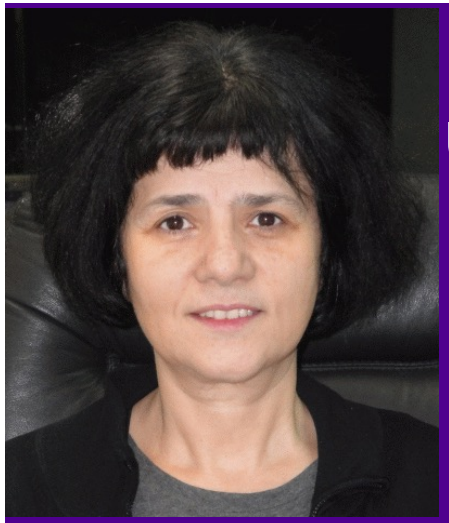
University of Ioannina

Universität der Bundeswehr München

Tampere University

Fairness and Explainability in AI

Models, Measures, and Mitigation Strategies



Evaggelia Pitoura

University of Ioannina & Archimedes/ Athena Research Center

Panayiotis Tsaparas

University of Ioannina & Archimedes/ Athena Research Center

Eirini Ntoutsi

University of the Bundeswehr

Kostas Stefanidis

Tampere University

Fairness and Explainability in AI

Lecture 1 - Bias and discrimination in AI systems: Sources of bias, definitions and models of fairness

Lecture 2 - Bias mitigation

Lecture 3. Solutions for mitigating unfairness in concrete contexts

- Fairness in rankings, recommendations, entity resolution, graphs

Lecture 4 - Explainable AI: Models and methods

Lecture 5 - Connections between fairness and explanations

Solutions for Mitigating Unfairness in Concrete Contexts

- Fairness in rankings
- Fairness in recommender systems
- Fairness in rank aggregation
- Fairness in entity resolution
- Fairness in networks

Fairness in Rankings

Abstractly, a fair ranking is one where the assignment of entities to positions is not unjustifiably influenced by the values of their protected attributes

Rankings

In many applications, the output is a *ranked list*

Items are ordered in descending order of some score, i.e., measure of the relative quality of the items, e.g., relevance to query

- E.g., Web search, job search applications, news feeds, recommendations, etc.

Formally, given a set items $\{i_1, i_2, \ldots i_N\}$, a *ranking* is *an assignment (mapping) of items to ranking positions*

Or we may have *pairs*

- E.g., $(x299, x78)$: $x299\ is\ more\ relevant\ than\ x78$

Rank	ID	Score
1	x299	0.98
2	x78	0.97
3	x45	0.97
4	x329	0.95
5	x23	0.92
6	x981	0.90
7	x665	0.88
8	x724	0.85
9	x87	0.84
10	x232	0.81
	.	
	.	
	.	

What is fairness? [DHP+12]

Individual-based fairness:

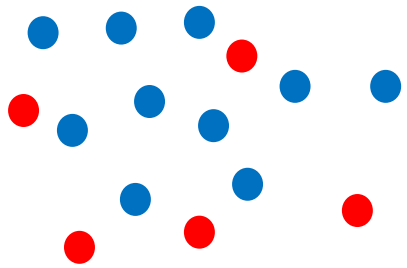
- Similar items should be treated similarly

Group-based fairness:

- Items partitioned into groups based on the value of one, or more of their protected attributes - All groups should be treated similarly

Two groups

- G^+ : Protected (minority) group
- G^- : Non protected (privileged) group



Fairness in ranking

Position bias: People tend to “see” only few top results

Rank	ID	Score
1	x299	0.98
2	x78	0.97
3	x45	0.97
4	x329	0.95
5	x23	0.92
6	x981	0.90
7	x665	0.88
8	x724	0.85
9	x87	0.84
10	x232	0.81
	.	
	.	
	.	

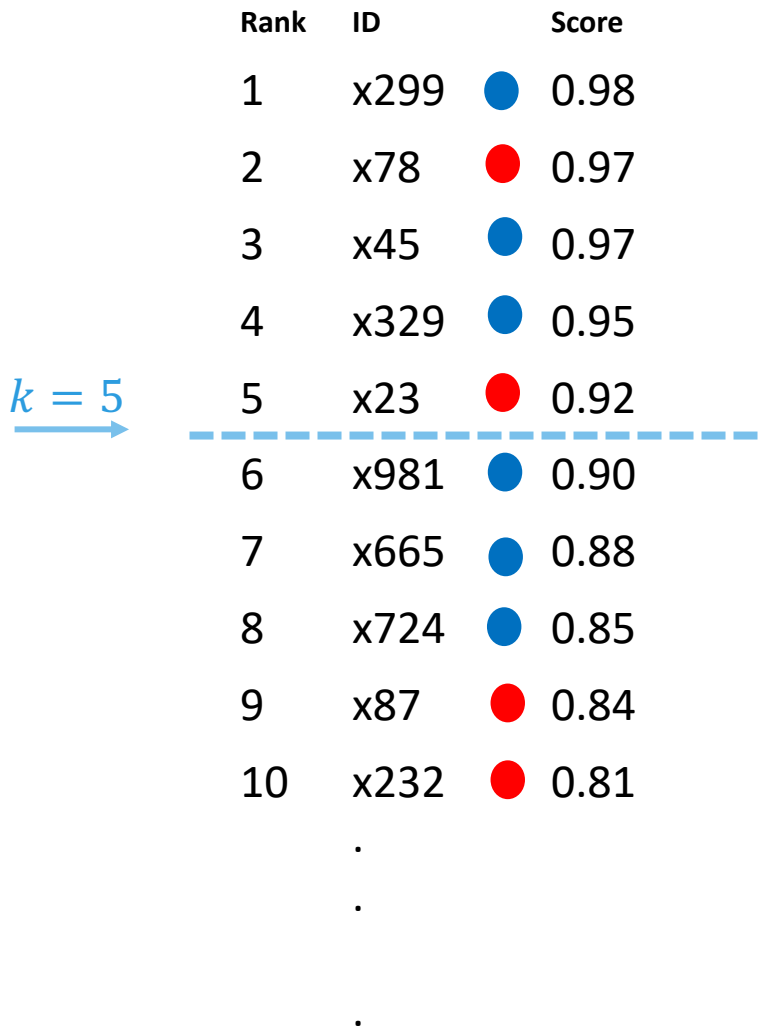
Fairness in ranking (in a nutshell):

- *Individual*: Items with similar relevance scores should receive similar “visibility”
- *Group*: The groups should receive similar “visibility”

Let us see a few example definitions

Fairness constraints [CSV18]

Fairness constraints: Given protected attributes, *an upper bound* U_{lk} and *a lower bound* L_{lk} on the number of items with attribute value *l* that can appear in the top *k* positions of the ranking



$U_{blue\ 5}$ = *3: At most 3 items with property blue in the top-5 positions*

$L_{red\ 5}$ = *1: At least 1 item with property red in the top-5 positions*

Discounted cumulative fairness [YS17]

Metrics are inspired by *Discounted Cumulative Gain (DCG)* commonly used to evaluate the quality of a ranking in information retrieval

$DCG_p(r)$: accumulate scores up to position *p with a logarithmic discount*

$$DCG_p(r) = rel_1 + \sum_{i=2}^{p} \frac{rel_i}{log_2(i+1)}$$

$$DGC_5(\mathrm{r}) = 0.98 + \frac{0.97}{log_2(3)} + \frac{0.97}{log_2(4)} + \frac{0.95}{log_2(5)} + \frac{0.92}{log_2(6)}$$

Normalized DCG (NDGC) $\quad NDCG_p(r) = \frac{DCG_p(r)}{opt\_DCG_p}$

Rank	ID	Score
1	x299	0.98
2	x78	0.97
3	x45	0.97
4	x329	0.95
$k = 5$ → 5	x23	0.92
6	x981	0.90
7	x665	0.88
8	x724	0.85
9	x87	0.84
10	x232	0.81
.		
.		
.		

Discounted cumulative fairness

Normalized discounted difference (rND)

Accumulate the number of items belonging to the protected group at discrete positions in the ranking (e.g., p = 5, 10, …) and discount these numbers accordingly *(Better to have many protected items in higher positions)*

$$rND(r) = \frac{1}{opt\_rND} \sum_{p=5,10,..}^{N} \frac{1}{log_2(p)} \left| \frac{|G^+_{1,..p}|}{p} - \frac{|G^+|}{N} \right|$$

$$rND(r) = \frac{1}{log_2(5)} \left| \frac{2}{5} - \frac{4}{10} \right| + \frac{1}{log_2(10)} \left| \frac{4}{10} - \frac{4}{10} \right|$$

	Rank	ID		Score
	1	x299	●	0.98
	2	x78	●	0.97
	3	x45	●	0.97
	4	x329	●	0.95
$k = 5$ →	5	x23	●	0.92
	6	x981	●	0.90
	7	x665	●	0.88
	8	x724	●	0.85
	9	x87	●	0.84
$k = 10$ →	10	x232	●	0.81
		.		
		.		

