Singapore Management University

Institutional Knowledge at Singapore Management University

Research Collection School Of Computing and Information Systems

School of Computing and Information Systems

12-2021

Robust bipoly-matching for multi-granular entities

Ween Jiann LEE
Singapore Management University, wjlee.2019@phdcs.smu.edu.sg

Maksim TKACHENKO
Singapore Management University, mtkachenko.2015@phdis.smu.edu.sg

Hady W. LAUW
Singapore Management University, hadywlauw@smu.edu.sg

Follow this and additional works at: https://ink.library.smu.edu.sg/sis_research

Part of the Databases and Information Systems Commons, and the Data Science Commons

Citation

LEE, Ween Jiann; TKACHENKO, Maksim; and LAUW, Hady W.. Robust bipoly-matching for multi-granular entities. (2021). *2021 IEEE International Conference on Data Mining ICDM: Auckland, Virtual, December 7-10: Proceedings*. 1192-1197.

Available at: https://ink.library.smu.edu.sg/sis_research/6434

This Conference Proceeding Article is brought to you for free and open access by the School of Computing and Information Systems at Institutional Knowledge at Singapore Management University. It has been accepted for inclusion in Research Collection School Of Computing and Information Systems by an authorized administrator of Institutional Knowledge at Singapore Management University. For more information, please email cherylds@smu.edu.sg.

Robust BiPoly-Matching for Multi-Granular Entities

Ween Jiann Lee
Singapore Management University
mail@wjl.ee

Maksim Tkachenko
Singapore Management University
maksim.tkatchenko@gmail.com

Hady W. Lauw
Singapore Management University
hadywlauw@smu.edu.sg

***Abstract*—Entity matching across two data sources is a prevalent need in many domains, including e-commerce. Of interest is the scenario where entities have varying granularity, e.g., a coarse product category may match multiple finer categories. Previous work in one-to-many matching generally presumes the 'one' necessarily comes from a designated source and the 'many' from the other source. In contrast, we propose a novel formulation that allows concurrent one-to-many bidirectional matching in any direction. Beyond flexibility, we also seek matching that is more robust to noisy similarity values arising from diverse entity descriptions, by introducing receptivity and reclusivity notions. In addition to an optimal formulation, we also propose an efficient and performant heuristic. Experiments on multiple real-life datasets from e-commerce sources showcase the effectiveness and outperformance of our proposed algorithms over baselines.**

***Index Terms*—entity resolution, matching, one-to-many, poly, bipoly**

I. Introduction

Entity matching identifies records associated with the same real-world entity. This work is concerned with matching across two data sources that are respectively duplicate-free. Most of the prior work presumes entities are of the same type and granularity, e.g., person to person or product to product, employing *bipartite matching* with one-to-one constraint. That assumption may not hold when entity mentions across sources are diverse. Consider a product with color variants. One e-commerce site may have a single listing, while another may list variants individually. For another example, a chapter in a textbook may span multiple chapters in another.

In contrast, *poly-matching* (Poly) allows an entity of coarser granularity (denoted *host*) from one source to match multiple entities of finer granularity (denoted *clients*) from the other source. However, prior works [1] often require hosts to come designatively and exclusively from one source, and clients from the other. We posit that such a requirement is overly restrictive when entities from the two sources have variable granularity. A more general formulation is *bidirectional poly-matching* (BiPoly), where host or client alike could come from any source. Figure 1 illustrates the matching of entities (product categories) from two e-commerce taxonomies. Mapping categories across the two sites would entail BiPoly.

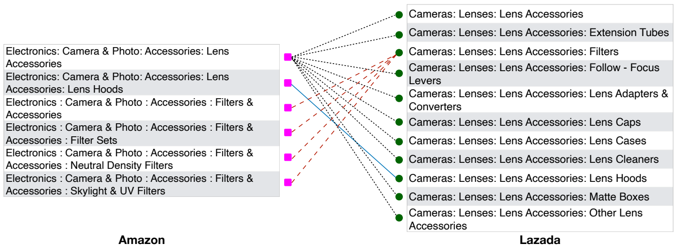

Electronics: Camera & Photo: Accessories: Lens Accessories
Electronics: Camera & Photo: Accessories: Lens Accessories: Lens Hoods
Electronics : Camera & Photo : Accessories : Filters & Accessories
Electronics : Camera & Photo : Accessories : Filters & Accessories : Filter Sets
Electronics : Camera & Photo : Accessories : Filters & Accessories : Neutral Density Filters
Electronics : Camera & Photo : Accessories : Filters & Accessories : Skylight & UV Filters
Cameras: Lenses: Lens Accessories
Cameras: Lenses: Lens Accessories: Extension Tubes
Cameras: Lenses: Lens Accessories: Filters
Cameras: Lenses: Lens Accessories: Follow - Focus Levers
Cameras: Lenses: Lens Accessories: Lens Adapters & Converters
Cameras: Lenses: Lens Accessories: Lens Caps
Cameras: Lenses: Lens Accessories: Lens Cases
Cameras: Lenses: Lens Accessories: Lens Cleaners
Cameras: Lenses: Lens Accessories: Lens Hoods
Cameras: Lenses: Lens Accessories: Matte Boxes
Cameras: Lenses: Lens Accessories: Other Lens Accessories
Amazon
Lazada

Fig. 1: A bidirectional poly-matching between the lens accessories categories in Amazon and Lazada.

