Mixed-Initiative Feature Engineering Using Knowledge Graphs

Martin Atzmueller
Tilburg University (CSAI)
Tilburg, Netherlands
m.atzmuller@uvt.nl

Eric Sternberg
University of Kassel (ITeG)
Kassel, Germany
est@cs.uni-kassel.de

ABSTRACT

This paper proposes a mixed-initiative feature engineering approach using explicit knowledge captured in a knowledge graph complemented by a novel interactive visualization method. Using the explicitly captured relations and dependencies between concepts and their properties, feature engineering is enabled in a semi-automatic way. Furthermore, the results (and decisions) obtained throughout the process can be utilized for refining the features and the knowledge graph. Analytical requirements can then be conveniently captured for feature engineering – enabling integrated semantics-driven data analysis and machine learning.

CCS CONCEPTS

• **Human-centered computing** → **Visualization**; • **Computing methodologies** → **Knowledge representation and reasoning**; **Machine learning**;

KEYWORDS

knowledge graph, feature engineering, machine learning

ACM Reference Format:
Martin Atzmueller and Eric Sternberg. 2017. Mixed-Initiative Feature Engineering Using Knowledge Graphs. In *K-CAP 2017: Knowledge Capture Conference, December 4–6, 2017, Austin, TX, USA.* ACM, New York, NY, USA, 4 pages. https://doi.org/10.1145/3148011.3154473

1 INTRODUCTION

For many areas of machine learning and data analysis such as predictive modeling, anomaly detection, or pattern mining, the majority of the applied methods still considers the data in isolated fashion. In contrast, an effective integrated approach is given by constructing a knowledge graph cf. e. g., [7, 13]: Here, the data is integrated in a comprehensive knowledge structure capturing the relations between concepts and their properties in an explicit way. Then, this structure can be exploited in order to facilitate machine learning and data analysis. However, the knowledge graph mainly focuses on the structuring of the concepts and their relations, while specific modeling tasks, as well as data characteristics (e. g., distributions, correlations) are typically not captured. This second step is the main focus of this paper.

K-CAP 2017, December 4–6, 2017, Austin, TX, USA

ACM ISBN 978-1-4503-5553-7/17/12…$15.00
https://doi.org/10.1145/3148011.3154473

Objectives. This paper adresses this issue by proposing an approach for mixed-initiative feature engineering using the explicitly represented kowledge in a knowledge graph. In particular, we present an according process model including a novel interactive visualization method for feature engineering, i. e., supporting feature construction and selection. In addition, the obtained feature engineering results can be also applied for refining the knowledge graph, e. g., by extending the graph or modifying its construction process. In this way, we provide both a knowledge-based and data-driven way for integrated data analysis and knowledge capture.

Contribution. Our contribution is summarized as follows: We present a mixed-initiative feature engineering approach using explicit knowledge captured in a knowledge graph. For that, we propose a process model integrating a novel interactive visualization method, for semi-automatic feature engineering. We discuss the different steps of the process and describe the visualization method in detail. Furthermore, we summarize first results in an industrial real-world context given by an (anonymized) case study.

The rest of the paper is organized as follows: Section 2 discusses related work. After that, Section 3 describes the proposed approach for mixed-initiative feature engineering using knowledge graphs. Next, Section 4 summarizes an anonymized case study applying the proposed approach. Finally, Section 5 concludes with a summary and interesting directions for future work.

2 RELATED WORK

Existing works utilizing semantic structures in data mining [10, 14, 16, 19] focuses on applying ontologies in the data mining step. However, so far the approaches only apply a "shallow" coupling. Here, advanced knowledge-rich representations like knowledge graphs [4, 17, 20] are a prominent research direction, e. g., [7, 13]. In contrast to existing approaches for feature engineering, e. g., [1–3, 20], this work proposes to use knowledge graphs for feature engineering in a mixed-initiative approach: Utilizing the formalized relations, a semi-automatic visualization method enables advanced feature engineering, knowledge capture, and refinement.