Normalized discounted KL divergence (rKL)

Use KL-divergence to compute the expectation of the difference between *the membership probability distribution of the protected group at discrete top-p positions* (for p =.5, 10, ..) and in the *over-all population*

Fairness of exposure

Counting items at discrete positions does not fully capture the fact that:

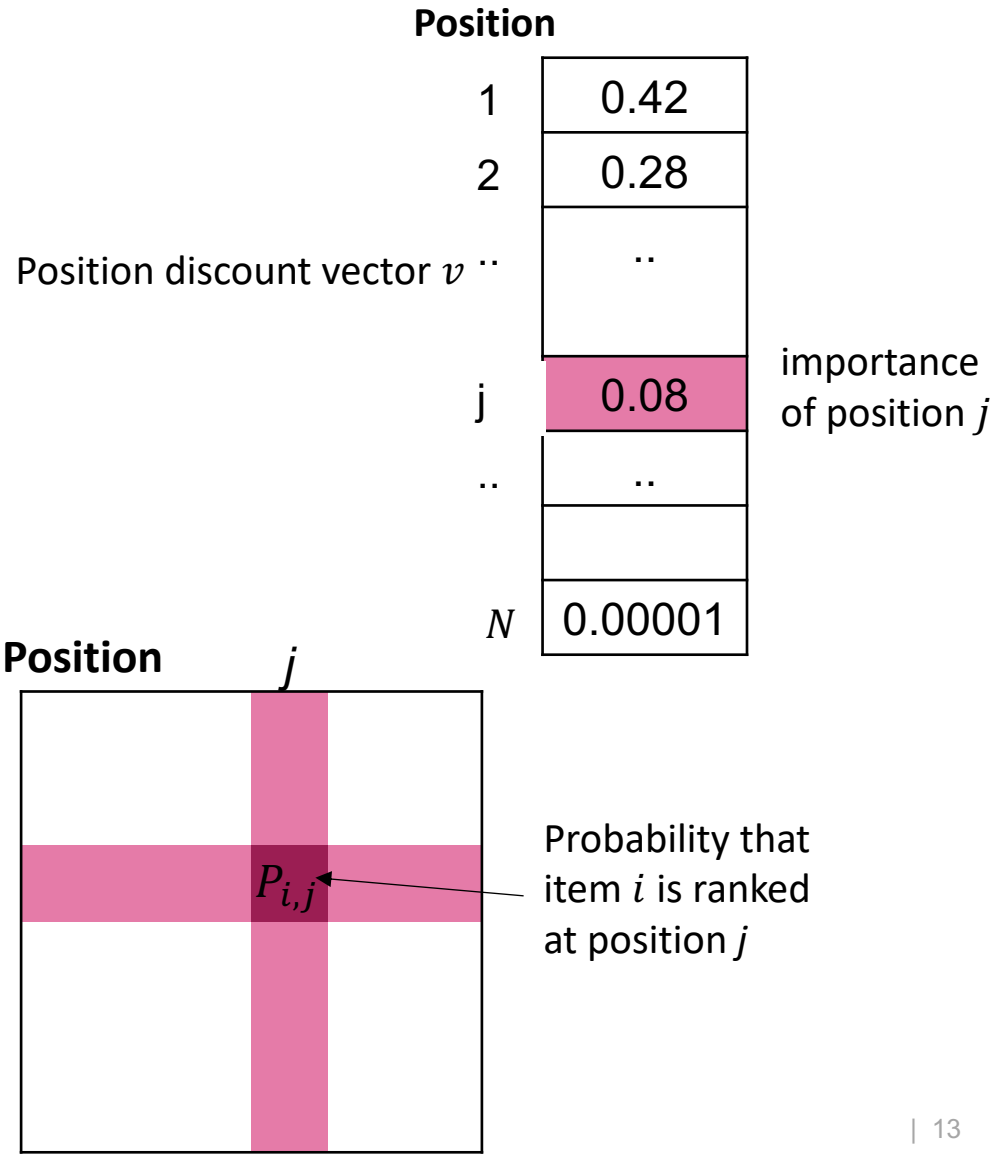
- *Small differences in relevance scores may* translate into *large differences in visibility/exposure* for different groups because of *position bias* that results in a large skew in the distribution of exposure

Fairness of exposure [SJ18]

Position discount vector v to capture position bias

- v_j represents the importance of position j (i.e., the fraction of users that examine an item at position j)

Probabilistic ranking of *N* items in N positions modeled as *a doubly stochastic NxN matrix* P, where $P_{i,j}$ is the probability that item i is ranked at position j

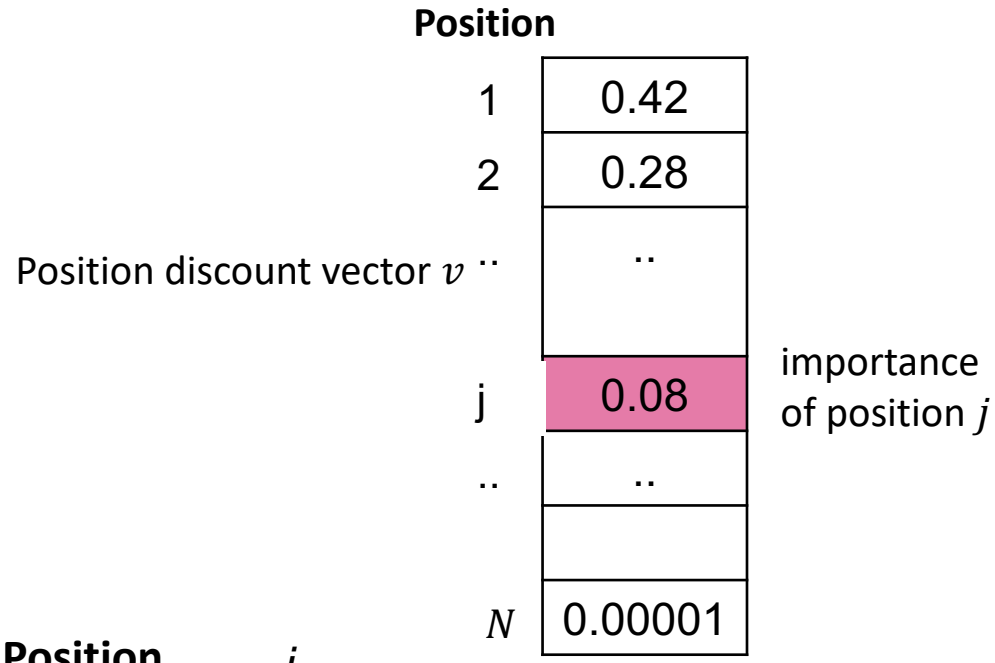


Probabilistic ranking matrix P

Fairness of exposure

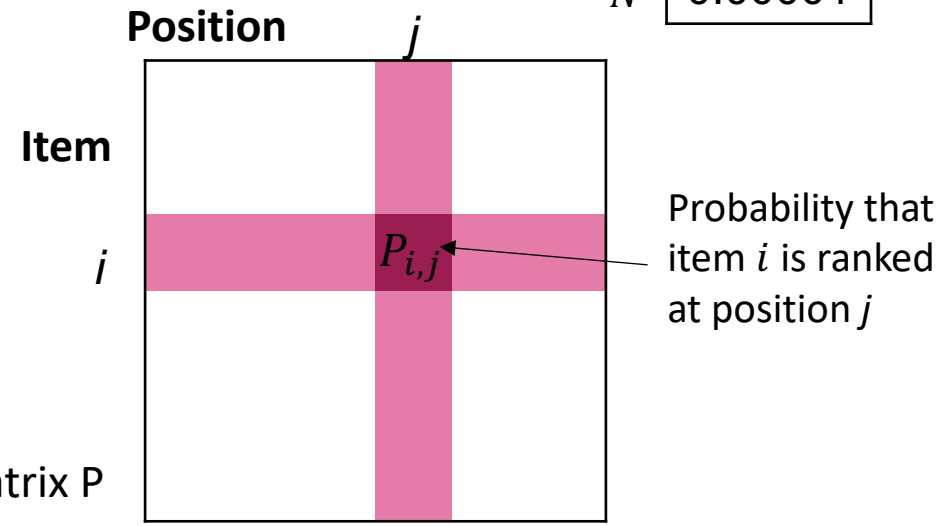
Item exposure

$$Exposure(i|P) = \sum_{j=1}^{N} P_{i,j}\, v_j$$



Group G_k exposure

$$Exposure(G_k|P) = \frac{1}{|G_k|} \sum_{i \in G_k} Exposure(i|P)$$



Probabilistic ranking matrix P

Fairness of exposure

- Demographic parity: the two groups get the same average exposure

$$\frac{Exposure(G^+|P)}{Exposure(G^-|P)} = 1$$

- Disparate treatment: the exposures for the two groups are proportional to their *average* *utility*

$$\frac{Exposure(G^+|P)}{Utility(G^+|q)} = \frac{Exposure(G^-|P)}{Utility(G^-|q)}$$

- Disparate impact: the *impact* (clickthrough rate (CTR) which depends on exposure and relevance) for the two groups are proportional to their average utility

$$\frac{CTR(G^+|P)}{Utility(G^+|q)} = \frac{CTR(G^-|P)}{Utility(G^-|q)}$$

Equity of attention [BGW18]

An idea similar to fairness of exposure but for *individual items*

Equity of attention: each item $\boldsymbol{i}$ receives attention $\boldsymbol{a_i}$ (e.g., exposure, views, clicks) that is proportional to its relevance $\boldsymbol{rel_i}$ for a given query

$$\frac{a_1}{rel_1} = \frac{a_2}{rel_2} \; \forall \; i_1, i_2$$

Unlikely to be satisfied in *any single ranking:* If multiple items have the same relevance score, yet obviously cannot occupy the *same ranking position*

