FairER: Entity Resolution With Fairness Constraints

Vasilis Efthymiou
vefthym@ics.forth.gr
FORTH-ICS
Greece

Kostas Stefanidis
konstantinos.stefanidis@tuni.fi
Tampere University
Finland

Evaggelia Pitoura
pitoura@cs.uoi.gr
University of Ioannina
Greece

Vassilis Christophides
vassilis.christophides@ensea.fr
ENSEA, ETIS
Cergy, France

ABSTRACT

There is an urgent call to detect and prevent "biased data" at the earliest possible stage of the data pipelines used to build automated decision-making systems. In this paper, we are focusing on controlling the data bias in entity resolution (ER) tasks aiming to discover and unify records/descriptions from different data sources that refer to the same real-world entity. We formally define the ER problem with fairness constraints ensuring that all groups of entities have similar chances to be resolved. Then, we introduce FairER, a greedy algorithm for solving this problem for fairness criteria based on equal matching decisions. Our experiments show that FairER achieves similar or higher accuracy against two baseline methods over 7 datasets, while guaranteeing minimal bias.

CCS CONCEPTS

• **Information systems → Entity resolution**.

KEYWORDS

entity resolution, algorithmic fairness

ACM Reference Format:
Vasilis Efthymiou, Kostas Stefanidis, Evaggelia Pitoura, and Vassilis Christophides. 2021. FairER: Entity Resolution With Fairness Constraints. In *Proceedings of the 30th ACM International Conference on Information and Knowledge Management (CIKM '21), November 1–5, 2021, Virtual Event, QLD, Australia.* ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/3459637.3482105

1 INTRODUCTION

Given the widespread adoption of data-driven systems for making decisions in real life, there is an urgent call for responsible data management [28]. As a matter of fact, downstream harms to particular individuals or groups, often blamed to "biased data"[1], should be detected and prevented at the earliest possible stage of data pipelines used to build automated decision-making systems [13].

In this paper, we focus on controlling the data bias in entity resolution (ER) tasks aiming to discover and unify records/descriptions from different data sources that refer to the same real-world entity. ER tasks are typically used to improve data quality by reducing data incompleteness (i.e., missing values), redundancy (i.e., duplicate values) or inconsistency (i.e., conflicting values) [7]. Failing to address bias in ER tasks may lead to systematic bias that jeopardize both accuracy and fairness of downstream data analysis [18, 27, 30].

We argue that, in order to mitigate data bias implications, ER systems should not only maximize accuracy, as in traditional ER, but also satisfy fairness constraints among the so-called *protected* and *non-protected* groups of entities in their output. Since several criteria of determining protected groups (e.g., race, gender) and fairness measures (e.g., equal representation of groups, similar error rates among groups) have been proposed in the literature [19], the problem of *fairness-aware ER* is a broad one. We introduce a general constraint-based formulation of the problem and then investigate in depth a specific instance of this formulation.

Recent research has investigated the explainability of entity matching decisions [2, 8, 26], but controlling and mitigating bias in ER tasks has not been studied so far. Previous work has focused exclusively on the assessment of demographic bias in word embeddings exploited in named entity recognition [21] and on string similarity functions in name matching tasks [14]. Fairness-aware ER addresses discrepancies in the size of the groups in input data (i.e., popularity bias), where the majority of resolved entities belongs to a specific group [20]. To the best of our knowledge, FairER is the first ER algorithm that satisfies fairness constraints. In summary, the contributions of this work are:

- We formalize the problem of fairness-aware ER as a submodular optimization problem with cardinality constraints, which is known to be NP-hard. This definition requires that the retrieved pairs of entities should maximize a cumulative similarity score, while ensuring fair matching decisions across different groups of interest.
- We introduce FairER, a greedy algorithm to solve the problem of fairness-aware ER for fairness criteria based on equal matching decisions. We prove that FairER provides a $1 - 1/e$ approximation of the optimal solution.
- We experimentally evaluate FairER against fairness-agnostic ER and fair ranking methods over 7 datasets, and show that it consistently achieves similar (or even higher) accuracy, while guaranteeing minimal bias, with a negligible time overhead.

