Tailoring Entity Matching for Industrial Settings

Nils Barlaug
Cognite
Lysaker, Norway
Norwegian University of Science and Technology
Department of Computer Science
Trondheim, Norway
nils.barlaug@ntnu.no

ABSTRACT

Entity matching has received significant attention from the research community over many years. Despite some limited success, most state-of-the-art methods see no widespread usage in industry.

In this paper, we present the author's PhD research, which aims at identifying issues that hold techniques and methods developed by the research community back from use in industry, and look at how they might be adapted to address those issues. In our proposed approach, we implement a modular framework, which will be used for real-world user testing and quantitative experiments of our adapted methods. We will have three main contributions from our research: 1) We develop a modular framework for interactive entity matching combining intra- and inter-session iterations. 2) We show how active learning methods for entity matching can be adapted to learn not only classification of matches but also classification of which records are of interest to the user jointly, and how it compares to current methods. 3) We show how deep learning can be used to synthesize interpretable rules for entity matching, and how it compares to traditional methods.

KEYWORDS

Entity Matching, Entity Resolution, Data Matching, Data Integration

ACM Reference Format:
Nils Barlaug. 2020. Tailoring Entity Matching for Industrial Settings. In *Proceedings of the 29th ACM International Conference on Information and Knowledge Management (CIKM '20), October 19–23, 2020, Virtual Event, Ireland*. ACM, New York, NY, USA, 4 pages. https://doi.org/10.1145/3340531.3418514

1 INTRODUCTION

1.1 Motivation

Digitalization is non-trivial and usually involves complex changes both at a technical and organizational level. This is no different for heavy-asset industry companies (e.g., shipping, utility, oil & gas, manufacturing), which are currently trying to move towards more modern ways of working and cooperating. While part of the challenge is to move data from an analog medium (pen and paper) over to a digital one, a big part of it is to utilize the data that are already available digitally. Industrial companies often have very complex operations, which need contributions from many different parts of the organization with different fields of expertise and workflows. Their data model is then correspondingly complex. Most of this data exist digitally, but often in tens or hundreds of silo systems (e.g., CAD models in one system, some maintenance logs in another, the rest of the maintenance logs in a third, sensor values in a fourth, ...). There is immense value in integrating these data sources, as it enables more efficient workflows across silos and makes information more accessible to those who need it (e.g., finding sensor values for the equipment mentioned in the maintenance logs). Unfortunately, these systems do not always have common identifiers. So one needs to find out which records refer to the same asset. Entity matching is, therefore, a central task in integrating industrial data sources. While entity matching as an academic problem is surprisingly well-studied [1], current solutions still leave much to be desired in terms of using them in practice. Most matching in the industry is still done ad-hoc on a case-by-case basis without really making use of the decades of research on the subject.

1.2 Overall goal

The research goal is to identify why state-of-the-art entity matching techniques and tools are usually not successfully applied in industry and what can be done to address some of these shortcomings — with a special focus on typical scenarios from the heavy-asset industry. The author has access to data, use cases, and domain experts from multiple heavy-asset industry companies.

1.3 Outline

In this paper, we start by defining the entity matching problem and briefly covering the state of the art. Since the initial phase of identifying shortcomings and exploring data is mostly done, we then present the most important findings to provide context for the proposed approach we present next. We continue with our proposed approach, our methodology, and a summary of where we are now before concluding and discussing how to proceed.

2 PROBLEM DEFINITION

Entity matching is the problem of identifying which records from two data sources refer to the same real-world entity. Given two sets A and B with records $a = (a_1, a_2, ..., a_n) \in A$ and $b = (b_1, b_2, ..., b_m) \in B$, the goal is to find the maximum subset $M \subseteq A \times B$ such that a and b refer to the same entity for all $(a, b) \in M$. There are several ways specific instances of the problem might vary. One or both data sources might contain duplicates, meaning multiple records from the same source might refer to the same entity. It might be because

