Decentralized Adaptive Ranking using Transformers

Marcel Gregoriadis, Quinten Stokkink, Johan Pouwelse

Presented at EuroMLSys ’25, March 31st 2025, Rotterdam, The Netherlands

Motivation

European Commission

EN

Search | Search

Home > Press corner > Commission opens formal proceedings against TikTok under DSA

Available languages: English

PRESS RELEASE | Dec 17, 2024 | Brussels | 3 min read

Commission opens formal proceedings against TikTok on election risks under the Digital Services Act

PAGE CONTENTS

Top

Quote(s)

Related topics

Print friendly pdf

Commissi
Digital Ser

Today, the
breach of t
assess and
recent Rom

The proceedings will focus on management of risks to elections or civic discourse, linked to the following areas:

- TikTok's **recommender systems**, notably the risks linked to the coordinated inauthentic manipulation or automated exploitation of the service.
- TikTok's policies on **political advertisements and paid-for political content**.

Motivation

Big Tech's recommender systems determine what we **see, read, believe, and vote.**

We aim to offer a decentralized alternative.

Specifically, we focus on the **problem of decentralized ranking of search results.**

Related Work

- Existing decentralized ranking algorithms are based on heuristics, often taking into account
  - Term-based metrics (e.g., BM25)
  - Resource availability (seeders)
  - Resource demand (leechers)
  - Freshness of a document
  - Collaborative filtering

Learning-to-Rank and the Design of DART

- Plenty of metrics to describe the relationship between query and document
- *Problem:* It is not obvious how they predict relevance
- *Solution:* Let a ML model learn from empirical data

List of documents (search results)

$$X = \{x_1, x_2, \ldots, x_n\} \quad \text{with} \quad x_i \in \mathbb{R}^f$$

Document-Query feature vector

Table 1. Features of a Document-Query Pair

ID	Description
0	BM25 score [25]
1–5	Term frequency (TF): min, max, mean, sum, and variance [26]
6–10	Inverse document frequency (IDF): min, max, mean, sum, and variance [26]
11–15	TF*IDF: min, max, mean, sum, and variance [26]
16	Cosine similarity of the TF*IDF 5-tuple
17	Number of query terms in the document title
18	Ratio of query terms in the document title
19	Number of characters in the document title
20	Number of terms in the document title
21	Number of terms in the query
22	Query matches document title exactly
23	Ratio of query terms matching the document title
24	Number of nodes storing the doc. (seeders) [23]
25	Number of nodes querying the doc. (leechers) [23]
26	Number of times the doc. has been clicked [9]
27	Number of times the document was selected when one of the document's terms was also part of the query terms (hit count) [16]
28	Document rank in the result, before re-ranking
29	Number of user-annotated document tags [23]
30	Freshness (time since document creation) [23, 30]

Learning-to-Rank and the Design of DART

We employ a context-aware ranker based on a transformer encoder with self-attention.