2 PROBLEM STATEMENT

Next, we define the problem of ER subject to fairness constraints.

Traditional Entity Resolution. *Real-world entities* are usually described in one or more sources by sets of attribute-value pairs. *Entity descriptions* may be structured in a tabular form (i.e., records) when stored in relational databases or semi-structured in a graph format (e.g., subject-predicate-object triples) when stored in knowledge graphs [7]. The objective of an ER algorithm is to discover

[1] An overloaded term encompassing among others societal and historical, representation and aggregation aspects [29].

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.
CIKM '21, November 1–5, 2021, Virtual Event, QLD, Australia
© 2021 Association for Computing Machinery.
ACM ISBN 978-1-4503-8446-9/21/11…$15.00
https://doi.org/10.1145/3459637.3482105

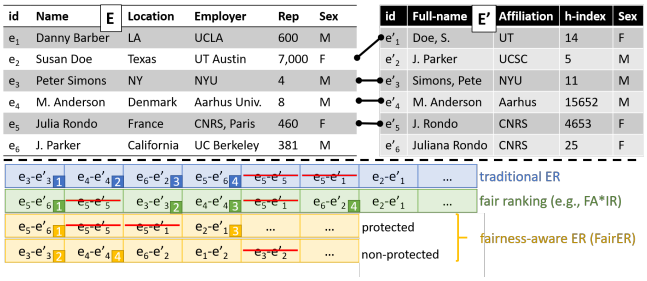

Figure 1: Top: Sources E and E' with matches. Bottom: Fairness-agnostic ER, fair, and FairER rankings (numbered).

pairs of descriptions e, e' that refer to the same real-world entities, called *matches*. For example, in Figure 1 we are interested in forming a team of experts based on some criteria (e.g., reputation and h-index) described in a developers platform E (e.g., Stack Overflow) and a researchers platform E' (e.g., Google Scholar). ER tries to find the matching pairs between E and E', depicted by a connected edge.

Deciding whether a pair of entity descriptions (e, e') refers to the same real-world entity relies on a scoring function s assessing the similarity of the attribute values and names used to describe e and e'. Scores of entity pairs are at the heart of main formulations of the ER problem using heuristic rules (e.g., MinoanER [10]), agglomerative clustering (e.g., SiGMa [17]), or binary classification (e.g., DeepMatcher [23], DeepER [9], and [12]). In the remaining of this paper, we are focusing on the definition of the core task of entity matching in unsupervised ER settings which do not require lots of accurately labeled training data (readers are referred to [11] for entity blocking, and [31] for entity clustering tasks):

Definition 2.1 (Fairness-agnostic ER). Given a set of candidate matches $C \subseteq E \times E'$ and a scoring function $s : E \times E' \rightarrow \mathbb{R}$, produce a subset $R \subseteq C$ of matches that maximizes the cumulative scores:

$$R = \underset{R^* \subseteq C}{\text{argmax}} \sum_{(e_i, e'_j) \in R^*} s(e_i, e'_j).$$

In fairness-agnostic ER, typically, the matching pairs are ranked $R = \left\langle (e_1, e'_1), \ldots, (e_k, e'_k) \right\rangle$ according to their scores. Additional constraints may be imposed to R according to specific assumptions pertaining to the application of an ER task. For example, when integrating two redundancy-free entity sources E and E', we want to match each entity from E to at most one entity from E'.

The above formulation of the ER problem aims to return the candidate pairs that are most likely to be true matches according to their similarity [1, 3], ignoring other qualitative features of those results, such as fairness or diversity.

Fair Entity Resolution. In this work, we are extending the typical quality assessment of ER results to also incorporate fairness. Thus, in a fair ER setting, the results that are retrieved first should not only be the ones that most likely correspond to matches, but they should also satisfy a given fairness constraint.

