CEUR-WS.org/Vol-2324/Paper00-InvTalk2-NPaton.pdf

Automating Data Preparation: Can We? Should We? Must We?

Norman W. Paton
School of Computer Science
University of Manchester
Manchester, UK
norman.paton@manchester.ac.uk

ABSTRACT

Obtaining value from data through analysis often requires significant prior effort on data preparation. Data preparation covers the discovery, selection, integration and cleaning of existing data sets into a form that is suitable for analysis. Data preparation, also known as data wrangling or extract transform load, is reported as taking 80% of the time of data scientists. How can this time be reduced? Can it be reduced by automation? There have been significant results on the automation of individual steps within the data wrangling process, and there are now a few proposals for end-to-end automation. This paper reviews the state-of-the-art, and asks the following questions: Can we automate data preparation – what techniques are already available? Should we – what data preparation activities seem likely to be able to be carried out better by software than by human experts? Must we – what data preparation challenges cannot realistically be carried out by manual approaches?

1 INTRODUCTION

It is widely reported that data scientists are spending around 80% of their time on data preparation[1, 2]. Likely data scientists can't realistically expect to spend 0% of their time on data preparation, but 80% seems unnecessarily high. Why is this figure so high? It isn't because there are no products to support data preparation; the data preparation tools market is reported to be worth $2.9B and growing rapidly [24].

It seems likely that data preparation is expensive because it is still in significant measure a programming task: data scientists either write data wrangling programs directly (e.g., [19]), use visual programming interfaces [18] to develop transformation scripts, or write workflows [35] to combine data preparation operations. This leads to a significant amount of work, as the activities facing the data scientist are likely to include the following:

- Data discovery: the identification of potentially relevant data sources, such as those that are similar to or join with a given target.
- Data extraction: obtaining usable data sets from challenging and heterogeneous types of source, such as the deep web.
- Data profiling: understanding the basic properties of individual data sets (such as keys) and the relationships between data sets (such as inclusion dependencies).
- Format transformation: resolving inconsistencies in value representations, for example for dates and names.
- Source selection: choosing the data sets that are actually suitable for the problem at hand, in terms of relevance, coverage and quality.
- Matching: identifying which properties of different sources may contain the same type of information.
- Mapping: the development of transformation programs that combine or reorganize data sources to remove structural inconsistencies.
- Data repair: the removal of constraint violations, for example between a zip code and a street name.
- Duplicate detection: the identification of duplicate entries for the same real world object within individual data sets or in the results of mappings.
- Data fusion: the selection of the data from identified duplicates for use in later steps of the process.

This is an intimidating collection of stages to approach manually. Quite a few of these are individually challenging, involving tasks such as the authoring of mappings or format transformation rules, and the setting of thresholds and parameters (e.g., for matching and duplicate detection). Current data preparation systems provide frameworks within which such tasks can be carried out, but typically the data scientist or engineer remains responsible for making many fine grained decisions.

Any data preparation solution in which the data scientist retains fine grained control over each of the many steps that may be involved in data preparation is likely to be expensive. As a result, the hypothesis of this paper is that automation of these steps, and indeed of complete data preparation processes, should be a priority. This could be expressed in the following principle for data preparation systems:

> Data preparation systems should involve the description of what is required, and not the specification of how it should be obtained.

To realise this principle in practice, all the steps described above need to be automated, informed by the description of what is required. Providing a description of what is required involves the data scientist in understanding the problem to be solved, but this is what the data scientist should be focusing on[3].

In this setting, we assume that it would be desirable for a system to take more responsibility for data preparation tasks than is currently the case, if there can be confidence that the system will perform as well as a human expert. As a result in this paper, we explore three questions:

- *Can we automate?* There has been previous work on automating individual data preparation steps. Section 2 reviews such work, and the information that is used to inform automation.