Given as input are similarity weights between any pair of entities across sources. One possible objective is to maximize the sum of similarity weights across client-to-host assignments, which is rational insofar as the input weights are reflective of true similarity. In practice, similarity weights are often derived from entity descriptions (e.g., product titles), with noises abound. We propose a more robust objective that additionally incorporates two types of rewards. *Reclusivity* encourages entities to refrain from participating in any matching, so incidental similarities would not result in false positives. *Receptivity* encourages entities to form a matching, inducing more connected components of smaller sizes while discouraging large clusters with many entities. https://www.overleaf.com/project/612ef8d891a5168973db170e

Contributions and Organization. In summary, the main contributions of this paper are as follow:

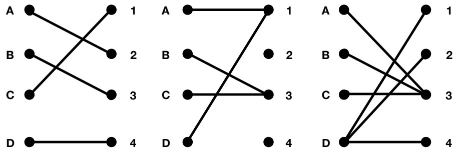
- (subsection II-A): We formalize the problem of BiPoly for multi-granular entities. The bidirectionality is novel, inducing a more general formulation than previously known bipartite and poly-matchings.
- (subsection II-B): We propose a robust optimization objective by incorporating receptivity and reclusivity notions to mitigate the effects of noisy similarity.
- (section III): We express an optimal formulation for BiPoly using Integer Linear Programming (ILP), which subsumes bipartite and poly-matchings as special cases.
- (section IV): We develop a computationally efficient greedy algorithm with a known approximation bound.
- (section V): We conduct experiments to demonstrate the effectiveness of our algorithms against baselines.

We cover the related literature in section VI and discuss the key findings as well as future work in section VII.

II. Problem Formulation

We are given two sets of entities, the left set $L = \{l_1, l_2, \ldots, l_m\}$ and the right set $R = \{r_1, r_2, \ldots, r_n\}$, that contain m and n records respectively. Informally, our objective is to find all pairs of matching multi-granular entities, $\rho \subseteq L \times R$, such that for every pair $\langle l, r\rangle \in \rho$ we can say that either l *is a part of* r, l *consists of* r, or l *is equivalent to* r.

h	1	2	3	4
A	0.9	0.9	0.8	0
B	0.5	0	0.7	0.3
C	0.4	0.2	0.5	0.4
D	0.8	0.5	0	0.8



	(a)	(b)	(c)
Bipartite	✔	✘	✘
Poly	✔	✔	✘
BiPoly	✔	✔	✔

Fig. 2: Results based on max-weighted Bipartite, Poly (one-to-many) and the BiPoly constrained matching.

We identify an entity matching by similarity function $h : L \times R \rightarrow [0, 1]$. For any $r \in R$ and $l \in L$, $h(r, l)$ is defined as $h(l, r)$. $h(l, r)$ ranges from 0 (l and r are not related) to 1 (highly related). Ideally, $\langle l, r\rangle \in \rho$ if and only if $h(l, r) \approx 1$. In practice, however, we deal with similarity functions – assumed specified as input – that often produce high similarity scores for entities that ought not to be included in ρ (i.e., false positives) or low similarity scores for entities that ought to be included in ρ (i.e., false negatives). Therefore, in the following sections, we introduce constraints that a valid matching must comply with, as well as a robust objective function.

A. Matching Constraints

Let $G(L, R)$ be a bipartite graph over the left L and right R entity sets, $L \cap R = \emptyset$. All pairs of entities between two parts are connected by an edge: $\langle l, r\rangle \in E(G)$. Each edge is weighted by the similarity score $h(l, r)$. Any $\rho \subseteq E(G)$ is a matching. We call a graph $G_\rho(L, R)$ with edges from ρ a matching-induced subgraph or simply matching subgraph. For any $l \in L$, let $\rho_l = \{\langle u, r\rangle \in \rho | u = l\}$. Analogously, for any $r \in R$, let $\rho_r = \{\langle l, u\rangle \in \rho | u = r\}$. ρ_u is essentially the set of all the edges connected with u. Since $L \cap R = \emptyset$, ρ_u is unambiguously defined over any $u \in L \cup R$.

To capture the relations of *is equivalent to* between L and R entities, it is apt to employ one-to-one constraint [2], [3].

Definition II.1 **(Bipartite Matching).** *$\rho \subseteq E(G)$ is a bipartite (one-to-one) matching in $G(L, R)$ if and only if $\forall u \in L \cup R, |\rho_u| \leq 1$.*

This is the most restrictive constraint. Figure 2 illustrates a toy example involving $L = \{A, B, C, D\}$ and $R = \{1, 2, 3, 4\}$. The adjacency matrix specifies the similarity between any pair of entities. Based on these scores, the maximum weight bipartite matching is $\rho = \{\langle A, 2\rangle, \langle B, 3\rangle, \langle C, 1\rangle, \langle D, 4\rangle\}$.

Bipartite matching is inappropriate for *consists of* or *is part of* relations involving variable granularity. We describe *poly-matching* as a relaxation to the one-to-one constraint [3]–[5].