Idea: Consider a sequence $\rho^1, \rho^2, \ldots \rho^m$ of rankings and ask that an item receives cumulative attention proportional to its cumulative relevance

Equity of amortized attention

Consider a sequence $\rho^1, \rho^2, \ldots \rho^m$ of rankings

Equity of amortized attention:

A sequence $\rho^1, \rho^2, \ldots \rho^m$ of rankings offers amortized equity of attention if each item receives *cumulative attention* proportional to its cumulative relevance, i.e.:

$$\frac{\sum_{l=1}^{m} a_1^l}{\sum_{l=1}^{m} rel_1^l} = \frac{\sum_{l=1}^{m} a_2^l}{\sum_{l=1}^{m} rel_2^l} \forall\ i_1, i_2$$

Allow to permute individual rankings so as to satisfy fairness requirements over time

Achieving Fairness in Rankings

Methods for achieving fairness can be distinguished as:

Pre-processing: Transform the data so that any underlying bias or discrimination is removed

In-processing: modify existing or introduce new algorithms that result in fair rankings

Post-processing: treat the algorithms for producing rankings as black boxes and modify their output to ensure fairness

Learning to rank
Linear ranking function

Learning to rank algorithms

- Learning to rank obtains a ranking function *f* that is learned by solving a minimization problem with respect to a *loss function* which most often is a measure of accuracy with respect to the training data

- Training data may be pair of items, item-scores, ranked lists

General approach: Extend the loss function by adding an *extra term* to ensure fairness

Extending the loss function in learning to rank

The DELTR approach [ZDC20]

Extends the ListNet learning to rank framework

- List-wise
- Training set: A query q and a *list of documents* ordered by their relevance to q
- Learn a ranking function f that minimizes a loss function L_{LN} that measures the extent to which the ordering $\hat{r}$ of documents induced by f for a query differs from the ordering r in which the documents appear in the training set for this query.

unfairness term

$$L_{DELTR}\big(r(q),\hat{r}(q)\big) = L_{LN}\big(r(q),\hat{r}(q)\big) + \gamma\, \mathrm{F}(\hat{r}(q))$$

γ depends on desired trade-offs between ranking utility and fairness

As a measurement of fairness democratic parity based on exposure is used

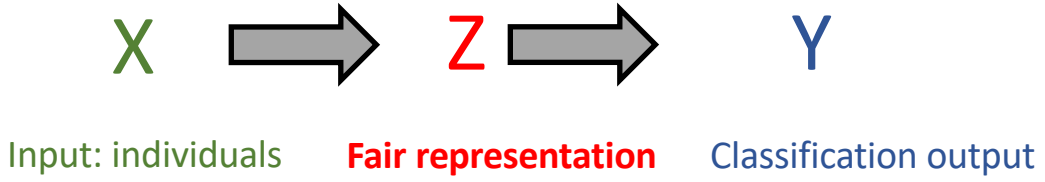
$$F\big(r(q)\big) = \max\big(0, exposure\big(G_0 \big| P_{\hat{r}(q)}\big) - exposure\big(G_1 \big| P_{\hat{r}(q)}\big)\big)^2$$

Squared hinge loss: a differentiable loss function that prefers rankings in which the exposure of the protected group is not less than the exposure of the non protected group but not vice versa

Learning fair representations

Extend learning algorithm for fair classification [ZWS+13]

- Basic idea: Introduce an intermediate level Z between the input space X that represents individuals and the output space Y that represents classification outcomes



Z: fair representation of X

- best encodes X and
- obfuscates any information about membership in the protected group

Z is a multinomial random variable of size k where each of the k values represents *a prototype (cluster)* in the space of X.

Learning fair representations

A learning system that minimizes the loss function

Quality of the encoding — Fairness — Accuracy

$$L = A_x L_x + A_z L_z + A_y L_y$$

Distance from points in X to their representation in Z should be small

Statistical parity

Prediction based on the representation should be accurate

Input: individuals — **Fair representation** — Classification output

A_x, A_z , A_y hyper-parameters that control the trade-off among the three objectives

Statistical parity $P(z = k|x \in G^+) = P(z = k|\, x \in G^-)\ \forall\ k$

The probability that a random element that belongs to the protected group of X maps to a particular prototype of Z is equal to the probability that a random element that belongs to the non-protected group of X maps to the same prototype

Learning fair representations

Modify the loss function to work for ranking [YS17]

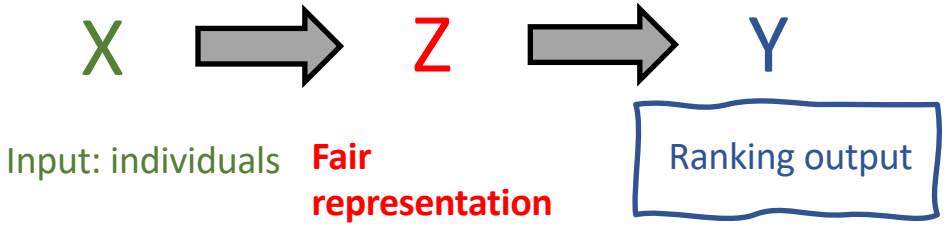
Quality of the encoding — Fairness — Ranking accuracy

$$L = A_x L_x + A_z L_z + A_y L_y$$

Distance from points in X to their representation in Z should be small

Statistical parity

Distance between the ground truth ranking and the estimated ranking should be small



Distance used:

average per-item score difference between the ground truth ranking and the estimated ranking

Other:

Position accuracy (per-item rank difference)

Kendall-τ distance

Spearman and Pearson's correlation coefficients

Adjusting the weights in ranking functions [AJS19]

For each item *i*, d scoring attributes {i[1], i[2], …, i[d]}
Linear ranking functions that use a weigh vector w = $\{w_1, w_2, \ldots, w_d\}$ to compute a utility (goodness) score for each item

$$f(i) = \sum_{j=1}^{d} w_j i[j]$$

Given a function f with weights w = $\{w_1, w_2, \ldots, w_d\}$, find a function f^* with weight vector $w^* = \{w_1^*, w_2^*, \ldots, w_d^*\}$ s.t.
$\cos(w, w^*)$ is minimized, and
f^* is fair

Post-processing

Generative process
Constraint optimization

Generative process

Input: a ranking and a fairness parameter f, $0 \leq f \leq 1$, that specifies

---

Start with an empty list
For each position j in the new ranking, perform a Bernoulli trial with probability *f*
    If the trial *succeeds*,
        the best available item *from the protected group* is selected;
    else,
        the best available item from the *non-protected group* is selected.

$f = 1$	All items in the protected group precede all items in the non-protected group
$f = 0$	All items in the non-protected group precede all items in the protected group
$f > 0.5$	Items in the protected group are preferred over items in the non-protected group
$f < 0.5$	All items in the non-protected group are preferred over items in the protected group

Generative process

Start with an empty list
For each position j in the new ranking, perform a Bernoulli trial with probability *f*
    If the trial *succeeds*,
        the best available item *from the protected group* is selected;
    else,
        the best available item from the *non-protected group* is selected.

Property: the relative order of two items that belong to the same group is not changed

Rank	ID	Group	Score
1	x299		0.56
2	x78		0.55
3	x45		0.45
4	x329		0.44
5	x23		0.44
6	x981		0.25
7	x665		0.23
8	x724		0.18
9	x87		0.16
10	x232		0.15

f = 1

Rank	ID	Group	Score
1	x78		0.55
2	x23		0.44
3	x87		0.16
4	x232		0.15
5	x299		0.56
6	x45		0.45
7	x329		0.44
8	x981		0.25
9	x665		0.23
10	x724		0.18

Rank	ID	Group	Score
1	x78		0.55
2	x299		0.56
3	x23		0.44
4	x45		0.45
5	x87		0.16
6	x329		0.44
7	x232		0.15
8	x981		0.25
9	x665		0.23
10	x724		0.18

f > 0.5

Generative process

Fair* presents a statistical test for this generative model that given a ranking determines the probability that the ranking was generated by the model [ZBC+17]:

- Given that at a specific position we have seen a specific number of items from each group, a one-tailed Binomial test is used to compare the null hypotheses that *the ranking was generated using the model with parameter $f* = f$, or with $f* < f$,* which would mean that the protected group is represented less than desired.

Constraint optimization problem

Many variants

Given a query q, a utility definition $U(r|q)$ of a ranking r and a fair ranking definition, find ranking r that

$$r = argmax_r U(r|q)$$
$$\text{s.t. } r \text{ is fair}$$

If unfairness measure instead of condition

Given a query q, a utility definition $U(r|q)$ of a ranking r and a fair ranking measure F, produce a ranking $\hat{r}$ such that that:

$$\hat{r} = argmax_{\hat{r}} F(\hat{r}|q)$$
$$\text{s.t. distance}(U(\hat{r}|q), U(r,q)) \leq \theta)$$

Constraint optimization (amortized fairness [BGW18])

Amortized individual fairness

Offline version

Given a ranking sequence $\rho^1, \rho^2, \ldots \rho^m$, produce a ranking sequence $\rho^{1*}, \rho^{2*}, \ldots \rho^{m*}$ so as to *minimize unfairness* subject to a *constraint in utility (quality) loss*

$$\text{minimize} \quad \sum_{i=1}^{N} |A_i - Rel_i|$$

$$\text{subject to} \quad \frac{NDCG(\rho^j)}{NDCG(\rho^{j*})} \geq \theta \quad \forall\, j$$

Constraint optimization

Online version

Given the ranking sequence $\rho^1, \rho^2, \ldots \rho^{l-1}$, seen so far, reorder the current ranking ρ^l so as to *minimize the unfairness seen so far* subject to a *constraint in utility (quality) loss of the current ranking*

$$\text{minimize} \quad \sum_{i=1}^{N} \left| \left(A_i^{l-1} + a_i^l\right) - \left(Rel_i^{l-1} + rel_i^l\right) \right|$$

$$\text{subject to} \quad \frac{NDCG(\rho^l)}{NDCG(\rho^{l*})} \geq \theta$$

Use *Integer Linear Programming (ILP)* to solve the online optimization problem:
Introduce N^2 decision variables $X_{i,j}$ set to 1 if item i is assigned to the ranking position j, and 0 otherwise.