[1] https://www.nytimes.com/2014/08/18/technology/for-big-data-scientists-hurdle-to-insights-is-janitor-work.html

[2] https://www.forbes.com/sites/gilpress/2016/03/23/data-preparation-most-time-consuming-least-enjoyable-data-science-task-survey-says/#6d86d9256f63

[3] We note that there are also likely to be further data preparation steps that are more closely tied to the analysis to be undertaken [29]. Such steps are also both important and potentially time consuming, and are likely to benefit from some level of automation, but are beyond the scope of this paper.

© 2019 Copyright held by the author(s). Published in the Workshop Proceedings of the EDBT/ICDT 2019 Joint Conference (March 26, 2019, Lisbon, Portugal) on CEUR-WS.org.

Stage	What is Automated	What Evidence is Used	Citation
Data discovery	The search for unionable data sets	A populated target table	[26]
Data extraction	The creation of extraction rules	Training data, feedback	[11]
Data profiling	Annotations on dependencies, keys, ...	The source data	[27]
Format transformation	The learning of transformations	Training examples	[15]
Source selection	Identification of the sources matching user criteria	Source criteria	[2]
Matching	Identification of related attributes	Schema and instances	[3]
Mapping	The search for queries that can populate a target	Source schema, target examples	[28]
Data repair	The correction of constraint violations	Master data	[13]
Duplicate detection	Setting parameters, comparison functions	Training data, feedback	[25]
Data fusion	Selection of values across sources	Other data sources	[12]

Table 1: Proposals for automating data preparation steps.

- *Should we automate?* There are likely data preparation steps for which automation can be expected to out-perform a human expert, and *vice versa*. Section 3 discusses which steps seem most amenable to automation, and the situations in which human experts are likely to be more difficult to replace.
- *Must we automate?*There are likely settings in which either the scale of the data preparation task or the resources available preclude a labour intensive approach. Section 4 considers where automation may be an imperative, i.e., where a lack of automation means that an investigation cannot be carried out.

2 CAN WE?

What might an automated approach to data preparation look like? In our principle for data preparation automation in Section 1, it is proposed that the user should provide a *description of what is required*. This means that it can be assumed that the user is familiar with the application in which data preparation is to take place, and can provide information about it.

What sort of information may be required by an automated system? Table 1 lists proposals for automating each of the data preparation steps listed in Section 1, and for each of these steps other proposals exist that either use different techniques to underpin the automation or different evidence to inform the automation.

It cannot be assumed that the automation of data preparation can take place without some supplementary evidence to inform the step. In Table 1, the *What Evidence is Used* column outlines what data is used by each proposal to inform the decisions it makes. In some cases, rather little supplementary data is required. For example *Data Profiling* acts on source data directly, and *Matching* builds on source data and metadata.

However, in general, automation stands to benefit from additional evidence. As an example, for *Format Transformation*, the cited method infers format transformation programs from examples. For example, if we have the source names *Robert Allen Zimmerman* and *Farrokh Bulsara*[4], and the target names *R. Zimmerman* and *F. Bulsara*, a format transformation program can be synthesized for reformatting extended names into abbreviated names consisting of an initial, a dot, and the surname [15]. So, to save the data scientist from writing format transformation programs, instead the program is synthesized. This reduces the programming burden, but still requires that examples are provided, which may be difficult and time consuming. However, there are also proposals for discovering the examples, for example making use of web tables [1] or instance data (such as master data) for a target representation [8].

An important feature of automated approaches is that they can often generate (large numbers of) alternative proposals. For example, a *mapping generation* process is likely to produce multiple candidate mappings, and a *duplicate detection* program can likely generate alternative comparison rules and thresholds. As a result, a popular approach is to generate a solution automatically, and then request feedback on the results. This feedback can then be used to refine individual steps within the automated data preparation process (e.g., for mapping generation [5, 9, 34] or for duplicate detection [14, 25]) or to influence the behaviour of the complete data preparation pipeline (e.g., [22]). The provision of feedback involves some effort from the user, but builds on knowledge of the domain, and does not require the user to take fine grained control over how data preparation is being carried out.