To quantify fairness in ER decisions, we are focusing on *equal decision measures* that allow us to examine the allocation of benefits and harms across groups by looking at the decision alone [22]. Group-based fairness definitions place matching entities into disjoint groups based on the values of their protected attributes and ask that all groups receive similar treatment, i.e., they have similar chances to be resolved. Specifically, the decision whether a pair (e, e') belongs to a protected group or not is assumed to be given by a Boolean function $pr : E \times E' \rightarrow \{true, false\}$, based on whether e and/or e' belong to the protected group. In this work, we assume that $pr(e, e') = true$, when e or e' belong to the protected group according to the value of one, or more protected attributes. A less error-tolerant protected group membership function on pairs would consider conjunction instead of disjunction. More challenges regarding this function arise, including handling missing or conflicting values, as well as learning it on the fly from the data. We leave the exploration of the impact of this decision as future work.

Ranked group fairness [5, 16, 32], employed in settings where the ordering of results is important, requires that a fairness constraint should be satisfied when considering the results within a given rank position. In our example, if we want to hire 4 experts, we may not only want the first 4 results to be matches, but also require that they equally represent males and females. To introduce a fairness-aware adaptation of Definition 2.1, we use an abstract fairness criterion F, and require that the provided solution R should satisfy F.

Definition 2.2 (Fairness-aware ER). Given a set of candidate matches $C \subseteq E \times E'$, a scoring function $s : E \times E' \rightarrow \mathbb{R}$, and a fairness criterion F, produce a ranking of matches $R \subseteq C$ that for any given rank position k, maximizes the cumulative scores:

$$R = \underset{R^* \subseteq C}{\text{argmax}} \sum_{(e_i, e'_j) \in R^*} s(e_i, e'_j)$$

$$\text{s.t. } R[k] \text{ satisfies } F,$$

where $R[k]$ are the k first results of R.

In the previous example, 3 out of 4 suggested pairs by traditional ER are male candidates (Figure 1). A fairness-aware ER algorithm targeting an equal representation of males and females would return a balanced ranking like the last ranking shown in the figure.

Note that the fairness-aware ER problem of Definition 2.2 is not the same as the fair top-k ranking problem [32], since in fairness-aware ER, the items for which a positive matching decision will be made, i.e., the subset of candidate matches that are returned as matches, are not known in advance, but may be dynamically updated based on previous decisions. For example, a fair ranking $R[4] = \left\langle (e_5, e'_6), (e_5, e'_5), (e_3, e'_3), (e_4, e'_4) \right\rangle$ (with two male and two female candidates) may not be a valid ER result, since returning both (e_5, e'_6) and (e_5, e'_5) violates the redundancy-free assumption.

3 FAIRER ALGORITHM

The fairness-aware ER problem introduced in Definition 2.2 is general enough to encompass different settings with respect to fairness criteria and matching constraints. In this section, we present a simple, yet highly efficient method, *FairER*, to solve an instance of this problem for a specific setting, as described next.

Inspired by the ranked group fairness constraint [32], we denote with $|R_p|/k$ (resp. $|R_n|/k$) the ratio of protected (resp. non-protected) group members in the first k results $R[k]$, and require that the protected and non-protected groups are equally represented in $R[k]$, i.e., that $|R_p|/k \approx |R_n|/k$. In other words, our ranked group fairness criterion F is defined as $|(|R_p|/k) - (|R_n|/k)| = \epsilon*$, where

$\epsilon*$ is the smallest possible ratio difference for a given k. As shown later, $\epsilon* = 0$ when k is even, and $\epsilon* = 1/k$, when k is odd.

As the base matching decision, we build on top of a simple algorithm (similar to existing approaches [4, 17]) that places candidate pairs in a priority queue (PQ) in descending scores and at each iteration, returns the top pair as a match. Then, it updates the queue by removing pairs containing descriptions that have been already matched, to respect the one-to-one matching constraint, and proceeds until the queue becomes empty. In FairER (Algorithm 1), we extend this method to use not only one, but two priority queues, one (Q_p) for the protected group and one (Q_n) for the non-protected group (Lines 6-10). The matching algorithm proceeds as usual, but in each iteration for one of the priority queues only (Lines 13-16), until a match is found (Lines 17-21). Then, it proceeds to the next non-empty priority queue (Lines 22-23), until k matches have been found (early termination - optional), or all queues are empty. The time complexity of Algorithm 1 is $O(|C|log|C|)$.

