Article

AMERICAN MARKETING ASSOCIATION

Journal of Marketing Research
2025, Vol. 62(6) 1101-1124

DOI: 10.1177/00222437251349798
journals.sagepub.com/home/mrj
Sage

Modeling Categorized Consumer Collections with Interlocked Hypergraph Neural Networks

Khaled Boughanmi, Asim Ansari, and Yang Li

Abstract

Consumers curate collections of items for various reasons and categorize them into subsets or categories based on different criteria as their collections grow. The items in a collection reflect a consumer's preferences, and the categories provide insights into the different contexts in which items are consumed. The authors develop a novel deep generative modeling framework that captures the network structure of consumer collections using multiple interlocked hypergraphs. This model employs message-passing variational autoencoders that leverage hypergraph structures and entity-specific covariates to generate probabilistic deep embeddings for consumers, items, and item categories. Applying this framework to digital music collections and playlists of music consumers, the authors demonstrate that the model outperforms several sophisticated benchmarks in predicting linkages within these collections. They then illustrate how this approach enables firms to generate novel personalized product bundles, recommend relevant items and bundles, and dynamically expand existing bundles with new items. Beyond the music application, this method is broadly applicable to other consumer collections, such as food recipes and content collections on social curation platforms like Pinterest.

Keywords

consumer collections, categorization, hypergraphs, generative neural networks, music industry, playlist generation, playlist recommendation

Online supplement: https://doi.org/10.1177/00222437251349798

Submitted July 28, 2023

Consumers curate product collections and organize them into subsets that represent distinct consumption contexts. A collection is defined as a set of items of the same type; for example, a consumer's book collection consists of books they own in their library. With the rapid advancement of technology, this practice has expanded to include digital collections. Notable examples of digital assets that people collect include music in the form of songs, albums, and playlists; e-books in Kindle libraries; YouTube video playlists; game-based virtual items and skins; digital art in the form of nonfungible tokens; digital trading cards; and podcasts. Consumers not only collect and categorize these items but also share them with others. Platforms such as Pinterest and Goodreads facilitate and profit from these social curation activities, highlighting the growing role of digital collections in consumer behavior.

As a collection grows with the addition of novel items, consumers tend to categorize and organize them into subsets or categories. They often rely on various organizational principles based on genres or subgenres; creators, authors, or artists; chronological order of production; underlying themes and experiences; or the ability of items to fulfill specific consumption goals (Ratneshwar et al. 2001). Such categorization enables consumers to retrieve items more efficiently, create customized bundles for various consumption contexts, and share their collections with like-minded individuals through discussions, online forums, and social media platforms.

With the growing popularity of digital collections, vast amounts of online data on consumer collections and their categorization are becoming increasingly accessible. Companies collect and profit from data on consumer playlists, podcast bundles, books, videos, bookmarked content, and other digital assets. Notable examples include Spotify and Last.fm, which track consumer music playlists; YouTube, which gathers data on video collections; and Amazon, which maintains records of books purchased and stored in Kindle libraries. The unique

Khaled Boughanmi is Assistant Professor of Marketing, SC Johnson Graduate School of Management, Cornell University, USA (email: kb746@cornell.edu). Asim Ansari is the William T. Dillard Professor of Marketing, Columbia University, USA (email: maa48@columbia.edu). Yang Li is Associate Professor of Marketing, Cheung Kong Graduate School of Business, China (email: yangli@ckgsb.edu.cn).

items within a consumer's collection reflect their tastes and preferences, while the categories within these collections provide valuable insights into how items are perceived and the contexts in which they are consumed.

Despite their business significance, the marketing literature has given limited attention to the quantitative modeling of consumer collections and their categorization structures. One of the key challenges lies in the complex structure of collection data, which typically comprises multiple data modalities, including structured information on product attributes and consumer characteristics as well as unstructured data such as textual descriptions and user-generated tags, especially in online digital collections. Also, collections exhibit multiple layers of heterogeneity—consumers differ not only in the specific items they collect but also in how they categorize these items into subsets with varying compositions. Since the same item can appear in multiple consumer collections and across different subsets or bundles, conventional network modeling approaches, such as those based on ordinary graphs, struggle to capture these complex relationships. This limitation arises as traditional graph-based models rely on dyadic edges that link only two nodes at a time, whereas collections, categories, and items exhibit higher-order relationships that require a more expressive representation.

