Know Your Dataset

Poonam Kumari
Supervised by Dr. Oliver Kennedy
State University of New York at Buffalo, Buffalo, NY, USA
{poonamku,okennedy}@buffalo.edu

ABSTRACT

It has become very easy to obtain a large dataset for experimental analysis. Along with the ease arises the need to document the dataset for future use. Apart from documentation another challenge is posed by unlabeled or poorly labeled. Absence of labels, leads to difficulty in accessing the data.

We propose the design of a system, the first part of the system called LOKI would serve as a knowledge base for storing rules and column-naming heuristics, as well as provide an interactive tool: the LOKI editor for populating the knowledge-base. The second part of the system called DOKI would help start documentation for a dataset.

PVLDB Reference Format:
Poonam Kumari, supervised by Dr. Oliver Kennedy. Know Your Dataset. *PVLDB*, 11 (8): xxxx-yyyy, 2018.
DOI: https://doi.org/TBD

1. MOTIVATION

Big datasets are available in abundance and are being used by data scientists and database community for research purpose. These datasets are often curated, analyzed and forgotten without any documentation about the dataset itself. We propose to end this cycle by designing a system with a central goal of inferring column names by creating a knowledge base, which would store a collection of rules and column-naming heuristics (LOKI: Label Once and Keep It), as well as help start the documentation for a dataset.

So when presented a new dataset our system would start with creating labels using LOKI which would also help an analyst start documenting the dataset. Apart from creating a knowledge base LOKI provides an interactive tool for populating the knowledge base.

2. RESEARCH QUESTIONS

Proceedings of the VLDB Endowment, Vol. 11, No. 8

DOI: https://doi.org/TBD

Given a dataset the user wants to analyze, the system performs two steps (1) create labels (2) help start the documentation. Once a dataset is documented the analyst can save it for future use.

LOKI helps in streamlining the process of developing schemas for unlabeled or poorly labeled datasets. It provides users with two modes of interaction: (1) A labeling interface that assists users in assigning names to existing columns of data, and (2) A discovery interface that helps users to search for columns representing particular concepts of interest. These interfaces are supported by a knowledge-base that combines expert-provided heuristics, learned characteristics, as well as historical feedback gathered from users about already-loaded datasets. Once the user has labeled or discovered a sufficient set of columns, LOKI generates appropriate data loading/initialization code (e.g., a CREATE TABLE or Spark DataFrame initializer).

Once a knowledge base is created, the next step is to start documenting the dataset. 1 illustrates the different components of dataset that LOKI would store.

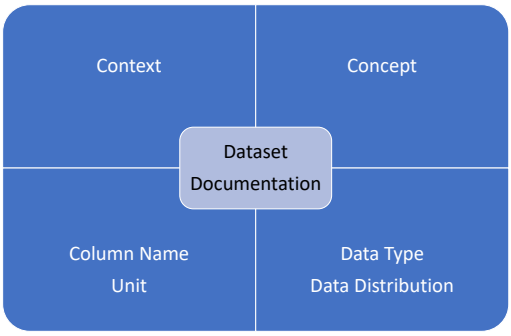

Figure 1: Overview of Dataset Documentation

If a user points LOKI at a new dataset, LOKI proposes a schema for it. Input for a labeling query that helps infer column name for a new dataset is a set of columns and the knowledge base. The goal of the query is to identify a list of N distinct concepts, one for each attribute, that best fit the data in columns. For achieving this goal LOKI takes into account the context, concept, column name (if any), unit used in the column data, type of data present and data distribution of the column.

EXAMPLE 2.1. *If LOKI is presented with 2, then the documentation would store details about columns and the dataset itself. For instance the values stored for column Fruithectares*

Fruithectares	Vegcount	Fruityieldtons
5.8	0	1.5
0	1.26	0
2.6	0	7.2
4.05	0.089	1.1
0.17	6.5	0.48

Table 1: CropYields Dataset

in 2 would be context: agriculture, column name: Fruithectares, unit: Hectares, Data type: Float, Data distribution: Uniform.

Context stores the details of the domain to which the dataset belongs. Concepts correspond to names. Since the same name may be used in different contexts, multiple concepts can use the same column name. Similarly, a single concept may be associated with multiple names. Unit refers to the mathematical unit associated with a column. e.g. Kilogram, Hectares, etc. Name stores the column name. Type of data present in the dataset whether it is numerical, categorical, datetime etc are stored as data type. Data distribution refers to the distribution that best describes a column. e.g Normal, uniform, zipfian etc.

Once LOKI starts documenting the dataset, analyst does not have to document it again, and is well versed with the dataset through information stored in knowledge base. The core component of LOKI is a knowledge-base that is used to identify column names. The knowledge-base is broken down into two parts, one heuristic-based, and the other feedback-based. In short, LOKI will allow users to assemble schemas on-demand, both (re-)discovering and incrementally refining schema definitions in response to changing data needs.