Algorithm 1: FairER

Input: Candidate matches $C = (E, E', s)$, k (optional)
Output: Ranked matches $R[k] = \langle (e_1, e'_1), \ldots, (e_k, e'_k) \rangle$

1 $R \leftarrow \emptyset$
2 $M \leftarrow \emptyset$ // matched nodes from E
3 $M' \leftarrow \emptyset$ // matched nodes from E'
4 $Q_p \leftarrow \emptyset$ // PQ for protected group in desc. score
5 $Q_n \leftarrow \emptyset$ // PQ for non-protected group in desc. score
6 **foreach** $c = (e, e', s(e, e')) \in C$ **do**
7 **if** *pr(c)* **then** // returns True if c is in protected group
8 $Q_p.put(c)$
9 **else**
10 $Q_n.put(c)$
11 $nextPr \leftarrow True$ // should the next pair be protected?
12 **while** $(Q_p \neq \emptyset$ ***or*** $Q_n \neq \emptyset)$ ***and*** $(|R| < k)$ **do**
13 **if** *nextPr* **then**
14 $c \leftarrow Q_p.pop()$ // the top protected pair
15 **else**
16 $c \leftarrow Q_n.pop()$ // the top non-protected pair
17 **if** $c.e \in M$ ***or*** $c.e' \in M'$ **then** // e or e' are matched
18 **continue**
19 $R.append(\langle (c.e, c.e') \rangle)$
20 $M \leftarrow M \cup \{c.e\}$
21 $M' \leftarrow M' \cup \{c.e'\}$
22 **if** $(nextPr$ ***and*** $Q_n \neq \emptyset)$ ***or*** $(\neg nextPr$ ***and*** $Q_p \neq \emptyset)$ **then**
23 $nextPr \leftarrow \neg nextPr$ // swap queues
24 **return** R

Proposition 3.1. *Algorithm 1 is a 1 - 1/e approximation to the problem of Definition 2.2, for F defined as* $|(|R_p|/k) - (|R_n|/k)| = \epsilon*$.

Proof (Sketch). Solving the problem of Definition 2.2 reduces to maximizing a monotone submodular function with cardinality constraints (k), which is an NP-hard problem [24]. The greedy approach of Algorithm 1 is a known 1 - 1/e approximation to this problem, noting that the next pair to be added on each iteration, is guaranteed to satisfy F. We omit the detailed proof due to limited space. We focus on the satisfaction of the fairness criterion F.

Table 1: Datasets and Protected Group Criteria.

Dataset	Pr. group criterion	$\lvert R_p^* \rvert, \lvert R_n^* \rvert$
BeerAdvo-RateBeer (D1)	"Red" in beer name	5, 9
iTunes-Amazon (D2)	"Dance" in genre	11, 16
Fodors-Zagats (D3)	type = "asian"	3, 19
DBLP-ACM (D4)	female last author	39, 405
DBLP-Scholar (D5)	"vldb j" in venue	80, 990
Amazon-Google (D6)	"Microsoft" in manufacturer	12, 222
Walmart-Amazon (D7)	category = "printers"	11, 182

Let $R_p \subseteq R[k]$ be the protected group matches and $R_n \subseteq R[k]$ be the non-protected group matches, where $|R_p| + |R_n| = k$, and let $|(|R_p|/k) - (|R_n|/k)| \geq \epsilon*$ for all possible solutions R^*. For a constant k, this absolute difference is minimized when $|(|R_p| - |R_n|)|$ is minimized. It is straightforward to show that when k is even, this equation is minimized for $|R_p| = |R_n| = k/2$, so $\epsilon* = 0$, while, when k is odd, it is minimized for $|(|R_p| - |R_n|)| = 1$, so $\epsilon* = 1/k$. This holds in the general case when $k \leq min(|Q_p|, |Q_n|)$. Otherwise, $\epsilon*$ is bigger. By construction, Algorithm 1 produces R_p and R_n, such that $|R_p| = |R_n| = k/2$, when k is even, and $|R_p| = (k+1)/2 = |R_n| + 1$, when k is odd. In both cases, $|(|R_p|/k) - (|R_n|/k)| = \epsilon*$. □

4 EXPERIMENTS

In this section, we present results comparing FairER to two baseline methods over 7 publicly available datasets.