Definition II.2 **(Poly-Matching).** *$\rho \subseteq E(G)$ is a poly-matching in $G(L, R)$ if and only if $\forall l \in L, |\rho_l| \leq 1$. When the constraint is imposed on the L, we say that the left L is matched into the right R.*

Entities on the right can be matched to multiple entities on the left, but entities on the left can be matched to at most one entity on the right. This aligns with the nature of a taxonomy that allows a record on the right to encompass multiple more records on the left. Continuing with the example in Figure 2, poly-matching allows A and D on the left to match the same entity on the right 1. Ditto, we have $\rho_3 = \{\langle B, 3\rangle, \langle C, 3\rangle\}$.

In addition to *is equivalent to*, Poly captures either *consists of* or *is part of*, but not both. Our proposed *bidirectional poly-matching* allows the matching to work in both directions.

Definition II.3 **(Bidirectional Poly-Matching).** *$\rho \subseteq E(G)$ is BiPoly in $G(L, R)$ if and only if $\forall\langle l, r\rangle \in \rho$, $|\rho_l| = 1 \vee |\rho_r| = 1$.*

Figure 1 depicts exactly this kind of matching. BiPoly is also the least restrictive matching as its solution space entails that of Bipartite and Poly as shown in Figure 2.

B. Objective Function

Having identified possible matching space $\mathfrak{P}$, we seek to find $\rho \in \mathfrak{P}$, which maximizes a particular objective function.

Max-Weight. The classical objective is to maximize the total similarity score, known as the maximum weight objective.

$$\rho = \underset{\rho' \in \mathfrak{P}}{\arg\max} \sum_{\langle l, r\rangle \in \rho'} h(l, r) \tag{1}$$

In practice, the similarity measure $h(l, r)$ (e.g., cosine similarity) is unlikely perfect. The entity representations used to measure that similarity (e.g., product title) are highly noisy. For instance, entities of a domain may contain a generic word (e.g., *camera*) which results in non-zero similarity.

Robust. We develop a robust version of the constraint objective, which introduces independently adjustable incentives to reduce the number of false positives and negatives. The rewards are cast upon the connected components in $\rho \in \mathfrak{P}$. Any matching-induced subgraph G_ρ consists of one or more connected components. We distinguish between two types of connected components. A component is *reclusive* if it contains exactly one entity u, i.e., $|\rho_u| = 0$. Otherwise, it is *receptive*, i.e., for each entity u in that component, we have $|\rho_u| \geq 1$.

We define the following robust objective with the reclusivity incentive function $\mathcal{F}_\Omega$ and receptivity incentive function $\mathcal{F}_H$:

$$\rho = \underset{\rho' \in \mathfrak{P}}{\arg\max} \sum_{\langle l, r\rangle \in \rho'} h(l, r) + \mathcal{F}_H\left(C(G_\rho)\right) + \mathcal{F}_\Omega\left(\bar{C}(G_\rho)\right), \tag{2}$$

where $C(\,\cdot\,)$ and $\bar{C}(\,\cdot\,)$ are the sets of receptive and reclusive components in a given graph respectively.

The function $\mathcal{F}_\Omega$ provides an incentive to each reclusive component. This reward can be positive, which encourages entities to stay reclusive unless they can form a match that captures a similarity score higher than the incentive. Alternatively, this reward can be negative, which discourages reclusivity.

Among receptive components, there is a trade-off between a smaller number of connected components of larger cardinalities, or a larger number of components of smaller cardinalities. This is managed by $\mathcal{F}_H$. If it offers positive rewards to receptive components, the outcome tends towards more receptive components, which may each tend to contain fewer entities. If it offers negative rewards, the converse applies.

TABLE I: Summary of Notations

Notation	Description
$h(l,r) \in [0,1]$	similarity score between l and r (input)
$v_u \in \{1,0\}$	variable whether u is a host/client
$z_u \in \{1,0\}$	variable whether u is a receptive/reclusive
$v_u = 1, z_u = 1$	indicating u is a receptive host
$v_u = 1, z_u = 0$	indicating u is a reclusive host
$v_u = 0, z_u = 0$	indicating u is a client
$x_{l \to r} \in \{0,1\}$	whether host r accepts request from client l
$x_{r \to l} \in \{0,1\}$	whether host l accepts request from client r
$\omega_u \in [-1,1]$	reclusivity reward offered to u (parameter)
$\eta_u \in [-1,1]$	receptivity reward offered to u (parameter)

III. Linear Programming Models

To articulate a concrete definition of robust BiPoly (other formulations represent special cases), we propose a binary linear program that lends itself to an optimal solution.

A. Formulation

For each connected component in a matching-induced subgraph, we designate exactly one node to be the *host*, and the rest are *clients*, indicated by v_u. A reclusive component has one host and no client. To differentiate reclusive from receptive host, we use binary variable z_u. To indicate connectivity within receptive components, for every $\langle l, r\rangle \in E(G)$, we introduce two mutually exclusive variables $x_{l \to r}$ and $x_{r \to l}$ that indicate whether l and r are connected. These variables and other relevant notations are summarized in Table I.

