Holistic AI × The Alan Turing Institute

Bias in Recommender Systems Part III

Content by: Sachin Beepath, Giulio Filippi, Cristian Munoz, Roseline Polle, Nigel Kingsman, Sara Zannone

Speaker: Giulio Filippi

Contents

III – Measuring Bias in Recommender Systems

- 1) Introduce metrics for user and item sides of bias.
- 2) Give some examples to help the students understand the metrics.
- 3) Show how we can compute these metrics with the holisticai library

Exposure distribution

– We first define the exposure distribution. The exposure distribution p_A of a group A is defined as

$$p_A(i) = \frac{N_A(i)}{N_A}$$

– Where $N_A(i)$ is the number of times item i is shown to group A and N_A is the total number of items shown to group A.

– The exposure distribution acts as an empirical probability of a group being shown each item.

Exposure distribution (example)

- We have two groups of users A and B and items 1, 2, 3.
- Group A is shown items 1, 2 and 3 respectively 1, 2 and 3 times.
- Group B is shown items 1, 2 and 3 respectively 2, 2 and 2 times.
- The exposure distribution for group A is then [1/(1+2+3), 2/(1+2+3), 3/(1+2+3)]
- The exposure distribution for group B is then [2/(2+2+2), 2/(2+2+2), 2/(2+2+2)]

User group fairness metric 1

- We want to measure whether the recommender system recommends items evenly to different groups of users.
- The Exposure Total Variation measures the total variation distance between the exposure distribution for groups A and B (e.g., male / female).
- $Total\ Variation = \frac{1}{2} \sum_{items\ i} |p_A(i) - p_B(i)|$
- The value of the metric ranges between 0 and 1 with 0 indicating an exact match and 1 indicating the distributions are as distant as they can be.

TOT VAR (example)

- Suppose there are 5 items. And we have computed the exposure distributions as $p_A = [0.1, 0.1, 0.1, 0.1, 0.6]$ and $\mathrm{p_B} = [0.2, 0.2, 0.2, 0.2, 0.2]$
- The total variation distance is
- $Total\,Variation = \frac{1}{2}(4 * |0.1 - 0.2| + |0.6 - 0.2|) = 0.4$
- 0.4 is a relatively high total variation, so this data is considered biased.

User group fairness metric 2

– The Exposure KL is the Kullback Leibler divergence from the item exposure distribution of group A to that of group B.

– $KL\ Div = \sum_{items\ i} p_A(i) log \frac{p_A(i)}{p_B(i)}$

– The KL doesn't work well if there are items that are not shown to group B.

– On the other hand, it is widely used in theoretical work, and an important metric to know about.

KL DIV (example)

- Suppose there are 5 items. And we have computed the exposure distributions as $p_A = [0.1, 0.1, 0.1, 0.1, 0.6]$ and $\mathrm{p_B} = [0.2, 0.2, 0.2, 0.2, 0.2]$
- The Kullback-Leibler Divergence is
- $KL\ Div = 4 * 0.1 * \log\left(\frac{0.1}{0.2}\right) + 0.6 \log\left(\frac{0.6}{0.2}\right) = 0.3819$
- Note: interpreting KL DIV is not always easy, except in relative terms.

Individual Item fairness metric 1

- The aggregate diversity is the proportion of all items that have been recommended to at least one user.
- $Agg\ Div = \frac{1}{|I|} \sum_{items\ i} 1[item\ i\ is\ shown]$
- We want the aggregate diversity to be close to 1, indicating almost all items are shown at least once.
- If the aggregate diversity is low, then a large proportion of items are not even given a chance.

AGG DIV (example)

- Suppose there are 10 items. And we have computed the item exposure distribution as $p = [0.1, 0.1, 0.1, 0.1, 0.6, 0, 0, 0, 0, 0]$.
- What is the aggregate diversity?
- As is seen from the distribution, 5 out of 10 items have nonzero probability of being shown.
- So $Agg\ Div = \frac{5}{10} = 0.5$

Individual Item fairness metric 2

- The gini index is a measure of how unequal a distribution is. We apply it to the exposure distribution of items.
- Suppose we have the overall exposure distribution q, and we sort items so that so that $q(i_k)$ increases with k.
- $GINI = \frac{1}{|I|-1} \sum_{k=1}^{|I|} (2k - |I| - 1) q(i_k)$
- An algorithm that recommends each item the same number of times (uniform distribution) will have a Gini index of 0 and the one with extreme inequality will have a Gini of 1.

