Individual Preference Probability Modeling and Parameterization for Video Content in Wireless Caching Networks

Ming-Chun Lee *Student Member, IEEE,* Andreas F. Molisch *Fellow, IEEE,* Nishanth Sastry, and Aravindh Raman

***Abstract*—Caching of video files at the wireless edge, i.e., at the base stations or on user devices, is a key method for improving wireless video delivery. While *global* popularity distributions of video content have been investigated in the past, and used in a variety of caching algorithms, this paper investigates the *statistical modeling of the individual user preferences*. With individual preferences being represented by probabilities, we identify their critical features and parameters and propose a novel modeling framework by using a genre-based hierarchical structure as well as a parameterization of the framework based on an extensive real-world data set. Besides, correlation analysis between parameters and critical statistics of the framework is conducted. With the framework, an implementation recipe for generating practical individual preference probabilities is proposed. By comparing with the underlying real data, we show that the proposed models and generation approach can effectively characterize individual preferences of users for video content.**

I. Introduction

Data traffic generated by the demand for video content in wireless networks has approximately doubled every year and is expected to continue to grow in the next several years [1]. Conventional approaches, such as using more efficient transceivers, densifying infrastructure, and/or using more spectrum, for supporting the increasing traffic are deemed insufficient or too expensive [2]–[4]. An important alternative that has emerged in the past years is video caching at the wireless edge. Leveraging unique features of video popularity and the low cost of storage resources, video caching has shown its potential and drawn wide attention [2]–[5].

Video content caching has been discussed in different networks with different equipments serving as storage resources [2]–[5]. Femtocaching was first proposed in [6] in which storage resources in low-cost helper nodes are exploited for content caching. This idea is then generalized to exploiting base stations (BSs) in heterogeneous networks in which storage resources of all types of BSs are used to cache video contents and provide the immediate service to users without using backhaul [7]–[9]. The combinations of BS-caching and femtocaching with other techniques, such as scheduling [10], multicasting [11], and multi-antenna processing [12], were also investigated. By exploiting storage resources in cellphones, tablets and laptops, video content caching in devices provides a more direct caching apporach [13], [14]. In this context, self-caching naturally offers video contents in users' own storages without consuming any resources [13], [14]. Moreover, when device-to-device (D2D) communications become widely available [15], D2D-caching, allowing the content accesses from neighboring devices, was first discussed in [16], and subsequently explored in many papers, e.g., [13], [14], [16]–[25]. In particular, the scaling laws of D2D-caching were investigated in [17], [18]. The trade-off between different parameters were investigated in [13], [18], [21], [22]. The influences of user mobility and uncertainty were discussed in [24]. The combination of storage on user devices together with coded multicast has been proposed [25].

While algorithms for wireless video content caching have been widely explored, most of the literature adopts a homogeneous popularity model, i.e., assumes all users have the same file popularity distribution for deciding the caching policy. Clearly this assumption violates the intuition that different users have different tastes and preferences, and the fact that different users have different preferences has been validated in various works [29]–[32]. Thus, designs adopting the homogeneous popularity model are restricted to some extent due to lack of considering the individual user preference. Some approaches exploiting individual preferences for caching or delivering contents in wireless caching networks have recently been discussed and gradually drawn attention [33]–[38]. Moreover, analyses of the individual preferences have demonstrated their capability of offering fundamental insights that might further enhance the system or strategy designs [29], [30].

Although recent literature starts to take individual preferences into consideration, the focus is basically on the policy design and network analysis based on certain abstract mathematical models without the support of real data. In fact, to our best knowledge, the statistical modeling for the individual user preferences based on real-world data has not been well investigated. This paper thus aims to fill this gap. Note that modeling the individual preference of a particular user for recommendation systems, also known as the "Netflix challenge", has been investigated intensely by using learning methods [30]–[32]. However, this is different from the need

This work was supported in part by the National Science Foundation (NSF). This work has been partly presented at the 2017 IEEE Global Communications Conference [45].

M.-C. Lee and A. F. Molisch are with the Department of Electrical and Computer Engineering, University of Southern California, Los Angeles, CA 90089, USA (e-mail: mingchul@usc.edu; molisch@usc.edu).

N. Sastry and A. Raman are with the Department of Informatics, King's College London, London, UK (e-mail: nishanth.sastry@kcl.ac.uk; aravindh.raman@kcl.ac.uk).

to find *statistics* of individual user distributions.[1]

Our model uses hierarchies of probabilities to represent preferences of users. Empirically, video files can be categorized into genres according to their features, and users might have strong preferences toward a few genres [30]. The overall request probability of a user for a file is then modeled as the probability that a user wants a specific genre, and the popularity of a file within this genre. Since the individual preference probabilities of users can be described by the individual popularity distributions and individual ranking orders, statistics of them are respectively investigated using the genre-based structure. We note that, in this paper, we implicitly denote the distribution as the rank-frequency distribution when we use the term: *popularity distribution*. We aim to extract the models and parameterization for these different statistics.

Such modeling and parameterization has to be based on real-world data to be meaningful. We are thus using data from an extensive dataset collected in the U.K. in 2014, namely the usage of the BBC iPlayer [27]–[30].[2] By observing the real data, we identify several important aspects of characterizing individual preferences, and propose a modeling framework for individual preference probabilities. Besides, parameterization of the proposed framework is provided via understanding and modeling the distributions of parameters in the framework. Moreover, to enhance the parameterization, correlations between different parameters and statistics used by the framework are also investigated. By following the modeling framework, an individual preference probability generation approach is proposed via judiciously linking the parameters and models together. We validate the proposed modeling and generation approach using real-world data. The validation results demonstrate that the proposed modeling and generation approach can effectively reproduce important features and statistics of the individual preference probabilities. Therefore the results can be helpful for designing, optimizing, analyzing, modeling, and simulating systems exploiting content caching and delivery as well as for studies of file popularities and user preferences. To be more specific, we have the following contributions:

- We propose a genre-based hierarchical modeling framework to enable the statistical descriptions of the individual preference probabilities. Specifically, we first identify that, instead of using the element-wise description, the individual preference probabilities of a user can be jointly described by the individual popularity distribution and individual ranking order. The genre-based hierarchical modeling approach is then applied to both of them. The corresponding statistical models for the individual popularity distribution and individual ranking order are thus proposed, respectively.
- Since each user owns a parameter set for describing their preference probabilities, the number of parameters for the whole user set is so large that they can only be numerically handled and gradually become impossible to handle when the number of users increases. To resolve this issue, a statistical parameterization is conducted for every parameter in the framework drastically condensing the description. Such statistical parameterization not only simplifies the representation of individual preferences but also enables the proposition of the individual preference generation without a huge parameter set.
- Correlation analyses between parameters in the framework is conducted. The results reveal the critical correlation features of individual preferences and enhance the parameterization.
- By exploiting the framework, modeling, parameterization, and correlation analysis in this work, a complete implementation recipe of individual preference probability generation approach is proposed.[3]
- All results need to be based on real-world data to be meaningful. Thus an extensive dataset from the BBC iPlayer is used for our investigations and validations of all propositions.

The remainder of the paper is organized as follows. Sec. II introduces the basic modeling concepts and describes the necessary tools for manipulating the dataset. The main modeling framework is presented in Secs. III and IV. In Sec. V, parameterization approaches and results are provided. Sec. VI offers the correlation analysis. We propose the individual preference probability generation recipe and conduct the corresponding numerical validations in Sec. VII. We summarize insights and discuss applications of our work in Sec. VIII. Sec. IX concludes this work. Various detail aspects are presented in the appendices.

II. Individual Preference Probability Modeling and Dataset Preparations

A. Modeling on Individual Preference Probability

In this work, we represent the individual preference by the individual user probability, which is defined as the probability that a specific user will in the future request a specific file for watching; multiple views by the same user are thus ignored (i.e., treated the same as single viewing). Since different users can have different preferences, preference probabilities of different users for the same file could be different. We assume that each file can be uniquely assigned to a genre, and there are G genres in the library. Therefore denoting M_g as the number of files in genre g, the total number of files in the library is given by $\sum_{g=1}^{G} M_g$. Given this library, we denote the preference probability of the file m in genre g for user k as $p_{g,m}^k$. Then the following properties must hold: $0 \leq p_{g,m}^k \leq 1, \forall g, m, k$ and $\sum_{g=1}^{G} \sum_{m=1}^{M_g} p_{g,m}^k = 1, \forall k$. We

[1]The Netflix challenge focuses on the per-user perspective while the statistical modeling in this work aims to statistically represent the preferences of the whole user set in the network.

[2]Though not being presented in the main part of this paper, the proposed modeling framework has been used to analyze another dataset in which the data is collected from social media. The results show that the general structure of our modeling framework, and the functional shape of the different curves, carry over very well. The specific parameterizations are, of course, different between the two data sets, since they describe different types of video services. The results of this additional dataset are provided with details in Appendix F.

[3]A complete code for the generation of the individual preference probabilities of users according to the data can be found in [50].

note that, when considering only probability representation for the individual preferences, the impact of loading of users on the system preference, i.e., the global popularity distribution, cannot be modeled. Therefore, the statistical modeling of loading of users is independently investigated as a constituent of system parameters in Sec. V.D, and the loading distribution is used when generating the global popularity. We also note that, roughly, the loading of a user is its number of accesses to the files in the dataset, and the precise definition will later be provided in Sec. V.D.

To characterize individual preference probabilities of users, two important features need to be characterized: individual popularity distributions of files and individual ranking orders of files. Different individual popularity distributions represent different *concentration* rates of popularity distributions that different users might have, and different individual ranking orders represent different preferences for files by ranking files differently. To clarify these concepts, we provide a simple example. We consider two users with different preferences. Suppose that $G = 1$ and $M_1 = 3$. Therefore there are three files in the library. Then suppose we have $p^1_{1,1} = 0.5$, $p^1_{1,2} = 0.3$, $p^1_{1,3} = 0.2$; and $p^2_{1,1} = 0.05$, $p^2_{1,2} = 0.7$, $p^2_{1,3} = 0.25$. Then note that these six popularity values are a complete description, but obviously such a description becomes impossible to handle when considering thousands of files and millions of users. From the description, it can be observed that their popularity distributions are somewhat different, i.e., 0.5, 0.3, 0.2 and 0.7, 0.25, 0.05, respectively, so that the second user has a stronger concentration than the first. In addition, the ranking orders are different, namely 1, 2, 3 and 2, 3, 1, respectively. It can thus be observed that the differences between preferences of users can be fully described by the differences of individual popularity distributions and individual ranking orders. While the above example gives *deterministic* descriptions, in the following we will aim for *stochastic* descriptions of these quantities.

To avoid confusions, in the following sections, we use global popularity/probability of genres/files to denote the popularity/probability of genres/files computed by taking all users into consideration. Conversely, the individual popularity/probability of genres/files is used to denote the popularity/probability computed by considering only a single specific user. In addition, without loss of generality, we consider the indices of genres to follow the descending order of the global popularities of genres, i.e., the global popularity of genre g is larger than the global popularity of genre $g + 1$ for all $1 \leq g \leq G$.

B. Dataset Descriptions and Preprocessing

This work uses an extensive set of real-world data, namely the dataset of the BBC iPlayer [29], [30]. The BBC iPlayer is a video streaming service provided by BBC (British Broadcasting Corporation) that provides video and radio content for a number of BBC channels without charge.[4] Content on the BBC iPlayer is basically available for 30 days after its first appearance [29]. We consider the two datasets covering June and July 2014, which include 192,120,311 and 190,500,463 recorded access sessions, respectively.[5] In each record, access information of the video content contains two important columns: *user id* and *content id*. *user id* is based on the long-term cookies that unqiuely (in an anonymized way) identify users. *content id* is the specific identity that uniquely identifies each video content separately. Although there are certain exceptions, *user id* and *content id* can generally help identify the user and the video content of each access. In addition to access identifications, video files in the BBC iPlayer are annotated with one or more genres.[6] More detailed descriptions of the BBC iPlayer dataset can be found in [29], [30].

To facilitate the investigation, preprocessing is conducted on the dataset. We first define "unique access". By observations, we notice that a user could access the same file multiple times, possibly due to temporary disconnnections from Internet and/or due to temporary pauses raised by users when moving between locations. Since a user is generally unlikely to access the same video after finishing to watch the video within the period of a month,[7] we assume that each user only needs to access a video once, and can cache it for any subsequent views.[8] We therefore consider multiple accesses made by the same user to the same file as a single unique access. We furthermore define a *regular user* as a user with more than 30 unique accesses in a month, and restrict our subsequent investigation to the regular users.[9] We note that the number of unique regular users in June and July are 384,596 and 369,105, respectively.

As described previously, a file could be annotated with no, one, or several genres, and the genre-wise classification is the foundation for characterizing preferences of users in our work. Hence if a file cannot be classified into any genre, i.e., if no genre is annotated on the file, the file is filtered out during

[4]We note that the BBC iPlayer is a massive application, ranked 2 in terms of the load it imposes on UKs networks. Only YouTube had more load than the dataset we consider. Thus, learning statistics and optimizing the network architecture for just this one application is a worthwhile endeavour because it can lead to huge impact on overall traffic of whole countries.

[5]Although what period of the dataset to choose should depend on the specific scenarios and applications, the choice of one-month is reasonable here since the BBC iPlayer assumes a weekly update and the average valid time for the files is approximately 30 days [29].

[6]Notice that there are certain files that are not annotated with any genre. We simply filter them out, as described in the following paragraph.

[7]This statement was partly supported by results in [29], [30]. Please refer to to Secs. III.A and V.A in [30] for details. Besides, by using the real data, we measure the rate that a user might watch the same video on different days within the same month, and obtain the result of approximate 6%, i.e., the combined number of minutes watched for a file from the successive sessions is usually not more than the total duration of the file.

[8]Using the approach that users cache on the first view and replay from local cache for the subsequent views, this can even be applied to any other dataset since the external access pattern is constructed to be similar to the one with the unique access property.

[9]We note that the definition of regular users in this paper is different from our conference version [45] in which the requirement was 100 unique accesses. Thus, the results in the conference version can be viewed as the results for high frequency users while the results here include more ordinary users.

the preprocessing. Besides, if a file is annotated with multiple genres, the file is considered shared by all annotated genres, i.e., each genre is considered accessed $\frac{1}{N}$ times when a file with N annotated genres is accessed. When considering the dataset constituted by regular users, the number of genres in the library is 110.

C. Kullback-Leibler Distance Based Parameter Estimation

In Secs. III and IV, we propose models to fit statistics acquired from the dataset. To find the suitable models and the parameters that best fit the models to the real data, the minimum Kullback-Leibler (K-L) distance approach is adopted and is given by

$$\hat{\mathbf{x}} = \arg\min_{\mathbf{x}} D_{\mathrm{KL}}(\mathbf{x}) = \sum_{m} p_m^{\text{real}} \log \frac{p_m^{\text{real}}}{p_m^{\text{model}}(\mathbf{x})}, \qquad (1)$$

where $\mathbf{x}$ is the vector representation of parameters, p_m^{real} is the probability of outcome m in real data, and $p_m^{\text{model}}(\mathbf{x})$ is the probability of outcome m characterized by the proposed model and $\mathbf{x}$. We note that $p_m^{\text{real}} \log \frac{p_m^{\text{real}}}{p_m^{\text{model}}(\mathbf{x})} = 0$ if $p_m^{\text{real}} = 0$ by definition; $\sum_m p_m^{\text{real}} = 1$; and $\sum_m p_m^{\text{model}}(\mathbf{x}) = 1$. To find a good model of the target statistics, the following steps are basically used: (I) we choose distributions based on visual inspection; (II) we confirm the fitness of the chosen distributions by the above K-L test. Thus, the main justification for adopting the models and set of functions proposed in this work is empirical.[10] We note that the efficacy of the K-L distance can be interpreted from the point of view of information theory. According to [39], the K-L distance is also known as the relative entropy. Thus the K-L distance between the model and the data reflects the increase of entropy when we approximate the distribution of the data by the model. The value of the K-L distance is the number of additional bits (nats) on average we need to use when the code designed for describing the random variable of the modeling distribution is used to describe the random variable embodied by the data. In other word, the K-L distance measures the inefficiency of the model for describing the real data. It should be noted that the K-L based estimation has pros and cons, and the corresponding discussions are provided in Appendix A.

D. Genre-Based Structure and Modeling

In this work, a genre-based structure is adopted for the proposed modeling. This structure is adopted both for pragmatic and fundamental reasons. From a practical point of view, a direct modeling of individual popularities would involve too many parameters (a similar reasoning underlies, e.g., cluster-based modeling of wireless propagation channels). Besides, according to the analysis in [29], [30] and our results, the users show strong preferences for a few specific genres. Thus, the ability to characterizing genre preferences is important for the model. More fundamentally, it is infeasible to formulate the statistics of individual user preferences on files by simply observing the accesses of users: in other words, a user does not have a *probability* to access a specific file - (s)he either requests it or does not. Therefore, instead of directly finding the statistics of file preferences, we first investigate the statistics of genre preferences of users, and then approximate the file preferences within each genre by using the conditional non-user-specific statistics of files in each genre.

