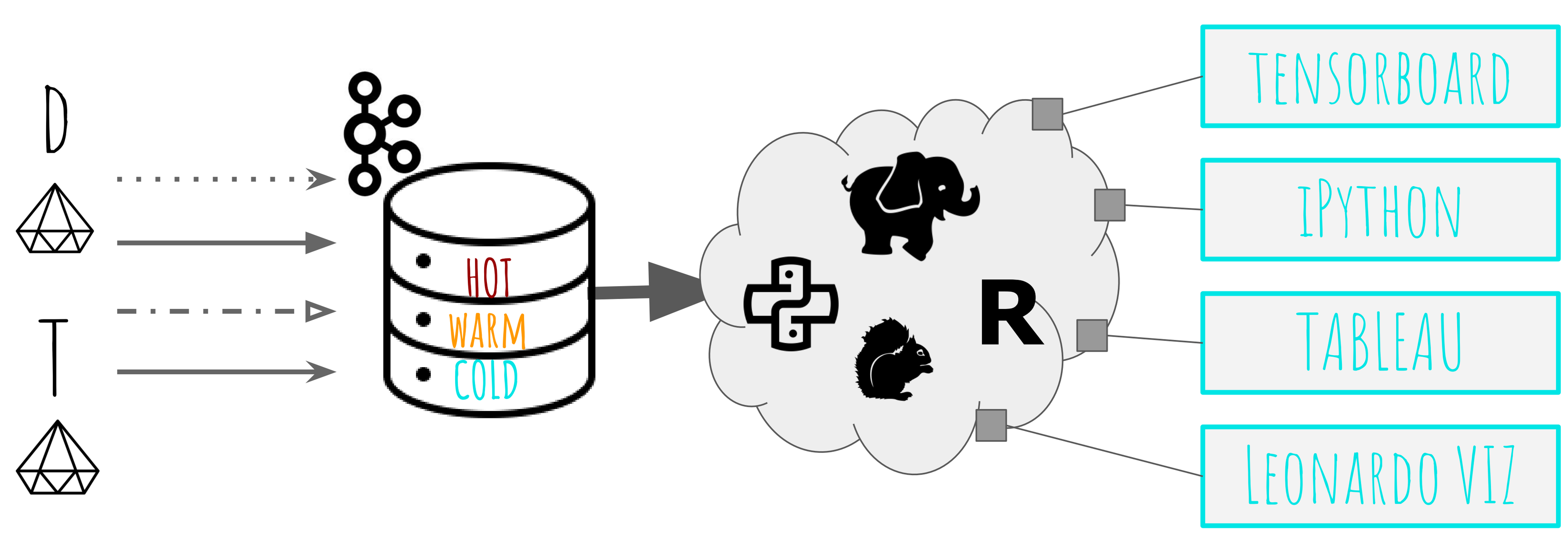


Figure 2 Exemplary data architecture with a central data store

Freya Behrens - freya.behrens@student.hpi.de
Bachelor

Ringvorlesung 'Data Engineering' WS 2017/18
Data Engineering/ Data Science Prozesse

References

[1] Usama Fayyad, Gregory Piatetsky-Shapiro, and Padhraic Smyth. 1996. The KDD process for extracting useful knowledge from volumes of data. *Commun. ACM* 39, 11 (November 1996), 27-34. DOI: http://dx.doi.org/10.1145/240455.240464
[2] Wirth, R. and Hipp, J., 2000, April. CRISP-DM: Towards a standard process model for data mining. In *Proceedings of the 4th international conference on the practical applications of knowledge discovery and data mining* (pp. 29-39)
[3] https://www.r-project.org
[4] https://github.com/correctiv/awb-notebook/blob/master/awb_meldungen.ipynb
[5] http://www.enersis.ch/portfolio/e-on-subsidiary-ekn-and-innogy-subsidiary-digikoo-rely-on-enersis/?lang=en
(Links last accessed 23.01.2017)

Hasso Plattner Institut