Eliciting Structure in Data

Anders Holst
RISE SICS, Sweden
anders.holst@ri.se

Ahmad Al-Shishtawy
RISE SICS, Sweden
ahmad.al-shishtawy@ri.se

Juhee Bae
University of Skövde, Sweden
juhee.bae@his.se

Mohamed-Rafik Bouguelia
CAISR, Halmstad, Sweden
mohbou@hh.se

Göran Falkman
University of Skövde, Sweden
goran.falkman@his.se

Sarunas Girdzijauskas
RISE SICS, Sweden
sarunasg@kth.se

Olof Görnerup
RISE SICS, Sweden
olof.gornerup@ri.se

Alexander Karlsson
University of Skövde, Sweden
alexander.karlsson@his.se

Sławomir Nowaczyk
CAISR, Halmstad, Sweden
slawomir.nowaczyk@hh.se

Sepideh Pashami
CAISR, Halmstad, Sweden
sepideh.pashami@hh.se

Alan Said
University of Skövde, Sweden
alansaid@acm.org

Amira Soliman
RISE SICS, Sweden
aaeh@kth.se

ABSTRACT

This paper demonstrates how to explore and visualize different types of structure in data, including clusters, anomalies, causal relations, and higher order relations. The methods are developed with the goal of being as automatic as possible and applicable to massive, streaming, and distributed data. Finally, a decentralized learning scheme is discussed, enabling finding structure in the data without collecting the data centrally.

CCS CONCEPTS

• **Human-centered computing** → **Visualization**; **Graphical user interfaces**; • **Mathematics of computing** → *Causal networks*; • **Theory of computation** → *Unsupervised learning and clustering*; • **Computing methodologies** → *Anomaly detection.*

KEYWORDS

Information Visualization; Clustering; Anomaly Detection; Causal Inference; Higher-Order Structure; Distributed Analytics.

ACM Reference Format:
Anders Holst, Ahmad Al-Shishtawy, Juhee Bae, Mohamed-Rafik Bouguelia, Göran Falkman, Sarunas Girdzijauskas, Olof Görnerup, Alexander Karlsson, Sławomir Nowaczyk, Sepideh Pashami, Alan Said, and Amira Soliman. 2019. Eliciting Structure in Data. In *Joint Proceedings of the ACM IUI 2019 Workshops, Los Angeles, USA, March 20, 2019* , 4 pages.

IUI Workshops'19, March 20, 2019, Los Angeles, USA

1 INTRODUCTION

Information visualization (IV) is concerned with the transformation of abstract data, information, and knowledge into interactive visual representations that amplify human cognition and assist humans in solving problems. IV has proven effective for presenting important information and for making complex analyses in solving big-data problems [7]. In contrast to traditional IV, in Visual Analytics (VA) the user is not just a consumer of information who interprets and acts on visually presented results, but is the driving force of the whole problem solving process, from problem formulation, data selection and cleaning, over hypothesis generation and testing, model building and steering computations, to interpreting results and communicating insights. The scope of VA is very broad, and the area has grown to encompass a wide array of topics. A unifying principle is the need for systems to leverage computational methods in data mining and machine learning for big data analysis. In a VA system, the human user and the computational process operate in coordination in an integrated fashion, forming a continuous and cooperative problem solving loop [3].

Here, we focus on understanding the underlying structure of real-world data sets before applying any artificial intelligence and machine learning algorithm. However, today, making sense of data is quite hard due to high-dimensional aspect of real-world data sets. Therefore it is important to find different structures in the data as automatic as possible to facilitate information visualization.

The endeavour is similar to that of the Automated Statistician [14] in that we want a data set to be automatically analyzed from several different aspects. However, while much of the efforts regarding the Automated Statistician is about characterizing time series or predicting an output variable

from inputs, we focus on unsupervised characterization of the data set as a whole. Also, rather than producing a fixed report, we focus on interactive and exploratory capabilities.

2 STRUCTURE IN THE DATA

"Structure" can mean many different things. The kind of structure we focus on here can be characterized as either *horizontal* or *vertical* properties of the data. Considering a (tabular) data set with different samples row-wise, and different features column-wise, then properties describing how samples relate to each other can be considered as horizontal properties, whereas properties of how different features relate to each other as vertical. Examples of the former kind of structure that we focus on are *clusters* among samples in the data, and finding *anomalous* samples or groups of samples. Examples of the latter kind of structure are relations between the features, such as *statistical correlation*, *causal relations*, and a higher order *similarity relations*.