Since the preference probabilities of a user are fully described by its corresponding individual popularity distribution and ranking order, we investigate statistics of the individual popularity distribution and ranking order using the genre-based structure in Secs. III and IV, respectively. To provide a clear overview of the proposed modeling framework, a simple two-part summary is provided as follows.

Firstly, to characterize the statistics of the individual popularity distribution, we use the following distributions and models:

- Size distribution (Sec. III.A): since each user is only interested in a small number of genres, we use size distribution to indicate the statistics of *how many genres a user is watching*.
- Individual genre popularity distribution (Sec. III.B): given the number of desired genres for a user, individual genre popularity distribution characterizes *how concentrated the preference for specific genres is*.
- Genre-based conditional popularity distribution (Sec. III.C): we use the genre-based conditional popularity distribution of each genre to approximate the file popularity distribution *within the corresponding genre*.

Secondly, to characterize the statistics of the individual ranking order, we use the following distributions and models:

- Size distribution (Sec. III.A): the size distribution is again used here because it indicates *how many genres we need to rank for a user*.
- Genre appearance probabilities (Sec. IV.A): Since only the desired genres of a user need to be ranked, for a given genre, we use genre appearance probabilities to characterize *the possibilities of genres that are desired by a user*.
- Genre ranking distribution (Sec. IV.B): For a genre, its genre ranking distribution characterizes *the probability distribution in terms of rank for the genre* given that the genre is desired by the user.
- We directly use the global ranking order of files *within each genre* to approximate the individual ranking order for files within the corresponding genre.

Following the description of framework, the parameterizations and correlation analyses of the framework are provided in Secs. V and VI, respectively. Then, to generate individual preference probabilities of a user, a generation approach is proposed and validated in Sec. VII. Specifically, the proposed generation approach first generates parameters according to the parameterization and correlation analysis results. Then individual popularity distributions and ranking orders are generated via using models in the framework. Finally, by linking individual popularity distributions and ranking orders, the desired preference probabilities are generated. The sketch of the generation approach is presented in Fig. 1. We note that

[10]The relevant empirical justifications for the proposed models are provided in Appendix E.

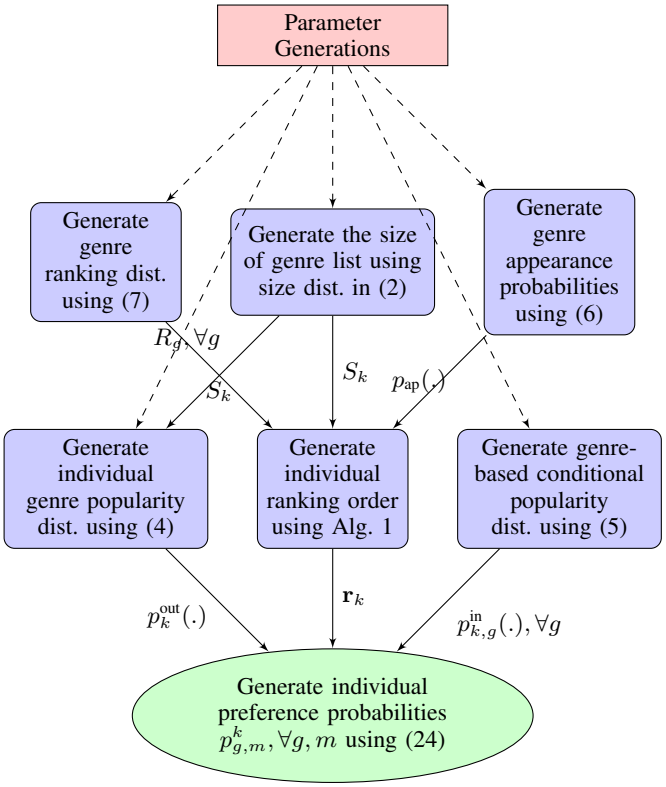


Fig. 1: Sketch of the individual preference probability generation of a user.

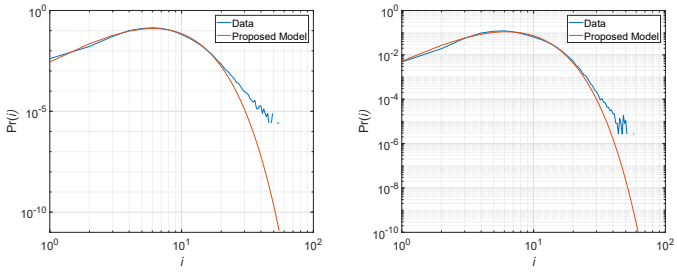
the parameter generations and the complete flow chart of the generation approach specifically used for the adopted dataset are further detailed in Sec. VII and Appendix D.

III. Proposed Modeling of Individual Popularity Distributions

In this section the genre-based structure is adopted and models for describing individual preference popularity are proposed. To be specific, the relevant statistics of genre popularity of users are first investigated. Then the genre-based conditional popularity distributions for files in each genre are investigated. We note that there are differences between the results here and in the conference version [45], whose focus was on the high frequency users.

A. Size Distribution

Here the size distribution is investigated and modeled. By observations from real data, we found that a user would usually access a small number of genres even if there are more than one hundred genres in the library, and even if we consider users that access the iPlayer more than 30 times per month. These observations can be intuitively explained by that people usually have their specific interests which constitute only a small portion of the whole entertainment palette.



(a) June. (b) July.

Fig. 2: Comparisons between the model and real data of size distributions.

To quantify these observations, the size distribution[11] is modeled as

$$\Pr(S_k = i) = \frac{f_i(a^{\text{Si}}, b^{\text{Si}})}{\sum_{j=1}^{M_{\text{Si}}} f_j(a^{\text{Si}}, b^{\text{Si}})} \sim \text{DGamma}(a^{\text{Si}}, b^{\text{Si}}, M_{\text{Si}}) \tag{2}$$

where $i = 1, 2, ..., M_{\text{Si}}$ and M_{Si} is the maximal possible number of genres accessed by a user in the dataset; S_k is the number of genres being accessed by user k; a^{Si} and b^{Si} are parameters that characterize the modeling distribution; and

$$f_i(a, b) = i^{a-1} \exp(-bi). \tag{3}$$

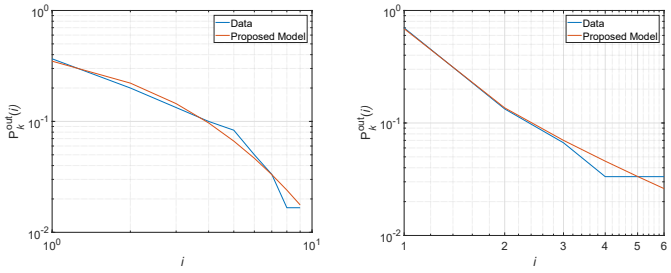
When i is a continuous variable instead of discrete, $f_i(a, b)$ follows the basic expression of Gamma distribution. As a result, (2) is named Discrete Gamma (DGamma) distribution. The fundamental characteristic of the DGamma distribution is that it is a hybrid power and exponential function, which is flexible to represent the cases that increase and decrease are according to the power law, the exponential law, and their mix. We compare the proposed model with the real distribution derived from the dataset in June and July in Figs. 2a and 2b, respectively. Parameters for the model are provided in Table IX in Appendix D. It can be observed that the model is able to well reproduce the size distribution from the real data except for the regime with very low probabilities.

B. Individual Genre Popularity Distribution

The popularity of a genre for a specific user k is defined as the ratio between the number of accesses to the genre by user k and the total number of accesses by the same user. Therefore characterizing the individual genre popularity distribution is to characterize the concentration level of individual popularity in terms of genres, in other words fitting the *sorted* distribution of the genre popularities for this user. The proposed model for this individual genre popularity distribution is the Mandelbrot-Zipf (MZipf) distribution [40]:

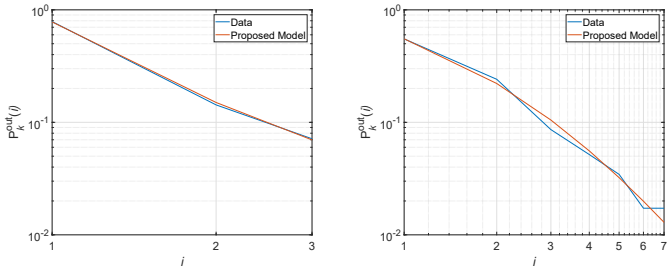
$$P_k^{\text{out}}(i) = \frac{(i + q_k^{\text{out}})^{-\gamma_k^{\text{out}}}}{\sum_{j=1}^{S_k} (j + q_k^{\text{out}})^{-\gamma_k^{\text{out}}}}, \tag{4}$$

[11] We model the number of genres accessed by the user as a random variable described by size distribution. Therefore although the size distribution is non-user-specific, different users can have different numbers of accessed genres.



(a) Case 1. (b) Case 2.

Fig. 3: Exemplary comparisons between the model and real data of individual genre popularity distributions in June.

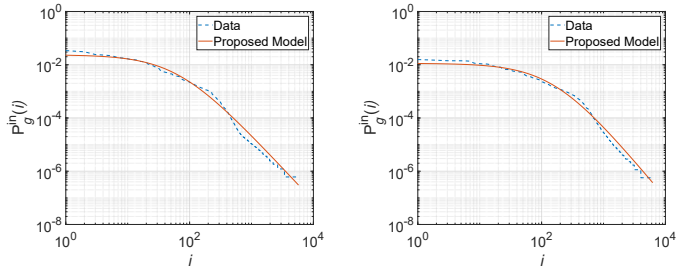


(a) Case 1. (b) Case 2.

Fig. 4: Exemplary comparisons between the model and real data of individual genre popularity distributions in July.

where S_k is the number of genres accessed by user k, $P_k^{\text{out}}(i)$ is the popularity of the ith ranked genre, γ_k^{out} is the Zipf factor, and q_k^{out} is the plateau factor. We note that the MZipf distribution degenerates to a Zipf distribution when $q_k^{\text{out}} = 0$. Since a specific user k would have a specific combination of γ_k^{out} and q_k^{out}, this renders the complete description of all γ_k^{out} and q_k^{out} impossible. As a result, to describe γ_k^{out} and q_k^{out} for all k, a statistical modeling for them is necessary and is presented in Sec. V.A.

In Fig. 3, we provide exemplary comparisons between the model and real data in June on a log-log scale. Parameters of the MZipf distribution are $\gamma_k^{\text{out}} = 5.5, q_k^{\text{out}} = 8.0$ and $\gamma_k^{\text{out}} = 1.2, q_k^{\text{out}} = -0.65$ for Figs. 3a and 3b, respectively. From both figures, it can be observed that the MZipf model can effectively characterizes the real data. In Fig. 4, we provide similar exemplary comparisons for July, and the parameters for Figs. 4a and 4b are $\gamma_k^{\text{out}} = 1.5, q_k^{\text{out}} = -0.5$ and $\gamma_k^{\text{out}} = 4.1, q_k^{\text{out}} = 3.0$, respectively. Again we observe the good fit between the model and real data. From Figs. 3 and 4, we can observe that the curve is concave-like when q_k^{out} is positive and convex-like when q_k^{out} is negative. Note that when $q_k^{\text{out}} = 0$ the curve is affine. Also note that the individual genre popularity distribution only specifies the *concentration rate* of the preference and does not specify *which* genre is the most popular one for this particular user; this aspect of genre ranking will be discussed in Sec. IV.



(a) June. (b) July.

Fig. 5: Exemplary comparisons between the model and real data of genre-based conditional popularity distributions.

C. Genre-Based Conditional Popularity Distribution

The genre-based conditional popularity distribution of a given genre is the conditional probability distribution under the condition that files are annotated with the given genre. We use this distribution to approximate the *per-user* conditional preference probabilities of files under the condition that the file is annotated with the desired genre. We emphasize that the approximation is due to the infeasibility of the direct characterization of user-based file preference statistics as discussed at the beginning of Sec. II.D. Since genre-based conditional popularity distributions are non-user-specific distributions, different users are assumed to have the same distribution for the same genre, though of course the *realizations* of what different users download are different.

To model the genre-based conditional popularity distribution of genre g, we propose to again use the MZipf distribution:

$$P_g^{\text{in}}(i) = \frac{(i + q_g^{\text{in}})^{-\gamma_g^{\text{in}}}}{\sum_{j=1}^{M_g} (j + q_g^{\text{in}})^{-\gamma_g^{\text{in}}}}, \tag{5}$$

where $P_g^{\text{in}}(i)$ is the popularity of the ith ranked file in genre g, γ_g^{in} is the Zipf factor, q_g^{in} is the plateau factor, and M_g is the number of files in genre g. We will again provide the statistical modeling for parameters in (5), and the results are presented in Sec. V.B. In Fig. 5, the model is compared with the real distribution of genre "factual". Parameters of the MZipf distribution for June and July are $\gamma_g^{\text{in}} = 2.5$, $q_g^{\text{in}} = 64$, $M_g = 5751$ and $\gamma_g^{\text{in}} = 2.8$, $q_g^{\text{in}} = 160$, $M_g = 6235$, respectively. From the figures, we observe that the MZipf distribution can effectively model the real distributions.[12]

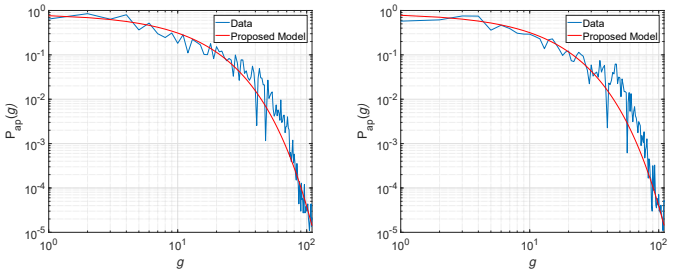
IV. Proposed Modeling of Individual Ranking Orders

In this section, the statistical modeling for individual ranking order is investigated. Identical to the individual popularity distribution case, a genre-based structure is adopted.

A. Genre Appearance Probability

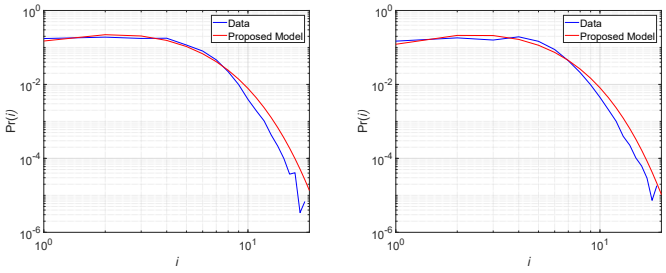
As elaborated in previous sections, the number of genres that a user might access is usually much smaller than the total number of genres in the library. Therefore for each user k,

[12]Of course, the MZipf distribution can effectively model other genres.



(a) June. (b) July.

Fig. 6: Comparisons between the model and real data of genre appearance probabilities.



(a) June. (b) July.

Fig. 7: Exemplary comparisons between the model and real data of ranking distributions.

we can obtain a genre list which collects the genres that are accessed by user k. The number of genres in the genre list of user k is by definition S_k.

Since the genre list of a user explicitly indicates the specific preference of that user on genres, characterizing statistics of the genre list is necessary. Thus, genre appearance probabilities are used. The appearance probability of genre g is defined as the probability of genre g to appear in genre lists of users, and it is the ratio between the number of times that genre g appears in genre lists of users and the number of total users. The proposed model[13] describing genre appearance probabilities is

$$P_{\text{ap}}(g) = N_{\text{ap}} \exp(-\gamma_{\text{ap}} g), \tag{6}$$

where $P_{\text{ap}}(g)$ is the appearance probability of genre g, γ_{ap} is the shaping parameter, and N_{ap} is the scaling parameter. The comparisons between the model and the real data is provided in Fig. 6, and the parameters of the model are offered in Table IX in Appendix D.

B. Genre Ranking Distribution

Given the genre list of a user, the ranking order of genres in the list characterizes the preference of a user. To investigate the statistics of the ranking order, we investigate the ranking distributions of genres. The ranking distribution of a genre g is defined as the distribution of ranks of genre g in genre lists of users conditioning on genre g appearing in those genre lists. By this definition, we denote $\Pr(R_g = i)$ as the probability of genre g to be the ith ranked genre when genre g appears in a genre list. The proposed model[14] for the distribution of this quantity is a DGamma distribution:

$$\Pr(R_g = i) = \frac{f_i(a,b)}{\sum_{j=1}^{G} f_j(a,b)} \sim \text{DGamma}(a_g^{\text{rk}}, b_g^{\text{rk}}, G). \tag{7}$$

The DGamma distribution in (7) follows the same expressions in (2) and (3). We will again provide the statistical modeling of the parameters in ranking distributions in Sec. V.C.

[13]The model here is different from the one in the conference version. In fact, the model in conference version can still be effective after adding an additional scaling parameter as in (6). However, (6) has a more compact expression.

[14]Note that the model for ranking distributions here is again different from the one in the conference version.

