Self-similar Epochs: Value in arrangement

Presented by Eliav Buchnik

Eliav Buchnik · Edith Cohen · Avinatan Hasidim · Yossi Matias

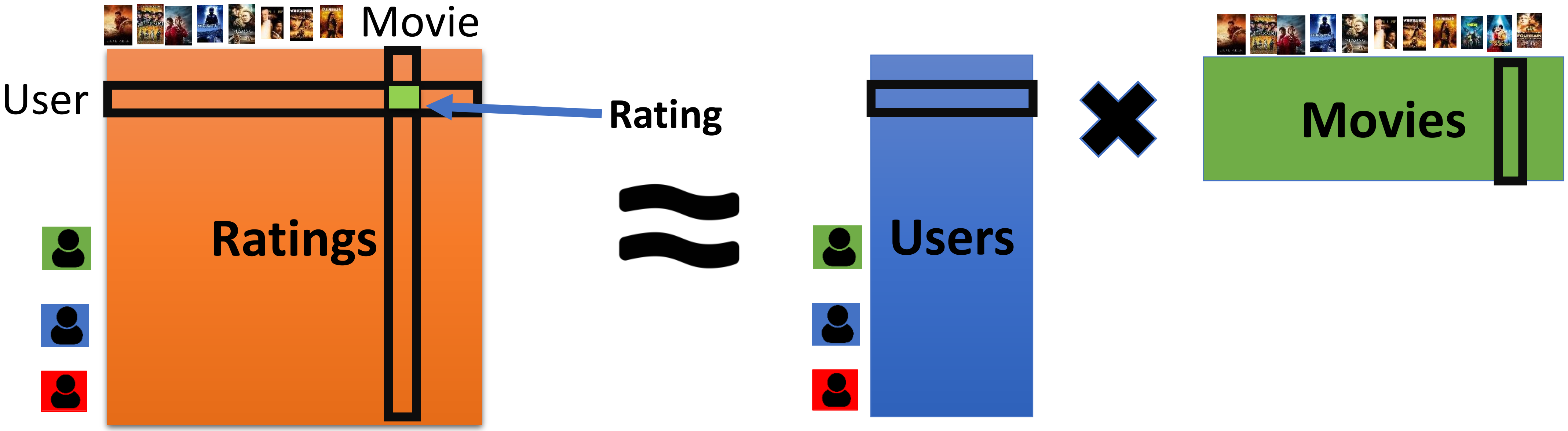
This work was supported by the ISF grant no. 1841/14

Arrangement methods of training examples for Stochastic Gradient Descent(SGD)

- We explore arrangement method of training examples as an optimization knob for SGD
- The common baseline is i.i.d. arrangement. The drawback is that sub-epochs lose the structure of the full data.
- We present *Self-Similar* arrangements.
    - ➢Keep the marginal distribution of training examples but sub-epochs do preserve the structure.
- Method can be combined with many other optimization knobs of SGD.
- Accelerate training time by **3%-37%**

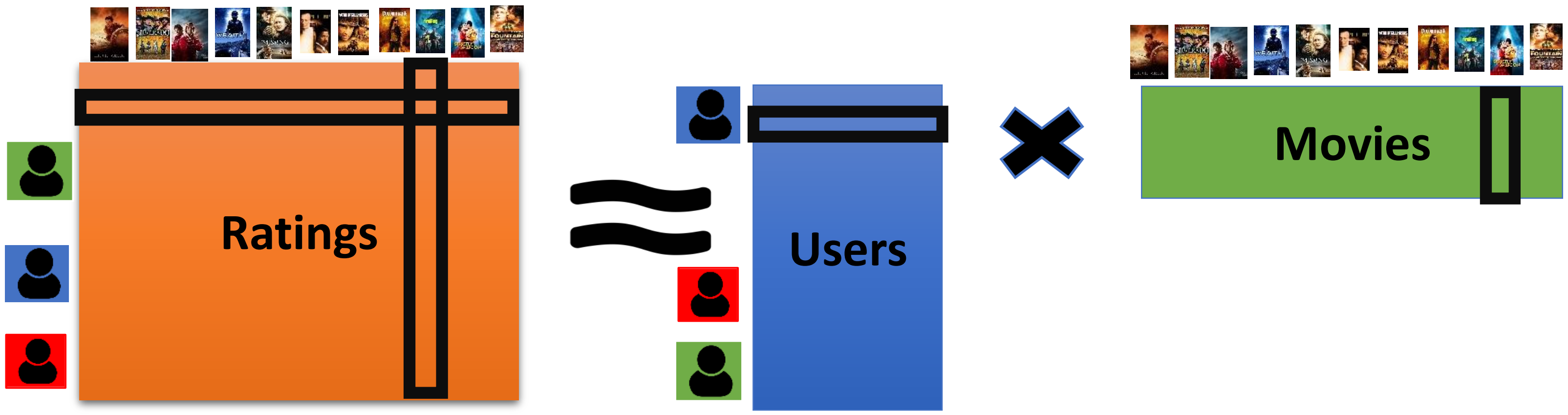
Test case - matrix factorization

Data is pairwise interactions: word co-occurrences, user-movie ratings/views:



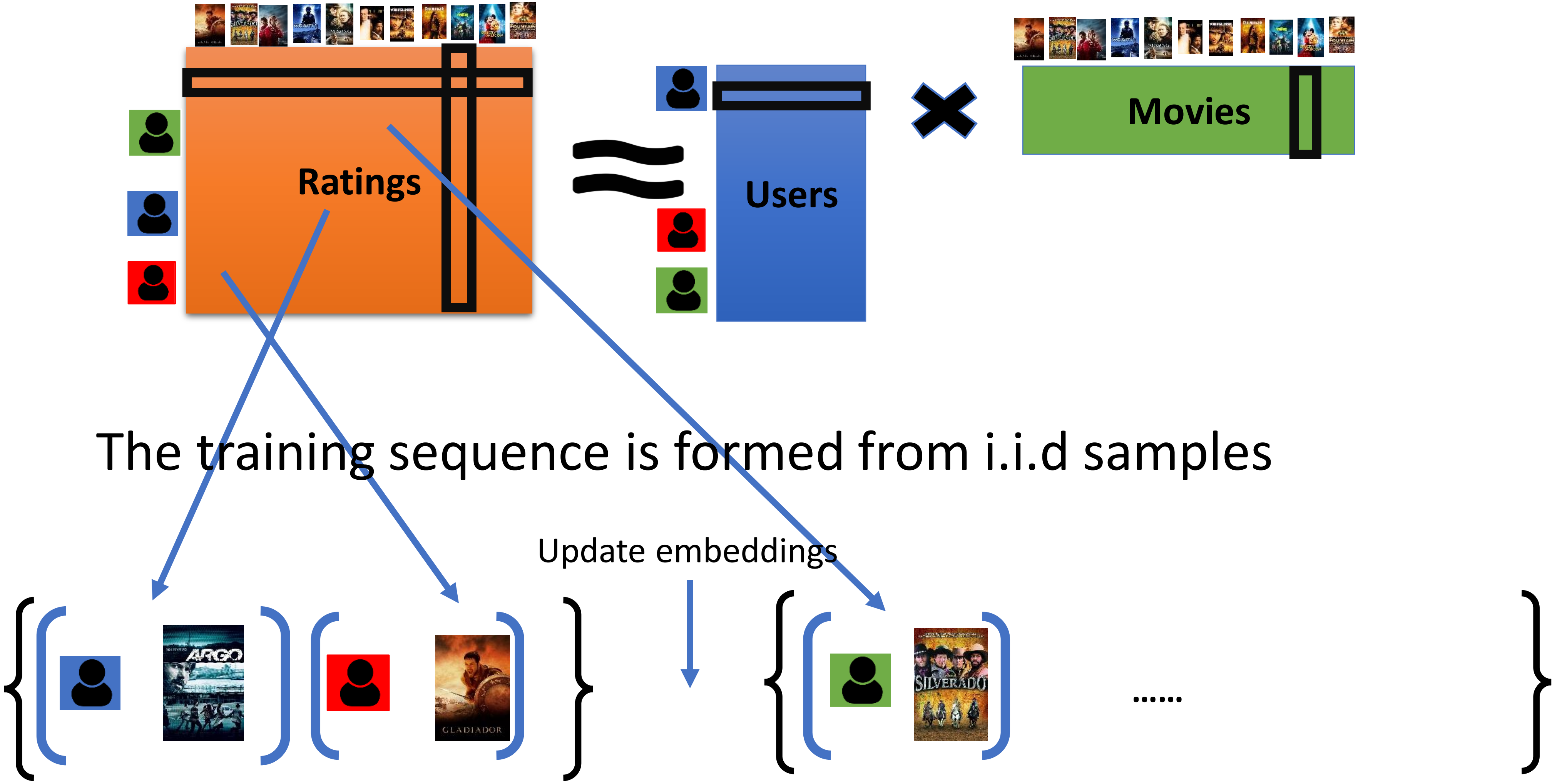
Produce an embedding vector for each entity (e.g. user or movie) so that interactions (e.g. views) can be recovered (and new ones predicted) from embeddings (e.g. SGNS by Mikolov et al., …)

