# FairER demo: Fairness-Aware and Explainable Entity Resolution

Nikolaos Fanourakis[1,2,†], Christos Kontousias[2,†], Vasilis Efthymiou[1], Vassilis Christophides[3] and Dimitris Plexousakis[1]

[1]FORTH-ICS, Heraklion, Greece
[2]University of Crete, Heraklion, Greece
[3]Lab. ETIS, CY Cergy Paris University, ENSEA, CNRS UMR 8051, France

**Abstract**
Entity resolution (ER) is the problem of identifying references to the same real-world objects, in disparate data sources. It has been recently shown that ER results are prone to bias, related to both factual and structural characteristics of the input data. In this work, we demonstrate an extended version of FairER, an open-source ER framework, that receives either tabular or knowledge graph data as input and produces fair and explainable results. Demonstration scenarios showcase some of its capabilities, while a public demo and video are available.

## 1. Introduction

Entity resolution (ER) is the task of identifying pieces of data that refer to the same real-world entity (e.g., a person, an organization), scattered across disparate data sources and formats, e.g., in tables or in knowledge graphs (KGs). Recent works have shown that the results of ER may be biased against some entity groups, based on some of their sensitive attributes [1, 2, 3] (e.g., gender, ethnicity), or even their structural representation in a KG [4] (e.g., central vs long-tail entities). In order to mitigate various forms of direct and indirect bias [4], we need to understand first how ER methods decide which entity pairs match [5].

In this demo, we introduce a unified platform that combines and extends several works for ER, namely FairER [2], extended to work not only on tabular, but also on KG data [6], with a sampling step, SUSIE [4], that deals with various structural bias settings in KG embedding-based ER algorithms [6]. Specifically, we are interested in statistical parity as the target fairness measure [2] and in a connectivity-based structural bias definition [4].

**Comparison with existing works.** A few end-to-end ER frameworks exist, with the most notable being Magellan [7] (now commercialized and closed-source), Dedupe[1], and JedAI [8]. Compared to existing works, this is the first end-to-end ER framework to incorporate fairness (both factual and structural) by design, offering a parameterized sampling for structural bias,

*ISWC 2023 Posters and Demos: 22nd International Semantic Web Conference, November 6–10, 2023, Athens, Greece*
†These authors contributed equally.
fanourakis@ics.forth.gr (N. Fanourakis); csd3965@csd.uoc.gr (C. Kontousias); vefthym@ics.forth.gr (V. Efthymiou); Vassilis.Christophides@ensea.fr (V. Christophides); dp@ics.forth.gr (D. Plexousakis)

[1]Forest Gregg and Derek Eder. 2022. Dedupe. https://github.com/dedupeio/dedupe

while also allowing the provision of visual explanations (e.g., from MOJITO [5]) for the matching decisions and a novel, intuitive explanation model for the fairness aspect.

**Contributions.** In summary, the contributions of this work are: (a) The first open-source, unified platform for fair and explainable ER on tabular data and on KGs. (b) A visual explanation method for the matching results, that incorporates both similarity and fairness assessments. (c) All the modules of our platform are easily extendable to include state-of-the-art methods, as well as various definitions of fairness.

The source code of our system[2], along with a running public demo[3] and a video demonstration[4] are available online.

## 2. System Overview

In this section, we provide background information from relevant works [2, 4, 6]. Then, we provide an overview of our system's pipeline, focusing on its most important components.

### 2.1. Background

FairER [2] defines fairness-aware ER as a ranking of entity pairs that maximizes the (cumulative) match likelihood score in its top-$k$ ranks, while satisfying a given (group) fairness condition $F$, such as statistical parity. That fairness condition $F$ can decide if the so-called protected pairs are fairly represented in the ER results, compared to non-protected pairs.

As a simplified example of how FairER works, consider two datasets to be matched, both describing people, with some of them being convicted criminals (which is the protected group in this example). If the fairness condition is statistical parity, requiring equal representation of protected and non-protected group members, then FairER will create two priority queues of candidate matching pairs - one for the convicted criminals and one for the rest of the people - sort them in descending order of match likelihood (see Similarity Scoring in Section 2.2), and interchangeably pick the most likely match from each queue (see Matching in Section 2.2).

Typically, the decision of whether an entity is protected or non-protected is given by the values of a sensitive attribute. However, in a recent work [4] we provided a more general, graph-theoretic definition of structural bias, which can also play the role of fairness condition $F$, based on the observation that KG nodes that belong to large (above a size threshold) connected components are more likely to be correctly matched. The same work [4] also introduces a sampling algorithm, SUSIE, that can take representative samples of both small and big connected components, and evaluate the robustness of ER works on structural bias [9].

### 2.2. System architecture

The high-level system architecture is illustrated in Figure 1 and briefly described next. The modular design of our architecture allows the seamless adaptation of new methods. Therefore, the selection of methods already implemented is just indicative and orthogonal to our framework.