In Fig. 7, exemplary comparisons between the model and real data are provided, and we again demonstrate the results of the genre "factual". Parameters for the model are $a_g^{\text{rk}} = 2.95$, $b_g^{\text{rk}} = 0.8$ for June and $a_g^{\text{rk}} = 2.65$, $b_g^{\text{rk}} = 0.75$ for July, and $G = 110$ for both June and July according to the dataset descriptions in Sec. II.B. The results show the good agreement between the model and real data.

V. Statistical Parameterization of the Proposed Modeling Framework

By using the framework and models, the individual preferences can be described via using distributions and probability models in Secs. III and IV, which greatly reduces the complexity for describing a dataset. However, the parameters in the proposed framework still need a numerical description as we have mentioned in our conference version [45], and this numerical description can gradually become impossible to handle when the number of users in a dataset increases. To further reduce the description complexity, in this section, the statistical representations of parameters in the modeling framework are proposed. We note that such representations are expressed either by well-known distributions or by certain specifically designed distributions.[15] Therefore, when dealing with well-known distributions, the standard maximum likelihood (ML) approach in Matlab (2017) is used to fit the real data; on the contrary, when dealing with special distributions, the K-L approach in Sec. II.C is used again. We note that all parameterization results in this section are provided quantitatively in Appendix D with complete details, including the confidence interval calculations. Thus, the details of the model curves in all figures are referred to Appendix D.

A. Statistical Modeling for Parameters of Individual Genre Popularity Distributions

In this subsection, the statistical model of the parameters of individual genre popularity distributions of users, i.e., γ_k^{out} and q_k^{out}, are provided. As indicated in Sec. III.B and in Figs. 3 and 4, the shapes of individual genre popularity distributions can be categorized into different types according to the sign

[15]Since our goal is to further reduce the complexity of expressing our proposed modeling framework, we aim to fit the real data at least to a certain degree even with some artificially constructed distributions.

of q_k^{out}. As a result, it is natural to also characterize the distributions of parameters differently according to the relevant types. Consequently, we provide four distributions which independently model the two parameters with two types: (i) γ_k^{out} whose relevant q_k^{out} is non-negative, i.e., $q_k^{\text{out}} \geq 0$; (ii) $q_k^{\text{out}} \geq 0$; (iii) γ_k^{out} whose relevant q_k^{out} is negative, i.e., $q_k^{\text{out}} < 0$; and (iv) $q_k^{\text{out}} < 0$. In the remaining article, the short-handed descriptions are used for them: (i) γ_k^{out} with non-negative type (NNT); (ii) q_k^{out} with NNT; (iii) γ_k^{out} with negative type (NT); and (iv) q_k^{out} with NT. We note that, by the dataset, the probabilities of having non-negative q_k^{out}, i.e., $q_k^{\text{out}} \geq 0$, when randomly picking a user are $P_{\text{NNT}}^{\text{out}} = 0.795$ and $P_{\text{NNT}}^{\text{out}} = 0.784$ in June and July, respectively. Thus the probability of picking a user who has a negative q_k^{out} is $1 - P_{\text{NNT}}^{\text{out}}$.

Here we provide the statistical models for γ_k^{out} and q_k^{out} with NNT. The model for γ_k^{out} with NNT is a mixed distribution whose probability density function (pdf) is

$$f_{\text{ga,NNT}}^{\text{out}}(x) = c_{1,\text{ga}}^{\text{out}} f_{\text{Gam}}(x; a_{1,\text{ga}}^{\text{out}}, b_{1,\text{ga}}^{\text{out}}) + (1 - c_{1,\text{ga}}^{\text{out}} - c_{3,\text{ga}}^{\text{out}}) \cdot f_{\text{unif}}(x; a_{2,\text{ga}}^{\text{out}}, b_{2,\text{ga}}^{\text{out}}) + c_{3,\text{ga}}^{\text{out}} f_{\text{Gam}}(x; a_{3,\text{ga}}^{\text{out}}, b_{3,\text{ga}}^{\text{out}}), \tag{8}$$

where $f_{\text{Gam}}(x; a, b)$ is a Gamma distribution with pdf

$$f_{\text{Gam}}(x; a, b) = \begin{cases} \dfrac{1}{b^a \Gamma(a)} x^{a-1} \exp\left(-\dfrac{x}{b}\right), & x \geq 0; \\ 0, & \text{otherwise}; \end{cases} \tag{9}$$

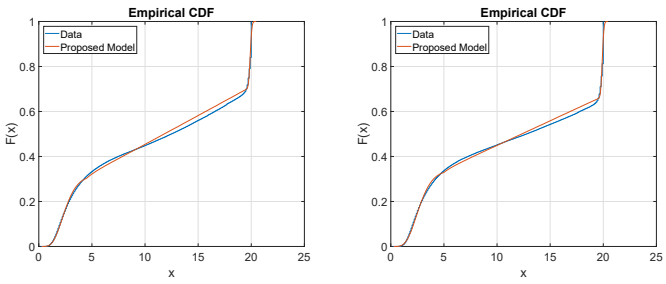
and $f_{\text{unif}}(x; a, b)$ is a uniform distribution with pdf

$$f_{\text{unif}}(x; a, b) = \begin{cases} \dfrac{1}{b-a}, & a \leq x \leq b; \\ 0, & \text{otherwise}. \end{cases} \tag{10}$$

From (8), we can observe that $f_{\text{ga,NNT}}^{\text{out}}(x)$ is a distribution summed by three different constituent distributions with $c_{1,\text{ga}}^{\text{out}}$, $1 - c_{1,\text{ga}}^{\text{out}} - c_{3,\text{ga}}^{\text{out}}$, and $c_{3,\text{ga}}^{\text{out}}$ being their weights, respectively. Then by carefully selecting parameters $a_{1,\text{ga}}^{\text{out}}, b_{1,\text{ga}}^{\text{out}}, ..., b_{3,\text{ga}}^{\text{out}}$, the distribution of γ_k^{out} with NNT is almost identical to a mixed distribution of three distributions with non-overlapping supports. This description can be more clear when observing Figs. 8 and 9. Similar to γ_k^{out} with NNT, the pdf of q_k^{out} with NNT is also a mixed distribution:

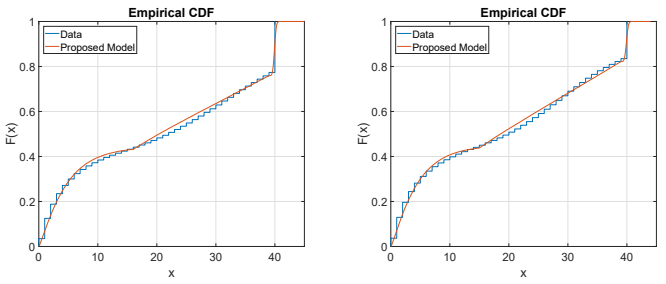
$$f_{\text{q,NNT}}^{\text{out}}(x) = c_{1,\text{q}}^{\text{out}} f_{\text{Gam}}(x; a_{1,\text{q}}^{\text{out}}, b_{1,\text{q}}^{\text{out}}) + (1 - c_{1,\text{q}}^{\text{out}} - c_{3,\text{q}}^{\text{out}}) f_{\text{unif}}(x; a_{2,\text{q}}^{\text{out}}, b_{2,\text{q}}^{\text{out}}) + c_{3,\text{q}}^{\text{out}} f_{\text{Gam}}(x; a_{3,\text{q}}^{\text{out}}, b_{3,\text{q}}^{\text{out}}), \tag{11}$$

where f_{Gam} and f_{unif} are defined in (9) and (10), respectively, and $a_{1,\text{q}}^{\text{out}}, b_{1,\text{q}}^{\text{out}}, ..., b_{3,\text{q}}^{\text{out}}, c_{1,\text{q}}^{\text{out}}, c_{3,\text{q}}^{\text{out}}$ are the modeling parameters. The proposed models of γ_k^{out} and q_k^{out} with NNT are compared with the real data in the form of cumulative distribution function (cdf) in Figs. 8 and 9, respectively, for both June and July. The parameters are given in Table IX in Appendix D. From the figures we can observe the effectiveness of the models, and that the cdfs can be regarded as a three-part function, which gives rise to the idea of using the mixed distribution. We note that since both γ_k^{out} and q_k^{out} with NNT are modeled using specifically designed distributions, they are fitted by the K-L approach.



(a) June. (b) July.

Fig. 8: Comparisons between the model and real data for the distribution of γ_k^{out} with NNT.



(a) June. (b) July.

Fig. 9: Comparisons between the model and real data for the distribution of q_k^{out} with NNT.

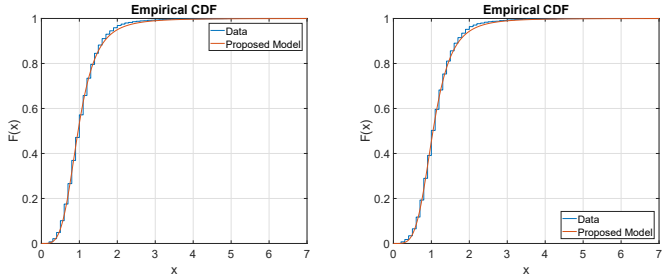
Now we provide the statistical modeling for γ_k^{out} and q_k^{out} with NT. The model for γ_k^{out} with NT is Loglogistic distribution, and the pdf of γ_k^{out} with NT is

$$f_{\text{ga,NT}}^{\text{out}}(x) = f_{\text{Logl}}(x; \mu_{\text{ga}}^{\text{out}}, \sigma_{\text{ga}}^{\text{out}}) = \frac{1}{\sigma_{\text{ga}}^{\text{out}}} \frac{1}{x} \frac{\exp(z)}{(1 + \exp(z))^2}, \tag{12}$$

where $z = \frac{\log(x) - \mu_{\text{ga}}^{\text{out}}}{\sigma_{\text{ga}}^{\text{out}}}$. Note that $f_{\text{Logl}}(x; \mu, \sigma)$ is the standard pdf expression of a Loglogistic distribution whose log-mean and log-scale parameter are μ and σ, respectively. The statistical model of q_k^{out} with NT is a variant of the Beta distribution, namely negative Beta distribution. The pdf of q_k^{out} with NT is

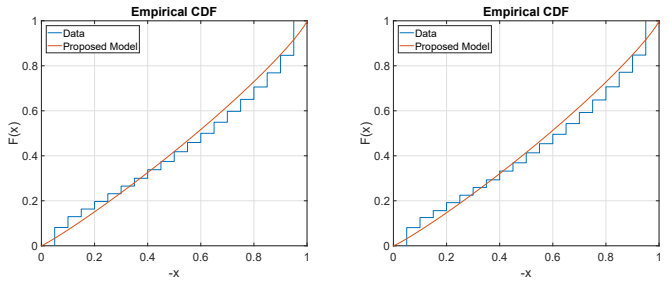
$$f_{\text{q,NT}}^{\text{out}}(x) = f_{\text{Beta}}(-x; a_{\text{q}}^{\text{out}}, b_{\text{q}}^{\text{out}}) = \begin{cases} \dfrac{(-x)^{a_{\text{q}}^{\text{out}}-1}(1+x)^{b_{\text{q}}^{\text{out}}-1}}{B(a_{\text{q}}^{\text{out}}, b_{\text{q}}^{\text{out}})}, & -1 < x < 0; \\ 0, & \text{otherwise}; \end{cases} \tag{13}$$

where $B(a, b)$ is the Beta function. Clearly, (13) is a Beta distribution in which the variable is $-x$ instead of x, and the domain of x is changed to $x \in (-1, 0)$ instead of $x \in (0, 1)$ accordingly. In Figs. 10 and 11, the comparisons between the models and real data are provided for γ_k^{out} and q_k^{out} with NT, respectively. We note that, different from their NNT counterparts, γ_k^{out} and q_k^{out} with NT are modeled using standard distributions and their variants. Therefore they are fitted by using a ML approach.



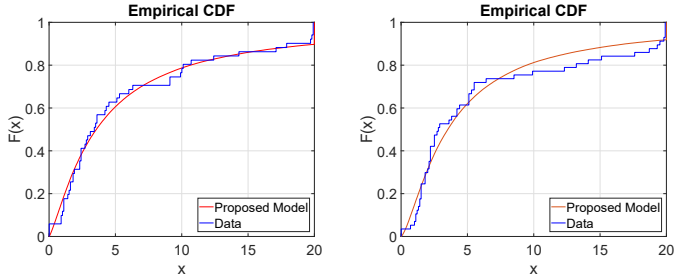
(a) June. (b) July.

Fig. 10: Comparisons between the model and real data for the distribution of γ_k^{out} with NT.



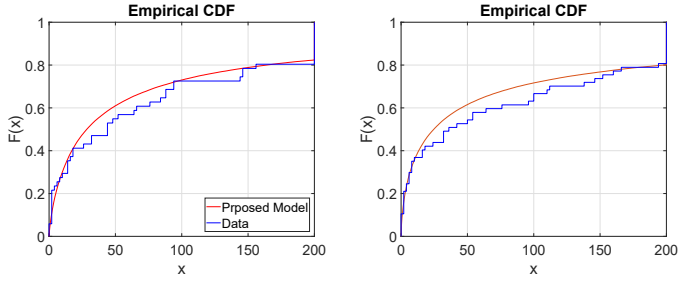
(a) June. (b) July.

Fig. 11: Comparisons between the model and real data for the distribution of q_k^{out} with NT.



(a) June. (b) July.

Fig. 12: Comparisons between the model and real data for the distribution of γ_k^{in} with NNT.



(a) June. (b) July.

Fig. 13: Comparisons between the model and real data for the distribution of q_k^{in} with NNT.

B. Statistical Modeling for Parameters of Genre-Based Conditional Popularity Distribution

We now present the statistical modeling for γ_g^{in} and q_g^{in}. We again consider γ_g^{in} and q_g^{in} with NNT and NT, respectively. We note that, according to the dataset, the probabilities of non-negative q_k^{in} when randomly picking a genre are $P_{\text{NNT}}^{\text{in}} = 0.66$ and $P_{\text{NNT}}^{\text{in}} = 0.71$ in June and July, respectively.

The γ_g^{in} with NNT is modeled by the variant of the Loglogistic distribution, namely, the shifted-and-truncated Loglogistic distribution. The pdf of γ_g^{in} is then given by

$$f_{\text{ga,NNT}}^{\text{in}}(x;\mu_{\text{ga}}^{\text{in}},\sigma_{\text{ga}}^{\text{in}},S_{\text{ga},}^{\text{in}},T_{\text{ga}}^{\text{in}}) = f_{\text{STLogl}}(x;\mu_{\text{ga}}^{\text{in}},\sigma_{\text{ga}}^{\text{in}},S_{\text{ga}}^{\text{in}},T_{\text{ga}}^{\text{in}})$$
$$\begin{cases} \dfrac{1}{\sigma_{\text{ga}}^{\text{in}}}\dfrac{1}{x-S_{\text{ga}}^{\text{in}}}\dfrac{\exp(z)}{(1+\exp(z))^2}, & S_{\text{ga}}^{\text{in}} < x < T_{\text{ga}}^{\text{in}}; \\ \displaystyle\int_{T_{\text{ga}}^{\text{in}}}^{\infty} \dfrac{1}{\sigma_{\text{ga}}^{\text{in}}}\dfrac{1}{x-S_{\text{ga}}^{\text{in}}}\dfrac{\exp(z)dx}{(1+\exp(z))^2}, & x = T_{\text{ga}}^{\text{in}}; \\ 0, & \text{otherwise}, \end{cases} \tag{14}$$

where $z = \frac{\log(x-S_{\text{ga}}^{\text{in}})-\mu_{\text{ga}}^{\text{in}}}{\sigma_{\text{ga}}^{\text{in}}}$, $S_{\text{ga}}^{\text{in}}$ is the shift of the original Loglogistic distribution, and $T_{\text{ga}}^{\text{in}}$ is the truncation parameter of the Loglogistic distribution. The q_g^{in} with NNT is modeled by the truncated Loglogistic distribution, i.e., the shifted-and-truncated Loglogistic distribution with zero shift. Thus the pdf of q_g^{in} with NNT is

$$f_{\text{q,NNT}}^{\text{in}}(x;\mu_{\text{q}}^{\text{in}},\sigma_{\text{q}}^{\text{in}},T_{\text{q}}^{\text{in}}) = f_{\text{STLogl}}(x;\mu_{\text{q}}^{\text{in}},\sigma_{\text{q}}^{\text{in}},S_{\text{q}}^{\text{in}}=0,T_{\text{q}}^{\text{in}}). \tag{15}$$

In Figs. 12 and 13, the models of γ_g^{in} and q_g^{in} with NNT are respectively compared with real data. We note that the model could be made more compact than the expression in (14) since $S_{\text{ga}}^{\text{in}}$ is close to zero when considering the adopted dataset. However, we include $S_{\text{ga}}^{\text{in}}$ in the model to preserve the flexibility when dealing with other potential datasets.[16]

The γ_g^{in} with NT is modeled by Weibull distribution, and its pdf is

$$f_{\text{ga,NT}}^{\text{in}}(x) = f_{\text{WB}}(x;a_{\text{ga}}^{\text{in}},b_{\text{ga}}^{\text{in}})$$
$$= \begin{cases} \dfrac{b_{\text{ga}}^{\text{in}}}{a_{\text{ga}}^{\text{in}}}\left(\dfrac{x}{a_{\text{ga}}^{\text{in}}}\right)^{b_{\text{ga}}^{\text{in}}-1}\exp\left[-\left(\dfrac{x}{a_{\text{ga}}^{\text{in}}}\right)^{b_{\text{ga}}^{\text{in}}}\right], & x \geq 0; \\ 0, & \text{otherwise}. \end{cases} \tag{16}$$

We note that $f_{\text{WB}}(x;a,b)$ is the pdf of a Weibull distribution with scaling parameter a and shaping parameter b. The q_g^{in} with NT is again modeled using negative Beta distribution in (13). To be specific, the pdf of q_g^{in} with NT is

$$f_{\text{q,NT}}^{\text{in}}(x) = f_{\text{Beta}}(-x;a_{\text{q}}^{\text{in}},b_{\text{q}}^{\text{in}}). \tag{17}$$

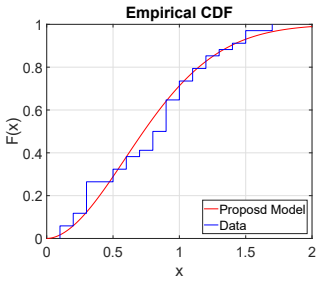
The proposed modeling for γ_g^{in} and q_g^{in} with NT are respectively compared with real data in Figs. 14 and 15.