CIKM '20, October 19–23, 2020, Virtual Event, Ireland

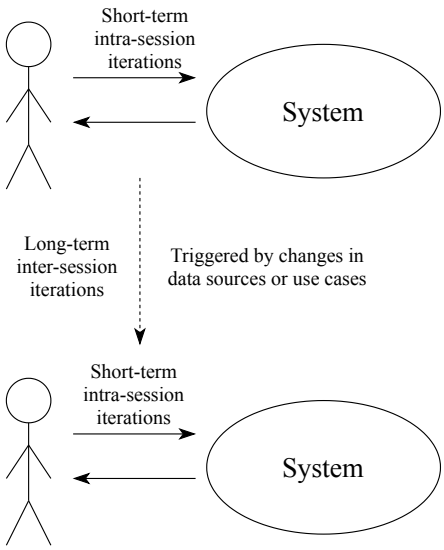

Figure 1: Illustration of entity matching as an iterative process at two levels. The user iterates in real-time within one session, but also iterates across sessions over time as use-cases and requirements change.

the data sources are dirty but also frequently occur because the data sources have a natural cardinality relationship between them (e.g., the same asset can be referenced multiple times in a database of sensors because assets can have multiple sensors). In the special case of $A = B$, one wants to find duplicates within the same data source (i.e., deduplication).

The problem is rarely solved in isolation. There are typically necessary steps both before and after an entity matching process, such as data exploration, pre-processing, and clustering of duplicates. And while we mostly consider these out of scope of our research, it is important to acknowledge their existence and the critical role they play in the overall process.

3 BACKGROUND AND STATE OF THE ART

The research history of entity matching is long, thus many aspects have been investigated in great depth. The problem and slight variations thereof have also been studied under multiple names in different fields, making it challenging to get a complete overview.

Certain prominent sub-tasks and technical challenges have received significant attention. String similarity often plays a central role in entity matching. A multitude of similarity metrics has been developed for different purposes, as well as efficient algorithms for calculating them [1]. Entity matching is generally computationally tricky because the number of possible matches is $O(|A| \times |B|)$. Techniques for reducing the potential number of matches to be evaluated are often referred to by the common term *blocking*, and many effective and efficient techniques have been developed [9]. Even with these techniques, it can still be quite computationally heavy. Therefore, efforts have been made to develop algorithms to scale across multiple cores and machines [2]. Since entity matching problem instances can vary significantly and be hard to tune correctly by hand, researchers have often used machine learning [1]. Recently, deep learning, in particular, has received increasing attention [3, 8, 11]. The community also see the need for interactively querying the user for examples, and so exciting work has been done within active learning [5, 7].

There is also a significant tradition for making end-to-end systems. JedAI [10] is a start-of-the-art, highly configurable system for scalable entity matching which can run on Spark. The user can choose among several workflows and configure each step through both graphical and programming interfaces. Magellan [6] is a popular state-of-the-art ecosystem of entity matching tools for data scientists. Users can try out different blockers and matchers, utilize builtin debugging helpers, and use the provided guides to work through the process. There exist additional packages for doing, for example, deep learning [8]. CloudMatcher [4] is a state-of-the-art entity matching platform for lay users. It supports interactive labeling and crowdsourcing.

4 CHALLENGES FOR INDUSTRIAL USE

As the author has observed from practical work with a wide range of heavy-asset companies, there are at least five typical issues that stop existing techniques and methods from being applied successfully.

- **Off-the-shelf implementations are scarce**: The number of openly accessible, well-documented, high-quality, production-ready libraries and systems is still very low — almost non-existing. This makes it a bigger investment with an increased risk of failure to develop and integrate an entity matching solution into an organization.
- **Domain experts depend on entity matching experts**: Most tools require the user to have specific knowledge about entity matching, which very few have. Other tools require the user to be able to program and otherwise have strong technical skills. When in addition significant domain knowledge is needed to understand the data sources, it can be challenging to scale up any efforts because the intersection of those who are capable of performing entity matching and those who actually have the necessary domain knowledge to understand the data is very small or empty. Domain experts, external consultants, or application developers are often the stakeholders and driving force behind a data integration effort. They become blocked by being dependent on someone with enough entity matching expertise, and should ideally instead be able to service themselves.
- **The iterative nature of the problem is not addressed**: Entity matching as a process in reality often needs to be iterative at two levels. First, the traditional act of performing entity matching typically takes several takes — since the user does not know the exact solution upfront. The user might have to go through the cycle of exploring the data, changing the solution, and checking the results several times before he or she gets it right. Regardless of whether the user is in a labeling flow or a configuration flow. Secondly, entity matching is rarely a one-time job. Data sources might change or matching mistakes discovered, which might necessitate tweaks or updates to the produced solution. Perhaps more importantly, one might not want to do the whole job at once.