[2]https://github.com/vefthym/fairER
[3]https://isl.ics.forth.gr/fairER/
[4]https://youtu.be/DTrf9sbmCZE

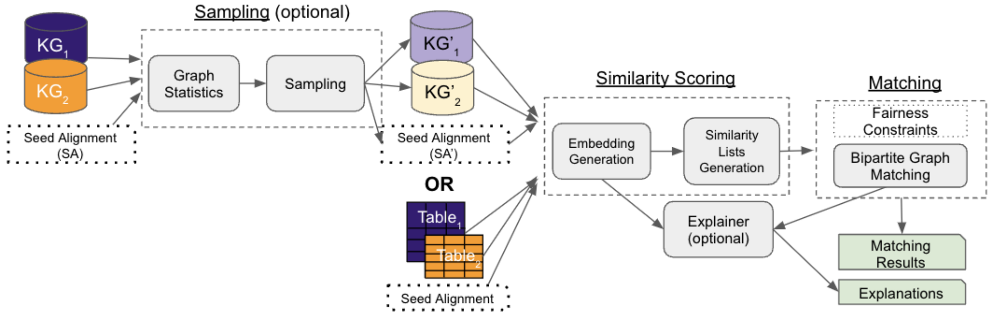



**Figure 1:** High-level system architecture. The input data can be in the form of KGs or tables.

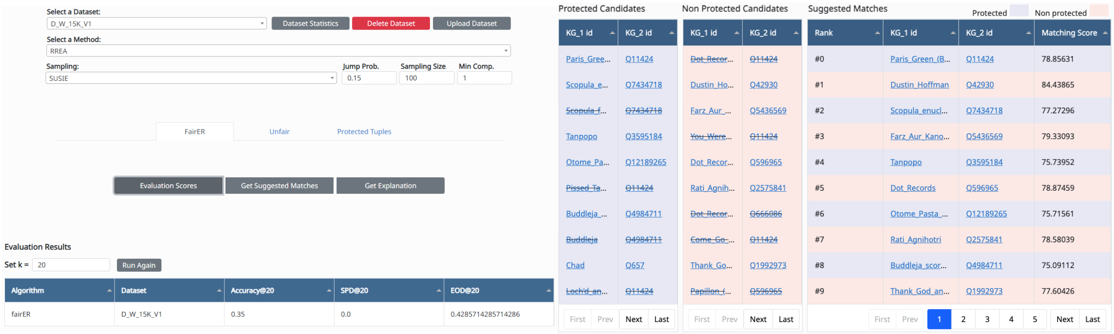
**Sampling.** The optional sampling component is responsible for selecting a subset of the input KGs (not yet applicable to tabular data), with the desired parameters (e.g., sample size, random walk jump probability, size threshold for considering a connected component as small or big). The sampling component follows the implementation of SUSIE [4], but instead of evaluating the robustness of ER methods, we use SUSIE in our framework to provide a fair representation of small connected components in the input data of ER.

**Similarity Scoring.** For assessing the matching scores between entity pairs, i.e., the likelihood that a pair is referring to the same real-world entity, we provide a wide selection of probabilistic and embedding-based matching algorithms for tabular and KG data (e.g., Deepmatcher [7], BERT-INT [10], PARIS [11]). Most embedding-based methods rely on cosine similarity on the entity embeddings that they create using pre-labeled training data (seed alignment). The different ways to create such embeddings is orthogonal to FairER and for knowledge graph embeddings, they are extensively described in our previous work [6].

**Matching.** Matching decisions [12] typically rely only on the similarity scores. However, FairER also considers fairness constraints before returning its results. Our current implementation uses an extension of the Unique Mapping Clustering algorithm [12] for the matching decisions, and statistical parity as the fairness constraint [2]. As mentioned before, FairER extends the Unique Mapping Clustering algorithm to operate on two, instead of one, priority queues, with each representing a different group. We note that FairER can be seamlessly extended to also operate with more than two groups and, consequently, priority queues.

In addition to providing a default criterion for splitting data into protected and non-protected groups for some selected datasets, our platform also allows its users to define their own, custom fairness criteria and test them with actual examples. The matching results can be viewed and downloaded raw, using our API, but the users can also see the evaluation results in terms of accuracy and fairness.

**Explanations.** Our framework offers two types of explanations, depending on the input data format: (a) explanations on the similarity scoring, relying on the deep matching explainability tool MOJITO [5], and (b) explanations on the fairness of the returned matches. The first one assigns scores to schema attributes, identifying the most important ones for match or no match. The latter also includes the similarity scores for both protected and non-protected matches, to give a broader picture (i.e., some matches may be ranked higher than others, even if they have




(a) Dataset selection and evaluation results. (b) Explaining the selection of suggested matches.

**Figure 2:** Screenshots from the framework for different menu options, for KG data.

lower similarity scores, in order to respect the fairness constraints). An example of the latter is shown in Figure 2b and described in Scenario 1 of Section 3.