To address this challenge, we introduce a novel generative deep learning framework that leverages interlocked hypergraphs to model digital consumer collections. Hypergraphs, a class of higher-order networks (Bianconi 2021), generalize ordinary graphs by incorporating vertices and hyperedges, where each hyperedge can connect multiple vertices simultaneously. This property makes hypergraphs particularly well-suited for representing complex relationships, such as songs grouped in a playlist or ingredients combined in a recipe.

In this article, we construct and interlock three hypergraphs to jointly model consumers, items, and item categories as heterogeneous but interconnected entities (Hajij et al. 2022; Papillon et al. 2023). Specifically, the first, a user–item hypergraph, captures preference structures by treating each consumer as a hyperedge that connects all the unique items in their collection. The second, a category–item hypergraph, characterizes consumption contexts by modeling categories as hyperedges over the items they contain. Finally, the third, a user–category hypergraph, represents consumers as hyperedges over their personal categories, capturing individual differences in how they organize and perceive items. The three hypergraphs are interlocked through the heterogeneous entities they jointly represent, enabling a richer and more flexible modeling approach.

To probabilistically characterize the relationships within these interlocked hypergraphs, we employ message-passing hypergraph variational autoencoders to generate deep embeddings (i.e., vector representations obtained from neural networks) for consumers, items, and item categories in a common latent space. The spatial positions of these embeddings effectively capture the underlying similarities among consumers, items, and categories. We show how digital platforms and e-commerce companies can leverage these embeddings to enhance personalized product recommendations, generate new product bundles, and extend existing categories with novel items.

We apply our framework to two substantive domains: music collections and recipe collections, with a primary focus on music collections in the main article and an analysis of recipe collections in the Web Appendix. Songs are typically consumed through albums and playlists. While albums are carefully compiled by artists, playlists are personalized bundles that music listeners use to categorize their song collections. Recent statistics indicate that half of all music consumption on streaming platforms occurs through albums and playlists (Jakobsen 2016). While album sales have been decreasing steadily since 2000 (IBISWorld 2025), playlists have become the principal mode of music listening since the revolution brought about by digital streaming services. Prebil (2023) reports that Spotify alone hosts more than four billion playlists that are curated for a variety of musical tastes, genres, and consumption contexts such as focus, party, chill, wellness, and sleep. Given the vast heterogeneity in musical tastes and the countless number of playlists that can be constructed, the generative design, expansion, and recommendation of personalized playlists have become central marketing strategies in the music streaming industry.

We showcase our modeling framework using data from multiple online sources, including Last.fm and Spotify. We augment playlist data with multimodal song features comprising user-generated tags and acoustical measures. The textual tags reflect how online listeners describe and categorize songs in terms of genres and listening contexts, while the acoustic features represent technical aspects of the music, such as key, time signature, and loudness. Our framework integrates these structured and unstructured data within the hypergraph setting to yield probabilistic embeddings of consumers, playlists, and songs. We benchmark our approach against several state-of-the-art benchmark models and show that it achieves superior performance in predicting song and playlist preferences. This improvement is primarily driven by the interlocked hypergraphs that effectively harness the rich information embedded in the features of the different entities. Finally, we illustrate how our framework can be applied to several types of downstream managerial tasks.

We make three main contributions. First, most narrowly, we extend the substantive literature on music consumption by providing a detailed empirical analysis of digital music collections. We focus on the diversity of musical preferences across consumers and consumption contexts, leveraging large-scale data from consumer-curated playlists. Our approach goes beyond traditional models by fusing multimodal data that includes both textual tags and acoustic features within the sophisticated network structure of music collections, providing a richer understanding of how consumers organize their music and how these organizational patterns reflect different listening contexts. The message-passing mechanism further enables the inference of latent representations even when music features are missing, improving our understanding of music in real-world applications where data sparsity is common. This comprehensive investigation provides new insights into the heterogeneity of consumer music preferences.

Second, at the core of this study, we make a methodological contribution by developing a novel interlocked hypergraph neural network framework to capture consumer preferences and behavior across collections. Our model yields deep representations of consumers, collections, and products in a unified latent space, such that entities with similar latent embeddings are more likely to be connected via hyperedges, signifying shared experiences or preferences. We are the first in marketing to leverage hypergraphs to model consumer data. Moreover, our use of message-passing mechanisms to learn deep embeddings of heterogeneous entities is novel in marketing, and the approach of interlocking multiple hypergraphs is also new to the broader computer science literature on topological neural networks (Hajij et al. 2022; Papillon et al. 2023).