Up to this point, the focus has been on automation of individual steps within the data preparation process; most results to date have involved individual steps, but there are now a few more end-to-end proposals. In Data Tamer [33][5], a learning-based approach is taken to instance-level data integration, in particular focusing on aligning schemas through matching, and bringing together the data about application concepts through duplicate detection and data fusion. In Data Tamer, the approach is semi-automatic, in that the automatically produced results of different steps are reviewed by users, so the principal forms of evidence deployed are feedback and training data. In VADA [21], all of format transformation, source selection, matching, mapping and data repair are informed by evidence in the form of the *data context* [20], instance values that are aligned with a subset of the target schema (e.g., master data or example values). Furthermore, feedback on the automatically produced results can be used to revisit several of the steps within the automated process [22]. These proposals both satisfy our principle that the user should provide information about the domain of application, and not about how to wrangle the data.

3 SHOULD WE?

In considering when or whether to automate data preparation steps, it seems important to understand the consequences for automation on the quality of the result, both for individual steps and for complete data preparation processes. In enterprise data integration, for example for populating data warehouses using

[4] The birth names of Bob Dylan and Freddie Mercury, in case you are wondering.

[5] Commercialised as Tamr: https://www.tamr.com/

ETL tools, the standard practice is for data engineers to craft well understood ETL steps, and to work on these steps and their dependencies until there is high confidence that the result is of good quality. It is then expected that analyses over the data warehouse will provide dependable results. This high-cost, high-quality setting is both important and well established, and may represent a class of application for which expert authoring of ETL processes will continue to be appropriate. In such settings, the warehouse is primarily populated using data from inside the organisation, typically from a moderate number of stable and well understood transactional databases, to support management reporting. However, there are other important settings for data preparation and analytics; for example, any analysis over a data lake is likely faced with numerous, highly heterogeneous and rapidly changing data sources, of variable quality and relevance, for which a labour-intensive approach is less practical. However, such data lakes provide new opportunities, for example for analysing external and internal data sets together [23]. In such a setting, an important question is: what are the implications for the quality of the result from the use of automated techniques?

It seems that there have been few studies on the effectiveness of automated techniques in direct comparison with manual approaches, but there are a few specific studies:

- **Format Transformation:** Bartoli *et al.* [4] have developed techniques for generating regular expressions from training examples, for extracting data such as URLs or dates from documents. In a study comparing the technique with human users, it was found that the generated regular expressions were broadly as effective (in terms of F-measure) as the most experienced group of humans, while taking significantly less time. There is also a usability study on semi-automatic approaches for format transformation [16], in which the system (Wrangler) suggests transformations to users. In this study, the users made rather sparing use of the system-generated transformations, and completion times were similar with and without the suggested transformations. This study at least calls into question the effectiveness of a semi-automated approach.
- **Mapping generation:** Qian *et al.* [30] have developed a system for generating schema mappings from example values in the target schema. An experimental evaluation found that mapping construction was substantially quicker when based on examples, than when using a traditional mapping development system, in which the user curates matches and is provided with generated mappings to refine [7]. This study at least suggests that the provision of instance data to inform automated data preparation may be a practical option.

Overall, the evidence on the quality of the results of automated data preparation in direct comparison with manual approaches seems to be quite hard to come by in the literature, and further studies would be valuable. However, research papers on automated techniques often report empirical evaluations of their absolute performance and/or performance against a computational baseline, which provides evidence that such techniques can provide respectable results. Furthermore, there are also empirical evaluations of the impact of feedback on results; these show significant variety. In some problems substantial improvements are observed with modest amounts of feedback (e.g., [25]) and in some cases more substantial samples are required (e.g., [31]). The amount of feedback required for refining a solution partly depends on the role it is playing, and it seems important to the cost-effectiveness of feedback collection for the same feedback to be used for more than one task [22]. We note that some feedback-based proposals obtain feedback on the final data product (e.g., [6, 9, 11, 34]), but that in some other proposals, the feedback is more tightly coupled to a single step in the data integration process (e.g., for entity resolution [25, 36]) or to the specific method being used to generate a solution (e.g., for matching [17] or mapping generation [10]).