Data integration efforts are often use-case driven. Therefore, some subsets of the data are more important than others. One wants to invest some extra resources to get high-quality matches for these subsets, while it is not worth the trouble to go beyond mediocre for the rest. Later, as use-cases change and new ones come along, one needs to raise the quality of other subsets of the data. See Figure 1 for an illustration.
Most existing systems operate within a single, one-time batch job framework. There is no notion of "changing" an already deployed solution, and no way of seeing the effect of the changes made. The user will simply have to start from scratch again.

- **Non-interpretable solutions**: Many machine-learning-based systems rely only on non-interpretable methods or on interpretable methods that, in practice, are hard to understand for users without machine-learning expertise. If matches produced from a system are going to be used for business-, health-, or safety-critical operations, they will have to be verified by someone with the appropriate authority. Moreover, if the model who produced them are not easy to interpret and verify, then each match has to be manually verified. Ideally, the user should be able to verify the model instead of the matches. It is often easier to construct interpretable models for heavy-asset industry data because it has a higher degree of structure and less noise than typical datasets seen in published research.
- **Systems are too rigid**: Too many systems do not offer users enough flexibility to reach their goal. The user might need to apply different methods to different subsets of the data. A black box solution can be acceptable in some cases, while some situations demand interpretable and trusted methods. And for particular tricky corner-cases, it is crucial to have an escape hatch to correct it manually.

Of course, most of these pain points have been addressed by varying degrees by existing work, but few efforts treat them holistically. For example, Magellan [6] offers flexibility and a short-term iterative workflow, but it is not suited for domain experts and lacks long-term iterative workflows. While CloudMatcher [4] can be used by domain experts, but are rigid. Current systems do not meet the constraints and requirements that typical scenarios in heavy-asset industry have, and they are mostly developed towards data with other characteristics (typical examples being publications and products).

5 APPROACH

Our approach is twofold and based on the findings in the section above. First, we implement a framework for flexible and transparent combination of matching strategies, real-time interactivity, and inter-session iterative workflows. In order for a system to be flexible, we view the ability to combine different matchers in a transparent way more modular and likely to be manageable than one big flexible (but monolithically) matcher. It also enables mixing and matching of non-interpretable and interpretable matchers, as well as matchers demanding different levels of technical expertise. We propose the matchers to be combined in prioritized order and each matcher operating on a defined subset of the data. To support intra-session iterations and to be user-friendly to domain experts, we emphasize the need to support real-time interactivity over batch-oriented processing. In addition, the system must have first-class support for inter-session iterative workflows. This means there must be native support changing and tweaking already deployed solutions, and easily see how those changes materialize. To the best of our knowledge, no one has tried to combine these three important aspects holistically in one system. The explicit support for inter-session iterations is the most novel aspect.

Secondly, we will adapt existing work within this framework — with a special focus on active learning approaches and deep learning. In particular, we attempt to adapt active learning methods to not only predict matches but also jointly predict whether a particular record is of interest — helping the user match within the scope of interest without spending unnecessary time on the rest. We base our model on current state-of-the-art deep learning models for active learning [e.g. 7]. The user can choose to further refine the scope or the match precision interchangeably. To our knowledge, jointly classifying records to determine scope at the same as classifying pairs of records as match or not using active learning is a novel approach.

While non-interpretable models are acceptable in some use-cases, others depend on stricter verification. So in addition, we propose to explore the possibilities of using deep learning to synthesize interpretable matching rules. A general disadvantage of deep learning models is their relatively low level of interpretability. One common way of addressing this is to make explainable models, which usually means the network will output not only a prediction but also an explanation of the prediction. Unfortunately, such an explanation has little value in formal verification processes, as there are no guarantees the explanations are true to how matches are actually made. Thus each match will have to manually verify regardless. The solution is to let the model produce interpretable rules. We will mainly explore different options on have to take input, produce output, and training strategy. To our knowledge, having deep learning models produce interpretable entity matching rules have not been done.

