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. But most of these datasets are unlabeled or poorly labeled. Absence of labels, leads to difficulty in accessing the data.

We propose the design of a system LOKI, which would serve as a knowledge base for storing column-naming heuristics, as well as an interactive tool: the LOKI editor for populating the knowledge-base.

Poonam:paraphrase, taken from HILDA paper

The LOKI editor primes the knowledge base by learning from example data (e.g., from open data portals), and assists domain experts in reviewing and refining the resulting heuristic naming schemes. We identify specific issues arising from training and show how the LOKI editor streamlines the process of manually repairing these issues.

PVLDB Reference Format:
Ben Trovato, G. K. M. Tobin, Lars Thørväld, Lawrence P. Leipuner, Sean Fogarty, Charles Palmer, John Smith, Julius P. Kumquat, and Ahmet Sacan. A Sample Proceedings of the VLDB Endowment Paper in LaTeX Format. *PVLDB*, 11 (8): xxxx-yyyy, 2018.
DOI: https://doi.org/TBD

1. MOTIVATION

Big datasets are available in abundance which 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. If the dataset needs to be used by another analyst. We propose to end this cycle by designing a system which would help with documenting the dataset. The system would consist of a knowledge base that would store different aspects of a dataset, for instance the data type of a column, data distribution, column name, etc. which can be used to document the dataset.

The knowledge base would store column-naming heuristics which would help user to identify schema names for new datasets. So when presented a new dataset our system would help document the dataset, create a knowledge base for storing column-naming heuristics and provide an interactive tool for populating the knowledge base.

Proceedings of the VLDB Endowment, Vol. 11, No. 8
Copyright 2018 VLDB Endowment 2150-8097/18/4.
DOI: https://doi.org/TBD

2. RESEARCH QUESTIONS

Given a dataset the user wants to analyze, our goal is to help the user document the dataset and reuse the dataset without additional effort in getting to know the data once again. Our first step towards building such a system is to create a knowledge base. Once the details about a dataset are stored and documented, analyst can reuse it without any difficulty. The analyst does not have to document the dataset again, and is well versed with the dataset through the information stored in knowledge base(data type, data distribution). The knowledge base would store expert inputs and feedback and also help the analyst in identifying schema names for new datasets. The system (referred to as LOKI) serves as a knowledge base and an interactive tool for populating the knowledge base. When a user first points LOKI at a new tabular data set, LOKI proposes a schema for it. It then collects feedback, both learning and also preserving schema metadata for later use.

Consider a scenario where an analyst has to deal with an unlabeled or poorly labeled dataset and analyze it. Since the dataset is unlabeled, the first step user needs to perform is to guess the schema names. We propose the design of a system LOKI, which would serve as a knowledge base for storing column-naming heuristics, as well as an interactive tool for populating the knowledge base.

Poonam:rewrite, taken from HILDA paper

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.

However, to accomplish this, we first need a knowledge-base of heuristics, domain knowledge, and dirty tricks. In principle, we might implement such a knowledge base as a neural network, train it on example data, and use it to suggest schemas. However, this approach suffers from a host of limitations: (1) Lack of generality: A neural network tuned for use with one type of training data may need to be retuned for other scenarios. (2) Lack of extensibility: Adding new training data or knowledge network misclassifies a column name, it can be hard to debug and refine the result. Instead, we opt for an interactive approach. We present the LOKI knowledge-base, a flexible, extensible infrastructure for collecting schema-naming heuristics.We show how simple, efficient off-the-shelf techniques can be used to quickly prime the knowledge-base using example data. Then, we introduce an interactive editor that we are designing as part of LOKI to allow domain experts to refine the knowledge-base. In particular, we focus on our preliminary work to streamline manual refinement of knowledge learned from one type of example data. We show how, the editor can be used to help experts to quickly identify and resolve errors and ambiguity in the LOKI knowledge-base. Concretely, the contributions of this paper include: (1) We introduce LOKI in Section 2 and detail the structure of its knowledge- base in Section 3. (2) We illustrate how the LOKI editor pre-populates the knowledge-base by learning from example data in Section 4. (3)We identify specific errors that arise in the training process and showhowthe LOKI editor facilitates efficient detection and manual repair of the error in Section 5.

Documentation would consist of storing the following information about the dataset:

- Context: Domain to which the dataset belongs.
- Concept: Noun that describes a column in the dataset.
- Unit: Mathematical unit associated with the column. e.g. Kilogram, Hectares, etc.
- Name: String name associated with the column. e.g. A column containing patients blood pressure might be named BP.
- Signature: Parameters used to describe a column. It is a combination of context, Range, Distribution, set of values and the type of data.
- Data type: The type of data in the dataset whether it is numerical, categorical, datetime etc.
- Data distribution: The distribution that best describes a column. e.g Normal, uniform, zipfian etc.
- Context: Domain to which the column belongs.
- Concept: Noun that describes a column.
- Unit: Mathematical unit associated with the column. e.g. Kilogram, Hectares, etc.
- Name: String name associated with the column. e.g. A column containing patients blood pressure might be named BP.
- Signature: Parameters used to describe a column. It is a combination of context, Range, Distribution, set of values and the type of data.
- Column:
- Data type:
- Data distribution:

2.1 Challenges

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

Poonam:rewrite, taken from HILDA paper

Data distributions have been used for similar purposes in other work. For example GestureQuery [4] uses data similarity between two attributes to select candidate attributes for an equi-join. To maximize the join arity, the system counts the number of times each value from one attribute appears in the other and a histogram is constructed from the counts for all of the values.

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. EXPERIMENTS

Poonam:rewrite, taken from HILDA paper

Describe how KB was created for datasets. System design from HILDA paper

5. RESEARCH PLAN

We plan several extensions as future work 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.

Poonam:for reference

To overcome the assumptions and limitations of the current framework we propose the following research plan: (1) We want to generalize candidate questions to include non-boolean atoms. In this way, same incomplete data in different candidate questions can be confirmed in one confirmation round. (2) The current framework assumes the probability distributions for incomplete data are given. However, this is not always feasible. We want to make use of the existing reliable data to estimate the probability distribution of incomplete data. (3) In many domains, there exist models that can be utilized to increase query result accuracy without consulting users. Currently, in the UDLM component, we mainly utilized the user-defined models to preprocess incomplete data at no cost. We want to explore how to learn models from system-user interactions. For instance, in data integration, possible schema matches are candidate models, one of which is a correct match. We can treat each candidate model as probabilistically related to the real model. We can integrate these candidate models with our current framework and adjust the confidence of a model on confirmation. The probability will converge on true model. This learned model will be utilized adaptively in subsequent queries. (4) Our current framework considers user feedback is always correct. However, humans may make mistakes. In crowdsourcing, humans may provide incorrect answers to tasks accidentally or on purpose. In medical expert systems, user feedback is obtained based on test results by machines. The test results may not be 100a user profile to record the credibility of user feedback. We need to decide how to integrate this profile with our frame work and based on the correctness of the decisions made

6. CONCLUSION

Poonam:for reference

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.

Mention Query model?

7. 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.

8. 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.