Color-blind baseline. This baseline is the classic paradigm of fairness-agnostic ER, solving Problem 2.1, in which the candidate matches are forwarded to matching and matches are returned ignoring fairness. We use it as a baseline for comparing fairness-aware to fairness-agnostic ER methods.

Fa*ir baseline. In this baseline, the candidate matches are first ranked in descending score order and then re-ranked (using FA*IR [32]) to respect the fairness constraints. The re-ranked list is then forwarded to matching to produce the final output of ER.

The problem with Fa*ir baseline is that it guarantees that the input to matching is fair, but the output may not be. On the other hand, it is straightforward to prove that FairER guarantees maximal fairness until no more protected group matches exist to be returned, i.e., until the priority queue for the protected group becomes empty.

Setup. Our open-source framework[2] uses DeepMatcher[3] (10 epochs and defaults for the other parameters, yielding similar scores to [23]) as the scoring function s, and FA*IR[4] (for Fa*ir baseline).

Datasets. All datasets are publicly available from DeepMatcher. We list them in Table 1, along with the criteria we employed for defining the protected group in each dataset, and the number of ground truth matches in the protected R_p^* and non-protected R_n^* groups. For detecting gender from authors' first names in DBLP-ACM dataset, we used the gender-guesser library[5]. This was not possible in DBLP-Scholar dataset, where first names are abbreviated. We note that those criteria are orthogonal to our approach and they need not be personal data, as we show in our examples.

[2] https://github.com/vefthym/fairER
[3] https://github.com/anhaidgroup/deepmatcher
[4] https://github.com/MilkaLichtblau/FA-IR_Ranking
[5] https://github.com/lead-ratings/gender-guesser

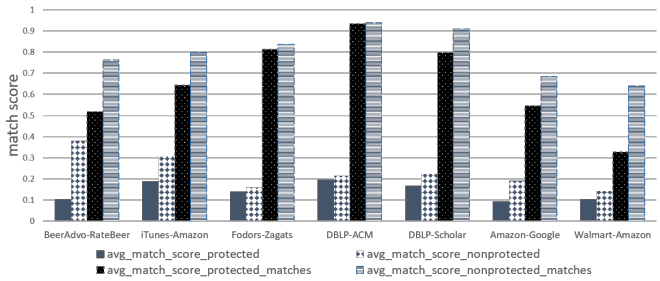

Figure 2: Average scores for prot. and non-prot. groups.

Measures. We evaluate the suggested method and baselines with respect to accuracy and fairness at the top-k positions of the returned results, for $k \in \{5, 10, 15, 20\}$. The employed measures are the following. *Accuracy@k*: out of k returned matches, how many are correct (i.e., in the ground truth of known matches)? *Bias@k*: following the fairness constraint F defined earlier as $|(|R_p|/k) - (|R_n|/k)| = \epsilon*$, we report the values of $(|R_p|/k) - (|R_n|/k)$, with negative values denoting favoring the non-protected group, zero implying no bias, and positive values favoring the protected group.

Scores of protected vs non-protected. Figure 2 shows the average scores for protected and non-protected groups, also presenting the scores only for known (from the ground truth) matches. As expected, matches have much higher average scores than non-matches. We observe that in all cases, matches belonging to the protected group (aka protected matches), shown in grey color, have lower average match scores than matches belonging to the non-protected group (aka non-protected matches), shown in blue color. The same observation holds when comparing protected (black) vs non-protected (orange) pairs (including matches and non-matches).