Fairness in rankings: Summary

Approaches depend both on the
- Definition of fairness
- Ranking algorithm

In-processing

Learning to rank
- Extent the objective function
- Introduce fair representations

Linear preference functions
- Adjust the weights

Post-processing

Generative process

Constraint optimization problem

References - Rankings

[DHP+12] Cynthia Dwork, Moritz Hardt, Toniann Pitassi, Omer Reingold, Richard S. Zemel: *Fairness through awareness*. ITCS 2012: 214-226

[CSV18] L. Elisa Celis, Damian Straszak, Nisheeth K. Vishnoi: *Ranking with Fairness Constraints*. ICALP 2018: 28:1-28:15

[YS17] Ke Yang, Julia Stoyanovich: *Measuring Fairness in Ranked Outputs*. SSDBM 2017: 22:1-22:6

[SJ18] Ashudeep Singh, Thorsten Joachims: *Fairness of Exposure in Rankings*. KDD 2018: 2219-2228

[BGW18] Asia J. Biega, Krishna P. Gummadi, Gerhard Weikum: *Equity of Attention: Amortizing Individual Fairness in Rankings*. SIGIR 2018: 405-414

[ZDC20] Meike Zehlike, Gina-Theresa Diehn, Carlos Castillo, *Reducing Disparate Exposure in Ranking: A Learning to Rank Approach*, WWW 2020

[ZWS+13] Richard S. Zemel, Yu Wu, Kevin Swersky, Toniann Pitassi, Cynthia Dwork: *Learning Fair Representations*. ICML (3) 2013: 325-333

[AJS+19] Abolfazl Asudeh, H. V. Jagadish, Julia Stoyanovich, Gautam Das: *Designing Fair Ranking Schemes*. SIGMOD Conference 2019: 1259-1276

[ZBC+17] Meike Zehlike, Francesco Bonchi, Carlos Castillo, Sara Hajian, Mohamed Megahed, Ricardo Baeza-Yates: *FA*IR: A Fair Top-k Ranking Algorithm*. CIKM 2017: 1569-1578

Fairness in Recommender Systems

In abstract terms, a recommendation is fair, if the values of the protected attributes of the users, or, the items, do not affect the outcome of the recommendation

Recommender Systems

Recommender systems aim at suggesting to users items of potential interest to them

Two main steps:

- Estimate a rating for each item and user
- Recommend to the user the item(s) with the highest rating(s)

Recommender Systems: Overall

Recommender systems retrieve interesting items for users based on their profiles and their history

- Depending on the application and the recommender, history may include explicit user ratings of items, or, selection of items (e.g., views, clicks)

In general:

- Recommenders estimate a score s(u, i) for a user u and an item i that reflects the preference of u for i, or, in other words, the relevance of i for u
- Then, a recommendation list is formed for u that includes the items having the highest estimated score for u
  - These scores can be seen as the utility scores in the case of recommenders

Fairness in Recommender Systems

In abstract terms, a recommendation is fair, if the values of the protected attributes of the users, or, the items, do not affect the outcome of the recommendation

Recommendations for different stakeholders:

- Consumers of recommendations
  - Recommenders care only for consumers fairness
    - A credit card company recommending consumer credit offers - No producer-side fairness issues since the products are coming from the same bank
- Providers/producers of data items to be recommended
- *System owners*
- *Regulators/auditors*
  - Decision making for data scientists, ML researchers, policymakers and governmental auditors

Stakeholders have a varying level of familiarity and expertise with the system and the underlying technologies

Providers/producers of data items to be recommended

- Fairness needs to be preserved for the providers only

Example:
Interest in ensuring market diversity and avoiding monopoly domination

- Online craft marketplace Etsy: the system wishes to ensure that new entrants to the market get a reasonable share of recommendations even though they have fewer shoppers than established vendors

<u>Consumers vs Producers fairness</u>:

Producers fairness is passive - Producers do not seek out recommendation opportunities but rather wait for users to come to the system and request recommendations

Can a recommender requires fairness for both consumers and providers?

Consider any domain in which both consumers and providers can belong to protected groups

- A rental property recommender
  - The recommender may treat minority applicants as a protected class and wish to ensure that they are recommended properties similar to white renters
  - The recommender may wish to treat minority landlords as a protected class and ensure that highly-qualified tenants are referred to them at the same rate as to white landlords
- Employment scenario

Ensuring Fairness in Recommenders

Fairness methods: Methods for achieving fairness in rankings and recommendations can be distinguished between:

- *Pre-processing*
  - Target at transforming the data so that any underlying bias or discrimination is removed
- *In-processing*
  - Target at modifying existing or introducing new algorithms that result in fair recommendations, e.g., by removing bias
- *Post-processing*
  - Treat the algorithms for producing recommendations as black boxes
  - To ensure fairness, modify the output of the algorithm

In-processing methods design fairness-aware algorithms, that is, algorithms that produce fair recommendations. E.g.:

- Use matrix factorization [YH17]
- Alter the objective of the algorithm to emphasize fairness, typically by adding regularization [KA+18, KA+18b]
- Incorporate randomness in variational autoencoders recommenders [BS19]

Recommendation in education in science, technology, engineering, and mathematics topics - STEM

- 2010 - Women accounted for only 18% of the bachelor's degrees awarded in CS
- The under representation of women causes historical rating data of CS courses to be dominated by men
- The learned model may underestimate women's preferences and be biased toward men
- If the ratings provided by students accurately reflect their true preferences, the bias in which ratings are reported leads to unfairness

Two forms of underrepresentation

- Population imbalance: different types of users occur in the dataset with different frequencies
  - Significantly fewer women succeed in STEM than those who do not; however more men succeed in STEM than those who do not

- Observation bias: certain types of users may have different tendencies to rate different types of items
  - Women are rarely recommended to take STEM courses, there may be significantly less training data about women in STEM courses

Value unfairness: Count inconsistency in estimation errors across the user types

- When one class of users is given higher or lower predictions than their true preferences
  - Male students are recommended STEM courses when they are not interested in STEM, while female students not being recommended even if they are interested

Absolute unfairness: Count inconsistency in absolute estimation error across user types

- A single statistic representing the quality of prediction for each user type
  - If female students are given predictions 0.5 points below their true preferences and male students are given predictions 0.5 points above their true preferences, there is no absolute unfairness
    - One type of user has the unfair advantage of good recommendation, while the other user type has poor recommendation

Underestimation unfairness: Count inconsistency in how much the predictions underestimate the true ratings

- Missing recommendations are more critical than extra recommendations
  - A top student is not recommended to explore a topic he/she would excel in

Overestimation unfairness: Count inconsistency in how much the predictions overestimate the true ratings

- Users may be overwhelmed by recommendations, so providing too many recommendations would be especially detrimental → big evaluation time

Non-parity unfairness: Count the absolute difference between the overall average ratings of disadvantaged users and those of advantaged users

Traditionally, the matrix-factorization targets at minimizing a regularized, squared reconstruction error

The above fairness metrics are used to augment the learning objective of MF, by helping reducing discontinuities in the objective, making optimization more efficient

Random variables X for users, Y for items and R for recommendation outcomes

Standard recommendations

In addition: sensitive feature S, i.e., information to be ignored in the recommendation process (e.g., user's gender, or item's popularity)

Standard Recommendations	→	Independence-enhanced recommendations
Dataset: D = {(xi, yi, ri)}	→	Dataset: D = {(xi, yi, ri, si)}
Prediction function: r(x, y)	→	Prediction function: r(x, y, s)

The goal is to achieve: Recommendation (or statistical) independence

- No information about a sensitive feature influences the outcome
- Recommendations are selected so as to satisfy a recommendation independence constraint

Adopting a regularizer imposing a constraint of independence while training a recommendation model