For visualization and browsing graphs and network structures [12], there are a variety of techniques, including visualizations for showing densely connected subgroups (as sets of nodes) [11], special cluster visualizations [8], and interactive techniques for analyzing connected graph structures [9]. In contrast, this paper focuses on feature engineering using knowledge graphs, visualizing feature dependencies to support feature selection and construction.

For feature engineering also explanation-awareness plays an important role. In particular, if explanations for the complete models, or parts thereof can be provided, then the acceptance can often be significantly improved, e. g., [6]. In the proposed approach, this is enabled by inspecting the underlying data and formalized relations at the relevant representational level and dimensions.

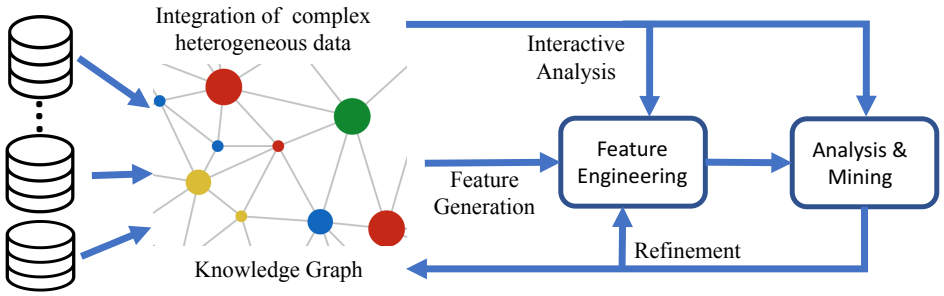

Figure 1: Overview on the proposed framework for mixed-initiative feature engineering using knowledge graphs.

3 METHOD

Below, we first outline and summarize the mixed-initiative approach for feature engineering using knowledge graphs. After that, we describe the novel visualization method in detail.

3.1 Overview

The proposed mixed-initiative process is depicted in Figure 1. We start with heterogeneous complex datasets that are integrated into a knowledge graph structure. After that, this knowledge structure can be applied for machine learning and data analysis. We do not describe the initial construction of the knowledge graph in detail. Instead, we focus on the subsequent feature engineering step, e. g., enabling feature selection and feature construction.

In this paper, we propose a mixed-initiative approach, such that first a set of relevant (initial) features is compiled (*feature generation*) using the concepts and relations modeled in the knowledge graph. This can happen, e. g., by selecting concepts of specific types, such as parts of specific machinery in industrial applications, and attributes describing their properties, as encoded by relations in the knowledge graph. Then, a semi-automatic process is initiated (i. e., for *feature engineering*), guided by interactive visualization as described below (*interactive analysis*). Finally, the engineered features are provided for data analysis and machine learning (*analysis and mining*). In addition, decisions taken during the *feature engineering* step and/or the *analysis and mining* step can be utilized for *refinement*: They can, for example, be (re-)integrated into the knowledge graph, such as formalizing/adding/extending annotations of features (represented by concepts) or refining the respective relations, e. g., based on the information that one feature is strongly correlated with another one. Furthermore, the knowledge graph structure itself, as well as the set of generated and/or selected features can be iteratively adapted based on the obtained insights.

We provide examples of such actions in the real-world case study sketched below. Finally, then also the knowledge graph itself can be applied for providing additional context regarding the results of the *analysis and mining* step, as well as during the *feature engineering* step, e. g., to provide explanations [6, 18]. This is facilitated by the links to the originating concepts and the covered data records, respectively, which can then be exploited, e. g., by exploring a node's neighborhood in the knowledge graph, or by browsing and/or inspecting the respective sets of objects [5].