To arrive at a robust BiPoly matching, we seek a configuration that fulfils the following linear program.

$$\begin{aligned} \max \sum_{l\in L}\sum_{r\in R} & h(l,r)\,(x_{l\to r} + x_{r\to l}) \\ & + \sum_{u\in L\cup R} \omega_u\,(v_u - z_u) + \sum_{u \in L\cup R} \eta_u z_u \qquad \text{s.t.} \end{aligned} \tag{3}$$

$$\textstyle\sum_{l\in L} x_{r\to l} + v_r = 1 \qquad \forall r \in R \tag{4}$$

$$\textstyle\sum_{r\in R} x_{l\to r} + v_l = 1 \qquad \forall l \in L \tag{5}$$

$$x_{r\to l} \le v_l \qquad \forall l \in L, \forall r \in R \tag{6}$$

$$x_{l\to r} \le v_r \qquad \forall l \in L, \forall r \in R \tag{7}$$

$$\textstyle\sum_{l\in L} x_{l\to r} \ge z_r \qquad \forall r \in R \tag{8}$$

$$\textstyle\sum_{r\in R} x_{r\to l} \ge z_l \qquad \forall l \in L \tag{9}$$

$$x_{l\to r} \le z_r \qquad \forall l \in L, \forall r \in R \tag{10}$$

$$x_{r\to l} \le z_l \qquad \forall l \in L, \forall r \in R \tag{11}$$

The first line of Equation 3 subject to the constraints reproduces the max-weight objective in Equation 1.

The first part of the second line is a formulation of $\mathcal{F}_\Omega$ from Equation 2. Each entity u is offered a reclusivity reward $\omega_u \in [-1, 1]$ (a parameter to be specified), which is earned only if u is a reclusive host, i.e., $(v_u = 1, z_u = 0)$. If $\omega_u > 0$, this incentivizes u to stay reclusive unless it can form a match that captures a similarity score higher than the incentive. When $\omega_u < 0$, it is encouraged to form a match, with ω_u compensating for low similarity scores. $\omega_u = 0$ is neutral.

The second part presents a formulation of $\mathcal{F}_H$ in Equation 2. A receptivity reward $\eta_u \in [-1, 1]$ (a parameter) is earned only if u is a receptive host. When $\eta_u > 0$, more receptive components are encouraged, which may reduce the cardinality of other receptive components. If $\eta_u < 0$, it consumes a part of the similarity within each component, discouraging any connection from being formed. $\eta_u = 0$ is neutral.

While unique rewards for every node u are possible, in practice we experiment with a simpler framework where rewards are tied for each part of $G(L, R)$: $\omega_l = \omega_L$ and $\eta_l = \eta_L$, $\forall l \in L$, analogously $\omega_r = \omega_R$ and $\eta_r = \eta_R$, $\forall r \in R$.

Equation 4 and 5 ensure that entities can only be a host or a client, and a client's request can only be accepted by a single host. Equation 6 and 7 state that only hosts can accept requests. These four constraints jointly enforce the bidirectional one-to-many restriction. Equation 8 through 11 ensure that $z_u = 1$ if and only if u is a host that has accepted one or more clients.

BiPoly-matching presents the loosest constraint setting. To recover poly-matching, we add linear constraint $x_{l\to r} = 0$ for all $\langle l, r\rangle \in E(G)$ blocking all the client requests from the left part of the graph L. To recover bipartite matching from poly-matching, we restrict the number of incoming requests for each host to at most one: $\sum_{r\in R} x_{r\to l} \le 1$ for all $l \in L$.

IV. Greedy Approximation

As the linear program may be intractable for large data, we present a greedy algorithm with approximation guarantee.

Algorithm. Let us consider the following minimization problem s.t. the constraints of the robust BiPoly-matching:

$$\begin{aligned} \min \sum_{l\in L}\sum_{r\in R} & (1 - h(l,r))\,(x_{l\to r} + x_{r\to l}) \\ & + \sum_{u\in L\cup R} (1-\omega_u)\,(v_u - z_u) + \sum_{u \in L\cup R} (1-\eta_u)\, z_u \end{aligned} \tag{12}$$

Expanding the summations and using Equations 4 and 5, one can show that this minimization problem is equivalent to the robust BiPoly. Since the variable weights are non-negative, the problem can be converted to an instance of weighted set cover: all the nodes in $L \cup R$ are to be covered with sets that yield minimal total cost. The collection of covering sets and their weights are defined as follows:

$$\mathcal{S} = \mathcal{S}_L \cup \mathcal{S}_R, \text{ where} \qquad \begin{matrix} \mathcal{S}_L = \big\{\{l\} \cup e \,\big|\, l \in L, e \in 2^R\big\} \\ \mathcal{S}_R = \big\{\{r\} \cup e \,\big|\, r \in R, e \in 2^L\big\} \end{matrix} \tag{13}$$

$$w_s = \begin{cases} 1 - \omega_u & \text{if } \{u\} = s \\ 2 - \max(\eta_l, \eta_r) - h(l,r) & \text{if } \{l\} = s \cap L \text{ and } \{r\} = s \cap R \\ |s| - \eta_l - \sum_{r\in s\setminus\{l\}} h(l,r) & \text{if } s \in \mathcal{S}_L \setminus \mathcal{S}_R \text{ and } \{l\} = s\cap L \\ |s| - \eta_r - \sum_{l\in s\setminus\{r\}} h(l,r) & \text{if } s \in \mathcal{S}_R \setminus \mathcal{S}_L \text{ and } \{r\} = s\cap R \end{cases}$$