Result list

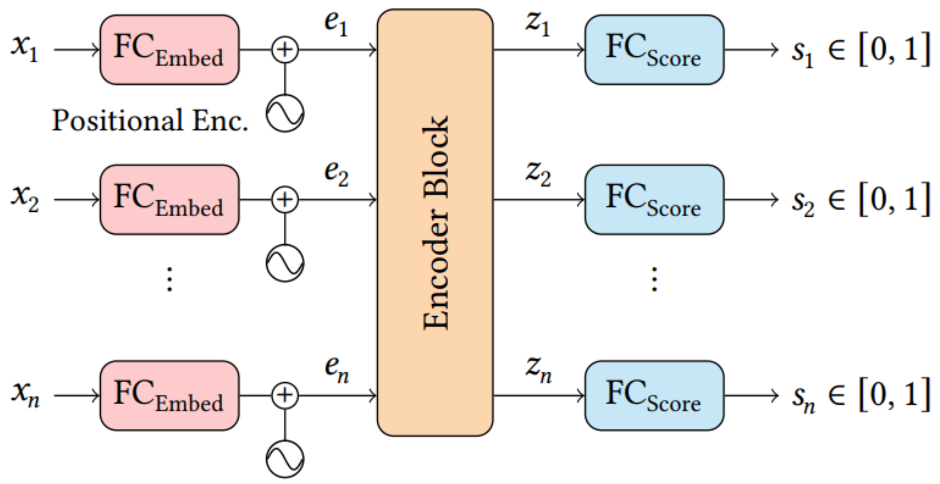


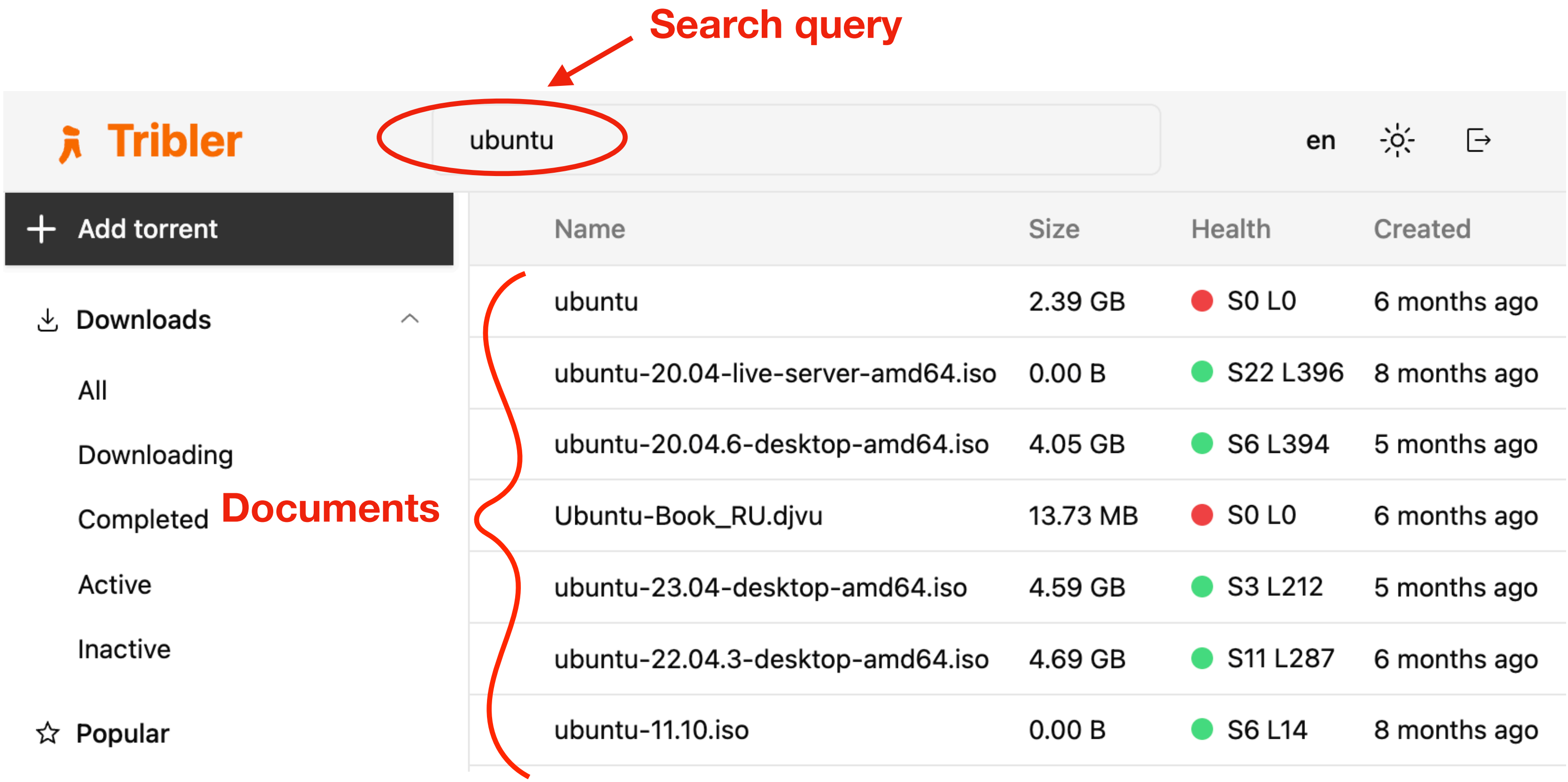
Figure 2. Model architecture and data flow.