C. Statistical Modeling for Parameters of Genre Ranking Distribution

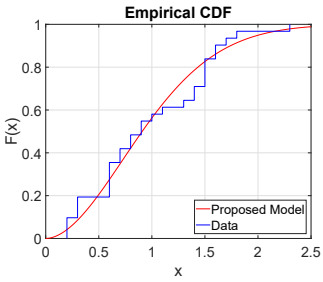
Here the statistical modeling for parameters a_g^{rk} and b_g^{rk} in ranking distributions are provided. The model for a_g^{rk} is Weibull distribution, and its pdf is

$$f_{\text{a}}^{\text{rk}}(x) = f_{\text{WB}}(x;\alpha_{\text{a}}^{\text{rk}},\beta_{\text{a}}^{\text{rk}}), \tag{18}$$

[16]We found $S_{\text{ga}}^{\text{in}}$ to be far from zero when we consider a one-week sub-dataset instead of the complete one-month dataset.

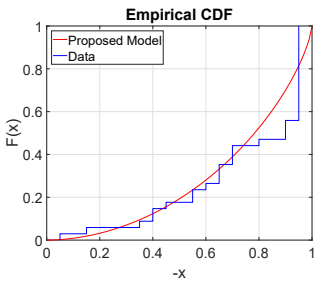


(a) June.

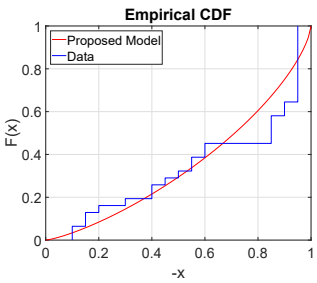


(b) July.

Fig. 14: Comparisons between the model and real data for the distribution of γ_k^{in} with NT.



(a) June.



(b) July.

Fig. 15: Comparisons between the model and real data for the distribution of q_k^{in} with NT.

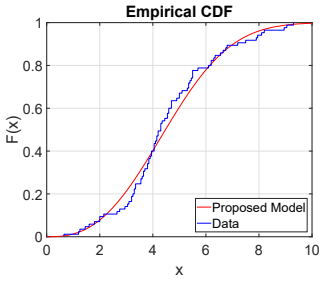
where $f_{\text{WB}}(x;\alpha_{\text{a}}^{\text{rk}},\beta_{\text{a}}^{\text{rk}})$ is described by (16). The model for b_g^{rk} is Gamma distribution, and its pdf is

$$f_{\text{b}}^{\text{rk}}(x) = f_{\text{Gam}}(x;\alpha_{\text{b}}^{\text{rk}},\beta_{\text{b}}^{\text{rk}}), \tag{19}$$

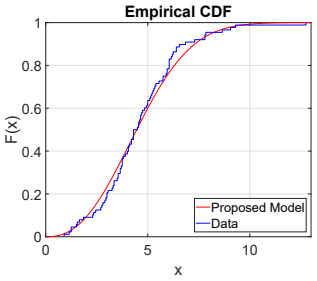
where $f_{\text{Gam}}(x;\alpha_{\text{b}}^{\text{rk}},\beta_{\text{b}}^{\text{rk}})$ is described by (9). The comparisons between the models and the real data for a_g^{rk} and b_g^{rk} are provided in Figs. 16 and 17.

D. Statistical Modeling for User Loading

Here the statistical modeling for the distribution of loading of regular users is provided. Although the loading of users is irrelevant to the individual preferences of users, for the system simulations and for the sake of generating the final global popularity, the characterization of user loading is necessary.

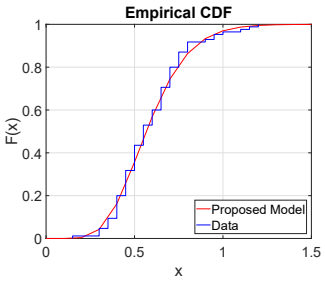


(a) June.

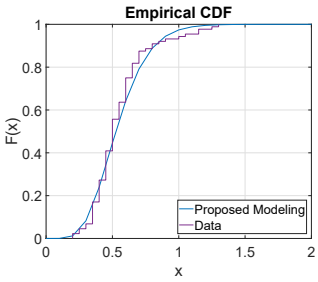


(b) July.

Fig. 16: Comparisons between the model and real data for the distribution of a_g^{rk}.



(a) June.



(b) July.

Fig. 17: Comparisons between the model and real data for the distribution of b_g^{rk}.

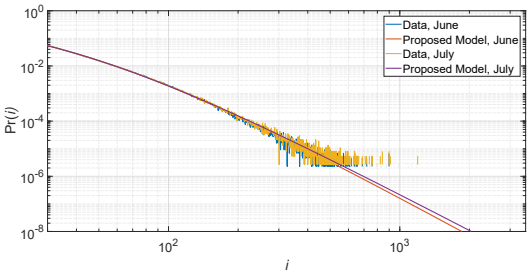


Fig. 18: Comparison between the model and real data of the loading distribution.

Based on the dataset, the loading of a user is given by the number of unique accesses of the user. Thus, the loading of users is always greater than or equal to 30 according to the descriptions in Sec. II.B. The loading distribution is then modeled by the shifted MZipf distribution:

$$\Pr(L_k = i) = \frac{(i - 29 + q^{Ld})^{-\gamma^{\text{Ld}}}}{\sum_{j=30}^{L}(j - 29 + q^{Ld})^{-\gamma^{Ld}}}, i \geq 30, \tag{20}$$

where L_k is the load of the user k and L is the maximum possible load; γ^{Ld} and q^{Ld} are the parameters for the distribution. In Fig. 18, the model is compared with real data.

VI. Correlation Analysis for Parameters of Proposed Modeling Framework

In this section, we investigate the correlation between parameters using both the Pearson correlation coefficient, i.e., linear correlation coefficient, and the Spearman rank correlation coefficient.[17] The reasons for using rank correlation are: (i) to provide the correlation analysis from a ranking perspective since ranking order is an important characteristic in our model; (ii) to allow reconstruction of the correlations between parameters characterized by arbitrary distributions. Note that, when parameters are non-Gaussian or not commonly used multivariate distributions, it is generally impossible to reconstruct their dependence by using only linear correlation information [41]. On the contrary, the reconstruction of rank correlation via copulas can be used generally for almost any distributions [41], [42]. As a result, knowing the rank

[17]Note that there are other types of rank correlation coefficient. The Spearman rank correlation coefficient is one of the most commonly used, and its definition is directly related to the linear correlation coefficient.

correlation is important and its properties are exploited by the proposed individual probability generation in next section.

The Pearson correlation coefficient is defined as

$$\rho_{\text{Ln}}(x,y) = \frac{\text{Cov}(x,y)}{\sigma_x \sigma_y}, \tag{21}$$

where $\text{Cov}(x,y)$ is the covariance of x and y; σ_x and σ_y are standard deviations of x and y, respectively. The Spearman rank correlation coefficient is defined as [43]

$$\rho_{\text{Rn}}(x,y) = \frac{\text{Cov}(r_x,r_y)}{\sigma_{r_x}\sigma_{r_y}}, \tag{22}$$

where r_x and r_y is the corresponding ranking of the original x and y, respectively; $\text{Cov}(r_x,r_y)$, σ_{r_x} and σ_{r_y} then are the covariance and standard deviations of r_x and r_y, respectively. We elaborate r_x using an example. Suppose we have three possible values 0.3, 0.5, 0.7 for x. Their corresponding ranking, i.e., values of r_x then are 1, 2, and 3, respectively. By comparing between (21) with (22), it can be observed that (22) is simply the linear correlation coefficient of the corresponding ranking values of x and y. We note that since the real data is used for conducting the correlation analyses, the corresponding sample-based approaches are then used for finding the results instead of the true expectations.

Here we discuss the correlations between parameters and present some insights. We generally aim to explore the correlation between parameters of the same distribution. This is relevant to determining whether a specific trend of the distribution (jointly determined by parameters of the same popularity distribution) appears more frequently for a popularity distribution. Note that, in this work, parameters are considered correlated only when their absolute values of linear and/or rank correlation coefficients are greater than 0.5. We then focus on the analytical discussions here, and the complete numerical descriptions are provided in Tables I, II, III, and IV in Appendix B.

The results are as follows. The γ_k^{out} and q_k^{out} with NNT are correlated positively in terms of both linear and rank correlation coefficients. In addition, the γ_k^{out} and q_k^{out} with NT are also correlated positively in terms of both linear and rank correlation coefficients. The results indicate that the γ_k^{out} and q_k^{out} could balance each other so that the case in which user has a single highly preferred genre with many other extremely low preferred genres seldom exists. Since a user's preference might be related to its loading and the number of genres of interest, we also explore whether γ_k^{out} and q_k^{out} are correlated to S_k and L_k. The results indicate that there is no significant correlation between them. Also, we notice that that S_k and L_k are not correlated with each other. Thus, even if a user only has a very few number of genres of interest, (s)he can still impose a very high load on the network and vice versa.

We now consider the parameters of genre-based conditional popularity distributions. For γ_k^{in} and q_k^{in} with NNT, the results show that there is only a slight correlation between them in terms of both linear correlation and rank correlation. For the case of γ_k^{in} and q_k^{in} with NT, again only a slight correlation is observed in terms of both linear and rank correlations. Finally we consider the parameters of ranking distributions, i.e., a_g^{rk} and b_g^{rk}. From the results, we observe that they are correlated with each other in terms of linear and rank correlations. Besides, since it is intuitive that a more popular genre (in terms of global ranking order) should have a higher likelihood of having a higher rank, we explore the ranking correlation between a_g^{rk}, b_g^{rk}, and the global ranking. Specifically, we relate the global rank of each genre with their a_g^{rk} and b_g^{rk} and compute the rank correlation. The result shows that the a_g^{rk} and b_g^{rk} are somewhat correlated to the global ranking, indicating that a genre with a higher global rank is likely to have a higher rank among the interests of the users.

VII. Proposed Individual Preference Probability Generation

In this section, we first propose an approach that can generate individual preference probabilities of users according to the models, parameterization, and correlation results in previous sections. Then the effectiveness of the proposed generation approach is validated by comparisons with real-world data.

A. Parameter Generation

To generate the individual preference probabilities of users, the first step is to generate the parameters used by the models in Secs. III and IV via using the results in Secs. V and VI. The parameter generation is based on the rank correlation results and the individual marginal distributions of the parameters. We therefore define the parameter generation function as

$$\mathbf{y}(\mathbf{x}) = \text{G}_{\text{para}}(\mathbf{C}_{\mathbf{x}}^{\text{Rn}}, \{f_{\mathbf{x}}\}), \tag{23}$$

where $\mathbf{x}$ is the parameters to be generated, $\mathbf{y}$ is the generated instance of $\mathbf{x}$, $\mathbf{C}_{\mathbf{x}}^{\text{Rn}}$ is the rank covariance matrix of $\mathbf{x}$, and $\{f_{\mathbf{x}}\}$ is the set of marginal distributions of $\mathbf{x}$. We note that the implementation recipe of $\text{G}_{\text{para}}(\mathbf{C}_{\mathbf{x}}^{\text{Rn}}, \{f_{\mathbf{x}}\})$ is provided in Appendix C. Also note that if a parameter in $\mathbf{x}$ is not correlated with other parameters, it is obvious that the parameter instance of $\mathbf{x}$ can be generated simply by its marginal distribution.

There are two types of parameters: (i) library-based parameters, and (ii) individual-based parameters. Since library-based parameters are determined at the beginning of the generation process of a library and is invariant across users, they either are directly given from the setup or only need a single generation for a particular library. By contrast, the individual-based parameters are generated independently for each user, and different users generally have their own instances of parameters. The library-based parameters are: G, $M_g, \forall g$, a^{Si}, b^{Si}, M_{Si}, γ_g^{in}, q_g^{in}, a_g^{rk}, $b_g^{\text{rk}}, \forall g$, γ_{ap}, N_{ap}, L, γ^{Ld}, and q^{Ld}. The individual-based parameters are: γ_k^{out} and $q_k^{\text{out}}, \forall k$.

Finally, we discuss the sensitivity of the statistics of the parameters with respect to the change of datasets based on the extensive real-world data in June and July. Indeed, the parameterization results show that the statistics of dataset of June and July are quite similar, and most of the fundamental statistics of the parameters in two datasets are close, including a^{Si}, b^{Si}, M_{Si}, a_g^{rk}, γ_{ap}, N_{ap}, L, γ^{Ld}, q^{Ld}, γ_k^{out} and q_k^{out}. In addition, the rest of the parameters are only different in part. Specifically, for γ_g^{in} and q_g^{in}, the differences lie only in the cases

considering their negative types; for b_g^{rk}, the difference lies only at the values of $\alpha_{\text{b}}^{\text{rk}}$. In conclusion, the fundamental statistics of the parameters in the framework is insensitive to the change of time in the scale of one month for our dataset.[18] This also leads to similar global popularity distributions, as we will see in Sec. VII.C. We should note that although their statistics are similar, they are actually different in many details, including the exact file and genre orders and the exact popularity distributions of a genre. We also stress that the conclusion here is only valid for the dataset adopted in our work, and the extension of this conclusion to other timeframes, and in particular to other types of video service such as YouTube or Netflix should undergo a careful examination.

B. Procedure of the Proposed Individual Preference Probability Generation Approach

Here the general procedure of the individual preference probability generation is elaborated. Note that the sketch of the generation approach has already been provided in Fig. 1. To generate the individual preference probabilities of users, we first prepare all the library-specific parameters. Then the genre-based conditional popularity distributions $P_g^{\text{in}}(.), \forall g$, genre appearance probabilities $P_{\text{ap}}(g), \forall g$, and ranking distributions $\{\Pr(R_g)\}, \forall g$, are generated according to (5), (6), and (7), respectively. Note that these distributions are library-specific and are invariant when generating individual preference probabilities of different users.

We next generate the individual popularity distribution of user k. The number of genres in the genre list of user k, i.e., S_k, is first generated according to (2). Then we generate γ_k^{out} and q_k^{out} according to the results in Sec. V.A and (23). Subsequently, the individual genre popularity distribution $P_k^{\text{out}}(.)$ is generated according to (4). The genre list and the individual ranking order of user k are generated according to the proposed ranking order generation approach in Alg. 1. The output of the ranking order generation process is the genre index vector $\mathbf{r}_k$ of user k, where $\mathbf{r}_k$ contains the indices of genres that appear in the genre list. Besides, the order of the indices in $\mathbf{r}_k$ is exactly the ranking order of corresponding genres. Therefore $\mathbf{r}_k$ uniquely specifies the genre list and the ranking order of user k. For example, suppose we have $G = 5$, $S_k = 3$, and $\mathbf{r}_k = [3, 2, 5]$. We know that the genre 2, 3, and 5 are genres in the genre list of user k; genre 3 is ranked 1st; genre 2 is ranked 2nd; and genre 5 is ranked 3rd for user k. For Alg. 1, we provide the following remarks: (i) $\|\mathbf{r}\|_0$ is equal to the number of non-zero entries in $\mathbf{r}$, where $\|\cdot\|_0$ is the L_0 norm; (ii) step 4 is to randomize the filling order of genres at each round; (iii) step 7 is to check whether the genre has already been filled into the genre list; (iv) step 10 is to check whether the selected genre should appear in the genre list, whether the ranking value R is less or equal to the size of the genre list, and whether the genre list is full; and (v) step 16 is to generate the final order of genres in the list according to the generated ranking values. For example, suppose $G = 5$, $S_k = 3$, and $\mathbf{r} = [0, 2, 1, 0, 2]$. We would have $\mathbf{r}_k = [3, 2, 5]$ according to step 16 in Alg. 1.