The weighted set cover framework is tightly related to the connected component interpretation of the robust BiPoly as outlined in subsection III-A. $\mathcal{S}$ defines a set of all possible connected components in $G(L, R)$ w.r.t. BiPoly as in Definition II.3. $\mathcal{S}_L$ identifies all possible connected components between the hosts from L and the clients from R including closed hosts, $\mathcal{S}_R$ does the same for the hosts from R. Weights $W = \{w_s\}_{s\in\mathcal{S}}$ are mapped to the contributions of these components as in the robust BiPoly objective. The first case for w_s identifies the contribution of reclusive hosts, the second counts the minimal contribution of the interchangeable host-client pairs, and the last two cases identify the contribution of the components that have multiple clients. Though the solution space is larger for the weighted set cover problem than for robust BiPoly, one can show that at least one of the minimizing solutions of the set cover problem consists of disjoint sets, thus, is convertible to a minimizing robust BiPoly solution.

BiPoly Set Cover adapts the greedy solver as in [6]. At each iteration, the procedure finds set $s \in \mathcal{S}$ minimizing its per node weight $w_s/|s|$ to cover nodes yet not assigned. To avoid enumerating all possible sets as defined in Equation 13, for each node $u \in L \cup R$ we maintain a list of potential clients in descending order of similarity $h(u,\cdot)$. At each round of the procedure, we retrieve a minimizing set in polynomial time.

Complexity Analysis. The complexity can be split into two: sorting clients for each host, accounting for $\mathcal{O}(nm\log nm)$ in worst-case, and selecting a minimizing set from $\mathcal{S}$.

The worst case is when every viable combination of hosts and clients is evaluated, but only a singleton host is selected in the round, and we always select a host from the bigger part. Let $a = \min(n,m)$, $b = \max(n,m)$, and $k = b - a$, then the computations for the first k iterations can be bounded by $\mathcal{O}\Big(a\sum_{i=1}^{k}(2b-i+1)\Big)$ or $\mathcal{O}(ab^2)$. For the next $2a$ iterations the procedure alternates between selecting hosts from L and R, thus, the number of operations for every two consecutive rounds can be bounded (up to a constant) by $2a(n+m)$, $2(a-1)(n+m)$, $2(a-2)(n+m)$ and so on, which in total is bounded by $\mathcal{O}\big((m+n)\sum_{i=1}^{a} i\big)$ or $\mathcal{O}\big(a^2(m+n)\big)$. Therefore, the worst case complexity of BiPoly is $\mathcal{O}\big((n+m)^3\big)$.

Approximation Bound. BiPoly has an approximation guarantee [7]: $log(n+m)$ to an optimal solution of Equation 12. Let $g^{\min}_{\text{greedy}}$ be the objective value with the greedy solution, $g^{\min}_{\text{opt}}$ be the optimal objective value, then:

$$g^{\min}_{\text{greedy}} \leq g^{\min}_{\text{opt}} \log{(n+m)}.$$

Since, the original objective is shifted by $n+m$, we can derive a lower bound for a greedy solution g_{greedy} of Program 3:

$$g_{\text{greedy}} \geq g_{\text{opt}} \log{(n+m)} + (n+m)\left(1 - \log{(n+m)}\right),$$

where g_{opt} is the optimal objective value for the robust BiPoly.

V. Experiment

We investigate the effectiveness of the proposed algorithm at how well it aligns multi-granular entities.

A. Datasets

As matching multi-granular entities is novel, we gather two real-world datasets with known ground truths as in Table II.

TABLE II: Summary of Datasets

Dataset	m	n	$\lvert e^L \rvert$	$\lvert e^R \rvert$	Matches
Cross-Platform	94	173	6,913	166,307	992
Multi-Lingual	119	169	710,475	20,000	2,315

Cross-Platform. The first dataset involves matching product categories across two e-commerce platforms. Such a scenario could occur when merging product taxonomies. This dataset has $m = 94$ categories from Amazon US (the left set L) and $n = 173$ categories from Lazada (the right set R).

Determining whether two categories match must go beyond superficial names and necessarily be informed by whether they contain the same products. We employ Mechanical Turk (MTurk) to manually identify matching *products*, and assess the matching of *categories* indirectly by how the latter facilitates the former. MTurk workers were instructed to select whether a candidate product matches the target product, where two products are considered a match only if they have the same brand, model, type, size, color, etc. Each task was assigned to three workers and the majority vote was accepted. Let e^L (resp. e^R) be the union of products under the category entities in L (resp. R). In total, 992 product matches are identified out of 15,516 pairs labeled, sampled from e^L and e^R.

Multi-Lingual. The second dataset again concerns category entities, but the respective sources correspond to two regional platforms of Amazon, namely the US (the left set L) and China (the right set R). Product categorization differs vastly between both regions. Amazon uses a unique identifier (UID), which we assume is relatively consistent across regions. In the case of Amazon China, we use machine translation with Microsoft Azure Translator to produce the English representations for similarity measurement with Amazon US vocabulary. In total, there are $2,315$ matching product pairs identified by UID.

The representation of a product entity is its title, while that of a category entity is a bag of words of product titles within the category. To derive the similarity for matching $h(l,r)$, we apply three normalized similarity functions, namely Jaccard Coefficient, Szymkiewicz–Simpson coefficient [8], and term frequency-inverse document frequency (TFIDF) cosine similarity, and select the best coverage for each method.