Relevance scores

Order results according to corresponding relevance score.

Tribler Software

Tribler is a decentralized file-sharing system based on the BitTorrent protocol with integrated search and anonymized filesharing. It has over 40k active users per month.



TU Delft

Dataset of Clicklogs

We compile a dataset of 9k clicklogs from search activities observed in the Tribler network.

Clicklog:

- User ID
- Search query
- Clicked document
- List of documents
- Timestamp

Document:

- Document ID
- Title
- Tags
- Seeders
- Leechers
- Size
- Creation Time
- Ranked Position

Decentralized System Model

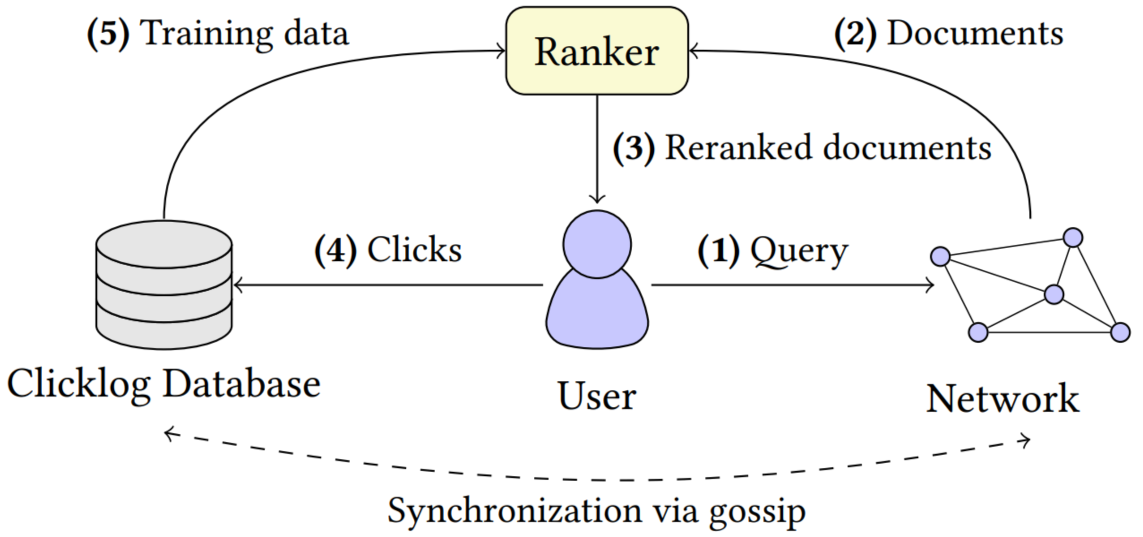


Figure 1. Overview of the system model and its interactions.

Experiment #1: Ranking Performance

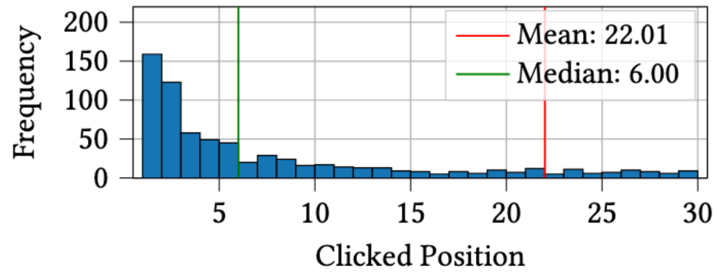
We split the entire dataset into context and test set (90:10).