Equipped with genre-based conditional probability distributions $P_g^{\text{in}}(.)$ and after the generations of the individual preference popularity $P_k^{\text{out}}(.)$ and the genre index vector $\mathbf{r}_k$, individual preference probabilities of user k can then be generated by[19]

$$p_{g,m}^k = f_{k,g}^{\text{out}} \times P_g^{\text{in}}(m), \tag{24}$$

where

$$f_{k,g}^{\text{out}} = \begin{cases} P_k^{\text{out}}(i), \text{ entry } i \text{ of } \mathbf{r}_k = g \\ 0, \text{ otherwise} \end{cases}. \tag{25}$$

Eq. (25) indicates that only genres indexed in $\mathbf{r}_k$ have non-zero preference probabilities, and the preference order is given by the order of indices in $\mathbf{r}_k$. For example, suppose that we have $G = 5$, $S_k = 3$, $P_k^{\text{out}}(1) = 0.5455$, $P_k^{\text{out}}(2) = 0.2727$, $P_k^{\text{out}}(3) = 0.1818$, and $\mathbf{r}_k = [3, 2, 5]$. Then $f_{k,1}^{\text{out}} = f_{k,4}^{\text{out}} = 0$, $f_{k,3}^{\text{out}} = P_k^{\text{out}}(1) = 0.5455$, $f_{k,2}^{\text{out}} = P_k^{\text{out}}(2) = 0.2727$, and $f_{k,5}^{\text{out}} = P_k^{\text{out}}(3) = 0.1818$. We note that, without loss of generality (for the proposed modeling framework), (24) assumes the indices of files within each genre to follow the descending order of the global popularities of files within the genre, i.e., $p_{g,m}^k \geq p_{g,m+1}^k, \forall k$. By combining (24) with (25), the individual preference probabilities $p_{g,m}^k, \forall g, m$ of user k can be obtained. By repeating the procedures in this section, individual preference probabilities of different users can be generated. We note that although L_k is not used for generating the individual preference of a user, it is used when generating the global preference of users since it indicates the traffic generated by each user. The global popularity distribution is generated by

$$p_{g,m}^{\text{Gb}} = \frac{\sum_{k=1}^K p_{g,m}^k \times L_k}{\sum_{g,m} \sum_{k=1}^K p_{g,m}^k \times L_k}, \tag{26}$$

where $p_{g,m}^{\text{Gl}}$ is the global preference probability of file m in genre g, K is the total number of users, and L_k is the loading of user k generated according to (20).

C. Numerical Validations

Here the generation approach is validated by comparing generated results to the underlying real data. To set up the generation approach, basic parameters of models used by the approach need to be specified and are provided in Table IX in Appendix D. We note that since the proposed modeling assumes each file has only one annotated genre while the files could indeed have more than a single annotated genre in the real data, there exists a mismatch for the number of files in each genre between the generation approach and the real data when considering they have the same number of total files. To calibrate, we conduct an adjustment on the numbers of files in genres of the generation approach so that the influences of the

[18]We note that all descriptions here can be quantified by comparing the values in the Tables in Appendix D.

[19]It can be observed that, with the proposed modeling and generator, the file m in genre g is ranked at the mth position in the genre-based conditional popularity distribution of genre g. This is because the non-user-specific genre-based conditional popularity distribution is used to approximate the user preferences of files within the genre, and this index arrangement is used for convenience and without loss of the generality.

Algorithm 1 Proposed Ranking Order Genreation Approach

Input: S_k
Init: a zero vector $\mathbf{r} = \mathbf{0}$
while $\|\mathbf{r}\|_0 < S_k$ **do**
  Create a random permutaion vector $\mathbf{P}_v$ with entries being $2, 3, ..., G$ and create an augmented vector $\mathbf{P} = [1|\mathbf{P}_v]$
  for $i = 1 \rightarrow G$ **do**
    $g = \mathbf{P}(i)$
    if $\mathbf{r}(g) = 0$ **then**
      $t \backsim$ binomial$(1, P_{\text{ap}}(g))$
      $R \backsim \text{DGamma}(a_g^{\text{rk}}, b_g^{\text{rk}}, G)$
      if $t = 1$ and $R \leq S_k$ and $\|\mathbf{r}\|_0 < S_k$ **then**
        $\mathbf{r}(g) = R$
      end if
    end if
  end for
end while
$\mathbf{r}_k$ = arrangement of indices of sort$(\mathbf{r}, \text{ascend})$. Break tie by putting the lower index at the lower order. Ignore indices with corresponding values being zero in $\mathbf{r}$.
return $\mathbf{r}_k$

multi-genre files are accommodated. To clarify the concept used for the calibration, the following example is provided. Suppose we have 1000 files in genre 1 and 2000 files in genre 2 according to real data, but the number of total files is only 2400 because there are 600 files annotated with both genres 1 and 2. This indicates we want $M_1 + M_2 = 2400$ for the generation approach. Then since there are 600 files to be shared by genres 1 and 2, we consider these files contribute $\frac{1}{2}$ to each genre, i.e., a file with 2 annotated genres is counted as $\frac{1}{2}$ file in each genre. Therefore after the calibration, we have $M_1 = 400 + 300 = 700$ and $M_2 = 1400 + 300 = 1700$.[20] Note that all the numbers of files $M_g, \forall g$, after calibration are provided in Tables XI and XII in Appendix D.

In addition to the calibration issue, since the global popularity distribution is highly sensitive to the parameters of genre-based conditional popularity distributions, i.e., $\gamma_g^{\text{in}}, \forall g$ and $q_g^{\text{in}}, \forall g$, to have a highly accurate generation for comparison purpose, in addition to providing the results generated purely by the statistical models in Sec. V.B, we also provide the results for which the numerical values directly derived from the dataset are used for the top 30 ranked genres, i.e., for $\gamma_g^{\text{in}}, g = 1, ..., 30$ and $q_g^{\text{in}}, g = 1, ..., 30$. Their values are provided in Table X in Appendix D. To clarify the sensitivity problem and the reason of using the numerical values, we stress that the real-world data is actually one of all possible instances that can be generated by the proposed generation approach. In other words, when using the statistical approach for generating certain parameters, we are comparing one particular *realization* of the file popularities with another realization (the measured data). Also, since the conditional genre-based popularity distributions whose corresponding genres are popular generally cover a wide range of files, the change of parameters of those popularity distributions indeed influences the final popularity significantly. Moreover, because the generation approach is non-linear, taking the average of the differently generated global popularity distributions does not actually give the equivalent result generated by the average of parameters.

[20]If there exist fractional numbers after the calibration, they are simply rounded to the nearest integer.

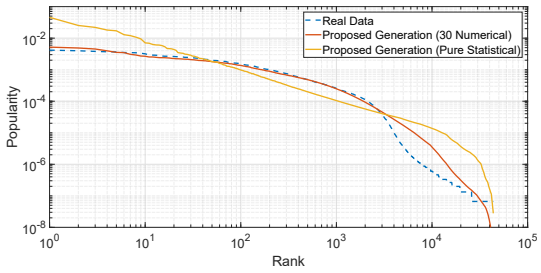


Fig. 19: Comparison between global popularity distributions from proposed generation approach and real data in June.

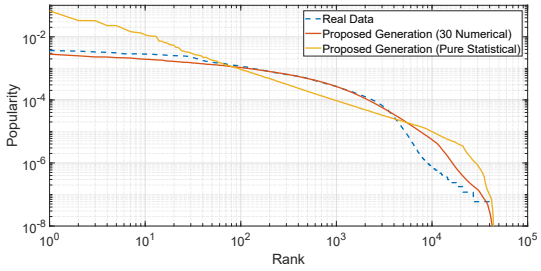


Fig. 20: Comparison between global popularity distributions from proposed generation approach and real data in July.

The validation of the individual components of the model has been provided in previous sections throughout the paper. Hence, as a validation of the complete model, we investigate whether averaging over the obtained individual user distributions provides the total popularity distribution that was independently extracted from the observed data. We note that the complete flow of the generation approach is provided in Fig. 31 in Appendix D, and it is particularly used for the validations in this work. Thus it is also a demonstration of how to use the proposed framework and generation approach to generate the individual preference probabilities and their corresponding global popularity distribution. Fig. 19 compares the global popularity of files of the dataset of June with the global popularity of files constructed by realizations generated by the generator; the same comparison for July is in Fig. 20. The results show good agreement between the model and the data in both figures when the numerical values of γ_g^{in} and q_g^{in} are used for the top 30 ranked genres. We note that the global popularity distributions in Figs. 19 and 20 are close because the statistics of parameters are insensitive to the monthly change as we discussed at the end of Sec. VII.A.

VIII. Summary of Insights and Applications

To finish our discussions, here we first summarize insights of this work and then discuss possible applications. To model the popularity distributions, we introduce the MZipf distribution, which is commonly used for modeling popularity distributions. For most cases of the results, the plateau factor of the MZipf distribution is positive, leading to a flat head followed by a steep tail. The flat head indicates that there is a group of popular files/genres with almost equal popularity and the steep tail represents a group of progressively less

popular files/genres. This implies that the requests are mainly from the group of popular files/genres, and the requests spread out evenly within such group. The results of the size and ranking distributions indicate that each person usually has only a handful of genres of interest. Besides, the genres of interests and the corresponding orders of preferences are different between different individuals. From the statistics of parameters, the positive correlation between parameters of individual genre popularity distributions indicates that the cases that a user has a single highly preferred genre with many other extremely low preferred genres seldom exists. Besides, we observe that the individual genre popularity distribution is not correlated to the number of genres of interest and the loading a user imposes on the network. Furthermore, we did not find obvious correlations between the number of genres of interest and the loading, indicating that even if a user only has a very small number of genres of interest, (s)he can still impose a very high load on the network and vice versa. Finally, we notice that the shape of a ranking distribution is slightly related to the global ranking of that genre. Specifically, when a genre has a higher global rank, it is more likely that its has a higher rank among the interests of the users.

The improved modeling of the popularity distribution, i.e., the modeling involving individual preferences, allows the following applications:

- Adjust the cached content individually to maximize the utility according to the estimated individual preferences of users [29], [30], [33], [46].
- Improve system performance by grouping users with similar preferences to enhance cooperation between users [34], [35], [47].
- Optimize the caching policy by considering the individual preferences of users [36]–[38].
- Decide where and how to store content in intermediate routers by considering the aggregates of interests at the edges of the network adopting an information-centric architecture [48].
- Adjust advertising campaigns to appeal better to consumers of such TV shows.
- Offer foundation of more accurate network analysis.

Recent literature has demonstrated that exploiting the information of individual preferences can further improve different aspects of the systems, while they either offered evaluations using simple individual preference modeling without support by real-world data or spent significant efforts on obtaining the data. Note that since collecting data is very challenging, sometimes, even if a research group spends significant efforts, the volume of the collected data might still be insufficient for providing reliable results. Our work can help in this situation. Based on the real-world data, the proposed modeling framework and parameterization can be used to generate the practical pseudo individual preference probabilities for verifying the investigations considering individual preferences. A demonstration of applying our work is provided in [37].

IX. Conclusions

This paper proposed what is to the best of our knowledge the first modeling framework and corresponding statistical models for individual preference probabilities of users for video content based on real-world data, and following the framework, parameterizations and correlation analyses are conducted. The parameterized model is able to reproduce critical statistics of the individual preferences, and therefore an individual probability generation approach is proposed by judiciously linking those statistics together. The modeling framework is based on, and parameterized by, extensive real-world data sets. The effectiveness of the proposed models and the generation approach was validated.

The framework and methodology presented in this work are capable of being used for other datasets, and the analysis methods and approaches adopted throughout the paper are extensible. Also, the flexibility of the proposed generation approach allows to replace particular fitting distributions if other data sets might indicate a need for such a replacement. In other words, any part of the models can be replaced if necessary, and the generation approach can still be effective as long as the logical flow and critical implementation steps are preserved. On the other hand, parameterization and correlation analysis results depend on the dataset, and there is no guarantee for extending those results to other datasets. Thus, careful examinations should be conducted when considering other datasets.

References

[1] "Cisco Virtual Networking Index: Global Mobile Data Traffic Forcast Updatae, 2016-2021," San Jose, CA, USA.

[2] N. Golrezaei, A. F. Molisch, A. G. Dimakis, and G. Caire, "Femtocaching and device-to-device collaboration: A new architecture for wireless video distribution," *IEEE Commmun. Mag.*, vol. 51, no. 4, pp. 142-149, Apr. 2013.

[3] X. Wang, M. Chen, T. Taleb, A. Ksentini, and V. Leung, "Cache in the air: Exploiting content caching and delivery techniques for 5G systems," *IEEE Commmun. Mag.*, vol. 52, no. 2, pp. 131-139, Feb. 2014.

[4] A. F. Molisch, G. Caire, D. Ott, J. R. Foerster, D. Bethanabhotla, and M. Ji, Caching eliminates the wireless bottleneck in video aware wireless networks, Adv. Elect. Eng., vol. 2014, Nov. 2014, Art. ID 261390.

[5] D. Liu, B. Chen, C. Yang, and A. F. Molisch, "Caching at the wireless edge: Design aspects, challenges, and future directions," *IEEE Commmun. Mag.*, vol. 54, no. 9, pp. 22-28, Sep. 2016.

[6] K. Shanmugam, N. Golrezaei, A. F. Molisch, A. G. Dimakis, G. Caire, "FemtoCaching: Wireless content delivery through distributed caching helpers," *IEEE Trans. Inf. Theory*, vol. 59, no. 12, pp. 8402-8413, Dec. 2013.

[7] C. Yang, Y. Yao, Z. Chen, and B. Xia, "Analysis on cache-enabled wireless heterogeneous networks", *IEEE Trans. Wireless Commun.*, vol. 15, no. 1, pp. 131-145, Jan. 2016.

[8] Z. Chen, J. Lee, T. Q. S. Quek, and M. Kountouris, "Cooperative caching and transmission design in cluster-centric small cell networks," *IEEE Trans. Wireless Commun.*, vol. 16, no. 5, pp. 3401-3415, May 2017.

[9] X. Li, X. Wang, K. Li, Z. Han, and V. C. M. Leun, "Collaborative multi-tier caching in heterogeneous networks: Modeling, analysis, and design," *IEEE Trans. Wireless Commun.*, vol. 16, no. 10, pp. 6926-6939, Oct. 2017.

[10] B. Zhou, Y. Cui, and M. Tao, "Optimal dynamic multicast scheduling for cache-enabled content-centric wireless networks," *IEEE Trans. Commun.*, vol. 65, no. 7 , pp. 2956-2970, Jul. 2017.

[11] Y. Cui and D. Jiang, "Analysis and optimization of caching and multicasting in large-scale Cache-Enabled heterogeneous wireless networks," *IEEE Trans. Wireless Commun.*, vol. 16, no. 1, pp. 250-264, Jan. 2017.

[12] A. Liu and V. K. N. Lau, "Exploiting base station caching in MIMO cellular networks: Opportunistic cooperation for video streaming," *IEEE Trans. Sig. Process.*, vol. 63, no. 1, pp. 57-69, Jan. 2015.

[13] N. Golrezaei, P. Mansourifard, A. F. Molisch, and A. G. Dimakis, "Base-Station assisted device-to-device communications for high-throughput wireless video networks," *IEEE Trans. Wireless Commun.*, vol. 13, no. 7, pp. 3665-3676, Jul. 2014.

[14] Z. Chen, N. Pappas, and M. Kountouris, "Probabilistic caching in wireless D2D networks: cache hit optimal vs. throughput optimal," *IEEE Commun. Lett.*, vol. 21, no. 3, pp. 584-587, Mar. 2017.

[15] K. Doppler, M. Rinne, C. Wijting, C. B. Ribeiro, and K. Hugl, "Device-to-device communication as an underlay to LTE-advanced networks,", *IEEE Commmun. Mag.,* vol. 47, no. 12, pp. 42-49, Dec. 2009.

[16] N. Golrezaei, A. F. Molisch, A. G. Dimakis, and G. Caire, "Femtocaching and device-to-device collaboration: A new architecture for wireless video distribution," *IEEE Commmun. Mag.*, vol. 51, no. 4, pp. 142-149, Apr. 2013.

[17] N. Golrezaei, A. G. Dimakis, and A. F. Molisch "Scaling behavior for device-to-device communications With distributed caching," *IEEE Trans. Inf. Theory*, vol. 60, no. 7, pp. 4286-4298, Jul. 2014.

[18] M. Ji, G. Caire, and A. F. Molisch "Wireless device-to-device caching networks: Basic principles and system performance," *IEEE J. Sel. Area Commun.*, vol. 34, no. 1, pp. 176-189, Jan. 2016.