B. Evaluation Measures

Intuitively two categories from different sources are well-matched if they collectively contain ground-truth product pairs.

Coverage. We define a matching ρ's goodness in terms of *coverage*. A better ρ recovers more ground-truth product pairs.

$$\text{product-pairs}(\rho) = \left\{\langle a,b\rangle \,\middle|\, a \in e^l, b \in e^r, \langle l,r\rangle \in \rho\right\}$$
$$\text{coverage}(\rho) = \frac{\sum_{\langle a,b\rangle \in \text{product-pairs}(\rho)} g(a,b)}{\sum_{\langle a,b\rangle \in \text{product-pairs}(L\times R)} g(a,b)} \qquad (14)$$

Technically the maximum coverage is 1. However, under the bidirectional one-to-many constraint, even if we use oracle-like $\tilde{h}(l,r)$, the *theoretical maximum achievable coverage* would be below 1. The coverage for a matching ρ with a noisy similarity $h(l,r)$ would be bounded by the coverage for the optimal max-weight matching $\tilde{\rho}$ with the prefect similarity $\tilde{h}$. In view of this, we present the fraction of actual $\text{coverage}(\rho)$ (using h) over its theoretical maximum $\text{coverage}(\tilde{\rho})$ (using $\tilde{h}$).

Tradeoff. Coverage alone is a one-sided measure. To appreciate this, we allude to the notion of blocking [9] in the product-to-product matching problem. Without blocking, we would compare all $|e^L| \times |e^R|$ pairs of products. With category matching as a blocking strategy, we compare only products across matched categories (much fewer). We can thus define *reduction* due to a category matching outcome ρ as follows.

$$\text{reduction}(\rho) = 1 - \frac{\sum_{\langle l,r\rangle \in \rho} |e^l| \cdot |e^r|}{|e^L| \cdot |e^R|} \tag{15}$$

The higher the reduction, the fewer product pairs to be compared, the more efficient is the product-to-product matching. Most methods we tested achieve a reduction of over 70%.

Since higher reduction often corresponds to lower coverage, a more balanced metric is the *tradeoff*, expressed as the harmonic mean of coverage and reduction, which has been established in product-to-product matching literature [10].

$$\text{tradeoff}(\rho) = \frac{2 \times \text{coverage}(\rho) \times \text{reduction}(\rho)}{\text{coverage}(\rho) + \text{reduction}(\rho)} \tag{16}$$

We define *tradeoff* as the $\text{tradeoff}(\rho)$ normalized by the theoretical maximum $\text{tradeoff}(\tilde{\rho})$ (under oracular $\tilde{h}$).

C. Optimal Solutions via Linear Program

Max-Weight Objective. We first conduct a comparison under the maximum weight objective (see subsection II-B). The baselines are Bipartite and Poly, which share the same objective and differ only in constraints. Table III (rightmost columns) summarizes the results. Evidently, constraints do matter. Due to the inherent multi-granularity of entities, Bipartite with the most restrictive one-to-one constraint has the lowest coverage. Poly-matching allows one-to-many and realizes more coverage, but is still limited by the restriction that the 'one' must come from one source (the source with fewer entities[1]). Because BiPoly factors in the possibility for hosts to flexibly come from either source, it attains the highest coverage. A similar trend manifests in the Tradeoff as well.

Robust Objective. For comparison under the robust objective (see subsection II-B), the baselines are now *robust* Bipartite and Poly-matching. For completeness, we include an additional method, namely UFLP [1], which constitutes a special case of robust Poly when $\omega_L = \omega_R = 0$, $\eta \leq 0$, and flipped similarity coefficients (every edge is weighted by $1 - h(l,r)$ as opposed to $h(l,r)$). We report the results for the best selection of the robust parameters (i.e., ω_L, ω_R, η_L, and η_R). The parameters are optimized simultaneously via grid search with a range from -1 to 1 in incremental steps of 0.1. Since Bipartite and Poly could produce only one-sided matchings, η_R does not influence the maximizing solution and can be set arbitrarily for these problems (we indicate this fact by a dash). As in Table III (middle columns), the most restrictive method, Bipartite, is the weakest. From UFLP to Poly to BiPoly, the constraints are increasingly flexible, and recall and tradeoff improve correspondingly.

TABLE III: Comparison on Linear Program

						Robust		Max-Weight	
	Constraint	ω_L	ω_R	η_L	η_R	Coverage	Tradeoff	Coverage	Tradeoff
Cross-Platform	Bipartite	0.3	0.3	0.0	—	57.70	70.49	55.61	68.66
	UFLP	0.0	0.0	-0.9	—	68.46	79.08	68.31	78.98
	Poly	0.5	0.5	0.7	—	70.70	80.87		
	BiPoly	0.1	0.1	0.2	0.2	**84.30**	**90.32**	**84.16**	**90.21**
Multi-Lingual	Bipartite	0.1	0.1	0.0	—	36.96	53.75	36.42	53.09
	UFLP	0.0	0.0	-0.8	—	71.83	89.19	63.27	81.55
	Poly	-0.8	-0.8	-1.0	—	71.83	89.19		
	BiPoly	0.0	0.0	0.7	0.7	**84.67**	**91.56**	**73.93**	**87.21**