3.2 Visualization

For inspecting and assessing the generated features, their characteristics and dependencies, we propose a visualization called the *feature matrix plot*, see Figure 2: It consists of several (sub-)plots, for each pair of features denoted by the corresponding row and column labels. As an example, we consider the feature pair A1/A3 (indicated by the red rectangle). The yellow subplot at (1,1) provides information about the domain of A1 via a histogram, accordingly the subplot at (3,3) for A3. The gray subplot at (1,3) shows a heat map, while the subplot at (3,1) provides information about missing values and the (Spearman) correlation between A1 and A3. A detailed view on two exemplary heat map subplots for the feature pairs A2/A4 and A2/A5 is given to the right of Figure 2: Each subplot shows a different grid, derived from nominal (or appropriately discretized intervals) of the two feature domains; the vertical partition comes from the "row" feature (here A2) and the horizontal partition from the "column" feature (here A4 resp. A5). Therefore, all partitionings of a subplot represent all possible value realizations for the given feature pair. The hotter the color, the larger the number of individuals in that partition. Each heat map is normalized so the highest occurrence is red and the lowest blue, the color gradient shows all possible pixel colors. The heat map can also be understood as an 2D Histogram (from above) in which the bins are represented by the single grid elements; the pixel color represents their height. For numeric features, we basically focus on the other subplots (below).

The subplots shown in gray depict three different types of information. First the background indicates the proportion between individuals used for generating the subplots and the individuals that have least one missing value for one of the paired features (indicated in yellow) – for data quality assessment and cross-checks. The colored oval in the center of the subplot displays the correlation between the paired features whereby the extent on the x-axis shows the correlation between the discretized intervals; the extent on the y-axis shows the correlation between the original (undiscretized) domains. Furthermore the oval's color provides information about the direction of a possible correlation (green=positive, red=negative); the color inside stands for the direction of the discretized intervals. The outer ovals' color shows the direction of the original domain. A gray color indicates that at least one of the paired features is nominal, whereas numerical and ordinal features provide richer visualizations. Overall, the idea here is to provide a fast visual method to detect, e. g., data quality and discretization problems via strongly skewed ovals and strongly correlated features by big circles.