Accuracy. Table 2 shows the accuracy@k results for all three methods. We observe that FairER yields lower accuracy than the other methods in very few occasions, while, surprisingly, it yields better accuracy than even the color-blind method in a few cases (Walmart-Amazon $k = 5$, Amazon-Google $k = 10$, DBLP-Scholar $k = 15$). Note that accuracy may change if a different matching algorithm than the one described in Section 3 is employed.

Table 2: Accuracy results.

Method	D1	D2	D3	D4	D5	D6	D7
Accuracy@5							
Color-blind	**1**	**1**	**1**	**1**	**1**	0.8	0
Fa*ir	**1**	**1**	**1**	**1**	**1**	**1**	0
FairER	**1**	**1**	**1**	**1**	**1**	0.8	**0.4**
Accuracy@10							
Color-blind	**0.9**	**1**	**1**	**1**	**1**	0.7	**0.4**
Fa*ir	**0.9**	**1**	**1**	**1**	**1**	0.6	**0.4**
FairER	**0.9**	**1**	0.8	**1**	**1**	**0.8**	**0.4**
Accuracy@15							
Color-blind	**0.73**	**1**	**1**	**1**	0.93	**0.73**	**0.6**
Fa*ir	**0.73**	**1**	0.93	**1**	0.93	0.71	**0.6**
FairER	0.66	**1**	0.66	**1**	**1**	**0.73**	0.33
Accuracy@20							
Color-blind	**0.65**	**1**	**1**	**1**	**0.95**	**0.75**	**0.65**
Fa*ir	**0.65**	**1**	0.85	**1**	**0.95**	0.72	0.6
FairER	**0.65**	0.95	0.65	**1**	**0.95**	0.7	0.4

Table 3: Fairness (bias@k) results. [-1,0): favoring non-protected, 0: no bias, (0,1]: favoring protected. The minimum possible |bias@k| values $\epsilon*$ are also presented.

Method	D1	D2	D3	D4	D5	D6	D7
$k = 5$ ($\epsilon* = 0.2$)							
Color-blind	-0.6	-0.6	-1	-1	-1	-1	-0.6
Fa*ir	-0.6	-0.6	-0.6	-0.6	-0.6	-0.6	-0.6
FairER	**0.2**	**0.2**	**0.2**	**0.2**	**0.2**	**0.2**	**0.2**
$k = 10$ ($\epsilon* = 0$)							
Color-blind	-0.6	-0.4	-1	-0.6	-0.8	-0.8	-0.8
Fa*ir	-0.6	-0.4	-0.6	-0.6	-0.6	-0.6	-0.6
FairER	**0**	**0**	**0**	**0**	**0**	**0**	**0**
$k = 15$ ($\epsilon* = 0.07$)							
Color-blind	-0.6	-0.33	-0.73	-0.6	-0.87	-0.87	-0.87
Fa*ir	-0.47	-0.33	-0.47	-0.47	-0.47	-0.57	-0.47
FairER	**0.07**	**0.07**	**0.07**	**0.07**	**0.07**	**0.07**	**0.07**
$k = 20$ ($\epsilon* = 0$)							
Color-blind	-0.4	-0.3	-0.7	-0.6	-0.9	-0.9	-0.9
Fa*ir	-0.4	-0.3	-0.4	-0.4	-0.4	-0.56	-0.4
FairER	**0**	**0**	**0**	**0**	**0**	**0**	**0**

Fairness Scores. The bias@k scores are presented in Table 3, with numbers closer to 0 denoting better fairness, while -1 and 1 denote the extreme cases of favoring the non-protected and protected groups, respectively. Note that in those reported numbers, we omit the absolute operation when considering the difference $(|R_p|/k) - (|R_n|/k)$, just for illustration purposes, to differentiate between favoring the protected or the non-protected group, which is otherwise not important in the targeted fairness criterion F.

The results confirm that FairER constantly provides the best possible bias@k scores $\epsilon*$ in all datasets. In cases where a perfect balance is not achievable, i.e., when k is odd, we note that FairER prefers to favor the protected group (positive bias@k scores). This can be reversed by initializing *nextPr* as false (Line 11, Algorithm 1).