Evaluation Metric:

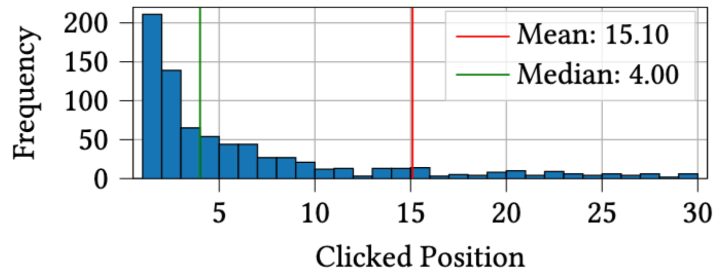
$$MRR = \sum_{clicklog} \frac{1}{\text{rank of relevant document}}$$

Table 5. Ranking Performance (Rec.=Recall)

Algorithm	MRR (± SD)	Rec.@1	Rec.@5	Rec.@10
Random	*0.18 ± 0.26*	0.15	0.31	0.42
G-Rank	0.25 ± 0.32	0.22	0.41	0.52
MAAY	0.27 ± 0.34	0.26	0.43	0.52
Panaché	0.28 ± 0.35	0.25	0.43	0.53
DINX	0.28 ± 0.34	0.26	0.46	0.55
DINX-s	0.31 ± 0.36	0.28	0.49	0.61
Tribler	0.32 ± 0.35	0.31	0.50	0.61
DART	**0.38 ± 0.37**	**0.38**	**0.61**	**0.73**



(a) Tribler

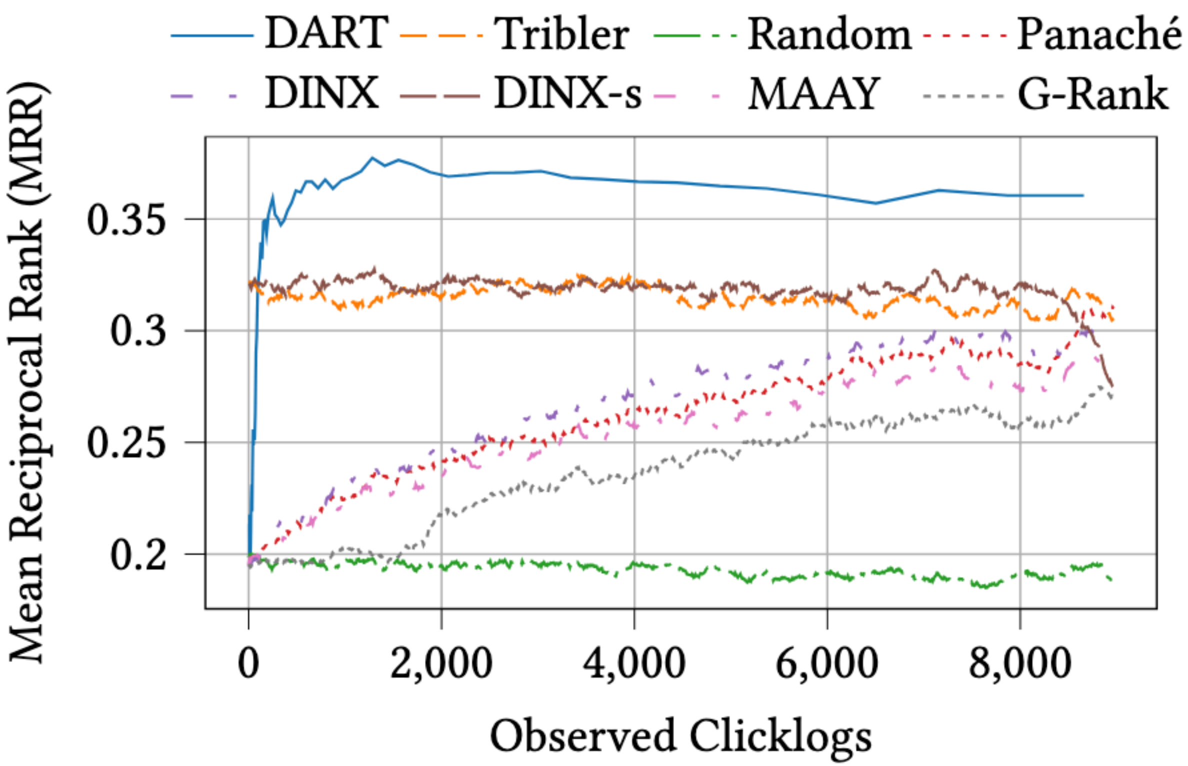


(b) DART

Figure 3. Comparison of clicked position distributions for Tribler and DART, truncated at position 30.