Third, our model has wide applicability across various consumption contexts where consumers curate item collections. While we demonstrate its effectiveness in music and recipe collections, the framework can be extended to model a broad range of consumer collections, such as grocery shopping baskets (Manchanda, Ansari, and Gupta 1999) and Pinterest boards featuring diverse item types. The managerial implications of our model are promising, as online platforms can leverage it for several key personalization tasks, including (1) generating personalized product bundles and categories targeted to different consumers and contexts, (2) expanding existing categories by identifying complementary items, and (3) recommending items and bundles based on diverse criteria. Overall, firms can harness these capabilities to enhance customer targeting, improve product recommendations, and foster deeper consumer engagement through personalized experiences.

The remainder of the article is structured as follows. The next section reviews the relevant literature, followed by a presentation of our modeling framework. We then describe the music data and report the empirical applications. In the penultimate section, we demonstrate the practical applications of our model for various managerial tasks. Finally, we conclude with a summary of our contributions, a discussion of the model's limitations, and directions for future research.

Literature Review

Our research builds on multiple intersecting streams in marketing and computer science, including studies on consumer collections and categorization, music consumption, generative latent space models, and deep representation learning for marketing problems. We briefly review these areas and highlight our contributions to each.

Consumer Collections and Categorization

Consumer researchers have examined various facets of consumer collections, including the motivations behind why individuals collect objects (Belk 2013), which range from pleasure and goal attainment (Belk 1995; Ijams Spaid 2018) to set completion (Carey 2008), social influences (Ijams Spaid 2018), and economic incentives (Burton and Jacobsen 1999). As our focus is on how consumers categorize their collections, research on consumer categorization (Hamilton, Puntoni, and Tavassoli 2010; Loken, Barsalou, and Joiner 2008), goal-derived categories (Ratneshwar et al. 2001), and assortment choice (Bradlow and Rao 2000) is particularly relevant.

In the context of music, consumers may create playlists that represent prototypical moods or genres, or build them around specific exemplar songs that define a category. Contextual shifts may prompt users to redefine playlists or create new goal-derived playlists. We add to this substantive literature by examining the deep structure of consumer collections and how hypergraphs can effectively model these structures. Our work addresses the motivational determinants of collections by showing that consumers' context-driven categorization and item organization reflect a nuanced interplay of pleasure, goal achievement, and self-expression goals. By leveraging interlocked hypergraphs and message-passing mechanisms, our framework reveals how items are grouped and used across diverse contexts to fulfill multiple consumption goals, supporting deeper insights into individual and collective consumer behavior. This approach not only contributes methodologically but also offers firms actionable tools for personalization and consumer engagement through tailored product categorization and bundling.

Music Consumption and Recommendation

Given our application on consumer music playlists, the marketing literature on music consumption and recommendation systems is highly relevant. Chung, Rust, and Wedel (2009) use hazard models to sequentially design playlists for handheld digital audio players, and Jain and Bradlow (2021) propose a hierarchical Bayesian model for tagged playlist generation. Our work differs from theirs in its ability to scale to large datasets and in its use of hypergraph neural networks to model the composition of consumer song collections and the relationships among songs across playlists. We also acknowledge other research on music in marketing, including studies on the dynamics of music consumption (Boughanmi and Ansari 2021) and the impact of digital platforms on listening behavior (Datta, Knox, and Bronnenberg 2018). In addition, our work is related to the broader marketing literature on recommendation systems and personalization (e.g., Ansari, Essegaier, and Kohli 2000; Ansari, Li, and Zhang 2018; Bodapati 2008; Chung and Rao 2012).

Representation Learning

Our research is closely tied to the computer science literature on representation learning, as well as the psychology literature on multidimensional scaling that yields item embeddings for capturing entity relationships in a latent space. We contrast conventional shallow embedding methods with deep embedding approaches that leverage structured network representations.

Shallow representation models. Shallow embedding techniques have been widely used in graph representation learning and recommendation systems. In the context of music playlists, La Gatta et al. (2022) employ random walks on hypergraphs to generate sequences of entities, which are then processed using a Word2vec model (Mikolov et al. 2013) to obtain shallow embeddings. Other common approaches include matrix factorization and neural collaborative filtering, which use multilayer perceptrons to encode entity similarities (He et al. 2017; Rendle et al. 2020). Hamilton (2020) reviews these methods and points out that shallow embeddings are statistically and computationally inefficient, as they fail to share model parameters across entities and do not incorporate entity features. They also tend to be transductive, meaning they cannot generate representations for new entities that are not encountered during model training.

