Expriment Design

Assigment 2 - Paper 11

Group 11

Summary

The paper presents a recommender system (RS) based on not only direct relationships between users and items, but also on the relationships between other implicit data such as `recency of liking interaction`, `popularity of the shared entity`, and `diversity of the explanation type`. The overall idea of the proposed RS is to improve explainability by providing insights into why specific items are suggested to users while keeping good recomendation quality.

- `Paths`: alternated sequence of entities and relationships that connect a user, a shared entity, and the recommended item.

`User 1` ---watched---> `Movie 1` ---directed_by---> `Director 1` (shared entity) ---directed---> `Movie 2` (recommended item)

- `Recency of liking interaction` Interactions that are more recent are more likely to be easier to catch for the user, while older interactions might not be remembered by the user and therefore won't be as valuable as recent interactions for explainability purpuses.

- `Popularity of the shared entity` The popularity of the shared entity can help users decide whether a product might be interesting for them. The more popular the shared entity (e.g., actor, director), the easier it is for users to understand the explanation, as they are more likely to recognize the entity and associate it with items they have already consumed.

- `Diversity of the explanation type` "Recommending items by only their predicted relevance increases the risk of producing results that do not satisfy users because the items tend to be too similar to each other". Depending of the dataset an explanation can be based on different entities director (directed_by), actor (acted_in), etc.

Reproducibility pros

- The repository has a `requirements.txt` file that lists all the libraries and their versions needed to run the code.
- The repository has a `README.md` file that explains how to run the main code and the preprocessing code.
- We have available pre-processed data for both the `LFM-1b` and `ML1M` datasets, also pretrained paths for both datasets.
- They explain clearly how datasets were preprocessed, including the order of the data (sorted by date), splited 70% (training), 20% (validation), and 10% (training).

Reproducibility issues

- The repository says that `python 3.6` or higher is required, but some versions of the libraries are not compatible with `python 3.6` or with higher versions like `python 3.13.0` or `python 3.9.6` . So far we found the minimum version of `python` that works is `3.7.1`.
- The original `LFM-1b` dataset is not available for download anymore (in the original source) due to license issues.
- In the paper is mentioned that some users are being removed if they do not provide 3 sensitive attributes (gender, age, interactions with items). However, if a user does not have gender or age it will be still mapped without those attributes cuasing an error when testing the recommender system. For interactions, the condition to remove a user is if this has less than 5 interactions with items.
- In the paper is mentioned that some items (movies or songs) are removed if they are not present as entities in the knowledge graph and their relationship types occur less than 200 times. But this preprocessing is not being applied when preprocessing the data.
- The generation of the data for `LFM-1b` is not correct, there are users (`uid`) linked to products (`pid`) that are not present in the dataset. This applies when generating the preprocessed data and with the available pre-processed data in the repository. (Only happens on "soft" optimizations "softETD", "softSEP", "softLIR", "ETDopt"

Other notes

- Metrics:
  - Utility of the recommendation: NDCG (Normalized Discounted Cumulative Gain)
  - Quality of the explanation: LIR (Latent Item Recency), SEP (Shared Entity Popularity), ETD (Explanation Type Diversity)