Should automation be focused on individual steps or on the end-to-end data preparation process? Likely this depends on the task and environment at hand. Where data preparation involves programming, data engineers have complete control over how the data is manipulated, and thus bespoke processing and complex transformations are possible. End-to-end automation will not be able to provide the same levels of customization as are available to programmers. As a result, there is certainly scope for automating certain steps within an otherwise manual process, although the potential cost savings, and synergies between automated steps, will not be as substantial as with end-to-end automation. Furthermore, we note that avoiding programming is a common requirement in self-service data preparation [32].

4 MUST WE?

Are there circumstances in which the only option is to automate? It seems that automation *must* be used if the alternative is to leave the task undone; in such situations, a best-effort automated approach creates opportunities for obtaining value from data that would otherwise be missed. Here are two situations where automation seems to be the only option:

- *The task presents challenges that are punishing for manual approaches.* The big data movement is associated with the production of ever larger numbers of data sources, from which value can potentially be achieved by bringing the data together in new ways. The *Variety*, *Veracity* and *Velocity* features of big data mitigate against the use of manual data preparation processes, where specific cleaning and integration steps may need to be developed for each new format of data set. It seems likely that manually produced data preparation tasks will always lag behind the available data. In particular, the growth of open data sets and the development of data lakes present opportunities for exploratory analyses that require flexible and rapid data preparation, even if the results may not be as carefully curated as a human expert could produce given sufficient time.
- *The resources are not available to enable a thorough, more manual approach.* The knowledge economy doesn't only consist of large enterprises; e.g., as noted in the UK government's Information Economy Strategy[6], *the overwhelming majority of information economy businesses – 95% of the 120,000 enterprises in the sector – employ fewer than 10 people.* As a result, many small and medium sized enterprises are active in data science, but cannot employ large teams or have large budgets for data preparation. For example, an e-Commerce start-up that seeks to compare its prices with those of competitors, or a local house builder that is trying to understand pricing trends in a region, may need to carry out analyses over a collection of data sets, but may not employ a team of data scientists.

[6] https://www.gov.uk/government/publications/information-economy-strategy

What about the individual steps within data preparation, from Table 1? Are there cases in which an automated approach seems the most likely to succeed? The following seem like cases where it may be difficult to produce good results without automation:

- *Matching:* Identifying the relationships between the attributes in n of sources involves n^2 comparisons; even manually curating the results of such automated comparisons is a significant task.
- *Mapping:* Exploring how data sets can be combined potentially involves considering all permutations; again, any manual exploration of how data sets can be combined for large numbers of sources seems likely to miss potentially useful solutions.
- *Entity Resolution:* Entity resolution strategies need to configure significant numbers of parameters (typically in all of blocking, pairwise comparison and clustering), as well as defining a comparison function; this is a difficult, multi-dimensional search space for a human to navigate.

These challenges at the level of individual steps are compounded when considering a pipeline of operations; we have the experience that the best results come when parameter setting across multiple steps is coordinated [25]. Again, manual multi-component tuning is likely to be difficult in practice.

5 CONCLUSIONS

This paper has discussed the hypothesis that data preparation should be automated, with the many components being configured for specific sets of sources on the basis of information about the target. Three questions have been considered:

Can we? There are significant results on the automation of individual steps, and several proposals for end-to-end automation, where the steps are informed by data about the intended outcome of the process, typically in the form of training data or examples. For the future, further work on each of the steps, for example to use different sorts of evidence about the target, should increase the applicability of automated methods. Early work on automating end-to-end data preparation seems promising, but there is likely much more to do.