Clusters

Unsupervised clustering of data is a frequently occurring task in statistical machine learning. Real world data is seldom completely homogeneous, but usually contains clusters representing different varieties of the entities under study [2]. The reasons to detect such clusters are plenty: clusters may represent a useful categorization of the data that is valuable to discern; the significance and value of the correlation between attributes may be misleading when there are clusters; or just as a mathematical trick that complex distribution can be approximated by a sum of simpler distributions. At the same time it is hard to reliably detect clusters. Most algorithms are "unstable" in the sense that different random starting points of iteration will lead to very different clustering results. It is also inherently hard to find the "best" number of clusters. Furthermore, in streaming data, the clusters may not even be fixed but change over time . Finally, in many situations, it is not even possible to objectively measure what the best clustering is, since there may be many equally good alternatives.

Anomalies

In many domains, various units, such as vehicles and vessels, are generating data over time. To diagnose a unit A at each time period T as anomalous or not, the data for A needs to be compared against the data from other similar groups of units over the same time period in an online fashion. Various representations or features can be extracted from each of these units which can capture different anomalies. Some anomalies might be relevant to inspect further, whereas others can be regarded as irrelevant from the human operator perspective. However, it is not easy to determine whether an event is anomalous or not and a human operator often needs to be involved in the process. With the help of a human operator, various deviation levels can be inspected, enriching the view of the potential important anomalies.

Causal structure

For systems that not only learn from data but actually act based on the resulting knowledge, it thus becomes more and more important to identify cause and effect. This is a very challenging task since in the general case it is easy to discover correlations between the attributes in data but is impossible to distinguish which is the cause or effect only based on observational data. There are however in many cases several hints in the data that can potentially be used. We propose a visualization tool which incrementally and gradually discovers causal relations between features as more data becomes available. The results are shown in a form of a causal graph [4] where the strength and uncertainty of causal and correlation links are marked by color and shading [1].

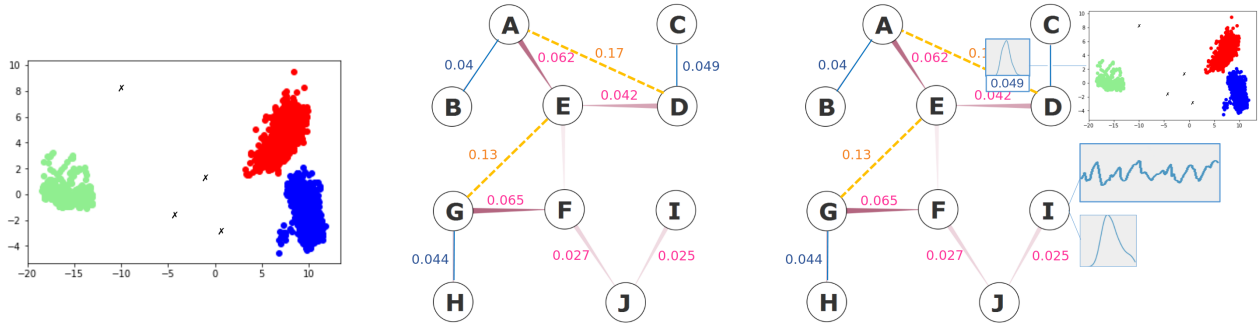
Higher-Order Structures

Appropriately representing data with regard to relevant features for the task at hand is often critical when applying machine learning. Specifically, by building hierarchical - or higher-order - representations, we are able to capture multiple abstraction levels in data, examine information from different perspectives and thereby capture various aspects of it. Such representations are applicable both to solve machine learning tasks at the appropriate granularity level and to perform exploratory data analysis on and across multiple levels of abstraction. In this context, we have proposed a fast and lightweight demonstrator estimating word similarities from the text by explicitly counting second-order co-occurrences [5]. It visualizes the similarity between words in a single pass over data in a form of a tree starting from leaves and building towards the root.

Visualization and Interaction

We have so far implemented separate modules for all the above analyses. It remains to make the different analyses accessible through a combined tool, to visualize and explore them simultaneously. Figure 1 illustrates one possible way of doing this. (The pictures are not representing any real data but are manually constructed to illustrate the principles.)