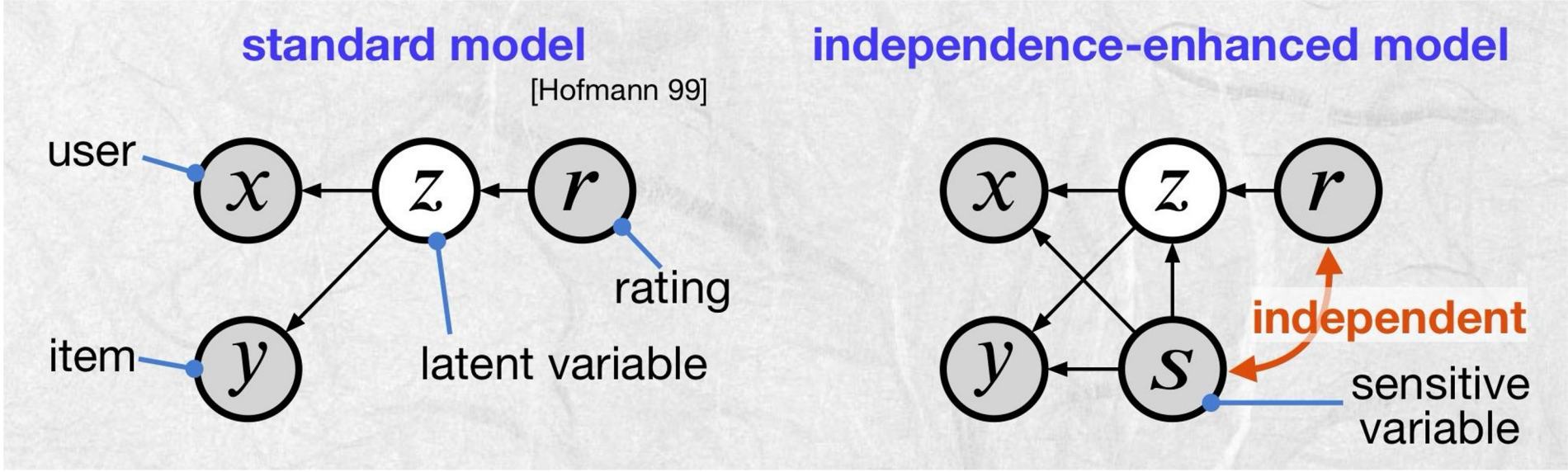
$\Sigma_D \text{loss}(r_i, r(x_i, y_i, s_i)) - \eta \text{ind}(R, S) + \lambda \text{reg}(\Theta)$

- loss: empirical loss
- η: independence parameter - control the balance between independence and accuracy
- ind: independence term - a regularizer to constrain independence
  - The larger value indicates that recommendation outcomes and sensitive values are more independent
- λ: regularization parameter
- Θ: L2 regularizer

Several alternatives for the independence term

The regularizer to constrain independence

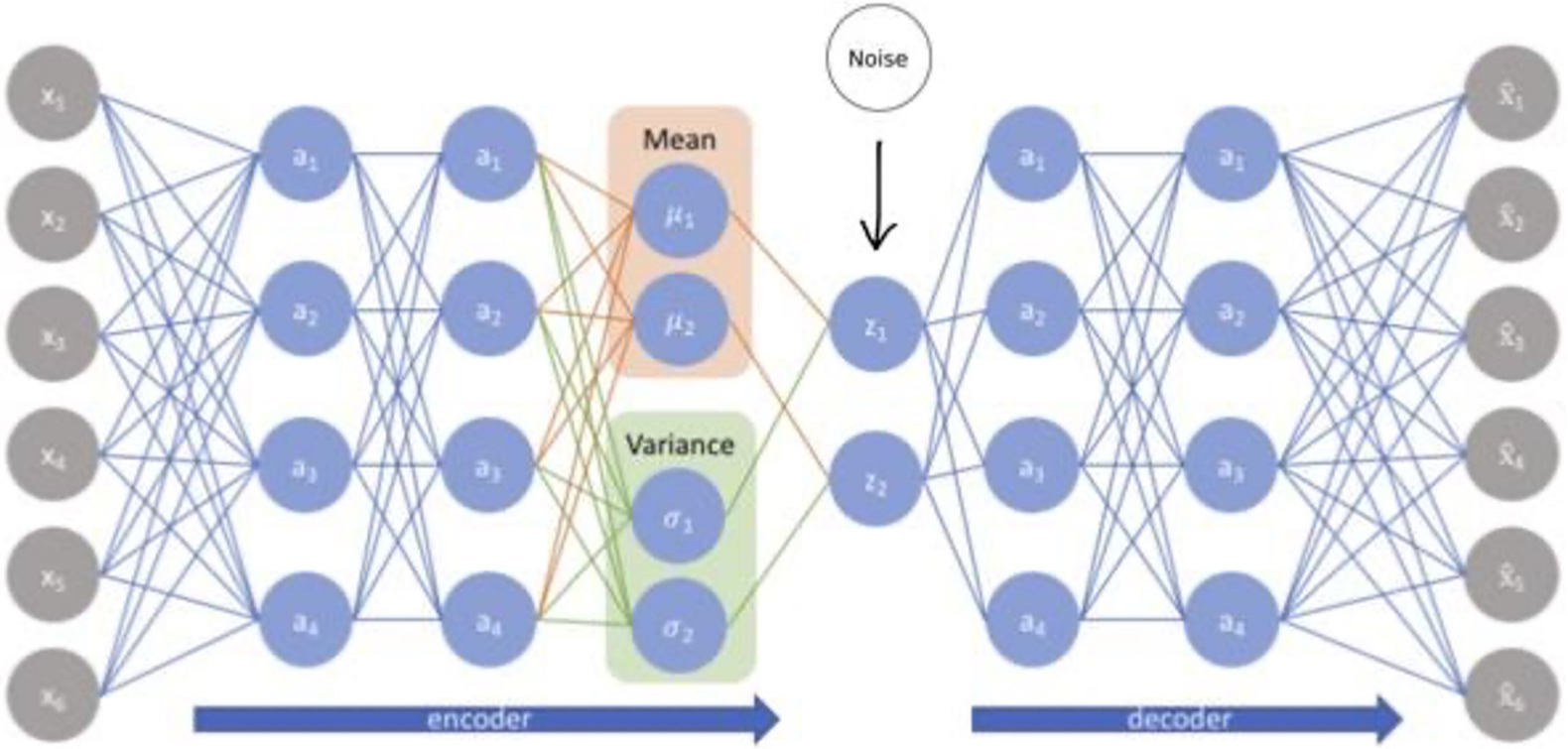
- Mutual information with histogram models
- Mean matching
  - Matching means of predicted ratings for distinct sensitive groups
- Mutual information with normal distributions
- Distribution matching with Bhattacharyya distance



A sensitive variable is added to a recommendation model so that it satisfies an independence constraint

Encoder: The input is mapped to a latent space (normal distributions) through hidden layers

Sampling Phase: Samples are drawn from the the distributions propagate to decoder



Decoder: The estimated output is compared with labels and propagates back

Explore the probability distribution learned in the training phase for varying ranking position in a collaborative manner

Post-processing methods modify the output of the recommender algorithms to ensure fairness:

- Calibrated recommendations

Results are fair if they achieve fair representation

- Results are evenly balanced, reflect population, user historical data

Re-ranking, aka *post-processing*

$I^* = \text{argmax}_I\ (1-\lambda)s(I) - \lambda CKL(p, q(I))$

- λ determines the trade-off between accuracy and calibration
- $s(I)$: the summation of the predicted relevance recommendation scores
- CKL: Kullback-Leibler divergence, i.e., *how similar are p and q?*

Fairness in Rank Aggregation

Fairness in Rank Aggregation

A new problem emerged!

WHEN? A number of ranked outputs is produced, and we need to aggregate these outputs to construct a new ranked consensus output

- Recently, some works study how to mitigate biases introduced during the aggregation phase
  - Mainly, **under the umbrella of group recommendations**, where instead of an individual user requesting recommendations from the system, the request is made by a group of users

Examples: A travel with friends // A movie to watch with the family // Music to be played in a car for the passengers

HOW? Apply a recommendation method to each member individually, and then aggregate the separate lists into one for the group

- Average scores for aggregations are enough?

Fairness in Group Recommendations

Fairness in Group Recommendations

Most works on group recommenders aim to maximize the group's overall satisfaction with the recommended list

This way, there could be one or more users that do not like the items in the list

- By using the average method, the opinion of some users can be lost

Need for fair group recommendations!

Intuitively: fairness attempts to minimize the feeling of dissatisfaction within group members

Assume a measure of quantifying the satisfaction, or utility, of a user (in a group) given a list of recommendations

- How relevant the K recommended items are to the user

Group utility, or social welfare: ways for averaging user utilities

Fairness: the balance of user utilities inside the group, i.e., fairness can be the minimum user utility

- *Intuitively, a list that minimizes the dissatisfaction of any user in the group can be considered as the most fair*

In this sense, fairness enforces the least misery principle among users utilities

Assume a user u in a group g and a set of items I (|I| = K) recommended to g

The individual utility U(u, I) : U×I → [0, 1] of the relevances rel(u, i), where i ∈ I, is defined as:

$$(1)\ \text{Average: } U(u,I) = \frac{1}{K \times rel_{max}} \sum_{i \in I} rel(u,i)$$

$$(2)\ \text{Proportionality: } U(u,I) = \frac{\sum_{i \in I} rel(u,i)}{\sum_{i \in I(u,K)} rel(u,i)}$$

I(u,K) denotes the set of items which are among the top-K favourite items of user u

Aggregate individual utilities as social welfare

The Social Welfare SW(g,I), is the overall utility of all users in g given group recommendations I

$$SW(g,I) = \frac{1}{|g|}\sum_{u \in g} U(u,I), \forall g, I$$

Fairness reflects the comparison between the utilities of users in the group

$$Least\ Misery:\ F_{LM}(g,I) = \min\{U(u,I), \forall u \in g\}$$
$$Variance:\ F_{Var}(g,I) = 1 - Var(\{U(u,I), \forall u \in g\})$$
$$Jain's\ Fairness:\ F_J(g,I) = \frac{(\sum_{u \in g} U(u,I))^2}{|U| \cdot \sum_{u \in g} U(u,I)^2}$$
$$Min-Max\ Ratio:\ F_M(g,I) = \frac{\min\{U(u,I), \forall u \in g\}}{\max\{U(u,I), \forall u \in g\}}$$