[19] M. Gregori, J. Gómez-Vilardebó, J. Matamoros, and D. Gündüz, "Wireless content caching for small cell and D2D networks," *IEEE J. Sel. Area Commun.,* vol. 34, no. 5, pp. 1222-1234, May 2016.

[20] D. Malak, M. Al-Shalash, and J. G. Andrews, "Optimizing content caching to maximize the density of successful receptions in device-to-device networking," *IEEE Trans. Commun.*, vol. 64, no. 10, pp. 4365-4380, Oct. 2016.

[21] M. Ji, G. Gaire, and A. F. Molisch, "The throughput-outage tradeoff of wireless one-hop caching networks," *IEEE Trans. Inf. Theory*, vol. 61, no. 12, pp. 6833-6859, Dec. 2015.

[22] B. Chen, C. Yang, and A. F. Molisch, "Cache-enabled device-to-device communications: Offloading gain and energy cost," *IEEE Trans. Wireless Commun.,* vol. 17, no. 7, pp. 4519-4536, Jul. 2017.

[23] B. Chen, C. Yang, G. Wang, "High-Throughput opportunistic cooperative device-to-device communications with caching," *IEEE Trans. Veh. Technol.,* vol. 66, no. 8, pp. 7527-7539, Aug. 2017.

[24] R. Wang, J. Zhang, S. H. Song, and K. B. Letaief, "Mobility-Aware Caching in D2D Networks," *IEEE Trans. Wireless Commun.,* vol. 16, no. 8, pp. 5001-5015, Aug. 2017.

[25] M. A. Maddah-Ali and U. Niessen, "Fundamental limits of caching," *IEEE Trans. Inf. Theory*, vol. 60, no. 5, pp. 2856-2867, May 2014.

[26] Y. Guo, L. Duan, and R. Zhang, "Cooperative local caching under heterogeneous file preferences," *IEEE Trans. Commun.*, vol. 65, no. 1, pp. 444-457, Jan. 2017.

[27] D. Karamshuk, N. Sastry, A. Secker, and J. Chandaria, "ISP-friendly peer-assisted on-demand streaming of long duration content in BBC iPlayer," *IEEE INFOCOM,* 2015.

[28] D. Karamshuk, N. Sastry, A. Secker, and J. Chandaria, "On factors affecting the usage and adoption of a nation-wide TV streaming service," *IEEE INFOCOM,* 2015.

[29] G. Nencioni, N. Sastry, G. Tyson, and *et. al.*, "SCORE: Exploiting global broadcasts to create offline personal channels for on-demand access," *IEEE/ACM Trans. Netw.*, vol. 24, no. 4, pp. 2429-2442, Aug. 2016.

[30] D. Karamshuk, N. Sastry, M. Al-Bassam, A. Secker, and J. Chandaria, "Take-Away TV: Recharging wok commutes with predictive preloading of catch-up TV content," *IEEE J. Sel Commun.*, vol. 34, no. 8, pp. 2091-2101, Aug. 2016.

[31] C. A. Gomez-Uribe and N. Hunt, "The netflix recommender system: Algorithms, business value, and innovation," *ACM Trans. Management Inf. Syst.,* vol. 6, no. 4, pp. 13:113:19, 2016.

[32] L. E. Chatzieleftheriou, M. Karaliopoulos, and I. Koutsopoulos, "Caching-aware recommendations: Nudging user preferences towards better caching performance," *in Proc. INFOCOM*, May 2017.

[33] W. Hoiles, O. N. Gharehshiran, V. Krishnamurthy, N.-D. Dáo, and H. Zhang, "Adaptive caching in the YouTube content distribution network: A revealed preference game-theoretic learning approach," *IEEE Trans. Cogn. Commun. Netw.*, vol. 1, no. 1, pp. 71-84, Mar. 2015.

[34] Y. Guo, L. Duan, and R. Zhang, "Cooperative local caching under heterogeneous file preferences," *IEEE Trans. Commun.*, vol. 65, no. 1, pp. 444-457, Jan. 2017.

[35] Y, Pan, C, Pan, H, Zhu, and *et. al.*, "On consideration of content preference and sharing willingness in D2D assisted offloading," *arXiv preprint*, arXiv:1702.00209, Feb. 2017.

[36] B. Chen and C. Yang, "Caching policy for cache-enabled D2D communications by learning user preference," arXiv preprint, arXiv:1707.08409v1, Jul. 2017.

[37] M.-C. Lee and A. F. Molisch, "Individual preference aware caching policy design for energy-efficient wireless D2D communications," *IEEE GLOBECOM*, Dec. 2017.

[38] D. Liu and C. Yang, "Optimizing Caching Policy at Base Stations by Exploiting User Preference and Spatial Locality," arXiv preprint, arXiv: 1710.09983v1, Oct. 2017.

[39] T. M. Cover and J. A. Thomas, *"Elements of information theory,"* John Wiley & Sons, Inc., 2006.

[40] M. Hefeeda and O. Saleh "Traffic modeling and proportional partial caching for peer-to-peer systems," *IEEE/ACM Trans. Netw.*, vol. 16, no. 6, pp. 1447-1460, Dec. 2008.

[41] P. Embrechts, F. Lindskog, and A. McNeil, "Modelling dependence with copulas and applications to risk management," *Handbook of Heavy Tailed Distributions in Finance,* vol. 1, pp. 329-384, 2003.

[42] "Generate correlated data using rank correlation", Mathworks, USA, 2017. [Online Available]: https://www.mathworks.com/help/stats/generate-correlated-data-using-rank-correlation.html.

[43] J.D. Gibbons, *Nonparametric statistical inference*, Marcel Dekker Inc., 1985.

[44] T. J. DiCiccio and B. Efron, "Bootstrap confidence intervals," *Statistical Science*, vol. 11, no.7., pp. 198-228, 1996.

[45] M.-C. Lee, A. F. Molisch, N. Sastry, and A. Raman, "Individual preference probability modeling for video content in wireless caching networks," *IEEE GLOBECOM*, Dec. 2017.

[46] A. Raman, N. Sastry, A. Sathiaseelan, A. Secker, and J. Chandaria, "Wi-Stitch: Content delivery in converged edge networks," *ACM SIGCOMM Workshop on MECOMM,* 2017.

[47] A. Raman, D. Karamshuk, N. Sastry, J. Chandaria, and A. Secker, "Consume local: Towards carbon free content delivery", *IEEE ICDCS,* Jul. 2018.

[48] G Tyson, N Sastry, R Cuevas, I Rimac, and A Mauthe, "A survey of mobility in information-centric networks," *Communications of the ACM,* vol. 56, no. 12, pp. 90-98, Dec. 2013.

[49] A. Raman, G. Tyson, and N. Sastry, "Facebook (A) Live? Are Live Social Broadcasts Really Broadcasts?" *in Proc. WWW 2018,* Apr. 2018.

[50] M.-C. Lee, A. F. Molisch, N. Sastry, and A. Raman, "Code for generating files with realistic popularity distribution," [URL]: https://wides.usc.edu/research_matlab.html.

APPENDIX A
CHALLENGES, LIMITATIONS, AND DRAWBACKS OF KULLBACK-LEIBLER DISTANCE BASED PARAMETER ESTIMATION

When dealing with a dataset with a large amount of raw data and without much understanding of the properties of the dataset, the estimation of the critical parameters is challenging. This is either because the exact properties are unclear or even because we do not know what parameter is most suitable one to estimate. In such a case, the K-L based estimation is useful because it is simple to implement and is intuitively relevant to the fundamental property of statistics, i.e., the K-L distance.

K-L based estimation also has its limitations and drawbacks. First of all, it could be subject to overfitting, and thus sometimes provides non-smooth results. For example, it can be observed from Figs. 12 and 13 that the tails of the curves are non-smooth. Besides, the performance of the estimation is difficult to analyze. In fact, we need to resort to a numerical approach, such as bootstrapping, to find the confidence intervals as presented in Appendix D. Moreover, it cannot provide any insight for choosing between different models. To be specific, if we keep adding more parameters into the modeling distribution, the K-L distance results of the K-L based estimation might keep improving, which indicates that the additional parameters can provide the better model. However, this might not be true because the increase of number of parameters in a model can have adverse impacts on other aspects, such as complexity and overfitting effect. From these discussions, it can be understood that there are non-trivial issues that can be more profoundly investigated for refining the framework. However, since our focus is to provide a complete modeling, parameterization, and preference generation framework based on the real-world data, treatments of those issues remain topics for future works.

APPENDIX B
DETAILS OF THE CORRELATION ANALYSIS RESULTS

Here the results of correlation analysis are reported in detail. To be specific, we provide several tables to present the correlation coefficients between different parameters and their corresponding 95% confidence intervals.

In Tables I and II, the linear correlation results, i.e., the Pearson correlation coefficients of parameters, are presented for June and July, respectively. Note that the confidence intervals in the tables are computed by using the standard tool in Matlab (2017). From the tables, we can observe that the confidence intervals regarding parameters of genre-based conditional popularity and genre ranking distributions are wide. The main reason could be that we do not have sufficient samples for them. In fact, we can only get around 50 to 110 samples for them because we only have $G = 110$ genres.

In Tables III and IV, the rank correlation results, i.e., the Spearman correlation coefficients of parameters, are presented for June and July, respectively. We note that the correlation coefficient and confidence intervals here are computed by following the definition in (22), i.e., we first convert the original values into their corresponding ranks, and then conduct the linear correlation computation.

TABLE I: Linear Correlation Results of June

Parameters	Correlation	Confidence Interval
γ_k^{out} NNT vs. q_k^{out} NNT	0.806	$[0.805, 0.807]$
γ_k^{out} NNT vs. S_k NNT	-0.080	$[-0.076, -0.072]$
γ_k^{out} NNT vs. L_k NNT	0.015	$[0.011, 0.019]$
q_k^{out} NNT vs. S_k NNT	-0.037	$[-0.041, -0.033]$
q_k^{out} NNT vs. L_k NNT	-0.003	$[-0.007, -0.001]$
S_k NNT vs. L_k NNT	0.236	$[0.232, 0.240]$
γ_k^{out} NT vs. q_k^{out} NT	0.578	$[0.573, 0.583]$
γ_k^{out} NT vs. S_k NT	-0.008	$[-0.016, 0.000]$
γ_k^{out} NT vs. L_k NT	0.036	$[0.029, 0.044]$
q_k^{out} NT vs. S_k NT	0.038	$[0.030, 0.046]$
q_k^{out} NT vs. L_k NT	0.016	$[0.008, 0.024]$
S_k NT vs. L_k NT	0.215	$[0.207, 0.222]$
γ_g^{in} NNT vs. q_g^{in} NNT	0.281	$[0.006, 0.517]$
γ_g^{in} NT vs. q_g^{in} NT	0.366	$[0.032, 0.627]$
a_g^{rk} vs. b_g^{rk}	0.570	$[0.406, 0.698]$

TABLE II: Linear Correlation Results of July

Parameters	Correlation	Confidence Interval
γ_k^{out} NNT vs. q_k^{out} NNT	0.812	$[0.811, 0.813]$
γ_k^{out} NNT vs. S_k NNT	-0.345	$[-0.349, -0.341]$
γ_k^{out} NNT vs. L_k NNT	0.091	$[0.087, 0.095]$
q_k^{out} NNT vs. S_k NNT	-0.166	$[-0.170, -0.162]$
q_k^{out} NNT vs. L_k NNT	0.067	$[0.063, 0.071]$
S_k NNT vs. L_k NNT	0.281	$[0.277, 0.285]$
γ_k^{out} NT vs. q_k^{out} NT	0.605	$[0.600, 0.610]$
γ_k^{out} NT vs. S_k NT	-0.030	$[-0.037, -0.022]$
γ_k^{out} NT vs. L_k NT	0.063	$[0.055, 0.070]$
q_k^{out} NT vs. S_k NT	0.188	$[0.180, 0.195]$
q_k^{out} NT vs. L_k NT	0.068	$[0.060, 0.075]$
S_k NT vs. L_k NT	0.266	$[0.260, 0.274]$
γ_g^{in} NNT vs. q_g^{in} NNT	0.301	$[0.044, 0.521]$
γ_g^{in} NT vs. q_g^{in} NT	0.286	$[-0.076, 0.581]$
a_g^{rk} vs. b_g^{rk}	0.605	$[0.453, 0.723]$

TABLE III: Rank Correlation Results of June

Parameters	Correlation	Confidence Interval
γ_k^{out} NNT vs. q_k^{out} NNT	0.714	$[0.712, 0.716]$
γ_k^{out} NNT vs. S_k NNT	-0.074	$[-0.078, -0.070]$
γ_k^{out} NNT vs. L_k NNT	0.022	$[0.018, 0.026]$
q_k^{out} NNT vs. S_k NNT	-0.024	$[-0.028, -0.020]$
q_k^{out} NNT vs. L_k NNT	-0.002	$[-0.006, 0.002]$
S_k NNT vs. L_k NNT	0.171	$[0.167, 0.175]$
γ_k^{out} NT vs. q_k^{out} NT	0.666	$[0.662, 0.671]$
γ_k^{out} NT vs. S_k NT	0.011	$[0.003, 0.019]$
γ_k^{out} NT vs. L_k NT	0.060	$[0.053, 0.068]$
q_k^{out} NT vs. S_k NT	0.044	$[0.036, 0.052]$
q_k^{out} NT vs. L_k NT	0.021	$[0.013, 0.029]$
S_k NT vs. L_k NT	0.155	$[0.147, 0.162]$
γ_g^{in} NNT vs. q_g^{in} NNT	0.386	$[0.123, 0.598]$
γ_g^{in} NT vs. q_g^{in} NT	0.341	$[0.003, 0.609]$
a_g^{rk} vs. b_g^{rk}	0.625	$[0.475, 0.739]$
a_g^{rk} vs. Global Rank	0.268	$[0.059, 0.455]$
b_g^{rk} vs. Global Rank	-0.303	$[-0.485, -0.096]$

TABLE IV: Rank Correlation Results of July

Parameters	Correlation	Confidence Interval
γ_k^{out} NNT vs. q_k^{out} NNT	0.748	$[0.746, 0.749]$
γ_k^{out} NNT vs. S_k NNT	-0.345	$[-0.349, -0.342]$
γ_k^{out} NNT vs. L_k NNT	0.128	$[0.124, 0.132]$
q_k^{out} NNT vs. S_k NNT	-0.125	$[-0.129, -0.121]$
q_k^{out} NNT vs. L_k NNT	0.082	$[0.078, 0.086]$
S_k NNT vs. L_k NNT	0.249	$[0.245, 0.252]$
γ_k^{out} NT vs. q_k^{out} NT	0.702	$[0.698, 0.706]$
γ_k^{out} NT vs. S_k NT	0.050	$[0.042, 0.058]$
γ_k^{out} NT vs. L_k NT	0.115	$[0.108, 0.123]$
q_k^{out} NT vs. S_k NT	0.210	$[0.202, 0.217]$
q_k^{out} NT vs. L_k NT	0.099	$[0.092, 0.107]$
S_k NT vs. L_k NT	0.274	$[0.267, 0.281]$
γ_g^{in} NNT vs. q_g^{in} NNT	0.503	$[0.279, 0.675]$
γ_g^{in} NT vs. q_g^{in} NT	0.301	$[-0.060, 0.592]$
a_g^{rk} vs. b_g^{rk}	0.610	$[0.459, 0.726]$
a_g^{rk} vs. Global Rank	0.146	$[-0.069, 0.345]$
b_g^{rk} vs. Global Rank	-0.385	$[-0.550, -0.190]$

APPENDIX C
GENERATION OF CORRELATED RANDOM SAMPLES WITH ARBITRARY DISTRIBUTIONS USING RANK CORRELATION

Here the implementation recipe of the rank-based parameter generation defined in (23) is provided. The generation is based on the Gaussian copula and rank correlation and can be realized by "Statistics and Machine Learning" toolbox in Matlab (2017).