5 CONCLUSION

In this paper, we have introduced the problem of fairness-aware ER and proposed a general constraint-based formulation. We have presented FairER, an algorithm that solves an instance of this problem for the case of fairness constraints expressed in terms of the cardinalities of the protected and non-protected groups in the output. We are currently working on extending our Algorithm 1 to more complex protected group criteria, such as handling missing and conflicting values for the protected attributes. As a future work, we plan to study mitigation of bias in other ER tasks, such as clustering and fusion. We additionally plan to study the impact of alternative fairness measures on ER, given that several competing definitions have been proposed in the literature that imply different, possibly mutually exclusive, understandings of fairness [6, 15, 22, 25].

ACKNOWLEDGMENTS

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

REFERENCES

[1] Arvind Arasu, Michaela Götz, and Raghav Kaushik. 2010. On active learning of record matching packages. In *SIGMOD*. 783–794.

[2] Andrea Baraldi, Francesco Del Buono, Matteo Paganelli, and Francesco Guerra. 2021. Using Landmarks for Explaining Entity Matching Models. In *EDBT*. 451–456.

[3] Kedar Bellare, Suresh Iyengar, Aditya G. Parameswaran, and Vibhor Rastogi. 2012. Active sampling for entity matching. In *KDD*. 1131–1139.

[4] Christoph Böhm, Gerard de Melo, Felix Naumann, and Gerhard Weikum. 2012. LINDA: distributed web-of-data-scale entity matching. In *CIKM*. 2104–2108.

[5] L. Elisa Celis, Damian Straszak, and Nisheeth K. Vishnoi. 2018. Ranking with Fairness Constraints. In *ICALP*, Vol. 107. 28:1–28:15.

[6] Alexandra Chouldechova. 2017. Fair Prediction with Disparate Impact: A Study of Bias in Recidivism Prediction Instruments. *Big Data* 5, 2 (2017), 153–163.

[7] Vassilis Christophides, Vasilis Efthymiou, Themis Palpanas, George Papadakis, and Kostas Stefanidis. 2021. An Overview of End-to-End Entity Resolution for Big Data. *ACM Comput. Surv.* 53, 6 (2021).

[8] Amr Ebaid, Saravanan Thirumuruganathan, Walid G. Aref, Ahmed Elmagarmid, and Mourad Ouzzani. 2019. EXPLAINER: Entity Resolution Explanations. In *ICDE*. 2000–2003.

[9] Muhammad Ebraheem, Saravanan Thirumuruganathan, Shafiq R. Joty, Mourad Ouzzani, and Nan Tang. 2018. Distributed Representations of Tuples for Entity Resolution. *Proc. VLDB Endow.* 11, 11 (2018), 1454–1467.

[10] Vasilis Efthymiou, George Papadakis, Kostas Stefanidis, and Vassilis Christophides. 2019. MinoanER: Schema-Agnostic, Non-Iterative, Massively Parallel Resolution of Web Entities. In *EDBT*. 373–384.

[11] Vasilis Efthymiou, Kostas Stefanidis, and Vassilis Christophides. 2020. Benchmarking Blocking Algorithms for Web Entities. *IEEE Trans. Big Data* 6, 2 (2020), 382–395.

[12] Cheng Fu, Xianpei Han, Le Sun, Bo Chen, Wei Zhang, Suhui Wu, and Hao Kong. 2019. End-to-End Multi-Perspective Matching for Entity Resolution. In *IJCAI*. 4961–4967.

[13] Kenneth Holstein, Jennifer Wortman Vaughan, Hal Daumé III, Miroslav Dudík, and Hanna M. Wallach. 2019. Improving Fairness in Machine Learning Systems: What Do Industry Practitioners Need?. In *CHI*.

[14] Alexandros Karakasidis and Evaggelia Pitoura. 2019. Identifying Bias in Name Matching Tasks. In *EDBT*. 626–629.