The horizontal properties relating samples to each other as described above, i.e. clusters and anomalies/outliers, are very natural to display in the same view (figure 1a), by showing each cluster in a different color and then assigning the outliers a (markedly) different color or symbol to make them stand out. Here it is illustrated by a scatter plot, but the same principle works also with time series.



(a) Horizontal properties (b) Vertical properties (c) Combined view of structure

Figure 1: A mock-up illustration of horizontal and vertical properties of the data, and a possible way to combine them in the same view.

The vertical properties relating features to each other, i.e correlations, causal relations and higher order relations, can also be visualized in the same view (figure 1b). Different shapes and colors can be used for the different types of relations. Here direct correlations (without established causal direction) are shown in blue, links with established causal direction as narrowing links in purple, and links representing higher order similarity in hatched orange.

Our suggestion for combining the horizontal and vertical views into one (figure 1c) is to start with the vertical view showing all features and their relations, and interactively popping up the horizontal properties, as scatter plots or time series, for those features or links that the user wishes to explore further. In this way all the data and its different types of structure is made readily accessible to the user to explore.

3 SHARING KNOWLEDGE WITHOUT SHARING DATA

The predominant way of using machine learning involves collecting all the data in a data center. The model is then trained on powerful servers. However, this data collection process is often privacy-invasive. Users may not want to share private data with companies, making it difficult to use machine learning in some situations. For example, users generate vast amounts of data through interaction with the applications installed on their mobile devices. This data is often deeply private in nature and should not be shared completely with a remote server. Even when privacy is not a concern, having to collect data in one central storage can be unfeasible. For example, self-driving cars generate too much data to be able to send all of it to a server.

Federated Learning approach provides distributed and privacy-friendly approach by allowing users to train models locally using their sensitive data, and communicate intermediate model updates to a central server without the need to

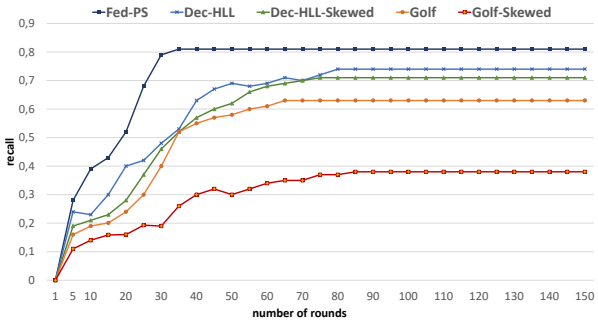


Figure 2: Comparative analysis of performing a clustering task using our proposed scheme (Dec-HLL), federated learning framework (Fed-PS) and gossip learning framework (GOLF). Fed-PS achieves the highest performance as a result of performing model aggregation centrally. Yet, our scheme achieves higher performance compared to P2P learning scheme used in Golf, particularly the performance gap increases when the data distribution is very skewed.

centrally store the data [8, 10, 13]. Specifically, users start by contacting the central server and downloading the learning algorithm and a global model that is common to all users. The algorithm trains its model locally on each device using user data and computes the update to the current global model. Afterwards, the new updates on the learning parameters obtained from the algorithm on the device of each user are sent to the central server for aggregation step. The server integrates the new learning outcomes as if it were directly trained on the data and sends the aggregated global model back to each user. This distributed approach for model computation diminishes the need to centrally store the data, hence, computation becomes distributed among the users and their personal data never leaves their devices.

In this context, we present a *decentralized learning scheme* that removes the dependency on a central aggregating authority in order to offer a more scalable Federated Learning approach. Our proposed scheme organizes user devices in a Peer-to-Peer (P2P) network allowing machine learning tasks to cooperate among each other using Gossip protocols [12]. Furthermore, our learning scheme operates as a decentralized learning and knowledge sharing approach that considers how strongly each local update is backed up by a representative volume of data. Particularly, our scheme integrates Hyperloglog [6] as a cardinality estimation mechanism to maintain the number of data items used in generating and incrementally updating the models exchanged among participating devices. Figure2 shows performance analysis results of our scheme (i.e., Dec-HLL) compared to two state-of-the-art techniques: federated learning scheme (i.e., Fed-PS) [9] and Gossip learning (i.e., Golf) [12]. As shown, our scheme achieves higher accuracy than P2P learning scheme adopted in Golf, specifically, our accuracy is much closer to the accuracy of Fed-PS when the data distribution is very skewed.