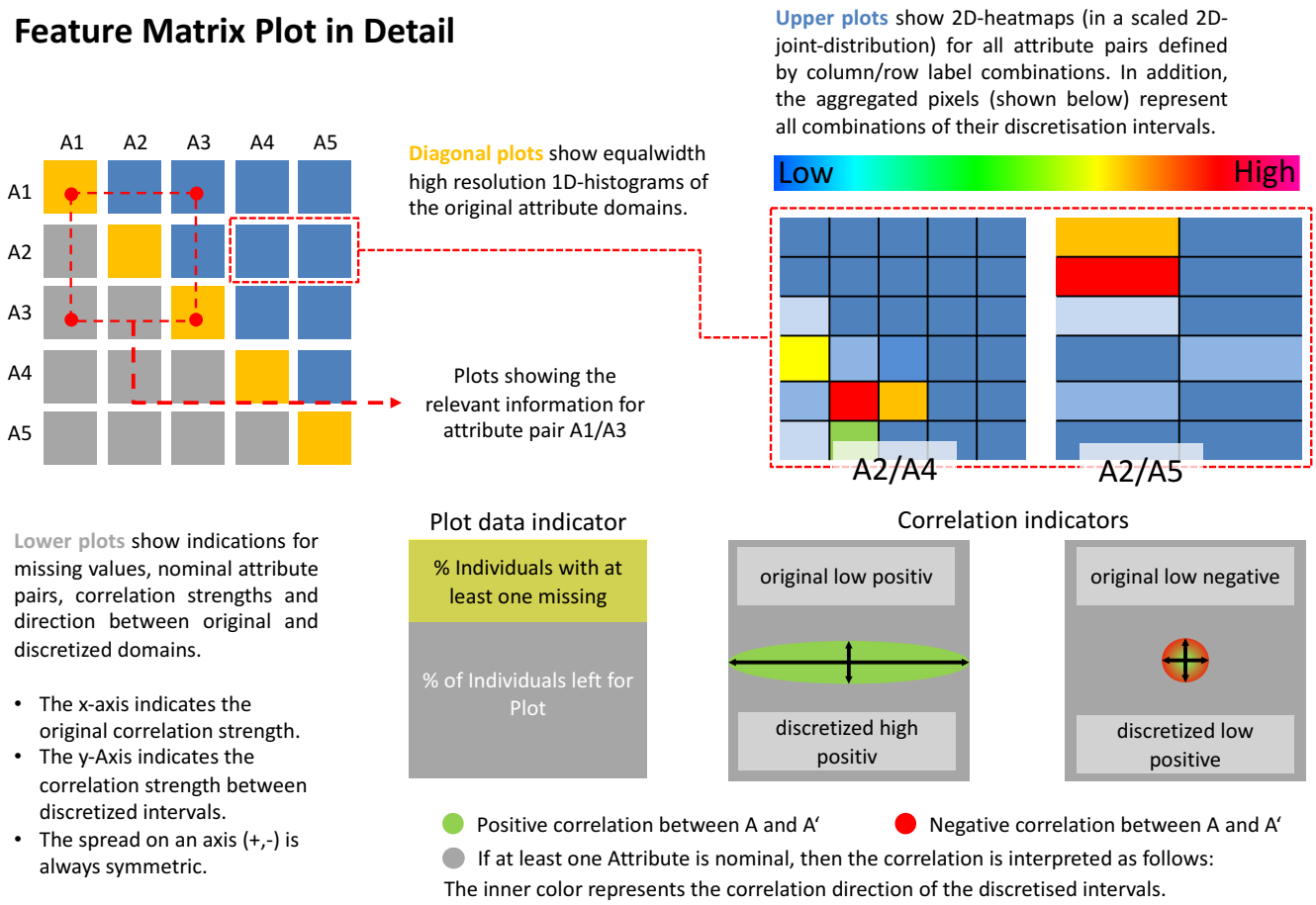

Figure 2: Schematic breakdown of the feature matrix plot for five attributes. The feature matrix plot is made up of three individual subplot types that provide valuable information, e. g., for feature engineering, knowledge graph optimization, discretization and spotting datasource problems.

4 CASE STUDY

The introduced semi-automatic approach utilizing the matrix plots for interactive analysis has been applied in an (anonymized) industrial case study, where data analysis was applied to productive logistic data. One major problem that occured in the iterative workflow was the large amount of concepts/features contained in the knowledge graph which had then to be "condensed" to a set of meaningful features that are relevant for the analysis. For that purpose, we applied the proposed approach including the novel visualization method. In particular, this allowed us to perform feature engineering and selection in an iterative process considering knowledge graph construction, feature construction, and refinement. By interpreting the visualization the set of features can be optimized, e. g., by removing strongly correlated attributes and detect/fix problematic domain discretizations before running costly data analysis tasks. If it is necessary to resign on features the plot can be utilized to focus on the most informative and available attributes. In that way, also the experts could be involved into this interactive visual process in order to provide feedback on the constructed feature set.

Furthermore, important feedback could be derived from the visualizations regarding the source data that was used for generating the knowledge graph. For that, basically a complex knowledge graph data structure was evolved from plain relational logistic data. The structure was a composition from two graphs that, e. g., represent the material flows, and composition of different processes and products. As a particularly important case, the 1D histogram was used to refine a central KPI feature, which was used for data analysis and further machine learning approaches. This KPI feature was constructed from the graphs' structure itself capturing the experts' beliefs. Then, using the visualization the relations, dependencies and correlations to other concepts from the knowledge graph could be checked, and expectations of the domain specialists could be verified. This allowed us to both construct the feature according to their domain knowledge and exploiting/refining the knowledge graph at the same time, e. g., by adding relations that were missing, or by removing incorrect relations and dependencies.

Figure 3 shows a feature matrix plot of an already improved feature set from a knowledge graph where 18 attributes have been selected. It is easy to see that the domain distributions are quite

exotic. Here, we also include some information on the "evolution" of some features. As an example, consider an earlier version of an important feature denoted by A16. Here, the distributions and correlations were not well expressed according to domain knowledge. The final refined version is given by A11; here, we can also note the absence of noise which directly visualizes the improved mapping of experts knowledge. A5 shows an discretization problem as it can be seen in all subplots A5 is involved. Here, all correlation ovals are extremely stretched on the x-axis, also the corresponding heatmaps show only a resolution of one "logical" pixel on the y-axis. The problem was that all individuals fall in only one discretization interval. The final feature set was improved by replacing the numeric A5 by an adapted nominal version (and by removing feature A16, as discussed above). Alltogether, the proposed approach proved crucial in feature engineering and in also refining the knowledge graph in order to be applicable for data analysis and machine learning.