Ensuring Fairness

Maximize social welfare and fairness

Use the following scheme to assign weights to each objective:

$$\lambda \cdot SW(g, I) + (1 - \lambda) \cdot F(g, I)$$

Greedy algorithm: Select an item that achieves the highest fairness (above function) when it is added to the current recommendation list

- Time-efficient, because of one item per round

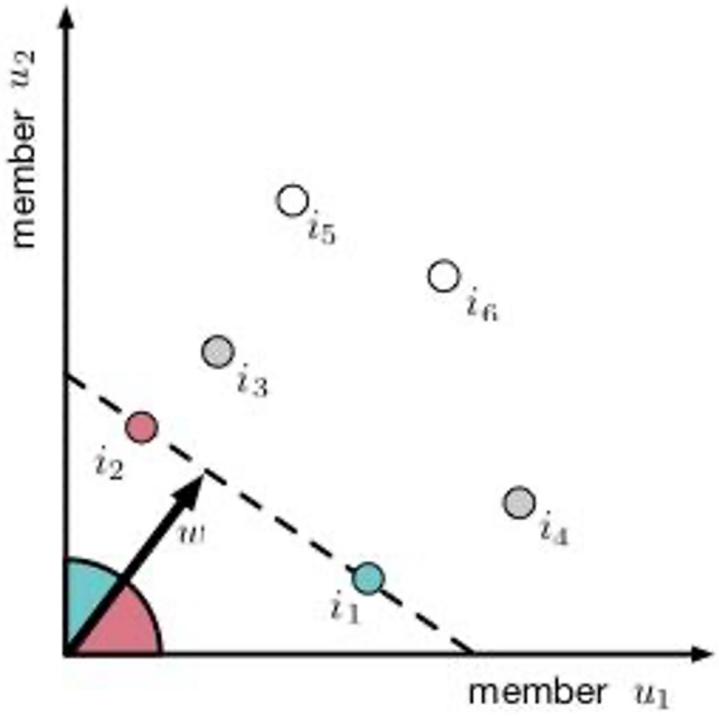
Alternatives via integer programming techniques

Fairness via Pareto

Items in space: each dimension corresponds to a group member u and its coordinate equals the rank rel(u,i) of th item i for u

Top-6 for u1: i2, i3, i5, i1, i6, i4, and for u2: i1, i4, i2, i3, i6

- Item i1 ranks 4th for u2 and 1st for u1, and is thus represented by the point (4,1)
- E.g., i1 is clearly better than another i4



We say that i dominates i′ for a group g, if for each user, item i ranks at least as good as i′, and there exists at least one user for whom i ranks better:

$$\forall u \in g : rel(u,i) \leq rel(u,i'), \text{ and } \exists u' \in g : rel(u',i) < rel(u',i')$$

Fairness via Pareto

The top items not dominated by any other item are called Pareto optimal

- Items i1 and i2 comprise the set of Pareto optimal items in the example

N-level Pareto optimal: contain items dominated by at most N − 1 other items

- Thus, the top-N choices are within the N-level Pareto optimal set
  - E.g., i3 is 2-level Pareto optimal as it is dominated by only i2

Ensuring Fairness

Impractical to identify the exact set of N-level Pareto optimal items

- It needs the ranks of each item to each user

Approximation:

- Request top-N′ recommendations for each user in the group, and take their union
  - N′>N is the largest number of items the system can recommend
- Identify the N-level Pareto optimal items among the N′ ones

Package-to-group recommendations

For a user u and a package P, P is m-proportional to u, if there exist at least m items in P that u likes

For a group g, the m-proportionality of P for g is defined as:

$$|gP| / |g|$$

where gP is the set of users in g for which P is m-proportional

Package-to-group recommendations

A user u in g is envy-free for an item i in P, if rel(u,i) is in the top-Δ% of the preferences in the set {rel(v,i) : v $\in$ g}
A package P is m-envy-free for u, if u is envy-free for at least m items in P

For a group of users g and a package P, the m-envy-freeness of P for g is defined as:

$$|gef| / |g|$$

where gef is the set of users in g for which P is m-envy-free

Ensuring Fairness

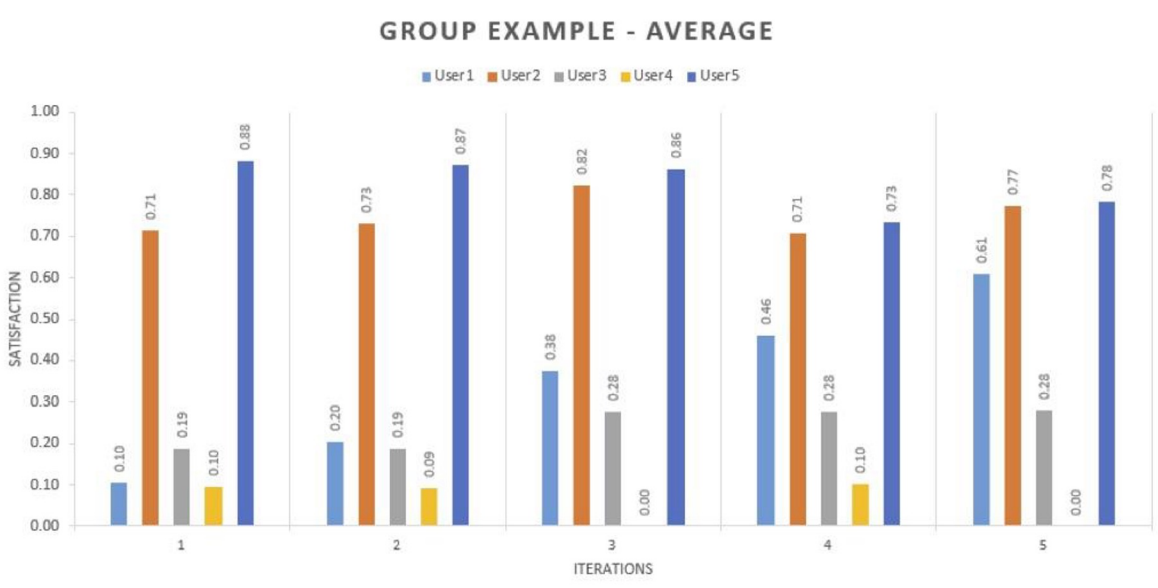
Fairness maximization

Construct P greedily

- In rounds, add to P the item that satisfies the largest number of non-satisfied users
  - Maximize: $fG(P,i) = |SatG(P \cup \{i\}) \setminus SatG(P)|$, at each round
    where SatG(P) denotes the users satisfied by P

- With category constraints: When selecting an item from a specific category, we remove the items of this category from the candidate set
- With distance constraints: Consider as candidate items only the items that when added to the existing solution satisfy the distance constraints

(*Un*)Fairness in Sequential



5 friends // watch a movie // top-10 // 5 iterations

Count satisfaction for each member: *How relevant are the group list's items, over the best items for each group member*

- **User 4** has a low satisfaction score: almost no interesting recommendations

The recommender is unfair to him/her - unfairness continues throughout the 5 iterations

Satisfaction & Disagreements

Satisfaction per iteration: directly compare the user's satisfaction from the group recommendations with the ideal case for that user

- pj(ui,dz): preference score of ui for item dz at iteration j

$$sat(u_i, Gr_j) = \frac{GroupListSat(u_i, Gr_j)}{UserListSat(u_i, A_{u_i,j})}$$

$$GroupListSat(u_i, Gr_j) = \sum_{d_z \in Gr_j} p_j(u_i, d_z)$$

$$UserListSat(u_i, A_{u_i,j}) = \sum_{d_z \in A_{u_i,j}} p_j(u_i, d_z)$$

Average for group satisfaction

Disagreements in the group: difference in the satisfaction scores between the most satisfied and the least satisfied user in the group

Fairness in Sequential Recommendations

Sequential hybrid aggregation method

A weighted combination of the average and minimum aggregations

$$score(G, d_z, j) = (1 - \alpha_j) * avgScore(G, d_z, j) + \alpha_j * leastScore(G, d_z, j)$$