Test case - matrix factorization



The training sequence is formed from i.i.d samples

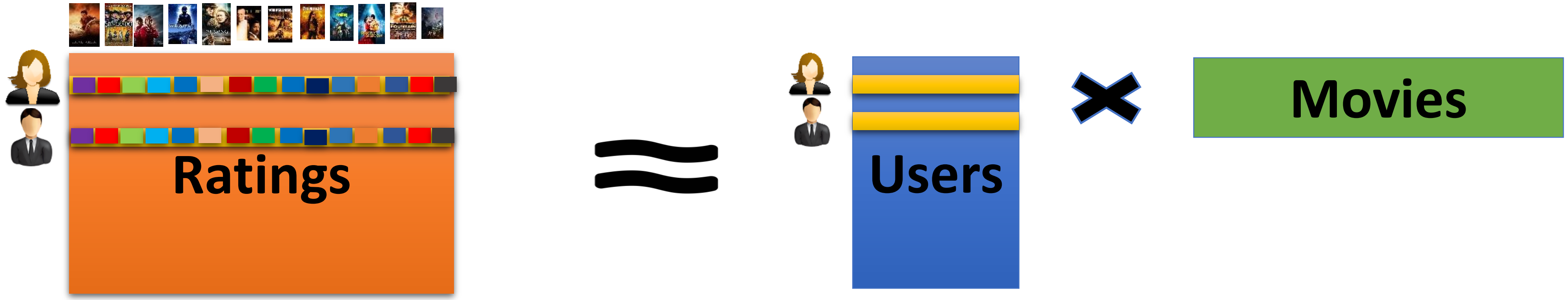
Test case - matrix factorization



The training sequence is formed from i.i.d samples

Motivation: Identical rows

Consider two users with identical movie preferences



Ideally the end result is two (nearly) identical embeddings

To recover this similarity from a sub-epochs we need it to contain examples where they rate the same movies.

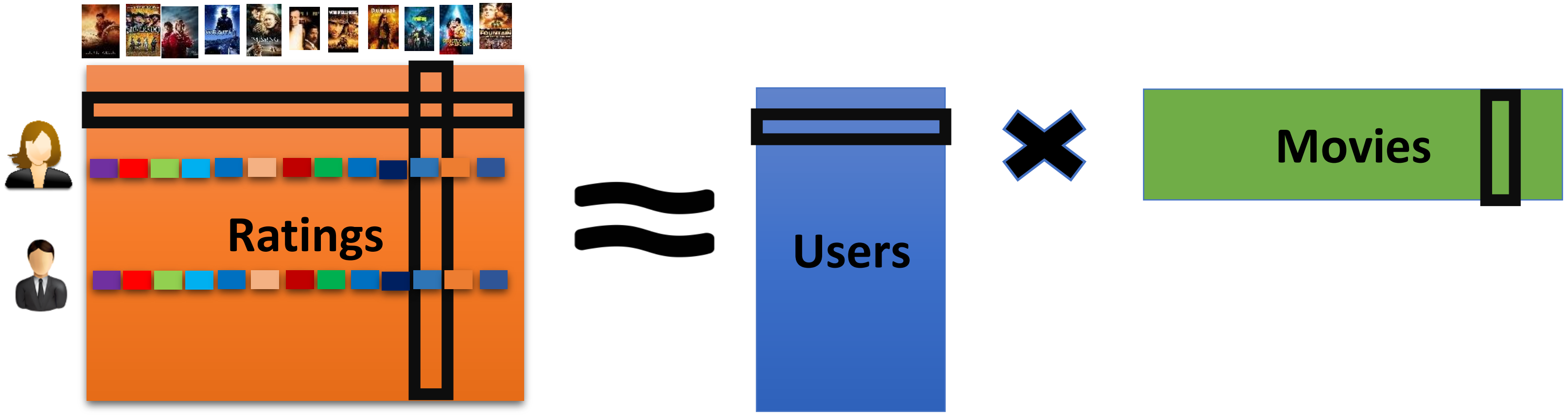
I.i.d arrangements: The samples of the two users are likely to be very different.

→ **Similarity structure is lost**

Self-similar epochs

<u>“self-similar”:=</u> preserves similarity structure in a sub epochs.