GINI (example)

- Suppose there are 4 items. And we have computed the item exposure distribution as $p = [0.1, 0.2, 0.3, 0.4]$. (Note: this is sorted in increasing order).
- Applying the formula from previous slide
- $GINI = \frac{1}{3}\ [(-3)0.1 + (-1)0.2 + (1)0.3 + (3)0.4 = \frac{1}{3}$
- 1/3 is a low gini index, so data is quite fair.

Individual Item fairness metric 3

- The average recommendation popularity stands true to its name, it measures on average how ‘popular’ is a recommendation.
- Let i_{uj} be the jth item (of K) recommended to user u. Let $N(i_{uj})$ be the total number of times this item appears in the training set.
- $AVG\ REC\ POP = \frac{1}{|U|} \sum_u \frac{1}{K} \sum_{j=1}^{K} N(i_{uj})$
- High values of this metric imply that on average we are recommending very popular items. Lower values are considered fairer.

AVG REC POP (example)

- Suppose there are 3 users and 3 items, with the following prediction matrix

$$M = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 0 \\ 1 & 0 & 0 \end{bmatrix}$$

- Item 1 is recommended 3 times, item 2 is recommended twice, item 3 is recommended once.
- $AVG\ REC\ POP = \frac{1}{3}\left[\frac{1}{3}(3+2+1) + \frac{1}{2}(3+2) + 1(3)\right] = \frac{15}{6} = 2.5$
- Note: there is no scale for this metric, one must interpret it relatively or as a fraction of the maximum popularity.

Computing Bias Metrics with holisticai library

– When computing recommender systems bias metrics with the holisticai library, we first need to understand the type of data we input.

– The input mat_pred is a matrix that contains a 1 for each (user, item) pair where the item is recommended to the user.

– The documentation of all the recommender bias metrics can be found here.

Computing Bias Metrics with holisticai library

– The first step is installing the library

```
pip install holisticai
```

– Once the library is installed, import the desired metric.

```python
from holisticai.bias.metrics import aggregate_diversity
aggregate_diversity(mat)
```

Python

```python
from holisticai.bias.metrics import gini_index
gini_index(mat)
```

Python

– Compute the value as shown above.

Computing metrics in batch with holisticai library

– There is also a helper to compute all bias metrics in batch. Suppose we are interested in all item-based metrics. We can compute them as follows.

```python
from holisticai.bias.metrics import recommender_bias_metrics
recommender_bias_metrics(mat_pred=mat, metric_type='item_based')
```

Python

– The output is a pandas DataFrame, e.g.,

Metric	Value	Reference
Aggregate Diversity	0.411355	1
GINI index	0.964424	0
Exposure Distribution Entropy	3.812282	-
Average Recommendation Popularity	5716.980253	-

Exercise Notebooks

– We have created exercise Notebooks for measuring Bias.

Next

- Part I – Introduction to Recommender Systems
- Part II – Fairness in Recommender Systems
- Part III – Measuring Bias in Recommender Systems
- **Part IV – Mitigating Bias in Recommender Systems**

References

[1] Dash et al, 2021, When the Umpire is also a Player

[2] Deldjoo et al, 2022, A Survey of Research on Fair Recommender Systems

[3] Sun et al, 2019, Debiasing the Human-Recommender System Feedback Loop in Collaborative Filtering

[4] V. Tsintzou, E. Pitoura, P. Tsaparas, 2019, Bias disparity in recommendation systems

[5] M. Mansoury, B. Mobasher, R. Burke, M. Pechenizkiy, 2019, Bias disparity in collaborative recommendation

[6] *Toshihiro Kamishima, Shotaro Akaho, Hideki Asoh, Jun Sakuma, 2018,* Recommendation Independence

Sources

https://fairness-tutorial.github.io/files/Tutorial_on_Fairness_in_Recommendation_Slides.pdf

https://ir.library.louisville.edu/cgi/viewcontent.cgi?article=1440&context=faculty

https://arxiv.org/pdf/2205.11127.pdf

https://arxiv.org/pdf/2005.01148.pdf

https://arxiv.org/pdf/2105.05779.pdf