Our proposed approach addresses the last four out of five challenges presented in the previous section and will make efforts to reduce them. The last (scarcity of off-the-shelf implementations) is a collective task for the community, but are easier solved when the others are not in the way.

6 METHODOLOGY

Since we are interested in reducing the gap between techniques developed within the research community and real-world use within industry, we want to evaluate in ways that better reflect real-world constraints and requirements. Towards this end, we use two main methodologies. The first being qualitative tests on real-world users with realistic use cases from heavy-asset industry, and the second being more classical quantitative experiments on industrial datasets with relevant evaluation metrics.

The main objective of testing in real-world scenarios with users is to assess the feasibility of the proposed framework and adapted methods in regards to being used in industry. It will reveal issues not easily measured in quantitative experiments. Such as whether

methods targeted towards non-technical users really can be used by users without technical expertise, whether the system is flexible enough, whether the feedback is fast enough that the user does not give up, or whether interpretable solutions really are interpretable by the targeted users. And importantly, whether the iterative workflow meets the requirements of the user's use case. We will provide select real-world users access to our system, and they will be tasked with going through specific, realistic use cases spanning multiple sessions concerning data they are familiar with. We will observe the whole process as well as collecting explicit feedback from users.

Most research within entity matching is done using quantitative experiments on datasets (both real and synthetic). We will do the same on our adapted methods using collected datasets from heavy-asset industry. The most prominent evaluation metric for entity matching is F_1, which is the natural metric to use for the quality of matches. In our case, where the user is interested in some subset of the data, it might be relevant to also measure F_1 for subset and the rest separately. Since we focus on constraints and requirements for industrial use, we will also measure two metrics targeted more specifically towards this. The first one being feedback latency as a metric — i.e., how long before the user finishes his/her actions until feedback on the action is received. For active learning it will be example selection latency. The second is quantitive interpretability. This has to be adapted depending on the method but will be based on Meduri et al. [7].

Our proposed active learning strategy of jointly classifying records of interest while classifying matches will be compared to solving the task separately as well simply ignoring which records are interesting.

7 RESULTS

We are still quite early in the process. However, we have finished the initial exploration phase, where we identify where current state-of-the-art from the research community falls short in being used in heavy-asset industry. The key findings are summarized in Section 4. While a wide range of typical heavy-asset industry companies is used in our exploration, we find it plausible that many of the same issues are relevant for other types of industry.

Recently, we have implemented the first version of our proposed framework system. Already included are a few traditional non-machine-learning-based matchers. Some limited preliminary user testing has been performed and shows promising results. But we see the need for accessible learning-based methods, and are currently in the initial phases of integrating an active learning workflow.

We have also invested some resources in collecting datasets, and we hope to publish some of them in the future when we finalize them.

8 CONCLUSION AND FUTURE WORK

In this paper, we have introduced the author's PhD research, including the proposed approach, methodology, what has been achieved so far. While research on entity matching has developed an extensive suite of specialized techniques and methods, they still see limited industrial use. So far, we have concluded that there are at least five pain points that stop widespread adoption of state-of-the-art methods in heavy-asset industry, which are summarized in Section 4. Furthermore, we have an initial implementation of our framework system that is promising, and we will use it for our future work.

The next steps in the research work will be to implement our proposed active learning approach, and then evaluate with both user tests and quantitative experiments as described above.

We expect our main research contributions to be how intra- and inter-session entity matching iterations can be combined, how active learning for entity matching can be adapted to take into consideration which records are of interest to the user or not, and how deep learning can be utilized to produce interpretable entity matching rules.

ACKNOWLEDGMENTS

Supervised by Professor Jon Atle Gulla. This work is supported by Cognite and The Research Council of Norway under Project 298998.