The literature on multidimensional scaling is also relevant, as it represents consumers and brands in a joint latent space based on perceptual similarities (Carroll and Green 1997). For instance, Jedidi and DeSarbo (1991) model ideal points to assess the impact of situational influences on consumer choice behavior. Most recently, machine learning generalization of these techniques has been introduced to incorporate richer data sources. One such development leverages textual data to analyze similarity between entities. For example, Boughanmi and Ansari (2021) employ a hierarchical Dirichlet process to embed albums in a joint latent space, with smaller Hellinger distances indicating albums described by similar user-generated tags.

Non-message-passing deep representation models. Researchers have proposed various approaches for producing deep representations. These approaches differ in whether they incorporate message-passing mechanisms to leverage graph structures or rely solely on neural network specifications to enforce parameter sharing across entities. Variational autoencoder (VAE) models have been introduced to leverage entity feature vectors and enable amortized inference for entity embeddings. In marketing, Dew, Ansari, and Toubia (2022) apply a multiview VAE to model similarities between logos in a latent space. However, traditional VAE models do not capture the higher-order graph structures present in consumer collections and are therefore limited in their expressivity (Kingma and Welling 2013). Sarker and Matin (2021) propose a hybrid neural collaborative filtering approach that combines a neural network with shallow embeddings for recommender systems. However, their model does not explicitly incorporate graph structures, reducing its ability to model complex relationships among entities.

Message-passing deep representation models. As we represent consumer collections using hypergraphs, the computer science literature on neural network–based graph representation learning is particularly relevant (Chami et al. 2022; Hamilton 2020). Kipf and Welling (2016b) introduce a variational graph autoencoder to predict link formation in graphs. However, as we demonstrate, standard graph convolutional models fail to adequately capture the higher-order relationships inherent in consumer collections.

Several researchers have extended graph-based models to hypergraphs. Feng et al. (2019), Bai, Zhang, and Torr (2021), and Yu et al. (2021) develop hypergraph convolutional networks for applications in citation networks, computer vision, and social recommendations, respectively. Meanwhile, Fan et al. (2021) combine an encoder–decoder structure with a hypergraph attention mechanism to model relationships in heterogeneous hypergraphs. In contrast, we use *interlocked* hypergraph neural networks that we specifically design to capture both the hierarchical and relational nature of consumer collections. We further elaborate these distinctions in the model comparison section, where we show that the model from Fan et al. lacks the expressivity needed to fully represent the intricate relationships within consumer collections due to its restrictive message-passing mechanism.

In the empirical applications section, we summarize the main conceptual differences between our model and the benchmark approaches used for comparison.

Other neural networks and generative models. Finally, we acknowledge the expanding role of deep learning in marketing research, including studies such as Fong, Kumar, and Sudhir (2024), which leverages convolutional neural networks to model emotional responses to music, as well as applications involving text analysis (Chakraborty, Kim, and Sudhir 2022; Liu, Lee, and Srinivasan 2019; Puranam, Kadiyali, and Narayan 2021), image processing (Dew, Ansari, and Toubia 2022; Hong and Hoban 2022; Liu, Dzyabura, and Mizik 2020), video modeling (Yang, Zhang, and Zhang 2024; Zhou et al. 2021), product assortments (Gabel and Timoshenko 2022), product design (Burnap, Hauser, and Timoshenko 2023), and market structure analysis (Yang, Zhang, and Kannan 2022). Our work is distinct in its focus on consumer collections and its use of interlocked hypergraph neural networks, a novel approach in marketing. In addition, we diverge from earlier approaches that relied on shallow embeddings, such as Word2vec (Mikolov et al. 2013) and neural matrix factorization (Dhillon and Aral 2021). In contrast, our deep embeddings incorporate both covariate information and message-passing mechanisms within hypergraph neighborhoods. We empirically demonstrate that these richer representations result in superior predictive performance, making them more effective for modeling consumer behavior and personalization strategies.