We consider the $\mathbf{x}$ whose dimension is N, i.e., we consider N parameters. We suppose the rank correlation of $\mathbf{x}$, i.e., $\mathbf{C}_{\mathbf{x}}^{\text{Rn}}$, and their marginal distributions $\{f_{\mathbf{x}}\}$ are given. To generate the dependent M instances of N parameters, the first step is to generate M samples of each parameter $\mathbf{v}_{\mathbf{x}_1}, ..., \mathbf{v}_{\mathbf{x}_N}$ using their marginal distributions $f_{\mathbf{x}_1}, ..., f_{\mathbf{x}_N}$, where $\mathbf{x}_n$ is the nth parameter in $\mathbf{x}$ and each $\mathbf{v}_{\mathbf{x}_n}$ is a vector with dimension M. This can be implemented using *randsrc* in Matlab. Then step 2 is to convert the $\mathbf{C}_{\mathbf{x}}^{\text{Rn}}$ into its corresponding linear correlation coefficient $\mathbf{C}_{\mathbf{x}}^{\text{Ln}}$ by exploiting properties of Gaussian copula. This can be implemented by using *copulaparam* in Matlab. Step 3 is to generate the Gaussian copula random numbers $\mathbf{u}_{\mathbf{x}_1}, \mathbf{u}_{\mathbf{x}_2}, ..., \mathbf{u}_{\mathbf{x}_N}$ using $\mathbf{C}_{\mathbf{x}}^{\text{Ln}}$, where the dimension of each $\mathbf{u}_{\mathbf{x}_n}$ is M. This step can be implemented using *copularnd* in Matlab. Step 4 is to find the orders of each $\mathbf{u}_{\mathbf{x}_n}$ in ascending order and record the indices of the orders using $\mathbf{i}_n$. Step 5 is to rearrange the order of each $\mathbf{v}_{\mathbf{x}_n}$ by using $\mathbf{i}_n$, and the rearrangement is to let $\mathbf{v}_{\mathbf{x}_n}$ follow the order of $\mathbf{u}_{\mathbf{x}_n}$. The psudocode of the generation approach is provided in Alg. 2 with the corresponding implementation functions in Matlab. Note that the final $\mathbf{y}_n$ in Alg. 2 is the samples for paramter $\mathbf{x}_n$, $\forall n$, and the collection of a n-tuple $(\mathbf{y}_1(m), \mathbf{y}_2(m), ..., \mathbf{y}_N(m))$ is a dependent instance of $\mathbf{x}$. We note that a simple but representative example for this generation approach can be found in [42].

Algorithm 2 Implementation Recipe of the Rank-Based Parameter Generation

1: $\mathbf{v}_{\mathbf{x}_n} = randsrc(M, f_{\mathbf{x}_n})$, $\forall n$; implemented using *randsrc* in Matlab.
2: $\mathbf{C}_{\mathbf{x}}^{\text{Ln}} = copulaparam(Gaussian, \mathbf{C}_{\mathbf{x}}^{\text{Rn}}, type, Spearman)$, $\forall n$; implemented using *copulaparam* in Matlab.
3: $\mathbf{u}_{\mathbf{x}_n} = copularnd(Gaussian, \mathbf{C}_{\mathbf{x}}^{\text{Ln}}, M)$, $\forall n$; implemented using *copularnd* in Matlab.
4: $\mathbf{i}_n = sort(\mathbf{u}_{\mathbf{x}_n})$, $\forall n$; implemented using *sort* in Matlab.
5: $\mathbf{y}_n(\mathbf{i}_n) = sort(\mathbf{v}_{\mathbf{x}_n})$, $\forall n$.

APPENDIX D
DETAILS OF THE PARAMETERIZTION RESULTS AND INDIVIDUAL PREFERENCE PROBABILITY GENERATION APPROACH

In this appendix, the complete parameterization results of the models are first provided. Then the complete implementation flow chart of the generation approach specifically used for the numerical results in Section VII.C is presented. This implementation can be regarded as an example of using the modeling framework and generation approach. Finally, we offer the additional details for generating the numerical results.

The complete parameterization results and their corresponding 95% confidence intervals are provided in Table IX at the end of the appendices. We note that when a parameterization is conducted by using ML approach, the confidence interval is provided simply by standard approach (from Matlab toolbox). However, when considering parameterization using K-L approach, the confidence interval calculation is via the bootstrapping [44], which is a Monte-Carlo based approach, with 1000 bootstrapping samples. Note that the bootstrapping confidence interval calculations can be implemented by using the function *bootci* in the "Statistics and Machine Learning" toolbox in Matlab (2017). The results are shown in Table IX for June and July, respectively. We note that since the parameters directly given by the setup or environment do not have confidence intervals, we note their confidence intervals as "None" in the tables.

With those fundamental parameters being specified, we then can generate the individual preference probabilities of users via using the modeling framework and generation approach. Therefore we provide a complete flow diagram of the implementation recipe of the generation approach in Fig. 31 at the end of the paper. In the figure, the rectangular blocks whose corners are sharp, i.e., the red blocks, correspond to the steps of the parameter generation; the rectangular blocks whose corners are round, i.e., the blue blocks, correspond to the steps of the main modeling framework; and the ellipse blocks, i.e., the green blocks, correspond to the final generated results. We note that the flow in Fig. 31 somehow corresponds to the adopted dataset in this work in the sense that the detailed structure of parameter generation is constructed according to the parameterization and correlation analysis results of the adopted dataset, i.e., we jointly generate those parameters that are correlated with one another. This also implies that if the parameterization and/or correlation results are different (when another dataset is used), the structure of the parameter generations needs to be fine-tuned accordingly. We also note that the $M_g, \forall g$, i.e., the numbers of files of each genre, are parameters determined by the emulation setup.

Finally, as mentioned in Sec. VII.C, the numerical validations in Figs. 19 and 20 require the specifically provided numerical values of $\gamma_g^{\text{in}}, q_g^{\text{in}}, g = 1, ..., 30$, and the calibrated numerical values of $M_g, \forall g$, of the dataset. Their values are

TABLE V: K-L Distance Results of Proposed Models in Secs. III and VI, and loading distribution

Modeling Target	Dataset	K-L Distance
Individual genre popularity	June	0.0277
Individual genre popularity	July	0.0279
Genre-based conditional popularity	June	0.0731
Genre-based conditional popularity	July	0.0770
Size distribution	June	0.0067
Size distribution	July	0.0070
Genre appearance probability	June	0.0701
Genre appearance probability	July	0.0832
Ranking distribution	June	0.0311
Ranking distribution	July	0.0321
Loading distribution	June	0.0015
Loading distribution	July	0.0021

respectively provided in Tables X, XI and XII at the end of the appendices.

APPENDIX E
EMPIRICAL JUSTIFICATIONS FOR THE PROPOSED MODELING

Here we report the K-L distance and K-S test results of our models. We first report the average K-L distance for models in Secs. III and IV, and the loading distribution in Table V. From the Table, we can see that the K-L distances are all small, indicating the effectiveness of the proposed models. For the statistical representation provided in Sec. V, our goal is to reduce the description complexity of the parameter set. We thus express the parameterization either by well-known distributions or by certain specifically designed distributions. For the specifically designed distributions, we provide the K-L distance to show that our representation is effective; for the well-known distributions, we conduct the K-S test at the standard significance level of 0.05 to show that our representations are effective. Note that when a K-S test cannot reject the null hypothesis, it indicates we cannot say the proposed modeling distribution is not equivalent to the statistics of the real data. The K-L distance results are shown in Table VI; and the K-S test results are shown in Table VII. All results show that our statistical representations are effective except for the γ_k^{out} NT and q_k^{out} NT. This is because, in these cases, we compare the quantized real data with continuous distributions, and the K-S test is conducted with a large number of samples (more than 70000 samples). Note that the parameterization of the real data is quantized because it is not possible to exhaustively search for the best parameters located in a real domain without quantization. However, although they did not pass the K-S test, we can still see from the Figs. 10 and 11 that the statistical representations fit the real data very well. To make the above statement more concrete, we compare γ_k^{out} NT and q_k^{out} NT with their corresponding quantized modeling distribution and show the results using K-L distance in Table VIII. We can see that the K-L distances are small, indicating good fit between the proposed modeling and real data.

TABLE VI: K-L Distance Results of the Specifically Designed Distributions in Sec. V

Modeling Target	Dataset	K-L Distance
γ_k^{out} NNT	June	0.0865
γ_k^{out} NNT	July	0.0808
q_k^{out} NNT	June	0.0352
q_k^{out} NNT	July	0.0361

TABLE VII: K-S Test Results of the Well-Known Distributions in Sec. V

Modeling Target	Dataset	K-S Test
γ_k^{out} NT	June	The test reject the null hypothesis
γ_k^{out} NT	July	The test reject the null hypothesis
q_k^{out} NT	June	The test reject the null hypothesis
q_k^{out} NT	July	The test reject the null hypothesis
γ_g^{in} NNT	June	The test cannot reject the null hypothesis
γ_g^{in} NNT	July	The test cannot reject the null hypothesis
q_g^{in} NNT	June	The test cannot reject the null hypothesis
q_g^{in} NNT	July	The test cannot reject the null hypothesis
a_g^{rk}	June	The test cannot reject the null hypothesis
a_g^{rk}	July	The test cannot reject the null hypothesis
b_g^{rk}	June	The test cannot reject the null hypothesis
b_g^{rk}	July	The test cannot reject the null hypothesis

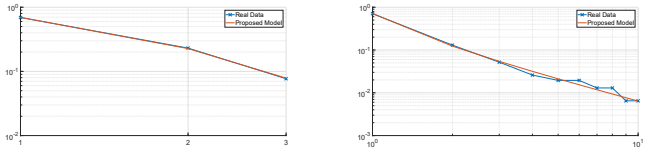
APPENDIX F
INDIVIDUAL PREFERENCE MODELING OF FACEBOOK DATASET

Here we use another dataset, which is from the records of Facebook, to validate our proposed modeling framework and show the generality and extensibility. In particular, we will see that the general structure of our modeling framework, and the functional shape of the different curves, carry over very well. The specific parameterizations are, of course, different between the two data sets, since they describe different types of video services.

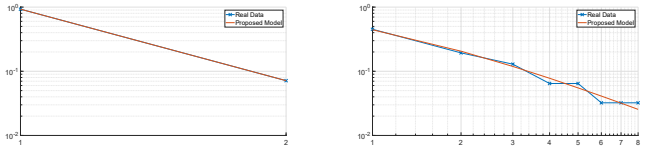
The Facebook dataset contains records from the on-demand video accesses, which are firstly generated in the form of live videos in a live social broadcast platform, Facebook Live [49], and then change to on-demand videos after the conclusion of the live broadcast. We collected a large-scale dataset comprising of interactions from users during eight months. As a part of the crawl, we collect all the comments made during the eight months for the videos that were made available after the live broadcast. Since comments are one of the major forms of the user interaction with the video, we use the records of comments as the indications of accesses from the users. While we realize that the commenters are only a subset of the viewers, this was the only way data could be obtained. Since we are interested in the genre of the user-access, we focus on the page videos, which are typically maintained by various organizations, e.g., political parties,

TABLE VIII: K-L Distance Results of the Quantized Well-Known Distributions in Sec. V

Modeling Target	Dataset	K-L Distance
γ_k^{out} NT	June	0.0188
γ_k^{out} NT	July	0.0192
q_k^{out} NT	June	0.0572
q_k^{out} NT	July	0.0527

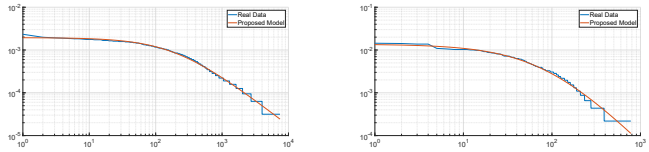


(a) Case 1. (b) Case 2.

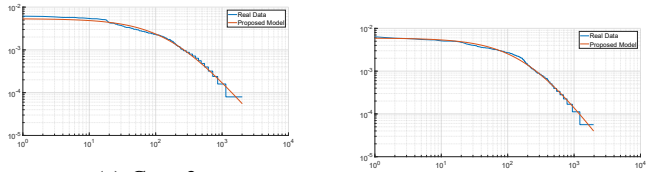


(c) Case 3. (d) Case 4.

Fig. 21: Exemplary comparisons between the model and real data of individual genre popularity distributions.



(a) Case 1. (b) Case 2.



(c) Case 3.

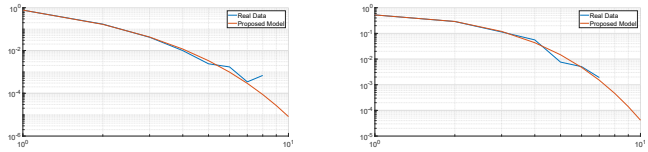
Fig. 22: Exemplary comparisons between the model and real data of genre-based conditional popularity distributions.

news channels, and sports teams, as the videos published from a page are usually tagged with the categories. Note that the number of genres in this dataset is 35. To this end, we have collected 3.8M users accessing 123K categorized videos, which we will use for the analysis of individual preference modeling.[21] While the amount of data is not as much as the BBC iPlayer and might not be able to provide reliable results, we believe it includes a significant proportion from a new and widely accessed live medium to indicate that our modeling approach is applicable to more scenarios than the BBC iPlayer. We emphasize that collecting data is very challenging, due to privacy considerations. We also note that although both the BBC iPlayer and Facebook datasets contain on-demand video streaming, they indeed feature different types of contents.

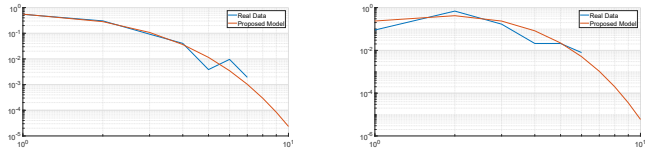
Based on the Facebook dataset, we apply the proposed modeling framework. We again consider the unique access feature and consider only those users that have at least 10 unique accesses. The genre-based structure is again used, and the modeling results are demonstrated in the following. Fig. 21 shows the results of the individual genre popularity distributions. From the figures, we can observe good matches between the real data and the proposed model. The results of the genre-based conditional popularity distributions are shown in Fig. 22. Again, we see that the proposed model can fit the real data. We note that considering all the data on hand, the MZipf distribution can effectively model all the distributions involving genre popularity.

In Fig. 23, we compare the real data with the proposed model of the ranking distribution. From the figures, we can see that all the figures show an excellent match between the proposed model and real data except for Fig. 23d, which was specifically chosen to demonstrate one of the few cases in which the real data does not match the proposed model so effectively. Rather, the results match the model proposed in our conference version [45], i.e., the double-sided Zipf distribution, which corresponds to the case where we consider only high frequency users; the possible explanation is that the Facebook dataset on hand might still not be large enough

[21]Though we collected data from so many users, only approximately seven thousand of them are useful for analysis, i.e., only few of them can provide at least 10 identifiable unique accesses.



(a) Case 1. (b) Case 2.



(c) Case 3. (d) Case 4.

Fig. 23: Exemplary comparisons between the model and real data of ranking distributions.

to include a sufficient amount of regular users, leading to the higher emphasis on high frequency users. We compare the proposed models of the appearance probabilities, size distribution, and loading distribution with the real data in Figs. 24, 25, and 26, respectively. From all figures, we can again observe excellent matches between the proposed models and the real data. Also, it is interesting to mention that, different from the results of the BBC iPlayer, the Facebook results demonstrate the concentration on the rank to be equal to one or two and the size of genre list to be equal to one. In other words, in the Facebook dataset, most people concentrate merely on one or two areas of interest.

Finally, we validate the proposed generation approach without considering the statistical representations of parameters proposed in Sec. V, i.e., we simply using the numerical values of the models in Secs. III and IV to generate the global popularity distribution. This is because from the results of the parameterization, the number of genres, i.e., 35 genres, appears to be insufficient to obtain the statistical representations of the parameters. This will be discussed more in the next paragraph. In Fig. 27, the global popularity distribution of the real data is compared with the one generated by the proposed approach. From the figure, we can see that the proposed approach can generate a result close to the real

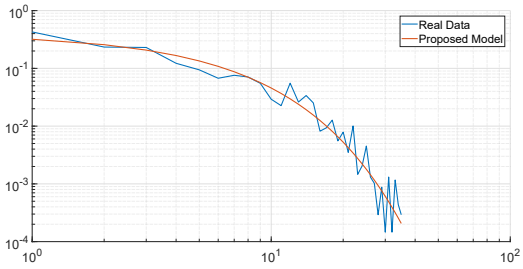


Fig. 24: Comparisons between the model and real data of genre appearance probabilities.

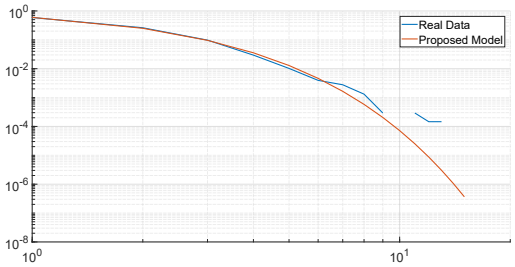


Fig. 25: Comparisons between the model and real data of the size distribution.

data. This validates the effectiveness of the proposed models and the generation approach. In summary, the above results validate that the proposed modeling framework is effective when another dataset is adopted. While of course this is not a conclusive proof that the framework will work for all possible datasets, it is at least indicative that two quite different types of video services can be described well by our model, and thus might serve for other researchers in the modeling of their data.

We now turn to the statistical representations of parameters in the modeling framework for the Facebook dataset, corresponding to Sec. V of the manuscript. In Fig. 28, the statistical representations of parameters of individual genre popularity distributions are demonstrated. It can be observed that the proposed statistical modeling is effective while the specific values of parameters are different. Also, we can see that the plot of the real data exhibits some jumps, indicating that the data volume of the Facebook dataset might not be sufficient. In

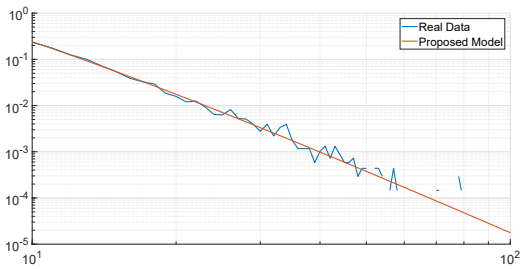


Fig. 26: Comparisons between the model and real data of the loading distribution.