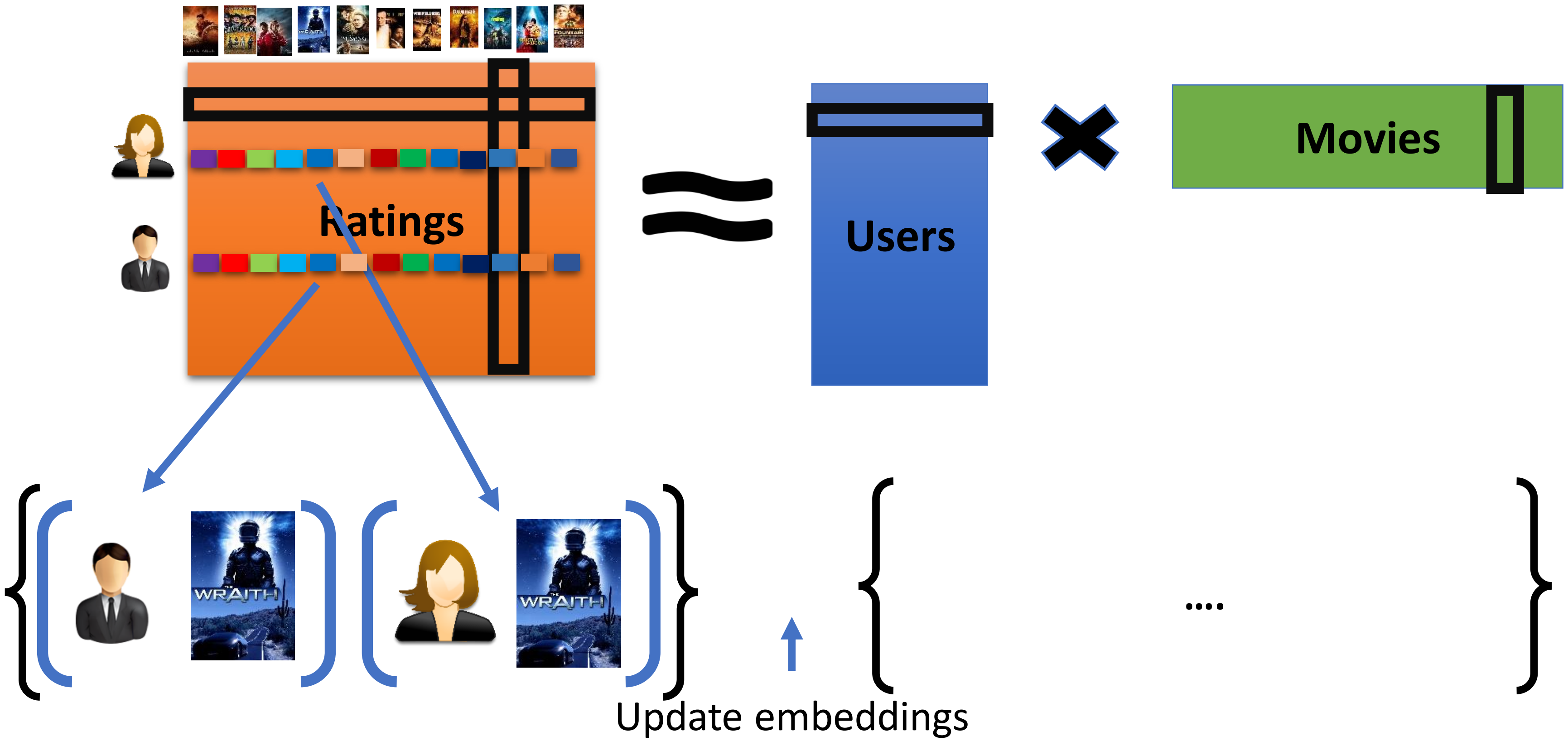
We hypothesize that “self-similar” arrangements will allow one epoch to act as multiple ones and thus help SGD converge faster.



Self-similar epochs

<u>“self-similar”:=</u> preserves similarity structure in a sub epochs.

We hypothesize that “self-similar” arrangements will allow one epoch to act as multiple ones and thus help SGD converge faster.



Properties of our arrangement method

- Does not change the marginal distribution of examples
- Sub-epochs preserve in expectation the weighted Jaccard similarities of pairs of rows and columns.
  - ❖ $J(u, v) = \frac{\sum_i \min(u_i, v_i)}{\sum_i \max(u_i, v_i)}$

Algorithms:

- Preprocessing step with cost linear in the sparsity of the matrix.
- During training the cost is $O(1)$ per example drawn

Results:

- **Acceleration of between 3%-37% in training time.**

Thank you!

Details at poster #60