Should we? A case can be made that automating many of the steps should be able to produce results that are at least as good as a human expert should manage, especially for large applications. There is a need for more systematic evaluation of automated techniques in comparison with human experts, to identify when automation can already be trusted to identify solutions that compete with those of experts, and those in which the automated technique or the evidence used can usefully be revisited.

Must we? There will be tasks that are out of reach for manual approaches. These may not only be the large and challenging tasks; if your budget is x and the cost of manual data preparation is $2x$, then the task is out of reach. As in many cases the available budget may be severely constrained, there is likely to be a market for automated techniques in small to medium sized organisations, where at the moment more manual approaches are rather partial (e.g. investigating only a small subset of the available data). In addition, with the data lakes market predicted to grow at a 28% compound annual growth rate to $28B by 2023[7], efficient techniques for exploratory analyses over data lakes are likely to be in growing demand.

Acknowledgement: Research into Data Preparation at Manchester is funded by the UK Engineering and Physical Sciences Research Council (EPSRC) through the VADA Programme Grant.

REFERENCES

[1] Ziawasch Abedjan, John Morcos, Ihab F. Ilyas, Mourad Ouzzani, Paolo Papotti, and Michael Stonebraker. 2016. DataXFormer: A robust transformation discovery system. In *32nd IEEE International Conference on Data Engineering, ICDE*. 1134–1145. https://doi.org/10.1109/ICDE.2016.7498319

[2] Edward Abel, John Keane, Norman W. Paton, Alvaro A.A. Fernandes, Martin Koehler, Nikolaos Konstantinou, Julio Cesar Cortes Rios, Nurzety A. Azuan, and Suzanne M. Embury. 2018. User driven multi-criteria source selection. *Information Sciences* 430-431 (2018), 179–199. https://doi.org/10.1016/j.ins.2017.11.019

[3] David Aumueller, Hong Hai Do, Sabine Massmann, and Erhard Rahm. 2005. Schema and ontology matching with COMA++. In *Proceedings of the ACM SIGMOD International Conference on Management of Data, Baltimore, Maryland, USA, June 14-16, 2005*. 906–908. https://doi.org/10.1145/1066157.1066283

[4] Alberto Bartoli, Andrea De Lorenzo, Eric Medvet, and Fabiano Tarlao. 2016. Inference of Regular Expressions for Text Extraction from Examples. *IEEE Trans. Knowl. Data Eng.* 28, 5 (2016), 1217–1230. https://doi.org/10.1109/TKDE.2016.2515587

[5] Khalid Belhajjame, Norman W. Paton, Suzanne M. Embury, Alvaro A. A. Fernandes, and Cornelia Hedeler. 2010. Feedback-based annotation, selection and refinement of schema mappings for dataspaces. In *EDBT*. 573–584. https://doi.org/10.1145/1739041.1739110

[6] Khalid Belhajjame, Norman W. Paton, Suzanne M. Embury, Alvaro A. A. Fernandes, and Cornelia Hedeler. 2013. Incrementally improving dataspaces based on user feedback. *Inf. Syst.* 38, 5 (2013), 656–687. https://doi.org/10.1016/j.is.2013.01.006

[7] Philip A. Bernstein and Laura M. Haas. 2008. Information integration in the enterprise. *CACM* 51, 9 (2008), 72–79. https://doi.org/10.1145/1378727.1378745

[8] Alex Bogatu, Norman W. Paton, and Alvaro A. A. Fernandes. 2017. Towards Automatic Data Format Transformations: Data Wrangling at Scale. In *Data Analytics - 31st British International Conference on Databases, BICOD*. 36–48. https://doi.org/10.1007/978-3-319-60795-5_4

[9] Angela Bonifati, Radu Ciucanu, and Slawek Staworko. 2014. Interactive Inference of Join Queries. In *17th International Conference on Extending Database Technology, EDBT*. 451–462. https://doi.org/10.5441/002/edbt.2014.41

