Multipartite Entity Resolution: Motivating a K-Tuple Perspective (Student Abstract)

Adin Aberbach, Mayank Kejriwal, Ke Shen
Information Sciences Institute, University of Southern California
4676 Admiralty Way, Suite 1001
Marina del Rey, CA 90292
adin.aberbach@gmail.com, kejriwal@isi.edu, keshen@isi.edu

Abstract

Entity Resolution (ER) is the problem of algorithmically matching records, mentions, or entries that refer to the same underlying real-world entity. Traditionally, the problem assumes (at most) two datasets, between which records need to be matched. There is considerably less research in ER when $k > 2$ datasets are involved. The evaluation of such multipartite ER (M-ER) is especially complex, since the usual ER metrics assume (whether implicitly or explicitly) $k < 3$. This paper takes the first step towards motivating a k-tuple approach for evaluating M-ER. Using standard algorithms and k-tuple versions of metrics like precision and recall, our preliminary results suggest a significant difference compared to aggregated pairwise evaluation, which would first decompose the M-ER problem into independent bipartite problems and then aggregate their metrics. Hence, M-ER may be more challenging and warrant more novel approaches than current decomposition-based pairwise approaches would suggest.

Introduction

ER research goes back more than 50 years and is surveyed by (Binette and Steorts 2022) and (Brizan and Tansel 2006). In the database community, ER is referred to as record linkage, but the problem spans many research communities, including Knowledge Graphs and the Web. There has been some research on M-ER, with the current state of the art being FAMER (Saeedi, Peukert, and Rahm 2017), a distributed clustering algorithm. However, the vast majority of ER algorithms and benchmarks are for $k \leqslant 2$ datasets.

The standard bipartite ER (B-ER) pipeline consists of both a blocking and a matching step. Given two datasets, each with N records, the naive approach compares each of the N^2 record pairs head on. To avoid quadratic complexity, the blocking step creates a reduced set of candidate pairs in linear or log-linear time. There are standard metrics for blocking that become more complicated for M-ER, but they are the not the focus of this paper. The matching step then outputs probabilistic binary predictions for the candidate set and is evaluated using precision and recall.

Precision and recall are natural metrics for B-ER because a prediction that two records match is always completely true or completely untrue. However, a prediction that more than two records match can be only partially correct. For example, the prediction that (x, y, z) are three matching records will be partially correct given that x matches y but z does not match x or y. Current literature only evaluates M-ER in a pairwise fashion as $\binom{k}{2}$ independent bipartite problems. In the above example, the prediction (x, y, z) would be decomposed into the predictions (x, y), (y, z), and (x, z), which have the binary property that makes precision and recall applicable.

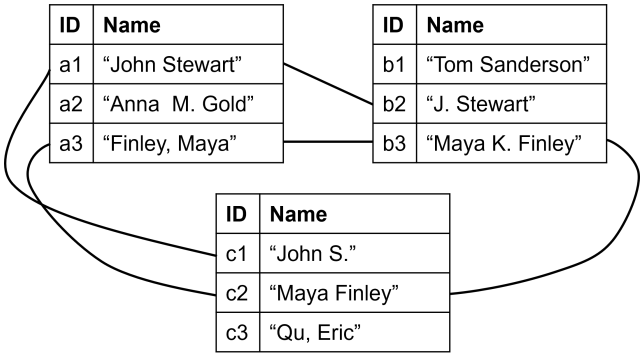

Figure 1: An illustrative toy example of $k = 3$ M-ER. Each table represents one of the three source datasets with connecting lines giving example bipartite match predictions.

However, there has been no in-depth justification for pairwise as the proper metric. One problem with it is that of transitivity. Take for example records a1, b2, c1 in Figure 1. The example predictions give (a1, b2) and (a1, c1) as matches. Say this were true. Then, both b2 and c1 represent the same real-world entity as a1. Therefore both b2 and c1 must represent the same real-world entity as well. However, (b2, c1) is not a predicted match in this example, creating a logical contradiction that should not be allowed. One possible alternative would be clustering metrics. However, most clustering metrics are designed for clusters that grow with the total number of nodes, and M-ER is an example of micro-clustering, where cluster sizes are roughly constant.

In this paper, we propose a k-tuple approach for evaluating M-ER. Given k datasets, a *k-tuple* is of the form $(x_1, ..., x_k)$, where x_i is a record from the ith dataset or ε, indicating that there is no entry from the ith dataset. In our

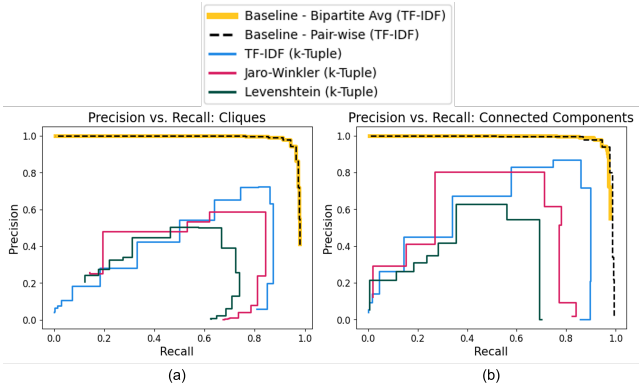

Figure 2: Comparison of precision-recall curves for pairwise and k-tuple M-ER evaluation.

approach, both the ground truth and the predicted matches will be sets of k-tuples. A k-tuple prediction will only be considered correct if the exact same k-tuple is present in the ground truth. Here, predictions are either right or wrong, so the standard definition of precision and recall can be used.

