Anas Buhayh, Matt Nicholson, Joshua Paup
INFO 5612 Recommender Systems
Dr. Robin Burke
Wednesday, 20 December 2023

CORGI-Lite: People Recommendation on hci.social

Introduction

Problem area

In recent years, federated social media platforms such as Mastodon have become an increasingly popular alternative to traditional social media platforms such as Twitter or Facebook [Nicholson 2023]. Traditional social media platforms have historically been profit-driven entities with total ownership of a single platform (or a collection of platforms such as Meta) where algorithmic, platform, and policy decisions are unilaterally made top-down with minimal user input. Federated social media platforms on the other hand are usually made up of loosely connected "instances" on a decentralized protocol (such as ActivityPub) where each instance is usually moderated by one user (or collection of instance members) who maintains autonomy over the instance.

One key distinction we would like to make between traditional and federated social media platforms is how recommender systems are deployed within both areas. Most federated social media platforms simply provide reverse chronological feeds for their users (often by choice). By contrast, the recommendation algorithms deployed by traditional platforms are highly complex systems that deliver personalized timelines to their users. The implementation details of these systems are often proprietary. This lack of transparency behind their recommendation algorithms has resulted in many users increasingly distrusting traditional platforms, as well as raising broader concerns about how the algorithm contributes to the growing issue of surveillance capitalism and other privacy violations.

We assume that users who have made the migration from more centralized platforms to federated alternatives view the harms of recommender systems and recommender systems themselves as inseparable. We believe that these can be decoupled.

What is CORGI?

CORGI (shorthand for "Community-Oriented Recommendation and Governance Infrastructure") is an ongoing project within That Recommender Systems Lab (TRSL) at the University of Colorado Boulder that hopes to tackle the tensions that we assume users of federated social media platforms maintain — more specifically, on Mastodon. CORGI attempts to complete three tasks: implement post and people recommendation within Mastodon instances, provide governance infrastructure to enable instance members to participate in the design and deployment of a recommendation system, and provide an algorithmic recourse dashboard for content creators to enable them to make sense of how their content is "affected" by the community's recommender system. This is a lofty goal. For the sake of this class, we limit our scope to recommendations for new users to follow.

Methods

Data Collection

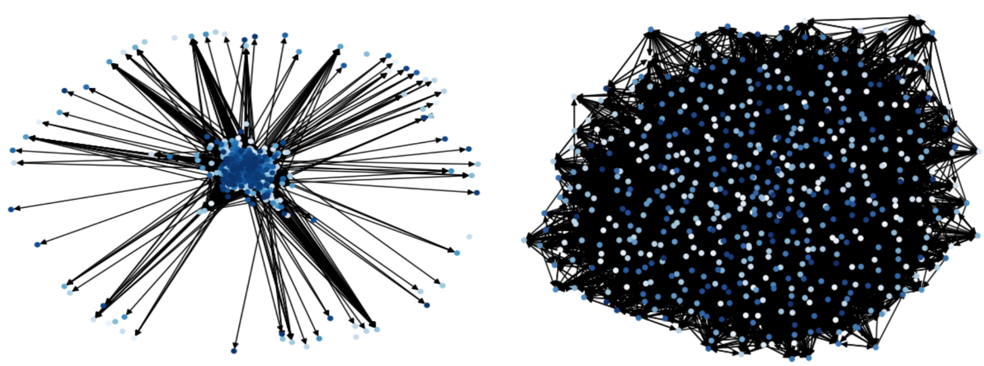
We scraped three days (October 17-20, 2023) of posts using the Mastodon API to build a set of seed accounts from which to collect followers and following lists for each user.

We chose hci.social for this project because it is a moderately sized (~2400 total users) and reasonably active (~700 monthly active users) community. Recognizing the common norms of privacy on the Fediverse, we limited our collection to hci.social exclusively, since we are members of that community, and discarded any nodes belonging to other instances. While we made this choice hoping to respect the local customs of the Fediverse, we recognize that we have a partial and limited view of the full network structure, as we undercount edges and likely underrepresented the importance of accounts with many links across instances.