[10] Angela Bonifati, Ugo Comignani, Emmanuel Coquery, and Romuald Thion. 2017. Interactive Mapping Specification with Exemplar Tuples. In *Proceedings of the 2017 ACM International Conference on Management of Data, SIGMOD Conference 2017, Chicago, IL, USA, May 14-19, 2017*. 667–682. https://doi.org/10.1145/3035918.3064028

[11] Valter Crescenzi, Paolo Merialdo, and Disheng Qiu. 2015. Crowdsourcing large scale wrapper inference. *Distributed and Parallel Databases* 33, 1 (2015), 95–122. https://doi.org/10.1007/s10619-014-7163-9

[12] Xin Luna Dong, Evgeniy Gabrilovich, Geremy Heitz, Wilko Horn, Kevin Murphy, Shaohua Sun, and Wei Zhang. 2014. From Data Fusion to Knowledge Fusion. *PVLDB* 7, 10 (2014), 881–892. https://doi.org/10.14778/2732951.2732962

[13] Wenfei Fan and Floris Geerts. 2012. *Foundations of Data Quality Management.* Morgan & Claypool.

[14] Chaitanya Gokhale, Sanjib Das, AnHai Doan, Jeffrey F. Naughton, Narasimhan Rampalli, Jude W. Shavlik, and Xiaojin Zhu. 2014. Corleone: hands-off crowdsourcing for entity matching. In *SIGMOD*. 601–612. https://doi.org/10.1145/2588555.2588576

[15] Sumit Gulwani, William R. Harris, and Rishabh Singh. 2012. Spreadsheet data manipulation using examples. *Commun. ACM* 55, 8 (2012), 97–105. https://doi.org/10.1145/2240236.2240260

[16] Philip J. Guo, Sean Kandel, Joseph M. Hellerstein, and Jeffrey Heer. 2011. Proactive wrangling: mixed-initiative end-user programming of data transformation scripts. In *Proceedings of the 24th Annual ACM Symposium on User Interface Software and Technology, Santa Barbara, CA, USA, October 16-19, 2011*. 65–74. https://doi.org/10.1145/2047196.2047205

[17] Nguyen Quoc Viet Hung, Nguyen Thanh Tam, Zoltán Miklós, Karl Aberer, Avigdor Gal, and Matthias Weidlich. 2014. Pay-as-you-go reconciliation in schema matching networks. In *IEEE 30th International Conference on Data Engineering ICDE*. 220–231. https://doi.org/10.1109/ICDE.2014.6816653

[18] Sean Kandel, Andreas Paepcke, Joseph M. Hellerstein, and Jeffrey Heer. 2011. Wrangler: Interactive Visual Specification of Data Transformation Scripts. In *CHI*. 3363–3372.

[19] Jacqueline Kazil and Katharine Jarmul. 2016. *Data Wrangling in Python.* O'Reilly.

[20] Martin Koehler, Alex Bogatu, Cristina Civili, Nikolaos Konstantinou, Edward Abel, Alvaro A. A. Fernandes, John A. Keane, Leonid Libkin, and Norman W. Paton. 2017. Data context informed data wrangling. In *2017 IEEE International Conference on Big Data, BigData 2017*. 956–963. https://doi.org/10.1109/BigData.2017.8258015

[7] https://www.marketresearchfuture.com/reports/data-lakes-market-1601

[21] Nikolaos Konstantinou, Martin Koehler, Edward Abel, Cristina Civili, Bernd Neumayr, Emanuel Sallinger, Alvaro A. A. Fernandes, Georg Gottlob, John A. Keane, Leonid Libkin, and Norman W. Paton. 2017. The VADA Architecture for Cost-Effective Data Wrangling. In *ACM SIGMOD*. 1599–1602. https://doi.org/10.1145/3035918.3058730

[22] Nikolaos Konstantinou and Norman W. Paton. 2019. Feedback Driven Improvement of Data Preparation Pipelines. In *Proc. 21st International Workshop on Design, Optimization, Languages and Analytical Processing of Big Data (DOLAP)*. CEUR.