[15] Jon M. Kleinberg, Sendhil Mullainathan, and Manish Raghavan. 2017. Inherent Trade-Offs in the Fair Determination of Risk Scores. In *ITCS (LIPIcs, Vol. 67)*. Dagstuhl, 43:1–43:23.

[16] Caitlin Kuhlman and Elke A. Rundensteiner. 2020. Rank Aggregation Algorithms for Fair Consensus. *Proc. VLDB Endow.* 13, 11 (2020), 2706–2719.

[17] Simon Lacoste-Julien, Konstantina Palla, Alex Davies, Gjergji Kasneci, Thore Graepel, and Zoubin Ghahramani. 2013. SiGMa: simple greedy matching for aligning large knowledge bases. In *KDD*. 572–580.

[18] Peng Li, Xi Rao, Jennifer Blase, Yue Zhang, Xu Chu, and Ce Zhang. 2021. CleanML: A Study for Evaluating the Impact of Data Cleaning on ML Classification Tasks. In *ICDE*.

[19] Karima Makhlouf, Sami Zhioua, and Catuscia Palamidessi. 2020. On the Applicability of ML Fairness Notions. *CoRR* abs/2006.16745 (2020).

[20] Edgar Meij, Tara Safavi, Chenyan Xiong, Gianluca Demartini, Miriam Redi, and Fatma Özcan. 2020. Proceedings of the KG-BIAS Workshop 2020 at AKBC 2020.

[21] Shubhanshu Mishra, Sijun He, and Luca Belli. 2020. Assessing Demographic Bias in Named Entity Recognition. *CoRR* abs/2008.03415 (2020).

[22] Shira Mitchell, Eric Potash, Solon Barocas, Alexander D'Amour, and Kristian Lum. 2021. Algorithmic Fairness: Choices, Assumptions, and Definitions. *Annual Review of Statistics and Its Application* 8, 1 (Mar 2021), 141–163.

[23] 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 *SIGMOD*. 19–34.

[24] George L. Nemhauser, Laurence A. Wolsey, and Marshall L. Fisher. 1978. An analysis of approximations for maximizing submodular set functions - I. *Math. Program.* 14, 1 (1978), 265–294.

[25] Evaggelia Pitoura, Kostas Stefanidis, and Georgia Koutrika. 2021. Fairness in Rankings and Recommendations: An Overview. *CoRR* abs/2104.05994 (2021).

[26] Kun Qian, Lucian Popa, and Prithviraj Sen. 2019. SystemER: A Human-in-the-loop System for Explainable Entity Resolution. *Proc. VLDB Endow.* 12, 12 (2019), 1794–1797.

[27] Sebastian Schelter, Yuxuan He, Jatin Khilnani, and Julia Stoyanovich. 2020. FairPrep: Promoting Data to a First-Class Citizen in Studies on Fairness-Enhancing Interventions. In *EDBT*. 395–398.

[28] Julia Stoyanovich, Bill Howe, and H. V. Jagadish. 2020. Responsible Data Management. *Proc. VLDB Endow.* 13, 12 (2020), 3474–3488.

[29] Harini Suresh and John V. Guttag. 2019. A Framework for Understanding Unintended Consequences of Machine Learning. *CoRR* abs/1901.10002 (2019).

[30] Ki Hyun Tae, Yuji Roh, Young Hun Oh, Hyunsu Kim, and Steven Euijong Whang. 2019. Data Cleaning for Accurate, Fair, and Robust Models: A Big Data - AI Integration Approach. In *DEEM@SIGMOD*. 5:1–5:4.

[31] Yan Yan, Stephen Meyles, Aria Haghighi, and Dan Suciu. 2020. Entity Matching in the Wild: A Consistent and Versatile Framework to Unify Data in Industrial Applications. In *SIGMOD*. 2287–2301.

[32] Meike Zehlike, Francesco Bonchi, Carlos Castillo, Sara Hajian, Mohamed Megahed, and Ricardo Baeza-Yates. 2017. FA*IR: A Fair Top-k Ranking Algorithm. In *CIKM*. 1569–1578.