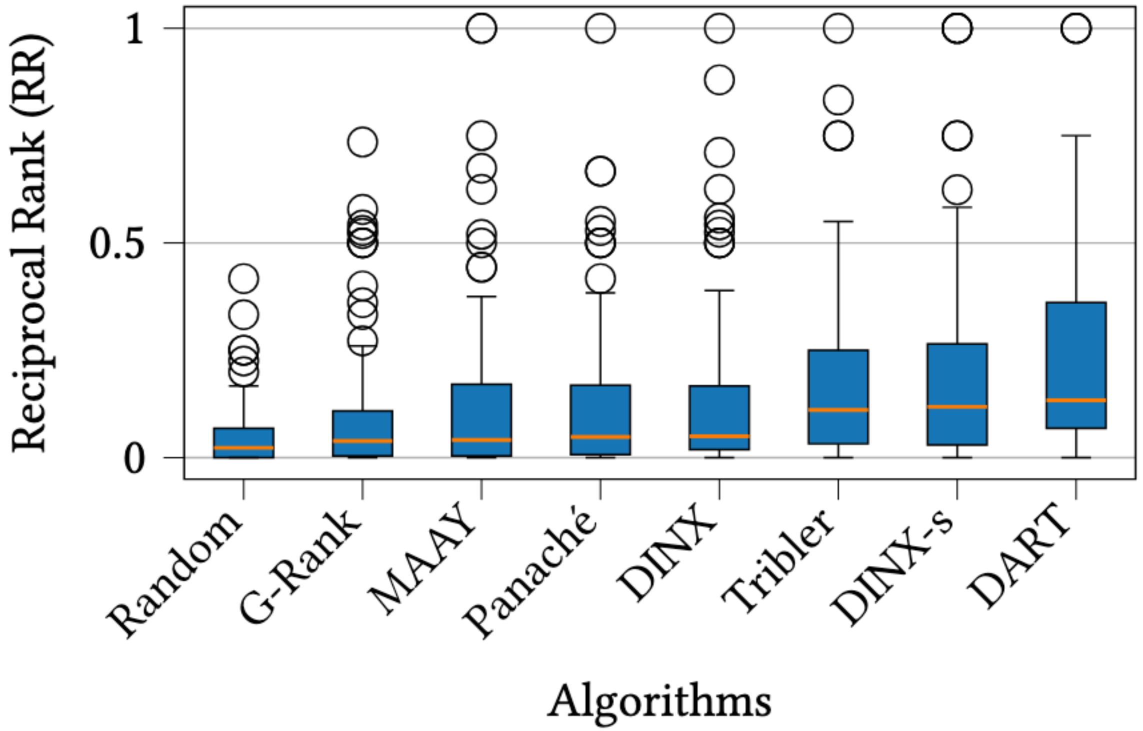
Experiment #2: Impact of Context Size on Performance

We gradually increase the context size and sample 100 clicklogs from the remaining dataset as test set.



Experiment #3: Decentralized Network Simulation

We simulate performance on a 90% split of user's personal clicklogs (with respect to chronology) after training on the total collective of clicklogs in the network.



Conclusion

- Decentralization of any algorithm is challenging and decentralized relevance ranking is a known difficult problem
- We believe that AI may underpin the revival of the P2P movement as Big Tech further gains dominance
- With DART, we established a **new baseline for decentralized relevance ranking**
- In future work, we want to focus on *scalability, privacy, attack-resilience*

Thank you for your attention!

Marcel Gregoriadis