[23] Jorn Lyseggen. 2017. *Outside Insight Navigating a World Drowning in Data*. Penguin.

[24] Ehtisham Zaidi Mark A. Beyer, Eric Thoo. 2018. *Magic Quadrant for Data Integration Tools*. Technical Report. Gartner. G00340493.

[25] Ruhaila Maskat, Norman W. Paton, and Suzanne M. Embury. 2016. Pay-as-you-go Configuration of Entity Resolution. *T. Large-Scale Data- and Knowledge-Centered Systems* 29 (2016), 40–65. https://doi.org/10.1007/978-3-662-54037-4_2

[26] Fatemeh Nargesian, Erkang Zhu, Ken Q. Pu, and Renée J. Miller. 2018. Table Union Search on Open Data. *PVLDB* 11, 7 (2018), 813–825. https://doi.org/10.14778/3192965.3192973

[27] Thorsten Papenbrock, Tanja Bergmann, Moritz Finke, Jakob Zwiener, and Felix Naumann. 2015. Data Profiling with Metanome. *Proceedings of the VLDB Endowment* 8, 12 (2015), 1860–1863. https://doi.org/10.14778/2824032.2824086

[28] Fotis Psallidas, Bolin Ding, Kaushik Chakrabarti, and Surajit Chaudhuri. 2015. S4: Top-k Spreadsheet-Style Search for Query Discovery. In *Proceedings of the 2015 ACM SIGMOD International Conference on Management of Data, Melbourne, Victoria, Australia, May 31 - June 4, 2015*. 2001–2016. https://doi.org/10.1145/2723372.2749452

[29] Dorian Pyle. 1999. *Data Preparation for Data Mining*. Morgan Kaufmann.

[30] Li Qian, Michael J. Cafarella, and H. V. Jagadish. 2012. Sample-driven schema mapping. In *Proceedings of the ACM SIGMOD International Conference on Management of Data, SIGMOD 2012, Scottsdale, AZ, USA, May 20-24, 2012*. 73–84. https://doi.org/10.1145/2213836.2213846

[31] Julio César Cortés Ríos, Norman W. Paton, Alvaro A. A. Fernandes, and Khalid Belhajjame. 2016. Efficient Feedback Collection for Pay-as-you-go Source Selection. In *Proceedings of the 28th International Conference on Scientific and Statistical Database Management, SSDBM 2016, Budapest, Hungary, July 18-20, 2016*. 1:1–1:12. https://doi.org/10.1145/2949689.2949690

[32] Rita L. Sallam, Paddy Forry, Ehtisham Zaidi, and Shubhangi Vashisth. 2016. *Market Guide for Self-Service Data Preparation*. Technical Report. Gartner.

[33] Michael Stonebraker, Daniel Bruckner, Ihab F. Ilyas, George Beskales, Mitch Cherniack, Stanley B. Zdonik, Alexander Pagan, and Shan Xu. 2013. Data Curation at Scale: The Data Tamer System. In *CIDR 2013, Sixth Biennial Conference on Innovative Data Systems Research, Asilomar, CA, USA, January 6-9, 2013, Online Proceedings*. http://www.cidrdb.org/cidr2013/Papers/CIDR13_Paper28.pdf

[34] Partha Pratim Talukdar, Marie Jacob, Muhammad Salman Mehmood, Koby Crammer, Zachary G. Ives, Fernando C. N. Pereira, and Sudipto Guha. 2008. Learning to create data-integrating queries. *PVLDB* 1, 1 (2008), 785–796. https://doi.org/10.14778/1453856.1453941

[35] Panos Vassiliadis. 2011. A Survey of Extract-Transform-Load Technology. *IJDWM* 5, 3 (2011), 1–27.

[36] Jiannan Wang, Tim Kraska, Michael J. Franklin, and Jianhua Feng. 2012. CrowdER: Crowdsourcing Entity Resolution. *PVLDB* 5, 11 (2012), 1483–1494. https://doi.org/10.14778/2350229.2350263