TABLE IV: Greedy Approximations with Robust Objective

	Constraint	ω_L	ω_R	η_L	η_R	Coverage	Tradeoff
Cross-Platform	Bipartite	0.2	0.2	0.0	—	54.26	67.42
	UFLP	0.0	0.0	-0.1	—	67.12	78.07
	Poly	0.1	0.1	0.1	—	67.12	78.13
	BiPoly	0.1	0.0	-0.1	-0.1	**82.66**	**89.47**
Multi-Lingual	Bipartite	0.2	0.2	0.0	—	27.55	41.96
	UFLP	0.0	0.0	-0.2	—	59.30	77.39
	Poly	-0.2	-0.2	-0.8	—	68.95	86.15
	BiPoly	0.2	0.2	0.0	0.2	**70.82**	**86.48**

We can compare Robust vs. Max-weight objectives in Table III. For instance, robust Bipartite outperforms max-weight Bipartite. The same generally holds for Poly and BiPoly as well. In particular, the robust objective makes a more significant difference on *Multi-Lingual* than on *Cross-Platform*, as the former has noisier similarities arising from the translation from Chinese to English when deriving similarity weights. The robust objective, designed to counter noises, apparently delivers on this count. BiPoly's coverage improves from 74% with max-weight to 85% with the robust objective. For subsequent discussion, we focus on the robust objective.

D. Greedy Approximation

We present the greedy solutions in Table IV. For one, BiPoly outperforms the baselines. For another, we can appreciate how closely the greedy solutions approach the optimal by comparing across Table IV and Table III. Observably, they get very close to the optimal on *Cross-Platform*, which we attribute to its relatively 'cleaner' and presumably more informative similarity weights. On the noisier *Multi-Lingual*, the gap is expectedly larger, but even so there is still a credible outperformance over the baselines by BiPoly. In both cases, the difference in tradeoff is less than coverage, as greedy is able to match records that contain more matching entities and only miss out on matches in records with far lower similarity.

E. Sensitivity Analysis

Figure 3 tracks coverage and tradeoff as reclusivity ω and receptivity η vary respectively on *Multi-Lingual* with BiPoly.

Varying Reclusivity ω (blue curves). Increasing ω when $\eta = 0$ tends to reduce coverage and correspondingly the tradeoff. At $\omega = -1$, there is a penalty for being a reclusive host. There are few singletons and the matched clusters result in high coverage. As we increase ω to 1, there is a reward to

[1] We have also investigated the reverse scenario where the 'one' must come from the source with more entities, which performs worse in all cases.

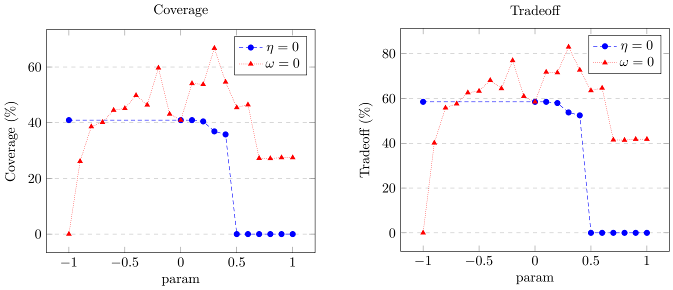


Fig. 3: Varying reclusivity ω and receptivity η for *Multi-Lingual* for BiPoly. *param* = ω when $\eta = 0$ and vice versa.

being reclusive, and progressively there are more singletons to the point where there is no connected component when $\omega = 1$.

Varying Receptivity η (red curves). When $\eta = -1$, there is a severe penalty to being a receptive host, and all nodes remain single. As η increases, some formerly reclusive hosts are incentivized to be receptive, attracting clients either from other former reclusive hosts or clients of other open hosts, thus increasing coverage and recall. Ultimately, when $\eta = 1$, we realize as many receptive hosts as possible, but each is part of a small 2-node cluster, reducing coverage and tradeoff.

VI. Related Work

Constraint matching stems from the assignment problem (AP) that assigns 'tasks' to 'agents' [11], [12]. One distinction is how task and agent in AP are designated apriori, whereas an entity could take on either host or client role in our problem. For entity matching, the emphasis is placed on cost-focused approaches to maximize similarity weights. Classical ones include one-to-one [11] and (unidirectional) one-to-many [13]. The classic AP is known as the weighted bipartite matching [14], the one-to-many AP is akin to max-weighted poly-matching. In turn, the aforementioned UFLP is an extension of generalized AP [15]. These instantiations of AP for entity matching have been included as baselines.

There are various directions in improving Entity Matching (EM) [16], [17]. One is to improve the similarity estimation either by better representation [18], or multiple attributes [2], [19], or employing supervised learning [20], [21]. Another direction is to improve the efficiency of the matching process, by using blocking [22], hashing [23], or end-to-end workflow [10]. Our work pursues an orthogonal direction in applying 'global constraints' to improve the quality of matching overall, particularly for multi-granular entities. Few prior works have studied leveraging global constraints. While [24], [25] use some constraints, their conditions apply between two records, e.g., *John Doe* and *J. Doe* must live at the same zip code to match. While [3] applied one-to-one and one-to-many constraints on matching, there have not been any studies on multi-granular datasets nor robustness.