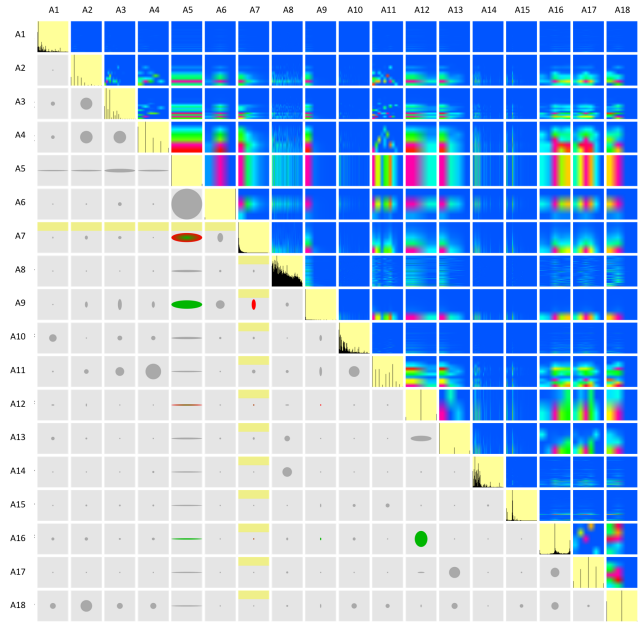

Figure 3: An example of a feature matrix plot: The plot has been generated from real-world (anonymized) industrial data in the domain of productive logistics.

5 CONCLUSIONS

This paper proposed an approach for mixed-initiative feature engineering using the explicitly represented kowledge captured by a knowledge graph. In particular, we presented an according process model including a novel interactive visualization method. In addition, the feature engineering results can be also applied for checking data and knowledge characteristics and refining the knowledge graph and feature sets, respectively. This was sketched in the context of an (anonymized) real-world industrial case study in the domain of productive logistics, where we also summarized first results and experiences. Here, we discussed the different steps of the process and described the visualization method in detail.

Our results indicate the efficacy of the proposed approach. The domain specialists could utilize the proposed approach very well. Guided by the visualization, they could easily interpret its results. Working together with the data scientists in the feature engineering phase, they were effectively supported using the proposed approach, especially applying the presented visualization method. Furthermore, the knowledge graph was also iteratively adapted and refined according to the outcomes of the modeling and analysis phases.

For future work, we aim to further extend explanation-aware approaches utilizing the semantic structures formalized in the knowledge graph in order to provide ad-hoc knowledge in context. This also connects to further visualization approaches for supporting detailed inspection of knowledge graph components, and its refinement [15]. Further interesting directions concern scalable machine learning and data analysis methods on complex knowledge graphs targeting predictive as well as descriptive approaches.