4 A PLATFORM FOR ELICITING STRUCTURE IN DATA

We have taken the first steps to implement a platform to simplify the above discussed exploratory data analysis and visual analytics. The purpose is to be able to get a first quick overview of various useful structures in a data set. To make it generally applicable, we base it on an existing big data platform HOPS [11], develop the code in Python, and do the interaction and visualization through Jupyter notebooks. This make development flexible and easy, and the chances of it to be useful for others. The intention is to publish the entire package as open source.

5 ACKNOWLEDGEMENTS

This research has been conducted within the "A Big Data Analytics Framework for a Smart Society" (BIDAF) project (http://bidaf.sics.se/) supported by the Swedish Knowledge Foundation.

REFERENCES

[1] Juhee Bae, Tove Helldin, and Maria Riveiro. 2017. Understanding Indirect Causal Relationships in Node-Link Graphs. *Computer Graphics Forum* (2017). https://doi.org/10.1111/cgf.13198

[2] Mohamed-Rafik Bouguelia, Alexander Karlsson, Sepideh Pashami, Sawomir Nowaczyk, and Anders Holst. 2018. Mode Tracking Using Multiple Data Streams. *Inf. Fusion* 43, C (Sept. 2018), 33–46. https://doi.org/10.1016/j.inffus.2017.11.011

[3] Remco Chang, David S. Ebert, and Daniel Keim. 2014. Introduction to the Special Issue on Interactive Computational Visual Analytics. *ACM Trans. Interact. Intell. Syst.* 4, 1, Article 3 (April 2014), 3 pages. https://doi.org/10.1145/2594648

[4] Diego Colombo and Marloes H. Maathuis. 2014. Order-independent Constraint-based Causal Structure Learning. *J. Mach. Learn. Res.* 15, 1 (Jan. 2014), 3741–3782.

[5] Olof Görnerup, Daniel Gillblad, and Theodore Vasiloudis. 2017. Domain-agnostic Discovery of Similarities and Concepts at Scale. *Knowl. Inf. Syst.* 51, 2 (May 2017), 531–560. https://doi.org/10.1007/s10115-016-0984-2

[6] Hazar Harmouch and Felix Naumann. 2017. Cardinality estimation: an experimental survey. *Proceedings of the VLDB Endowment* 11, 4 (2017), 499–512.

[7] Daniel A. Keim, Huamin Qu, and Kwan-Liu Ma. 2013. Big-Data Visualization. *IEEE computer graphics and applications* 33 4 (2013), 20–1.

[8] Jakub Konecnỳ, H Brendan McMahan, Daniel Ramage, and Peter Richtárik. 2016. Federated optimization: Distributed machine learning for on-device intelligence. *arXiv preprint arXiv:1610.02527* (2016).

[9] Mu Li, David G Andersen, Jun Woo Park, Alexander J Smola, Amr Ahmed, Vanja Josifovski, James Long, Eugene J Shekita, and Bor-Yiing Su. 2014. Scaling Distributed Machine Learning with the Parameter Server.. In *OSDI*, Vol. 14. 583–598.

[10] H Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, et al. 2016. Communication-efficient learning of deep networks from decentralized data. *arXiv preprint arXiv:1602.05629* (2016).

[11] Salman Niazi, Mahmoud Ismail, Seif Haridi, Jim Dowling, Steffen Grohsschmiedt, and Mikael Ronström. 2017. HopsFS: Scaling Hierarchical File System Metadata Using NewSQL Databases.. In *FAST*. 89–104.

[12] Róbert Ormándi, István Hegedűs, and Márk Jelasity. 2013. Gossip learning with linear models on fully distributed data. *Concurrency and Computation: Practice and Experience* 25, 4 (2013), 556–571.

[13] Virginia Smith, Chao-Kai Chiang, Maziar Sanjabi, and Ameet S Talwalkar. 2017. Federated multi-task learning. In *Advances in Neural Information Processing Systems*. 4424–4434.

[14] Christian Steinrucken, Emma Smith, David Janz, James Lloyd, and Zoubin Ghahramani. [n. d.]. The Automatic Statistician. 175–188.