June 2022

Universal Content Recommender

Zhiwei Gu

Will Xi Zhou

Jiahui Liu

Zemin Zhang

Universal Content Recommender

ABSTRACT

Online content providers typically provided personalized content recommendations. Content is available across many domains or verticals. Within each vertical, content can occur in clusters. A cluster is a set of content with correlated viewership. Certain metrics such as user engagement, user appeal, etc., apply to all content verticals while some metrics such as re-engagement, subscriptions, etc., apply only to specific verticals. The vertical-specific metrics are not comparable across verticals. This disclosure describes techniques to rank clusters within verticals that may have different and incomparable metrics or objectives. In a first pass, the clusters are ranked by metrics common across multiple verticals. In a second pass, the clusters with each vertical are re-ranked by metrics or objectives core to that vertical.

KEYWORDS

- Multi-objective optimization
- Recommendation system
- Ranking algorithm
- Content vertical
- Content metric
- Incomparable metric
- Domain-specific metric
- User engagement
- User interaction

BACKGROUND

Online content providers such as streaming services, video hosting and sharing services, application stores, game stores, deal websites, e-commerce sites, online booksellers, ad servers, etc., recommend content based on an understanding of the user's profile. Content provided by such services can comprise different domains or verticals. For example, a movie streaming service may provide content in verticals such as drama, comedy, romance, action, etc. Within each vertical, content clusters are often present. For example, in this context, a cluster can be a set of content with correlated viewership.

Certain metrics or objectives such as user engagement, user appeal, etc., apply to all content verticals. Some metrics e.g., re-engagement, subscriptions, etc., may apply only to specific verticals, and are not comparable across verticals. For example, in an application store that includes the verticals 'games' and 'productivity,' a subscription metric with value 37 on the productivity vertical is not comparable to a re-engagement metric of value 10 on the games vertical: an app with a high number of subscriptions cannot be said to be better than another app with a low (or high) re-engagement metric.

The problem is to rank content with different types of metrics (and purposes) that are served on the same page. Such a problem, known as multi-objective optimization (MOO) has been addressed, e.g., in [1], which lacks ground truth to the weights used in MOO, thus providing an experimental and unscalable solution; in [2], which combines multiple objectives into a single one, resulting, again, in an unknown ground truth of how to weight the differing objectives; in [3], which outputs a combined score using a combination function in the form of laborious, manually-tuned, weighted multiplication; etc.

DESCRIPTION

This disclosure describes techniques to rank clusters within verticals that may have different and incomparable metrics or objectives. In a first pass, clusters are ranked by metrics common to the verticals (if any). In a second pass, the clusters are re-ranked by metrics or objectives core to each vertical. The techniques are deterministic and do not involve multiple rounds of experimentally selected weights or the use of randomization.

Clusters by Vertical 102a	Common Metric 102b	Boost 102c	Core Vertical Metric 102d
A1	0.11	1x	38 (A-metric)
A2	0.08	1x	67 (A-metric)
B1	0.10	1x	0.02 (B-metric)
B2	0.04	1x	0.03 (B-metric)
C	0.03	3x	N/A (no vertical-specific metric)

Vertical A
Vertical B
Vertical C

First pass 104

Clusters by Vertical	Common Metric	Boost	Core Vertical Metric
A1	0.11	1x	38 (A-metric)
B1	0.10	1x	0.02 (B-metric)
C	0.09 (=0.03x3)	3x	N/A (no vertical-specific metric)
A2	0.08	1x	67 (A-metric)
B2	0.04	1x	0.03 (B-metric)

Second pass 106

Clusters by Vertical	Common Metric	Boost	Core Vertical Metric
A2	0.08	1x	67 (A-metric)
B2	0.04	1x	0.03 (B-metric)
C	0.09 (=0.03x3)	3x	N/A (no vertical-specific metric)
A1	0.11	1x	38 (A-metric)
B1	0.10	1x	0.02 (B-metric)