REFERENCES

[1] Taqwa Ahmed Alhaj, Maheyzah Md Siraj, Anazida Zainal, Huwaida Tagelsir Elshoush, and Fatin Elhaj. 2016. Feature Selection Using Information Gain for Improved Structural-Based Alert Correlation. *PloS One* 11, 11 (2016), e0166017.
[2] Martin Atzmueller, Joachim Baumeister, Achim Hemsing, Ernst-Jürgen Richter, and Frank Puppe. 2005. Subgroup Mining for Interactive Knowledge Refinement. In *Proc. AIME (LNAI 3581)*. Springer Verlag, Heidelberg, Germany, 453–462.
[3] Martin Atzmueller, Naveed Hayat, Andreas Schmidt, and Benjamin Klöpper. 2017. Explanation-Aware Feature Selection using Symbolic Time Series Abstraction: Approaches and Experiences in a Petro-Chemical Production Context. In *Proc. IEEE INDIN*. IEEE Press, Boston, MA, USA.
[4] Martin Atzmueller, Benjamin Kloepper, Hassan Al Mawla, Benjamin Jäschke, Martin Hollender, Markus Graube, David Arnu, Andreas Schmidt, Sebastian Heinze, Lukas Schorer, Andreas Kroll, Gerd Stumme, and Leon Urbas. 2016. Big Data Analytics for Proactive Industrial Decision Support. *atp edition* 58, 9 (2016).
[5] Martin Atzmueller and Frank Puppe. 2008. A Case-Based Approach for Characterization and Analysis of Subgroup Patterns. *Appl. Intell.* 28, 3 (2008), 210–221.
[6] Martin Atzmueller and Thomas Roth-Berghofer. 2010. The Mining and Analysis Continuum of Explaining Uncovered. In *Proc. AI-2010*. London, UK.
[7] Christian Bizer, Jens Lehmann, Georgi Kobilarov, Sören Auer, Christian Becker, Richard Cyganiak, and Sebastian Hellmann. 2009. DBpedia - A Crystallization Point for the Web of Data. *Web Semantics* 7, 3 (2009), 154–165.
[8] Brigitte Boden, Roman Haag, and Thomas Seidl. 2013. Detecting and Exploring Clusters in Attributed Graphs: A Plugin for the Gephi Platform. In *Proc. ACM CIKM*. ACM, New York, NY, USA, 2505–2508.
[9] Duen Horng Chau, Leman Akoglu, Jilles Vreeken, Hanghang Tong, and Christos Faloutsos. 2012. TourViz: Interactive Visualization of Connection Pathways in Large Graphs. In *Proc. ACM KDD*. ACM, New York, NY, USA, 1516–1519.
[10] Dejing Dou, Hao Wang, and Haishan Liu. 2015. Semantic Data Mining: A Survey of Ontology-based Approaches. In *IEEE ICSC*. IEEE, 244–251.
[11] Santo Fortunato. 2010. Community Detection in Graphs. *Physics Reports* 486, 3-5 (2010), 75 – 174.
[12] Ivan Herman, Guy Melançon, and M Scott Marshall. 2000. Graph Visualization and Navigation in Information Visualization: A Survey. *IEEE Transactions on Visualization and Computer Graphics* 6, 1 (2000), 24–43.
[13] Johannes Hoffart, Fabian M Suchanek, Klaus Berberich, and Gerhard Weikum. 2013. YAGO2: A Spatially and Temporally Enhanced Knowledge Base from Wikipedia. *Artificial Intelligence* 194 (2013), 28–61.
[14] Ahsan Morshed, Ritaban Dutta, and Jagannath Aryal. 2013. Recommending Environmental Knowledge as Linked Open Data Cloud using Semantic Machine Learning. In *Proc. IEEE ICDEW*. IEEE, 27–28.
[15] Heiko Paulheim. 2017. Knowledge Graph Refinement: A Survey of Approaches and Evaluation Methods. *Semantic web* 8, 3 (2017), 489–508.
[16] Heiko Paulheim and Johannes Fümkranz. 2012. Unsupervised Generation of Data Mining Features from Linked Open Data. In *Proc. WIMS*. ACM, 31.
[17] Petar Ristoski and Heiko Paulheim. 2016. Semantic Web in Data Mining and Knowledge Discovery: A Comprehensive Survey. *Web Semantics* 36 (2016), 1–22.
[18] Ilaria Tiddi, Mathieu d'Aquin, and Enrico Motta. 2015. An Ontology Design Pattern to Define Explanations. In *Proc. K-Cap*. ACM, New York, NY, USA.
[19] Anze Vavpetic, Vid Podpecan, and Nada Lavrac. 2014. Semantic Subgroup Explanations. *Journal of Intelligent Information Systems* 42, 2 (2014), 233–254.
[20] Xander Wilcke, Peter Bloem, and Victor de Boer. 2017. The Knowledge Graph as the Default Data Model for Learning on Heterogeneous Knowledge. *Data Science* Preprint (2017), 1–19.