Dynamic α in each iteration

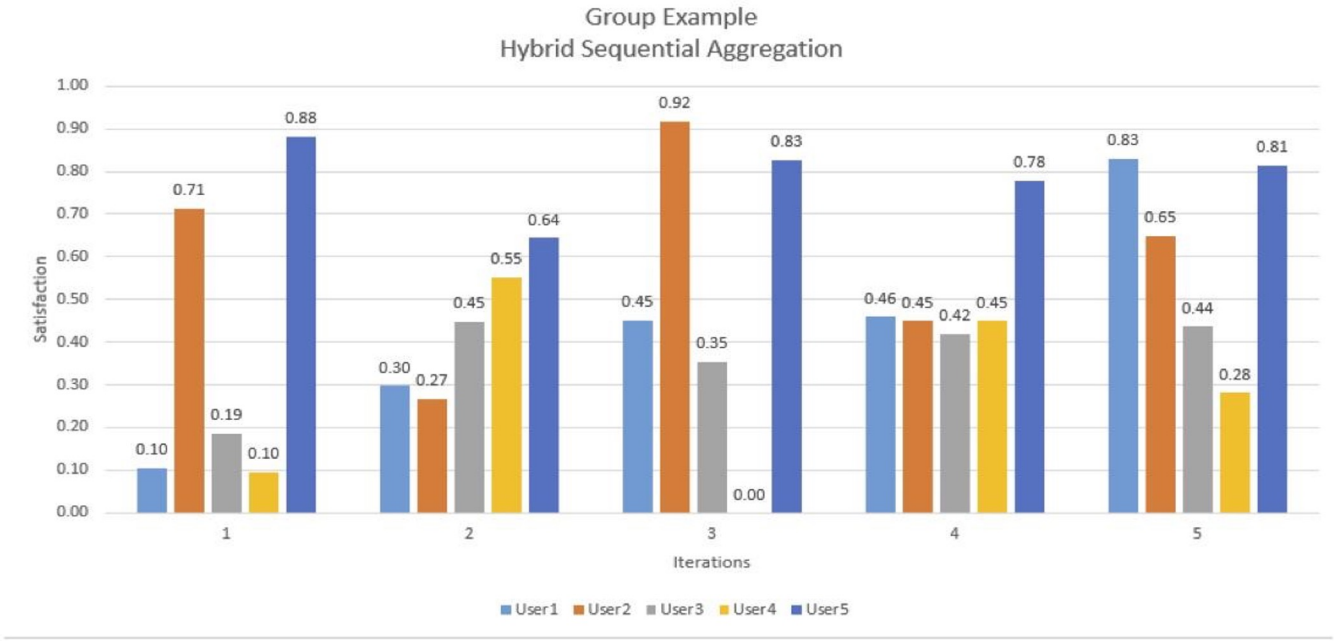
Subtract the minimum satisfaction score of the group members in the previous iteration from the maximum score

$$\alpha_j = max_{u \in G} sat(u, Gr_{j-1}) - min_{u \in G} sat(u, Gr_{j-1})$$

- For an extremely unsatisfied user in a previous iteration
  - α takes a high value and promotes that user's preferences
- For equally satisfied users at the last round
  - α takes low values, use a close to the average aggregation, everyone is treated as an equal

Fairness in Sequential Recommendations

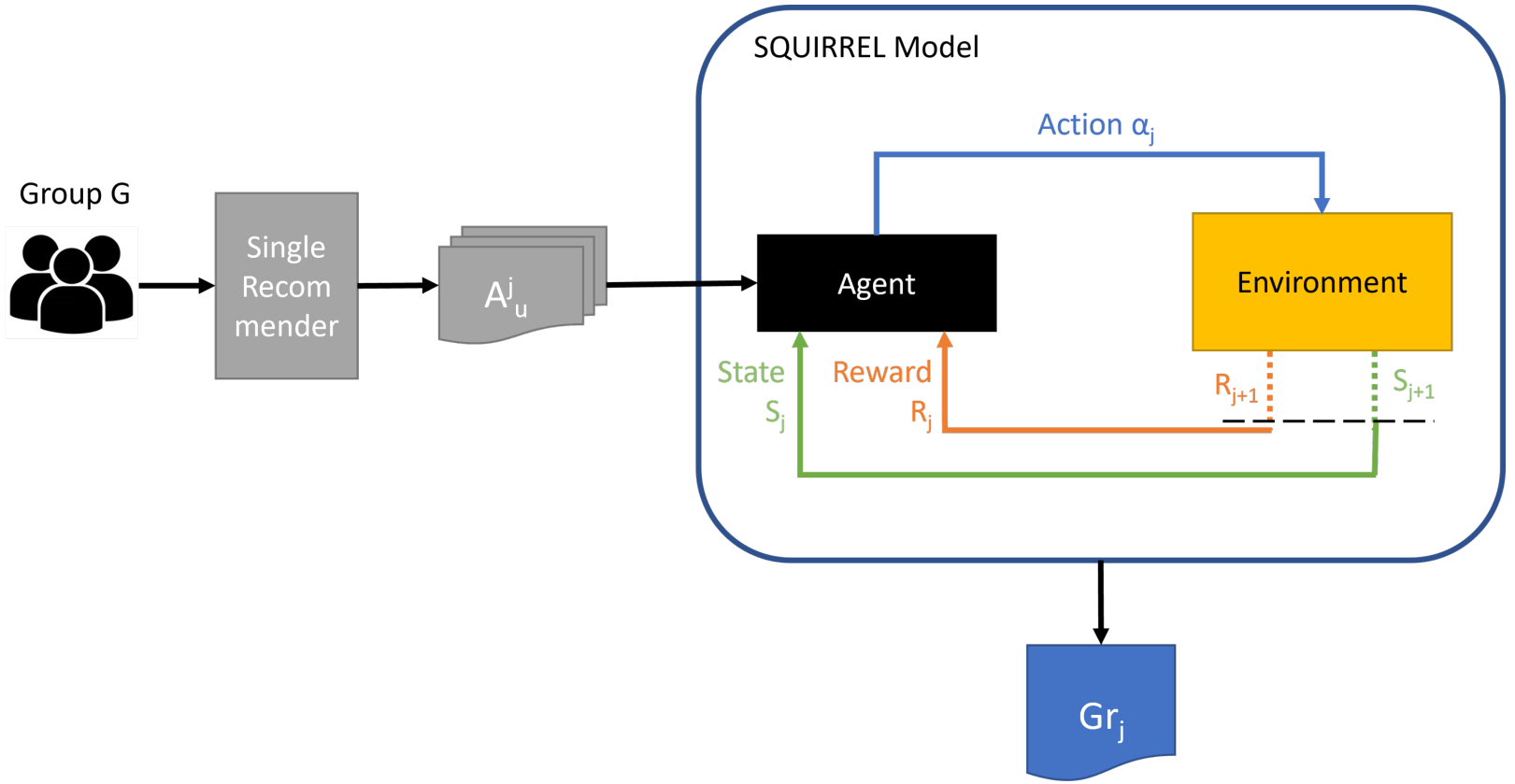
A group member that was not satisfied in the previous iteration, is satisfied in the next



User 4: In the first iteration has a low satisfaction score, and in the second has a higher one

- Improvement over the previous results, where User 4 was always the least satisfied member of the group

The SQUIRREL Model



State	Action	Reward
The overall satisfaction of each group member	Six group recommendation methods	Satisfaction R_s Satisfaction + disagreement R_{sd}

Fairness in Recommenders: Summary

- All existing learning and linear preference functions in-processing approaches target *group* and *producer* fairness
- Most approaches consider a single output - with few exceptions
- Focus on all different options of fairness definitions, namely, *individual* or *group*
- Many approaches treat the algorithms for producing recommendations as black boxes
  - They can lead to unpredictable losses in accuracy

References – Recommender Systems & Rank Aggregation

- [CD+16] L.Elisa Celis, Amit Deshpande, Tarun Kathuria, and Nisheeth K. Vishnoi. 2016. How to be Fair and Diverse? CoRR abs/1610.07183 (2016).
- [SR+19] Babak Salimi, Luke Rodriguez, Bill Howe, and Dan Suciu. 2019. Interventional Fairness: Causal Database Repair for Algorithmic Fairness. In SIGMOD.
- [S18] Harald Steck. 2018. Calibrated recommendations. In RecSys. 154–162.
- [YH17] Sirui Yao and Bert Huang. 2017. Beyond Parity: Fairness Objectives for Collaborative Filtering. In NIPS. 2921–2930.
- [BS19] Rodrigo Borges and Kostas Stefanidis. 2019. Enhancing Long Term Fairness in Recommendations with Variational Autoencoders. In MEDES.
- [XM+17] Lin Xiao, Zhang Min, Zhang Yongfeng, Gu Zhaoquan, Liu Yiqun, and Ma Shaoping. 2017. Fairness-Aware Group Recommendation with Pareto-Efficiency. In RecSys.
- [S19] Dimitris Sacharidis. 2019. Top-N Group Recommendations with Fairness. In SAC.
- [SQ17] Dimitris Serbos, Shuyao Qi, Nikos Mamoulis, Evaggelia Pitoura, and Panayiotis Tsaparas. 2017. Fairness in Package-to-Group Recommendations. In WWW.
- [SN+20] Maria Stratigi, Jyrki Nummenmaa, Evaggelia Pitoura, and Kostas Stefanidis. 2020. Fair Sequential Group Recommendations. In ACM SAC.
- [KA+18b] T. Kamishima, S. Akaho, H. Asoh, and J. Sakuma. 2018. Recommendation independence. In FAT.
- [B17] Robin Burke. 2017. Multisided Fairness for Recommendation. CoRR abs/1707.00093
- [KC11] Faisal Kamiran, Toon Calders. Data preprocessing techniques for classification without discrimination. Knowl. Inf. Syst. 33(1): 1-33 (2011)
- [CW+17] F. du Pin Calmon, D. Wei, B. Vinzamuri, K. N. Ramamurthy, K. R. Varshney. 2017. Optimized Pre-Processing for Discrimination Prevention. In NIPS.
- [ZW+13] R. S. Zemel, Y. Wu, K. Swersky, T. Pitassi, C. Dwork. 2012. Learning Fair Representations. In ICML.
- [RG+19] B. Rastegarpanah, K. P. Gummadi, M. Crovella. 2019. Fighting Fire with Fire: Using Antidote Data to Improve Polarization and Fairness of Recommender Systems. In WSDM.
- [FF+15] M. Feldman, S. A. Friedler, J. Moeller, C. Scheidegger, S. Venkatasubramanian. 2015. Certifying and Removing Disparate Impact. In KDD.