REFERENCES

[1] Peter Christen. 2012. *Data Matching: Concepts and Techniques for Record Linkage, Entity Resolution, and Duplicate Detection.* Springer-Verlag, Berlin Heidelberg. https://doi.org/10.1007/978-3-642-31164-2

[2] Vassilis Christophides, Vasilis Efthymiou, Themis Palpanas, George Papadakis, and Kostas Stefanidis. 2019. End-to-End Entity Resolution for Big Data: A Survey. *arXiv:1905.06397 [cs]* (Nov. 2019). http://arxiv.org/abs/1905.06397 arXiv: 1905.06397.

[3] Muhammad Ebraheem, Saravanan Thirumuruganathan, Shafiq Joty, Mourad Ouzzani, and Nan Tang. 2018. Distributed representations of tuples for entity resolution. *Proceedings of the VLDB Endowment* 11, 11 (July 2018), 1454–1467. https://doi.org/10.14778/3236187.3236198

[4] Yash Govind, Mingju Sun, Erik Paulson, Palaniappan Nagarajan, Paul Suganthan G. C., AnHai Doan, Youngchoon Park, Glenn M. Fung, Devin Conathan, and Marshall Carter. 2018. Cloudmatcher: a hands-off cloud/crowd service for entity matching. *Proceedings of the VLDB Endowment* 11, 12 (Aug. 2018), 2042–2045. https://doi.org/10.14778/3229863.3236255

[5] Jungo Kasai, Kun Qian, Sairam Gurajada, Yunyao Li, and Lucian Popa. 2019. Low-resource Deep Entity Resolution with Transfer and Active Learning. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, Florence, Italy, 5851–5861. https://doi.org/10.18653/v1/P19-1586

[6] Pradap Konda, Jeff Naughton, Shishir Prasad, Ganesh Krishnan, Rohit Deep, Vijay Raghavendra, Sanjib Das, Paul Suganthan G. C., AnHai Doan, Adel Ardalan, Jeffrey R. Ballard, Han Li, Fatemah Panahi, and Haojun Zhang. 2016. Magellan: toward building entity matching management systems. *Proceedings of the VLDB Endowment* 9, 12 (Aug. 2016), 1197–1208. https://doi.org/10.14778/2994509.2994535

[7] Venkata Vamsikrishna Meduri, Lucian Popa, Prithviraj Sen, and Mohamed Sarwat. 2020. A Comprehensive Benchmark Framework for Active Learning Methods in Entity Matching. In *Proceedings of the 2020 ACM SIGMOD International Conference on Management of Data (SIGMOD '20)*. Association for Computing Machinery, Portland, OR, USA, 1133–1147. https://doi.org/10.1145/3318464.3380597

[8] Sidharth Mudgal, Han Li, Theodoros Rekatsinas, AnHai Doan, Youngchoon Park, Ganesh Krishnan, Rohit Deep, Esteban Arcaute, and Vijay Raghavendra. 2018. Deep Learning for Entity Matching: A Design Space Exploration. In *Proceedings of the 2018 International Conference on Management of Data - SIGMOD '18*. ACM Press, Houston, TX, USA, 19–34. https://doi.org/10.1145/3183713.3196926

[9] George Papadakis, Dimitrios Skoutas, Emmanouil Thanos, and Themis Palpanas. 2019. Blocking and Filtering Techniques for Entity Resolution: A Survey. (May 2019). https://arxiv.org/abs/1905.06167v3

[10] George Papadakis, Leonidas Tsekouras, Emmanouil Thanos, Nikiforos Pittaras, Giovanni Simonini, Dimitrios Skoutas, Paul Isaris, George Giannakopoulos, Themis Palpanas, and Manolis Koubarakis. 2020. JedAI^3 : beyond batch, blocking-based Entity Resolution. https://doi.org/10.5441/002/EDBT.2020.74 Version Number: 1 type: dataset.

[11] Chen Zhao and Yeye He. 2019. Auto-EM: End-to-end Fuzzy Entity-Matching using Pre-trained Deep Models and Transfer Learning. In *The World Wide Web Conference on - WWW '19*. ACM Press, San Francisco, CA, USA, 2413–2424. https://doi.org/10.1145/3308558.3313578