2.1 Challenges

LOKI inferences column names

1. Column might match on multiple signatures
2. Similar concepts with different signatures
3. Similar signatures for different concepts
4. Insufficient signal for signature based matching. -¿types of signature insufficient
5. different signatures combine differently to id concepts
6. performance

3. RELATED WORK

Other works have used data distribution for similar purposes. For example GestureQuery [4] uses data similarity between two attributes to select candidate attributes for an equi-join.

Wrangler [3] and Potter's wheel [6] detect data domains through inclusion functions (e.g. regular expressions). Wrangler in particular infers the data type of a column and highlights errors based on inconsistent data types. Wrangler also has several operators like split and unfold that create new columns. The split operator decomposes composite data values into component distributions. The unfold operator reverses a table pivot, collapsing data laid out as key-value pairs into columns. A useful application of the LOKI knowledge-base that we hope to explore in future work is using it to detect opportunities for applying such operators.

An orthogonal approach to modeling and matching columns is to use ontologies, which express entities, facts, relations between facts and properties of relations Ontologies like Yago [1] could be used to identify semantic properties that relate columns.

A data summary called the data describer is used in [5]. The data types, correlations and distributions of the attributes in a private dataset are listed. Each attribute is categorized into either numerical or non-numerical. If non-numerical attribute cannot be parsed as datetime then it is considered to be a string.

Data describer takes in a CSV file and infers the data types and domains. The attribute datatypes are parsed as numerical, datetime or string. We are inferring the datatypes as well. When run in correlated attribute mode, data describer provides corelation between attributes. We could use this functionality in LOKI.

PADS [2] helps users to understand the layout and meaning of data by designing syntactic descriptions of the data. Based on the syntax, accumulators track the number of good values, the number of bad values, and the distribution of legal values. This technique could be used in LOKI to help capture expert knowledge.

There is an increasing number of datasets in which well-structured attributes (with or without a name) can be identified, each containing a set of values called a domain. There is lack of schema description in most of the datasets. LSH Ensemble is used in [7] to find domains that maximally contain a query dataset, which can help to find datasets that best augment a given set of data.

4. RESEARCH PLAN

We plan several extensions to LOKI focused on building and refining a knowledge-base for storing column-naming heuristics. Use of contextual information: Contextual information such as ontologies, units and data domains could be used to augment the LOKI knowledge base (KB). We could use a network of semantic relations such as BabelNet strengthen data models for training the KB as well as providing curation recommendations to experts. [2] Recommendation of columns for query: Concepts in the KB could be used to recommend columns that could be in a query based on the columns that are already present. Smarter Matchers:We would develop matchers which regocognize a wider spectrum of data. For example, regular expression matchers could be used to detect geolocation data. Matchers which can identify synonymous labels could help experts in the curation process.

5. CONCLUSION

This paper proposes the design of a system which would help build a knowledge base along with an interactive tool for populating the knowledge base.

6. ACKNOWLEDGMENTS

This work was supported by NSF Awards IIS-1750460, ACI-1640864 and by a gift from Oracle. The conclusions and

opinions in this work are solely those of the authors and do not represent the views of the National Science Foundation or Oracle.

7. REFERENCES

[1] M. Fabian, K. Gjergji, W. Gerhard, et al. Yago: A core of semantic knowledge unifying wordnet and wikipedia. In *16th International World Wide Web Conference, WWW*, pages 697–706, 2007.

[2] K. Fisher and R. Gruber. Pads: a domain-specific language for processing ad hoc data. In *ACM Sigplan Notices*, volume 40, pages 295–304. ACM, 2005.

[3] S. Kandel, A. Paepcke, J. Hellerstein, and J. Heer. Wrangler: Interactive visual specification of data transformation scripts. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 3363–3372. ACM, 2011.

[4] A. Nandi, L. Jiang, and M. Mandel. Gestural query specification. *Proceedings of the VLDB Endowment*, 7(4):289–300, 2013.

[5] H. Ping, J. Stoyanovich, and B. Howe. Datasynthesizer: Privacy-preserving synthetic datasets. In *Proceedings of the 29th International Conference on Scientific and Statistical Database Management*, page 42. ACM, 2017.

[6] V. Raman and J. M. Hellerstein. Potter's wheel: An interactive data cleaning system. In *VLDB*, volume 1, pages 381–390, 2001.

[7] E. Zhu, F. Nargesian, K. Q. Pu, and R. J. Miller. Lsh ensemble: Internet-scale domain search. *Proceedings of the VLDB Endowment*, 9(12):1185–1196, 2016.