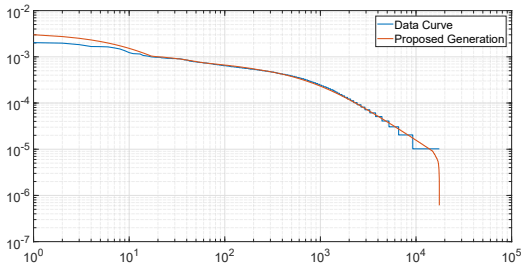
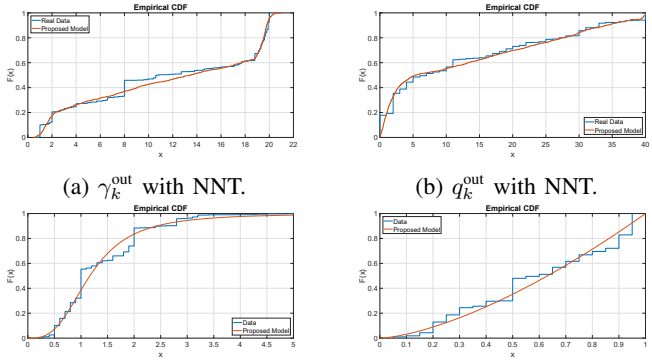


Fig. 27: Comparison between global popularity distributions of files from proposed generation approach and real data.



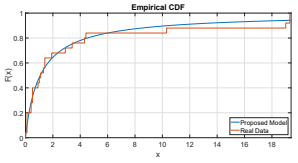
(a) γ_k^{out} with NNT. (b) q_k^{out} with NNT.

(c) γ_k^{out} with NT. (d) q_k^{out} with NT

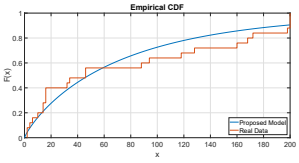
Fig. 28: Comparisons between the model and real data.

Fig. 29, the statistical representations of parameters of genre-based conditional popularity distributions are demonstrated. From the figures, we can see that the proposed model is effective, while the amount of data is somewhat insufficient because we can only have 35 different genres in the Facebook dataset.

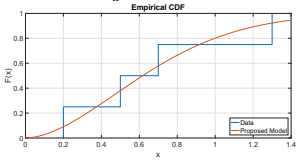
Finally, we present the statistical representations of parameters of ranking distributions in Fig. 30. From the figures, we can see that the proposed model cannot successfully characterize the negative part of a_g^{rk} of the real data. Note that the proposed model can match the real data if we exclude the negative values of a_g^{rk} in the real data. Although the results in Figs. 29 and 30 show that the proposed models might be able to characterize the real data in some cases, the results are considered unreliable because we only have 35 genres. On the other hand, since we only have a small number of genres, the complexity of using the purely numerical values instead of the statistical representations for these parameters might be acceptable in this case. Overall, the proposed statistical representations of parameters are effective in several cases, while there are some exceptions to handle. That being said, the parameterization results in terms of the specific shaping of the statistical representations are different from those in the BBC iPlayer. This reflects the statement in the main body of the manuscript that "the conclusion of the specific values in the modeling is valid only for the adopted dataset, and its extension of this conclusion to other types of video service should undergo careful examination."



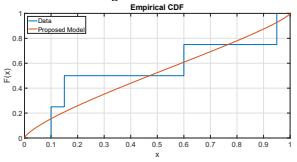
(a) γ_g^{in} with NNT.



(b) q_g^{in} with NNT.

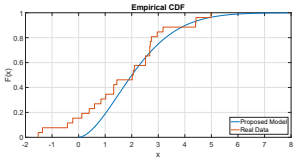


(c) γ_g^{in} with NT.

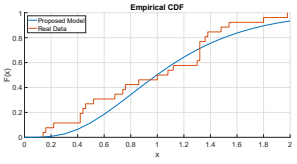


(d) q_g^{in} with NT

Fig. 29: Comparisons between the model and real data.



(a) α_g^{rk}.



(b) b_g^{rk}.

Fig. 30: Comparisons between the model and real data.

TABLE IX: Parameterization Results

Parameter	Value (June)	Confidence Interval (June)	Value (July)	Confidence Interval (July)
G	110	None	110	None
a^{Si}	4.95	$[4.93, 4.97]$	4.10	$[4.10, 4.10]$
b^{Si}	0.65	$[0.65, 0.65]$	0.50	$[0.50, 0.50]$
M_{Si}	55	None	64	None
Γ_{ap}	0.10	$[0.10, 0.10]$	0.10	$[0.1, 0.1]$
N_{ap}	0.84	$[0.84, 0.84]$	0.86	$[0.86, 0.86]$
$P_{\text{NNT}}^{\text{out}}$	0.784	None	0.795	None
$a_{1,\text{ga}}^{\text{out}}$	7.31	$[6.70, 8.38]$	7.91	$[7.31, 9.03]$
$b_{1,\text{ga}}^{\text{out}}$	0.35	$[0.27, 0.39]$	0.34	$[0.27, 0.38]$
$a_{2,\text{ga}}^{\text{out}}$	4.5	$[3.87, 4.82]$	4.7	$[4.12, 5.00]$
$b_{2,\text{ga}}^{\text{out}}$	19.6	$[19.55, 19.67]$	19.6	$19.54, 19.70]$
$a_{3,\text{ga}}^{\text{out}}$	24360	$[21764, 27952]$	25148	$[21853, 30849]$
$b_{3,\text{ga}}^{\text{out}}$	0.0008	$[0.0007, 0.0009]$	0.0008	$[0.0007, 0.0009]$
$c_{1,\text{ga}}^{\text{out}}$	0.31	$[0.29, 0.33]$	0.33	$[0.30, 0.34]$
$c_{3,\text{ga}}^{\text{out}}$	0.30	$[0.29, 0.31]$	0.34	$[0.33, 0.35]$
$a_{1,\text{q}}^{\text{out}}$	1.20	$[1.14, 1.25]$	1.24	$[1.20, 1.30]$
$b_{1,\text{q}}^{\text{out}}$	3.88	$[3.41, 4.35]$	3.60	$[3.06, 3.87]$
$a_{2,\text{q}}^{\text{out}}$	16	$[14.7, 17.4]$	15	$[13.5, 15.8]$
$b_{2,\text{q}}^{\text{out}}$	39	$[39, 39]$	39	$[39, 39]$
$a_{3,\text{q}}^{\text{out}}$	26836	$[26109, 27521]$	23713	$[23027, 24393]$
$b_{3,\text{q}}^{\text{out}}$	0.0015	$[0.0015, 0.0015]$	0.0017	$[0.0016, 0.0017]$
$c_{1,\text{q}}^{\text{out}}$	0.44	$[0.43, 0.45]$	0.45	$[0.43, 0.46]$
$c_{3,\text{q}}^{\text{out}}$	0.24	$[0.24, 0.24]$	0.18	$[0.18, 0.18]$
$\mu_{\text{ga}}^{\text{out}}$	-0.034	$[-0.037, -0.031]$	0.042	$[0.039, 0.045]$
$\sigma_{\text{ga}}^{\text{out}}$	0.247	$[0.246, 0.248]$	0.232	$[0.230, 0.233]$
$a_{\text{q}}^{\text{out}}$	1.080	$[1.067, 1.092]$	1.099	$[1.086, 1.111]$
$b_{\text{q}}^{\text{out}}$	0.850	$[0.839, 0.861]$	0.855	$[0.844, 0.867]$
$P_{\text{NNT}}^{\text{in}}$	0.71	None	0.66	None
$\mu_{\text{ga}}^{\text{in}}$	1.27	$[0.90, 1.63]$	1.26	$[0.94, 1.58]$
$\sigma_{\text{ga}}^{\text{in}}$	0.80	$[0.37, 1.40]$	0.72	$[0.58, 0.89]$
$S_{\text{ga}}^{\text{in}}$	0.1	None	0.1	None
$T_{\text{ga}}^{\text{in}}$	20	None	20	None
$\mu_{\text{q}}^{\text{in}}$	3.36	$[2.77, 3.96]$	3.19	$[2.52, 3.87]$
$\sigma_{\text{q}}^{\text{in}}$	1.26	$[1.00, 1.59]$	1.52	$[1.21, 1.89]$
S_{q}^{in}	0.0	None	0.0	None
T_{q}^{in}	200	None	200	None
$a_{\text{ga}}^{\text{in}}$	0.88	$[0.74, 1.07]$	1.10	$[0.91, 1.35]$
$b_{\text{ga}}^{\text{in}}$	1.87	$[1.42, 2.47]$	1.85	$[1.39, 2.46]$
a_{q}^{in}	1.87	$[1.18, 2.97]$	1.28	$[0.61, 2.70]$
b_{q}^{in}	0.72	$[0.37, 1.41]$	0.68	$[0.29, 1.61]$
$\alpha_{\text{a}}^{\text{rk}}$	5.13	$[4.72, 5.59]$	5.18	$[4.73, 5.68]$
$\beta_{\text{a}}^{\text{rk}}$	2.66	$[2.26, 3.13]$	2.43	$[2.08, 2.83]$
$\alpha_{\text{b}}^{\text{rk}}$	9.26	$[6.89, 12.45]$	7.59	$[5.68, 10.13]$
$\beta_{\text{b}}^{\text{rk}}$	0.063	$[0.047, 0.086]$	0.072	$[0.054, 0.098]$
γ^{Ld}	4.6	$[4.54, 4.73]$	4.4	$[4.34, 4.54]$
q^{Ld}	64	$[62.84, 67.68]$	60	$[58.86, 62.74]$
L	3197	None	3080	None

TABLE X: Numerical Results of γ_g^{in} and q_g^{in}

g	γ_g^{in} (June)	q_g^{in} (June)	γ_g^{in} (July)	q_g^{in} (July)
1	1.9	200	1.9	200
2	5.2	200	2.2	110
3	2.9	146	4.5	200
4	2.5	64	2.8	160
5	2.5	200	2.6	200
6	10.1	200	5.6	200
7	6.1	94	3.7	152
8	4.6	32	5.4	138
9	2.5	16	2.6	50
10	5.4	200	2.3	18
11	4.4	88	2.9	32
12	2.4	6	5.6	54
13	2.4	14	6.5	200
14	17.2	200	18.8	200
15	1.9	14	1.9	6
16	10.8	200	20.0	96
17	9.2	66	17.7	200
18	1.7	4	15.2	166
19	4.3	44	1.6	8
20	20.0	156	8.6	64
21	10.2	48	3.9	42
22	20.0	76	10.0	200
23	12.5	52	2.3	16
24	3.6	26	19.9	76
25	3.5	8	19.6	36
26	1.6	2	5.2	32
27	20.0	88	5.2	54
28	13.0	12	2.6	8
29	9.2	84	13.3	100
30	14.4	146	1.5	4

TABLE XI: Numerical Results of M_g of June

g	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
M_g	2955	996	1103	4548	820	882	842	617	2080	344	1379	1564	3042	223	1694	238	283
g	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34
M_g	3025	1016	56	203	80	166	787	357	1818	43	16	191	277	604	12	294	56
g	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51
M_g	58	228	16	244	1565	795	1889	208	57	147	16	367	25	776	202	18	32
g	52	53	54	55	56	57	58	59	60	61	62	63	64	65	66	67	68
M_g	412	359	12	267	954	5	23	250	39	6	357	7	439	13	208	7	103
g	69	70	71	72	73	74	75	76	77	78	79	80	81	82	83	84	85
M_g	97	58	34	116	4	131	3	60	58	16	38	34	50	62	25	1	3
g	86	87	88	89	90	91	92	93	94	95	96	97	98	99	100	101	102
M_g	17	37	12	5	22	11	19	25	18	4	7	12	5	5	4	4	6
g	103	104	105	106	107	108	109	110									
M_g	6	6	2	4	1	1	2	1									

TABLE XII: Numerical Results of M_g of July

g	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
M_g	2827	1468	1034	5042	865	1004	568	738	1501	2033	3234	617	330	91	3046	160	210
g	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34
M_g	259	1722	280	889	259	1648	87	33	134	271	829	194	1760	45	331	323	171
g	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51
M_g	3	58	231	26	1544	1666	19	520	13	34	18	11	46	365	219	44	541
g	52	53	54	55	56	57	58	59	60	61	62	63	64	65	66	67	68
M_g	60	341	16	3	29	601	297	9	252	40	411	1104	373	309	79	6	227
g	69	70	71	72	73	74	75	76	77	78	79	80	81	82	83	84	85
M_g	23	9	40	38	200	56	81	3	24	11	120	14	3	16	91	4	16
g	86	87	88	89	90	91	92	93	94	95	96	97	98	99	100	101	102
M_g	7	46	47	44	7	23	14	5	1	18	9	7	9	4	6	11	4
g	103	104	105	106	107	108	109	110									
M_g	5	3	3	4	4	4	1	1									

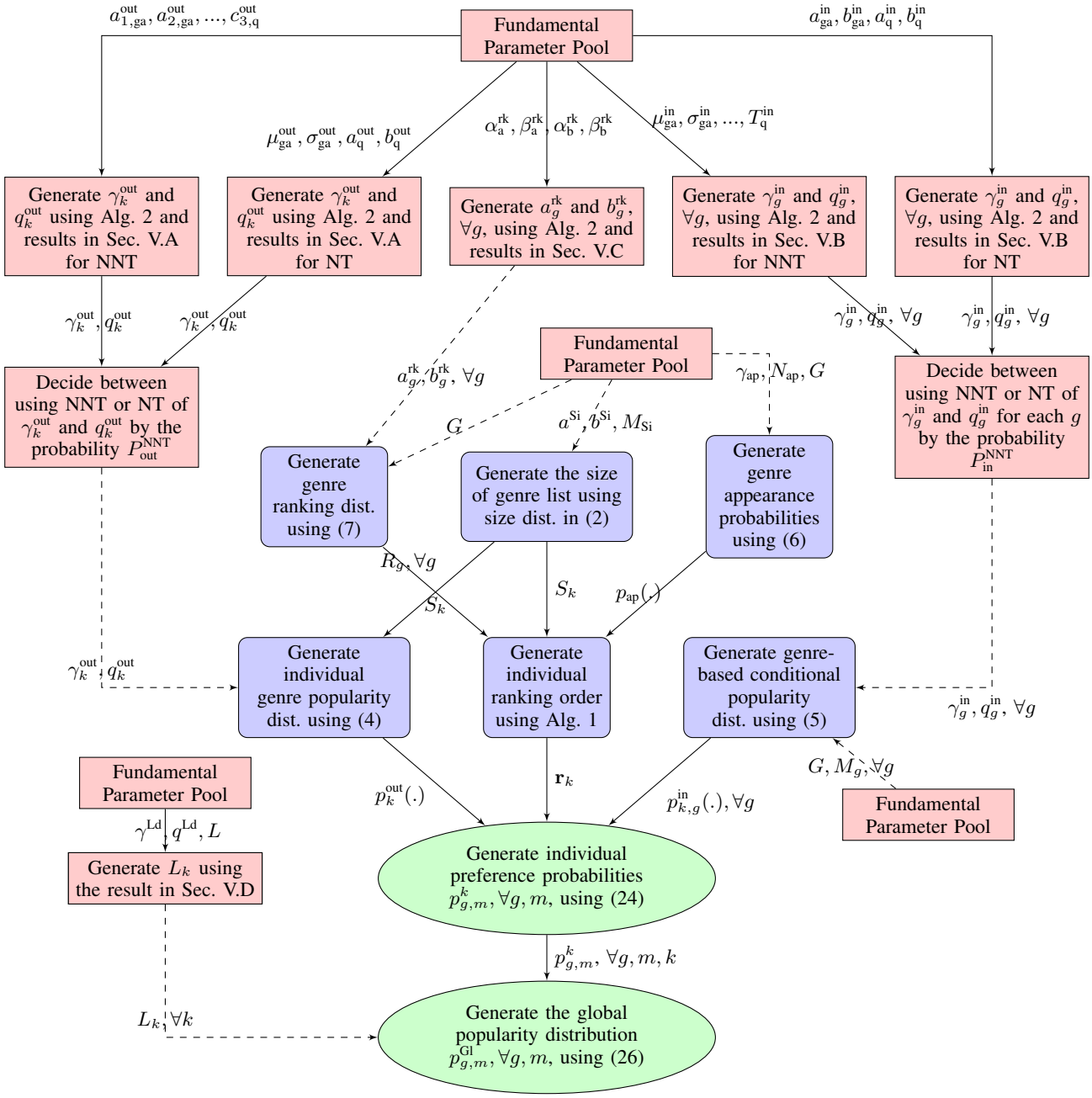


Fig. 31: Complete flow diagram of the individual preference probability and global popularity distribution generation.