VII. Conclusion

We address bidirectional poly-matching of multi-granular entities. Key to its robustness are novel notions of reclusivity and receptivity, which cooperatively help to counter the noisy similarity. We develop both an optimal solution via linear programming as well as an efficient greedy algorithm. Comprehensive experiments validate our contributions and shed light on the workings of the algorithms on real-world datasets.

References

[1] G. Cornuéjols, G. Nemhauser, and L. Wolsey, "The uncapicitated facility location problem," Cornell University Operations Research and Industrial Engineering, Tech. Rep., 1983.

[2] A. Hashemi, M. B. Dowlatshahi, and H. Nezamabadi-Pour, "A bipartite matching-based feature selection for multi-label learning," *IJMLC*, vol. 12, no. 2, pp. 459–475, 2021.

[3] J. Gemmell, B. I. Rubinstein, and A. K. Chandra, "Improving entity resolution with global constraints," *arXiv:1108.6016*, 2011.

[4] A. Khan, A. Pothen, M. Mostofa Ali Patwary, N. R. Satish, N. Sundaram, F. Manne, M. Halappanavar, and P. Dubey, "Efficient approximation algorithms for weighted b-matching," *SIAM Journal on Scientific Computing*, vol. 38, no. 5, pp. S593–S619, 2016.

[5] A. Dutta and A. Asaithambi, "One-to-many bipartite matching based coalition formation for multi-robot task allocation," in *ICRA*, 2019.

[6] V. Chvatal, "A greedy heuristic for the set-covering problem," *Mathematics of operations research*, vol. 4, no. 3, pp. 233–235, 1979.

[7] P. Slavık, "A tight analysis of the greedy algorithm for set cover," *Journal of Algorithms*, vol. 25, no. 2, pp. 237–254, 1997.

[8] M. Vijaymeena and K. Kavitha, "A survey on similarity measures in text mining," *MLAIJ*, vol. 3, no. 2, pp. 19–28, 2016.

[9] G. Papadakis, E. Ioannou, T. Palpanas, C. Niederee, and W. Nejdl, "A blocking framework for entity resolution in highly heterogeneous information spaces," *IEEE TKDE*, vol. 25, no. 12, pp. 2665–2682, 2012.

[10] G. Papadakis, L. Tsekouras, E. Thanos, G. Giannakopoulos, T. Palpanas, and M. Koubarakis, "The return of jedai: End-to-end entity resolution for structured and semi-structured data," *PVLDB*, vol. 11, no. 12, 2018.

[11] D. W. Pentico, "Assignment problems: A golden anniversary survey," *EJOR*, vol. 176, no. 2, pp. 774–793, 2007.

[12] T. Öncan, "A survey of the generalized assignment problem and its applications," *INFOR*, vol. 45, no. 3, pp. 123–141, 2007.

[13] H. Zhu, M. Zhou, and R. Alkins, "Group role assignment via a kuhn–munkres algorithm-based solution," *IEEE TSMC*, vol. 42, no. 3, 2011.

[14] H. W. Kuhn, "The hungarian method for the assignment problem," *Naval research logistics quarterly*, vol. 2, no. 1-2, pp. 83–97, 1955.

[15] G. T. Ross and R. M. Soland, "Modeling facility location problems as generalized assignment problems," *Management Science*, vol. 24, no. 3, pp. 345–357, 1977.

[16] H. Köpcke and E. Rahm, "Frameworks for entity matching: A comparison," *DKE*, vol. 69, no. 2, pp. 197–210, 2010.

[17] O. Hassanzadeh, F. Chiang, H. C. Lee, and R. J. Miller, "Framework for evaluating clustering algorithms in duplicate detection," *PVLDB*, vol. 2, no. 1, pp. 1282–1293, 2009.

[18] M. Ebraheem, S. Thirumuruganathan, S. Joty, M. Ouzzani, and N. Tang, "Distributed representations of tuples for entity resolution," *Proceedings of the VLDB Endowment*, vol. 11, no. 11, pp. 1454–1467, 2018.

[19] C. Fu, X. Han, J. He, and L. Sun, "Hierarchical matching network for heterogeneous entity resolution," in *IJCAI*, 2020, pp. 3665–3671.

[20] D. Zhang, Y. Nie, S. Wu, Y. Shen, and K.-L. Tan, "Multi-context attention for entity matching," in *WebConf*, 2020, pp. 2634–2640.

[21] C. Zhao and Y. He, "Auto-em: End-to-end fuzzy entity-matching using pre-trained deep models and transfer learning," in *WebConf*, 2019.

[22] G. Papadakis, J. Svirsky, A. Gal, and T. Palpanas, "Comparative analysis of approximate blocking techniques for entity resolution," *PVLDB*, vol. 9, no. 9, pp. 684–695, 2016.

[23] L. Paulevé, H. Jégou, and L. Amsaleg, "Locality sensitive hashing: A comparison of hash function types and querying mechanisms," *Pattern recognition letters*, vol. 31, no. 11, pp. 1348–1358, 2010.

[24] S. E. Whang, O. Benjelloun, and H. Garcia-Molina, "Generic entity resolution with negative rules," *VLDBJ*, vol. 18, no. 6, 2009.

[25] S. Chaudhuri, A. Das Sarma, V. Ganti, and R. Kaushik, "Leveraging aggregate constraints for deduplication," in *SIGMOD*, 2007.