The goal of this paper is not to propose a new method for M-ER. Instead, we demonstrate that the M-ER problem may be more difficult than previously thought and that its evaluation is more complicated than B-ER and needs to be investigated more thoroughly.

Preliminary Experiments

To investigate the difference between k-tuple and pairwise evaluation of M-ER, we set up an experiment using $k = 4$ source datasets. Due to limited existing M-ER data, we generate synthetic corrupted datasets with GeCo (Tran, Vatsalan, and Christen 2013). Each record represents a person and contains a phone number, first name, last name, and SSN.

Our simple M-ER algorithm starts by performing $\binom{k=4}{2}$ independent B-ER tasks to construct a pairwise linked graph. We use MinHash locality sensitive hashing for bipartite blocking and Levenshtein, Jaro-Winkler, or TF-IDF cosine similarity for bipartite matching. Levenshtein and Jaro-Winkler are both ways to estimate the edit distance between two strings. TF-IDF converts strings to vectors using term frequency inverse document frequency and these vectors are then compared using cosine similarity.

To make multipartite predictions, we cluster the pairwise linked graph using either connected components or cliques. For conventional pairwise predictions, P_{pair} contains every inter-dataset record pair where there is a cluster that contains both records. For k-tuple predictions, $P_{k\text{-tuple}}$, we extract all k-tuples with a maximal number of non-ε entries from each cluster. This means that no two k-tuples extracted from one cluster will have a subset/superset relationship.

For the ground truth, we start with the true matching k-tuples denoted $T_{k\text{-tuple}}$. We then decompose each k-tuple match into pairs to get T_{pair}.

For each matching method, we computed three precision-recall curves (Figure 2). "Bipartite Avg" is the average performance from all $\binom{k}{2}$ B-ER tasks. "Pair-wise" gives results for the M-ER task using P_{pair} and T_{pair}. Finally, "k-Tuple" gives results for the M-ER task using $P_{k\text{-tuple}}$ and $T_{k\text{-tuple}}$.

"Bipartite Avg" and "Pair-wise" are both baselines, with "Pair-wise" telling us how the results would be interpreted in the existing M-ER literature. To save space in Figure 2, we only show the baseline curves for the matching method that achieved the highest F-measure, which is TF-IDF for both connected components ("Bipartite Avg": 0.960, "Pair-wise": 0.961) and cliques ("Bipartite Avg": 0.960, "Pair-wise": 0.960).

"k-Tuple" performs well below the baselines for both combination methods. The baselines are essentially solved problems with F-measures greater than 0.95, whereas "k-tuple" has ample room for improvement. Connected components (Figure 2.b) achieves F-measures of (TF-IDF: 0.864, Jaro-Winkler: 0.755, Levenshtein: 0.609). Cliques (Figure 2.a) F-measures are somewhat lower: (TF-IDF: 0.785, Jaro-Winkler: 0.692, Levenshtein: 0.573).

Moreover, "k-Tuple" has a narrower zone for good performance and consistently falls off more steeply and at lower recall. Also, "k-Tuple" precision grows with recall up until the peak F-measure, because at low recalls, the model will predict incomplete k-tuples that will be counted as incorrect.

We have demonstrated that, in some cases, k-tuple evaluation provides quantitatively and qualitatively different results than pairwise evaluation and that optimizing one will not necessarily optimize the other. Yet, problematically, all current literature looks solely at pairwise evaluation without any consideration of whether it is a suitable metric.

We have also demonstrated that when starting a M-ER problem with all $\binom{k}{2}$ bipartite tasks, the end result will vary based on the combination method. This indicates that it might make more sense to perform the entire ER pipeline using k-tuples instead of starting with pairs.

In the future, we will explore how the above results change with more advanced ER algorithms like LLMs, construct new and more flexible k-tuple metrics for M-ER, design algorithms for a k-tuple ER pipeline, and evaluate current SOTA M-ER algorithms using k-tuples.

References

Binette, O.; and Steorts, R. C. 2022. (Almost) all of entity resolution. *Science Advances*, 8(12): eabi8021.

Brizan, D. G.; and Tansel, A. U. 2006. A. survey of entity resolution and record linkage methodologies. *Communications of the IIMA*, 6(3): 5.

Saeedi, A.; Peukert, E.; and Rahm, E. 2017. Comparative evaluation of distributed clustering schemes for multi-source entity resolution. In *Advances in Databases and Information Systems: 21st European Conference, ADBIS 2017, Nicosia, Cyprus, September 24-27, 2017, Proceedings 21*, 278–293. Springer.

Tran, K.-N.; Vatsalan, D.; and Christen, P. 2013. GeCo: an online personal data generator and corruptor. In *Proceedings of the 22nd ACM international conference on Information & Knowledge Management*, 2473–2476.