Additionally, many accounts on Mastodon opt out of sharing their social graph publicly (Matt is one of them). When called via API, these accounts return empty lists for both their followers and following. These edges can be partially reconstructed from the followers lists of people they follow and the following lists of people who follow them, provided that at least one of the parties is public. This serves as a stark reminder of the limits of individualized approaches to privacy [Marwick and boyd 2014].

Data cleaning and formatting

From this collection of account-level followers and following lists, we construct a network and accompanying adjacency matrix (Figure 1).



	Initial Hairball	Densified Graph
Nodes	1,528	588
Edges	38,064	25,373
Avg Degree	49.8	86.3
Reciprocity	0.581	0.653

Figure 1. Our initial hairball (left) and densified graph (right)

Note that in the initial graph of 1,528 nodes, we have a great number of low-degree peripheral nodes. These accounts represent accounts within hci.social that follow no other accounts in hci.social. To densify the graph, we iteratively removed nodes with fewer than 15 out-links, and we were left with only 588. For the sake of this project, we felt comfortable with the active exclusion of nearly two-thirds of the initial graph because we would likely take a different recommendation approach for these nodes. We leave further exploration of this “cold-start” problem as future work.

Data splitting

To construct train-test splits for this network, we selected one-third of the out-links from each node. These edges make up the test set. These edges were then removed from the original network, and the resultant graph was the train set. We performed this across five folds of the data.

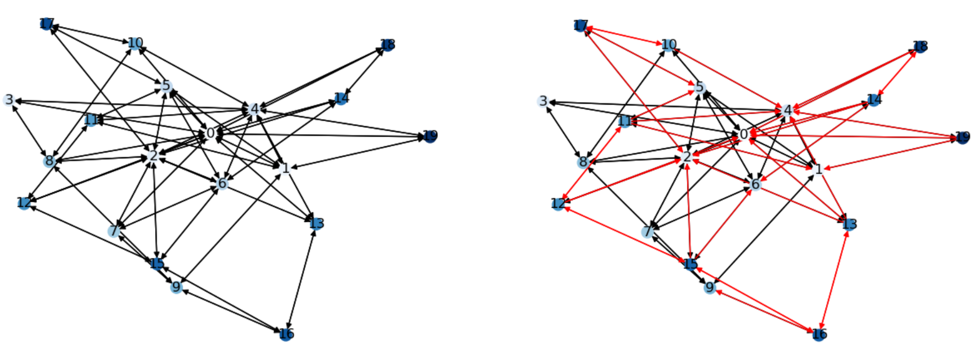


Figure 2. Depicting the data splitting process on a much smaller random graph using Barabási–Albert preferential attachment. We expect the full hci.social graph to have similar characteristics. The image on the left depicts the full graph, and the image on the right depicts an example of a train-test split, where edges in red are removed from the training graph and are used as a test set.

The choice of five-fold cross-validation was more pragmatic than principled – we certainly could have performed leave-one-out cross-validation (by removing a single node and all its links from a fold of the training data) but chose not to because of computation and time constraints.

Model training

We implement Personalized PageRank, LightGCN, and Factorization Machines to recommend users to follow. Personalized PageRank can be thought of as a “random walk” (or transition probability between nodes i and j) over the full set of nodes within the instance network [Page et al. 1999]. LightGCN can be thought of as neighborhood aggregation through linearly propagating the user and item embeddings within the instance network [He et al. 2020]. Factorization Machines can be thought of as the prediction of some underlying low-dimensional structure among the interactions between users and their items (in this case, who the users are following) within the instance network [McAuley 2022].

PageRank

The PageRank algorithm highlights the importance of the nodes based on how many connections it has. We trained the model on a network edge list representing the connection between the users. The output predictions contain nodes (recommended people to follow) and the probability that you’ll end up on that node.