Fairness in Entity Resolution

Or Entity Resolution with Fairness Constraints

Entity Resolution (ER)

ER: Identify entity descriptions from different data sources that refer to the same real-world entity

Improves data quality by reducing

- Data incompleteness (missing values)
  Redundancy (duplicate values)
  Inconsistency (conflicting values)

E

id	Name	Location	Employer	Rep	Sex
e_1	Danny Barber	LA	UCLA	600	M
e_2	Susan Doe	Texas	UT Austin	7,000	F
e_3	Peter Simons	NY	NYU	4	M
e_4	M. Anderson	Denmark	Aarhus Univ.	8	M
e_5	Julia Rondo	France	CNRS, Paris	460	F
e_6	J. Parker	California	UC Berkeley	381	M

E'

id	Full-name	Affiliation	h-index	Sex
e'_1	Doe, S.	UT	14	F
e'_2	J. Parker	UCSC	5	M
e'_3	Simons, Pete	NYU	11	M
e'_4	M. Anderson	Aarhus	15652	M
e'_5	J. Rondo	CNRS	4653	F
e'_6	Juliana Rondo	CNRS	25	F

Traditional ER: Example

E

id	Name	Location	Employer	Rep	Sex
e_1	Danny Barber	LA	UCLA	600	M
e_2	Susan Doe	Texas	UT Austin	7,000	F
e_3	Peter Simons	NY	NYU	4	M
e_4	M. Anderson	Denmark	Aarhus Univ.	8	M
e_5	Julia Rondo	France	CNRS, Paris	460	F
e_6	J. Parker	California	UC Berkeley	381	M

E'

id	Full-name	Affiliation	h-index	Sex
e'_1	Doe, S.	UT	14	F
e'_2	J. Parker	UCSC	5	M
e'_3	Simons, Pete	NYU	11	M
e'_4	M. Anderson	Aarhus	15652	M
e'_5	J. Rondo	CNRS	4653	F
e'_6	Juliana Rondo	CNRS	25	F

desc. score

Traditional ER	R[4]	
e_3-e'_3	e_3-e'_3	3 male
e_4-e'_4	e_4-e'_4	
e_6-e'_2	e_6-e'_2	
e_5-e'_6	e_5-e'_6	1 female
~~e_5-e'_5~~		
~~e_5-e'_1~~		
e_2-e'_1		

Fairness-aware ER: Intuition

The retrieved results should not only be the most likely matches,
but they should also satisfy a given *fairness constraint*

Fairness in ER decisions: equal decision measures that allow us to examine the allocation of benefits and harms across groups by looking at the decision alone

- Group-based fairness: disjoint groups (protected vs non-protected)
  - All groups should receive similar treatment, i.e., have similar chances to be resolved
  - Open question: how do we decide if an entity pair is protected or not?
    - Conjunctive/disjunctive decision? missing values? conflicting values? on-the-fly decisions?

Ranked group fairness: a fairness constraint should be satisfied when considering the results within a given rank position

Fairness-aware ER: Definition

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:

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

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

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

Fairness-aware ER: Example

E

id	Name	Location	Employer	Rep	Sex
e_1	Danny Barber	LA	UCLA	600	M
e_2	Susan Doe	Texas	UT Austin	7,000	F
e_3	Peter Simons	NY	NYU	4	M
e_4	M. Anderson	Denmark	Aarhus Univ.	8	M
e_5	Julia Rondo	France	CNRS, Paris	460	F
e_6	J. Parker	California	UC Berkeley	381	M

E'

id	Full-name	Affiliation	h-index	Sex
e'_1	Doe, S.	UT	14	F
e'_2	J. Parker	UCSC	5	M
e'_3	Simons, Pete	NYU	11	M
e'_4	M. Anderson	Aarhus	15652	M
e'_5	J. Rondo	CNRS	4653	F
e'_6	Juliana Rondo	CNRS	25	F

desc. score ↓

FairER Q_p	FairER Q_n	R[4]	
e_5-e'_6	e_3-e'_3	e_5-e'_6	female
~~e_5-e'_5~~	e_4-e'_4	e_3-e'_3	male
~~e_5-e'_1~~	e_6-e'_2	e_2-e'_1	female
e_2-e'_1	e_1-e'_2	e_4-e'_4	male
e_2-e'_5	~~e_3-e'_2~~		

Fairness in Entity Resolution: Summary

- Fairness-aware ER: A general constraint-based formulation
- Only an instance of this problem is solved
    - Fairness expressed as cardinality constraints of protected and non-protected group members in the output

- More complex protected group criteria to come
- Bias mitigation in other ER tasks (blocking, fusion)
- Impact of alternative fairness measures on ER

Fairness in Networks

E.g., in social nets, nodes correspond to people and edges to connections between them

Group-based setting: nodes belong to groups based on the value of one of their sensitive attributes, e.g., based on their gender or race

Study fairness with respect to *node centrality*, i.e., to whether nodes belonging to different groups hold equally central positions in the network

Fairness in Networks

As nets evolve, biases arise in the degree centrality of nodes belonging to different groups

How to measure node centrality

- Degree of the node, i.e., the number of its neighbours
  - E.g., in a social net, degree centrality considers the number of followers
- Page-rank (PR) centrality of its neighbours
  - E.g., in a social net, the PR centrality of a node considers not only how many followers the node has but also the PR centrality of these followers

Fairness in Networks

PR solution: Assign a weight P(u) to each node u that indicates the significance of u in the net, with the sum of the weights assigned to all nodes being equal to 1

- Protected, or red group R, unprotected, or blue group B
  - P(R) and P(B) denote the total weight that PR assigns to the nodes in the red and blue group

Given a fairness policy expressed with a parameter φ

- There is φ PR fairness, if the red PR is equal to φ
  - E.g., by setting φ = 0.5, we ask that both groups are equally important.
  - E.g., for r be a fraction of the total number of nodes, by setting $\varphi = r$, we ask that the red nodes have a share in the weights proportional to their share in the population [demographic parity]

Fairness in Networks

Personalized PR

- Each node i assigns a weight P$i(u)$ to each node u in the network
- P$i(u)$ indicates the significance that node u has for node i
  - A measure of proximity between source node i and node u
- P$i(R)$ and P$i(B)$ denote the weight that node i allocates to the red and blue groups and ask that P$i(R)$ is equal to φ

Intuitively, fairness of Pi implies that node i weights the red and blue groups fairly

- P$i(R)$ is a measure of how a specific node i weights the red group, while P(R) captures the weight that the network as a whole places on the red group

Link Recommendations for Fair Nets

“Correct” the network so that the PR algorithm produces fair weights

- Recommend people to follow in a social net: Recommend links that if accepted, the fairness of the network will improve

Edge importance:

- The most important edges in terms of fairness are edges that connect nodes whose neighbourhoods are of a “different colour”

References – ER & Networks

- V. Efthymiou, K. Stefanidis, E. Pitoura and V. Christophides. FairER: Entity Resolution With Fairness Constraints. Proceedings of *ACM CIKM* 2021
- C. Dwork, M. Hardt, T. Pitassi, O. Reingold, R. S. Zemel, Fairness through awareness, in: Innovations in Theoretical Computer Science, 2012, pp. 214–226.
- D. F. Gleich, Pagerank beyond the web, SIAM Review 57 (2015) 321–363.
- C. Avin, B. Keller, Z. Lotker, C. Mathieu, D. Peleg, Y. A. Pignolet, Homophily and the glass ceiling effect in social networks, in: ITCS, 2015, pp. 41–50.
- A. Stoica, C. J. Riederer, A. Chaintreau, Algorithmic glass ceiling in social networks: The effects of social recommendations on network diversity, in: WWW, ACM, 2018, pp. 923–932.
- S. Tsioutsiouliklis, E. Pitoura, P. Tsaparas, I. Kleftakis, N. Mamoulis, Fairness-aware pagerank, in: WWW, ACM, 2021, pp. 3815–3826.
- S. Tsioutsiouliklis, E. Pitoura, K. Semertzidis, P. Tsaparas, Link recommendations for pagerank fairness, in: WWW, ACM, 2022, pp. 3541–3551.
- F. Fabbri, F. Bonchi, L. Boratto, C. Castillo, The effect of homophily on disparate visibility of minorities in people recommender systems, in: ICWSM, 2020.
- T. A. Rahman, B. Surma, M. Backes, Y. Zhang, Fairwalk: Towards fair graph embedding, in: IJCAI, 2019, pp. 3289–3295.
- A. Khajehnejad, M. Khajehnejad, M. Babaei, K. P. Gummadi, A. Weller, B. Mirzasoleiman, Crosswalk: Fairness-enhanced node representation learning, in: AAAI, 2022, pp. 11963– 11970.
- D. Liben-Nowell, J. M. Kleinberg, The link prediction problem for social networks, in: CIKM, ACM, 2003, pp. 556–559.
- J. Ali, M. Babaei, A. Chakraborty, B. Mirzasoleiman, K. P. Gummadi, A. Singla, On the fairness of time-critical influence maximization in social networks, in: ICDE, IEEE, 2022, pp. 1541–1542.
- S. Haddadan, C. Menghini, M. Riondato, E. Upfal, Repbublik: Reducing polarized bubble radius with link insertions, in: WSDM, ACM, 2021, pp. 139–147.
- E. Pitoura, Social-minded measures of data quality: Fairness, diversity, and lack of bias, ACM J. Data Inf. Qual. 12 (2020) 12:1–12:8.

Fairness and Explainability in AI

Models, Measures, and Mitigation Strategies

NEXT – Explainable AI: Models and methods