## 3. Demonstration

Here, we briefly describe two indicative demonstration scenarios, which are also covered in the supplementary material, explaining how the ISWC attendees will interact with our system. In what follows, we will focus more on the case of matching data coming from KGs, but most of what is described next also applies to the tabular data case. One difference is the explanation component, for which we utilize MOJITO [5] in the tabular data case. Some of the features described below are shown in Figure 2.

**Scenario 1 (Walk-through/Author-driven).** An attendee selects one of our pre-loaded tabular or KG datasets and clicks on "Fairness Conditions" to see the default fairness conditions, based on which entities and entity pairs are considered as protected or non-protected. Then, the attendee can run the matching algorithm and see the matching results from FairER, as well as other baseline methods. Finally, the attendee can receive visual explanations of the matching decisions. E.g., in Figure 2b, the user can see how protected and non-protected candidates are ranked separately, as well as how they are ranked in the FairER results to respect the fairness condition, along with their protected/non-protected color-coding and matching score.

**Scenario 2 (Usage/Attendee-driven).** An ISWC attendee uploads one new dataset (that they bring, or we provide) and after viewing a random entity from each of the two data sources, enter their desired protected conditions, for both individual entities and entity pairs (currently supporting conjunctive and disjunctive conditions for pairs). They can even check the condition for a single entity or a pair of entities that they manually type (which may not even exist in the dataset), or load such an entity (pair) from a file. The attendee then clicks on "Dataset Statistics" to check the number of protected and non-protected pairs, the average similarity scores of protected vs non-protected matches (and similarly for non-matches). The features described in Scenario 1 are also applicable for a user-provided dataset.

## 4. Conclusion

We have presented the first unified framework for end-to-end entity resolution (ER) that is fair and explainable by design. This framework builds upon and largely extends works for direct (FairER [2]) and indirect (SUSIE [4]) bias, while it reuses works on explainable ER (MOJITO [5]). We plan to extend the implementation with more definitions of fairness, as well as design a new explainability method that interprets not only the similarity-based matching decision, but also the fairness aspect.

## Acknowledgments

This work has received funding from the Hellenic Foundation for Research and Innovation (HFRI) and the General Secretariat for Research and Technology (GSRT), under GA No 969.

## References

[1] N. Shahbazi, N. Danevski, F. Nargesian, A. Asudeh, D. Srivastava, Through the fairness lens: Experimental analysis and evaluation of entity matching, PVLDB 16 (2023).

[2] V. Efthymiou, K. Stefanidis, E. Pitoura, V. Christophides, FairER: Entity resolution with fairness constraints, in: CIKM, 2021.

[3] A. Karakasidis, E. Pitoura, Identifying bias in name matching tasks, in: EDBT, 2019.

[4] N. Fanourakis, V. Efthymiou, V. Christophides, D. Kotzinos, E. Pitoura, K. Stefanidis, Structural bias in knowledge graphs for the entity alignment task, in: ESWC, 2023.

[5] V. D. Cicco, D. Firmani, N. Koudas, P. Merialdo, D. Srivastava, Interpreting deep learning models for entity resolution: an experience report using LIME, in: aiDM@SIGMOD, 2019.

[6] N. Fanourakis, V. Efthymiou, D. Kotzinos, V. Christophides, Knowledge graph embedding methods for entity alignment: experimental review, Data Min. Knowl. Discov. 37 (2023) 2070–2137.

[7] A. Doan, P. Konda, P. S. G. C., Y. Govind, D. Paulsen, K. Chandrasekhar, P. Martinkus, M. Christie, Magellan: toward building ecosystems of entity matching solutions, Commun. ACM 63 (2020) 83–91.

[8] G. Papadakis, L. Tsekouras, E. Thanos, G. Giannakopoulos, T. Palpanas, M. Koubarakis, The return of jedai: End-to-end entity resolution for structured and semi-structured data, Proc. VLDB Endow. 11 (2018) 1950–1953.

[9] Z. Sun, Q. Zhang, W. Hu, C. Wang, M. Chen, F. Akrami, C. Li, A benchmarking study of embedding-based entity alignment for knowledge graphs, Proc. VLDB Endow. 13 (2020) 2326–2340.

[10] X. Tang, J. Zhang, B. Chen, Y. Yang, H. Chen, C. Li, BERT-INT: A bert-based interaction model for knowledge graph alignment, in: IJCAI, 2020.

[11] F. M. Suchanek, S. Abiteboul, P. Senellart, PARIS: probabilistic alignment of relations, instances, and schema, Proc. VLDB Endow. 5 (2011) 157–168.

[12] G. Papadakis, V. Efthymiou, E. Thanos, O. Hassanzadeh, P. Christen, An analysis of one-to-one matching algorithms for entity resolution, VLDB J. (2023).