Factorization Machine

The factorization machines are supervised algorithms that capture interactions between features in sparse data. In preparing the data for this algorithm, we added binary target variables to the train, test, and anti-test data. The people the user is following can be seen as item interactions. We also needed to extend the data to represent all possible interactions between users to build our sparse dataset for the factorization machine. We used the PyFm package to fit the model and composed a list of predictions representing a link connection between two users and a value representing the probability of that link.

Graph Convolutional Networks

We used the Light GCN module as our third model. GCN models learn patterns from graph data. Like the factorization machine, the directional edges can be seen as user-item interactions. We’ve also extended the dataset here to include a binary target variable representing if the user is following a person. The output variable for GCN was a value representing the probability of a user-item link, which in our case is a user-user link.

Evaluation

We evaluated the three models using 5-fold cross-validation on the test and anti-test data. We calculated the NDCG@K and the Precision@K for each of the algorithms. We also exported a list of predictions for the three of us to compare our results and see what people we recommended.

Results

Both PageRank and Light GCN achieved relatively high scores compared to the Factorization Machine. On the other hand, the factorization machine achieved a significantly higher precision score than the other two algorithms (Figure 3).

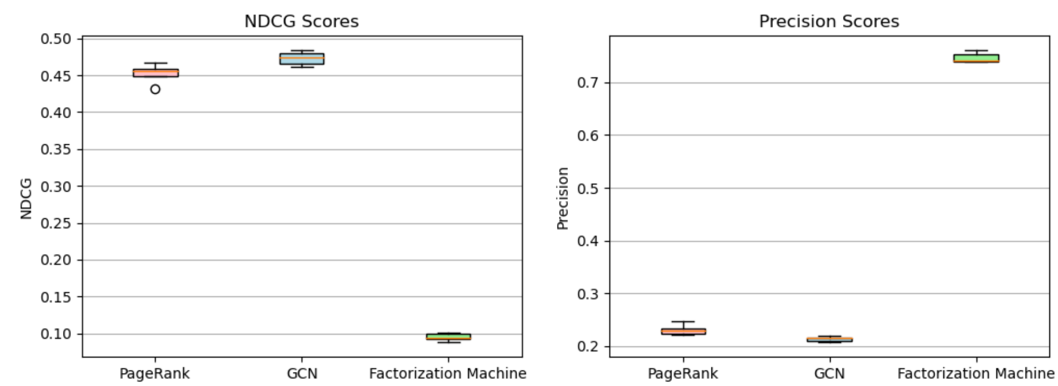


Figure 3. A box plot of NDCG scores (left) and Precision (right) across 5 folds of the test data.

We are unsure why our Factorization Machine machine implementation is significantly different from the rest. The low NDCG might be because we had to extend our user-item interaction to include all the possible interactions making the database more sparse, which might have led to lower NDCG. We are not quite sure why our precision is as high as it is. We suspect that this pairing of metrics for the same set of predictions is a mathematical impossibility, but we are not sure where our implementation mistake occurs.

An observation we made when examining our own prediction lists is that we had many predicted users in common, and that these users all had comparatively high in-degrees. A possible explanation is that the three of us know and follow similar people, however, this area requires more evaluation and experimentation.

Discussion

Recommendation in federated social media is materially different from the platform-centric social media settings in which recommender systems are often deployed and theorized. Specifically, the federated landscape features a distinct set of social (e.g. norms) and technical (e.g. data sharing and partial views of content between instances) considerations. In our project, these considerations were largely out of scope. However, we recognize that these are the critical challenges for deploying a system that balances utility and responsibility. We give attention to several dimensions here.

Transparency and Privacy

A common critique of more centralized platforms is a lack of transparency in their implementations of recommender systems. In the Fediverse, where open-source software is the norm, we (and instance maintainers) could disclose the conditions and results of our experiments, along with the implementation decisions we ultimately make.