Fig. 1: Two-pass ranking for universal recommendation

Fig. 1 illustrates universal recommendation using a two-pass ranking procedure, per techniques of this disclosure. Verticals are determined by categorizing clusters of content based on their objectives. This results, in this example, in the verticals A, B, and C. Each vertical, by definition, has clusters (102a), e.g., vertical A has clusters A1 and A2; vertical B has clusters B1 and B2; and vertical C has a single cluster. The verticals can have one or more common metrics (102b) that are comparable across verticals. A common metric may have a boost (102c), e.g., a factor that amplifies or attenuates the common metric of a vertical. Each vertical has a core metric (102d) that is not comparable across verticals. For example, vertical A has an A-metric that is not comparable to the B-metric of vertical B, while vertical C is not associated with a vertical-specific metric.

In a first (inter-vertical) pass (104), verticals are ranked by their common metric(s). As explained earlier, example metrics common among verticals include user interaction, user appeal, etc. In this example, although vertical C has a relatively low common metric of 0.03, it receives a 3x boost, such that it ranks above cluster A2 (with common metric 0.08) and below cluster B1 (common metric 0.10).

In a second (intra-vertical) pass (106), verticals as a whole maintain their rankings from the first pass, while clusters within each vertical are re-ranked based on their core metrics. In this example, clusters A1 and A2 interchange positions due to the relative sizes of their A-metrics; clusters B1 and B2 interchange positions due to the relative sizes of their B-metrics; while there are no changes in cluster C ranking. The number of clusters in a given vertical can optionally be limited to a predetermined maximum.

Some advantages of the two-pass universal recommender that can rank clusters across verticals include:

- *No arbitrary assignment of weights to incomparable verticals*: Unlike other techniques, no arbitrary, manually-assigned weights are assigned to metrics of incomparable verticals to force translatability across verticals.
- *No iterative experimentation to determine weights that translate across verticals*: Since no weights are used, the multiple rounds of experimentation typically used to determine weights that translate metrics across verticals are obviated.
- *Scalability*: The introduction of a new vertical-specific or common metric does not result in re-tuning weights, since, contrary to traditional techniques, weights are themselves absent. The procedure for ranking clusters across verticals remains unchanged: as explained before, clusters are ranked by their common metric; then, while maintaining the ranks of their respective verticals, re-ranked by their core metrics.
- *Stability*: Traditional weighted-combination techniques entail the re-tuning of weights based on data distributions, e.g., a change in the statistics of data necessitates a re-tuning of weights. Since the described techniques do not use weights, they are robust to long-term change in distributions (statistics) of data. No weight re-tuning or run-time adjustments are required after launch.

CONCLUSION

This disclosure describes techniques to rank clusters within content verticals that may have different and incomparable metrics or objectives. In a first pass, the clusters are ranked by metrics common across multiple verticals. In a second pass, the clusters with each vertical are re-ranked by metrics or objectives core to that vertical.

REFERENCES

[1] Yuyan Wang, Yuanchi Ning, Isaac Liu, and Xian Xing Zhang, "Food discovery with Uber eats: recommending for the marketplace," https://eng.uber.com/uber-eats-recommending-marketplace/ accessed Feb. 6th, 2022.

[2] Carmel, David, Elad Haramaty, Arnon Lazerson, and Liane Lewin-Eytan. "Multi-objective ranking optimization for product search using stochastic label aggregation." In *Proceedings of The Web Conference 2020*, pp. 373-383. 2020.

[3] Zhao, Zhe, Lichan Hong, Li Wei, Jilin Chen, Aniruddh Nath, Shawn Andrews, Aditee Kumthekar, Maheswaran Sathiamoorthy, Xinyang Yi, and Ed Chi. "Recommending what video to watch next: a multitask ranking system." In *Proceedings of the 13th ACM Conference on Recommender Systems*, pp. 43-51. 2019.