However, these considerations also collide with the Fediverse-specific norms of privacy and concerns about "gaming" the algorithm [Cotter 2019]. It is currently not clear how to balance these competing factors. Further, since the Fediverse is not a monolith, recommender system decisions within one instance impact users on other instances, an interconnectedness not seen in more centralized settings.

Interoperability and Participatory Design

Interoperability is commonly cited as an advantage of the Fediverse [Doctorow 2023], where users can leave an instance if they are displeased with how the instance is running, and still keep their social graph. Because users can leave an instance with few consequences, instance maintainers in theory are accountable to their users, however, in practice, these instances often operate as fiefdoms [Schneider 2022]. As this project progresses, we should invest in participatory governance structures, so that we diffuse this concentrated power among instance members.

Limitations

In this project, we limited our data collection and our recommendation approach to a single instance. Within this instance, we found that the degree distribution roughly followed a power-law distribution, and popularity is a rough proxy for importance or notability. In a truly federated context, where some authoritative accounts (e.g. some municipal government accounts, the BBC, etc) exist in their own servers with few to no outlinks, it is less clear that our approach of link-based methods would work quite as effectively due to differences in network structure. We must develop an approach that responsibly uses the data of other instances and addresses quirks in broader network structure when expanding this system across instances.

Future Work and Conclusion

This project represents a small step towards a much larger goal. For the context of this project, we collect data in a batch and evaluate it offline. In the future world where this recommender system is deployed, we might be able to evaluate the quality of our predictions based on actual user behavior. We similarly ignore the cold start problem in this setting. Finally, evaluation is not a straightforward endeavor in this setting and the notion of what makes a "good" recommendation is a contested topic.

Recognizing that recommendation is a deeply socio-technical task that engages multiple stakeholders, additional research would better inform how we might design a system in Mastodon. For example, we might conduct need-finding interviews with Mastodon users to better understand norms of data uses or perceived harms and benefits of recommender systems in a federated social media context. Additionally, we might explore model evaluation criteria beyond accuracy (e.g. novelty, diversity, fairness) and partner with instance maintainers to develop transparency and participatory governance tools to oversee these experiments and implementations. We look forward to engaging with these more nuanced aspects of this project moving forward.

References

Cotter, K. (2019). Playing the visibility game: How digital influencers and algorithms negotiate influence on Instagram. New Media & Society, 21(4), 895–913. https://doi.org/10.1177/1461444818815684
Doctorow, C. (2023). The internet con: How to seize the means of computation. Verso.

He, X., Deng, K., Wang, X., Li, Y., Zhang, Y., & Wang, M. (2020). LightGCN: Simplifying and Powering Graph Convolution Network for Recommendation (arXiv:2002.02126). arXiv. http://arxiv.org/abs/2002.02126

Marwick, A. E., & boyd, danah. (2014). Networked privacy: How teenagers negotiate context in social media. New Media & Society, 16(7), 1051-1067. https://doi.org/10.1177/1461444814543995

McAuley, J. (2022). Personalized Machine Learning. Cambridge University Press, 145.

Nicholson, M. N. (2023, October 5). An Exploration of the Twitter to Mastodon Migration. Medium. CUInfoScience. https://medium.com/cuinfoscience/an-exploration-of-the-twitter-to-mastodon-migration-21c15c4336f2

Page, Lawrence; Brin, Sergey; Motwani, Rajeev and Winograd, Terry, The PageRank citation ranking: Bringing order to the Web. 1999. http://dbpubs.stanford.edu:8090/pub/showDoc.Fulltext?lang=en&doc=1999-66&format=pdf

Schneider, N. (2022). Admins, mods, and benevolent dictators for life: The implicit feudalism of online communities. New Media & Society, 24(9), 1965–1985. https://doi.org/10.1177/1461444820986553

Appendix A – Link to GitHub

https://github.com/nicholson2208/rec-sys-final-project