Modeling Framework

Our modeling framework leverages the idea that the contextual preferences of consumers are expressed through their item collections and their categorization structure. As collections of different consumers often share common items, the entire corpus of collections across consumers exhibits a network structure of relationships among various entities. This structure is illustrated in Figure 1, which shows the collections of two consumers, u_1 and u_2. Consumer u_1's collection contains three items, $\{s_1, s_2, s_3\}$, organized into two categories, p_1 and p_2, whereas consumer u_2's collection includes four items, $\{s_3, s_4, s_5, s_6\}$, grouped into two categories, p_3 and p_4. As the figure shows,

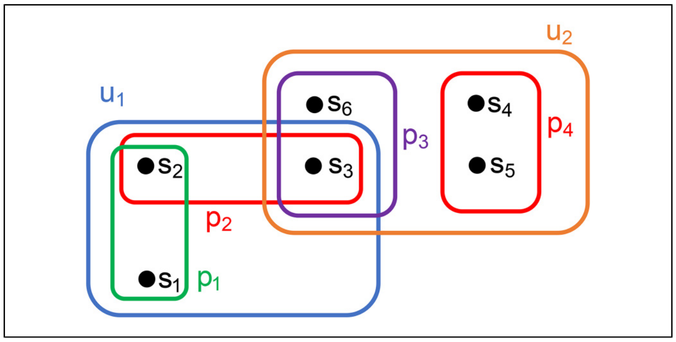


Figure 1. Two Collections (u_1, u_2), Four Categories ($p_1, \ldots, p_4$), and Six Items ($s_1, \ldots, s_6$).

an item can appear in the collections of multiple consumers (e.g., s_3 in u_1 and u_2) and may also belong to multiple categories within a single consumer's collection (e.g., s_2 in p_1 and p_2) or across different consumers (e.g., s_3 in p_2 and p_3). The number of items, number of categories, and category sizes within a consumer's collection can vary across consumers, resulting in a complex, higher-order network structure among these entities. Conventional graph models, which represent relationships as dyadic edges between vertices, are insufficient for capturing these higher-order relationships, where entities are interconnected in multiway interactions rather than simple pairwise links. To model this complexity, we leverage higher-order graph structures using hypergraphs and develop a novel framework of multiple interlocked hypergraphs to capture the intricate structure of consumer collections. We begin by introducing the concept of a single hypergraph before expanding to our full interlocked framework.

Hypergraphs

Hypergraphs are generalizations of ordinary graphs and are used to represent higher-order networks and relationships. Formally, a hypergraph is defined as $\mathcal{H} = \{\mathcal{V}, \mathcal{E}, X^v, X^e\}$, where $\mathcal{V}$ is the set of vertices, $\mathcal{E}$ is the set of hyperedges, X^v is the matrix of features describing the vertices, and X^e is the matrix of features for the hyperedges. A hyperedge is a set that contains any number of vertices and represents a relationship that is simultaneously shared by its vertices. For example, a category can be viewed as a hyperedge that connects items that belong to a shared consumption context. The set $\mathcal{E}$ therefore consists of nonempty subsets of $\mathcal{V}$, that is, $\mathcal{E} \subseteq \mathcal{P}(\mathcal{V}) \backslash \varnothing$, where $\mathcal{P}(\mathcal{V})$ is the power set of $\mathcal{V}$ (the set of all subsets of $\mathcal{V}$) and $\varnothing$ is the null set. The left portion of Figure 2 shows a simple hypergraph example with vertices ($v_1, \ldots, v_6$) and hyperedges ($e_1, \ldots, e_4$).

When ordinary graphs are used to represent higher-order structures, they often fail to preserve key aspects of the rich relationships between different vertices. For instance, the relationships in the hypergraph of Figure 2 can be alternatively illustrated by decomposing the hypergraph into two ordinary graphs: one illustrating the connections between vertices and another depicting the relationships among hyperedges, as shown on the right side of Figure 2. In the first graph (top right), vertices that belong to common hyperedges are linked by ordinary dyadic edges, while in the second graph (bottom right), hyperedges that share vertices are connected through dyadic links. However, these two ordinary graphs are inferior in their capacity to encode information, compared with the hypergraph.

First, the holistic structure is lost. For instance, the distinct contexts connecting vertices v_1 and v_2, and v_1 and v_5, are erased in the ordinary graph representation. Second, the degree of overlap between entities is not preserved. For example, hyperedges e_2 and e_4 share two vertices in the hypergraph, and hyperedges e_1 and e_3 share only one vertex. Yet, in their ordinary graph representation, information about the varying degree of overlap is lost. Finally, both ordinary graphs are limited to pairwise interactions and fail to capture higher-order relationships, which are essential for modeling complex consumer collections.

Moreover, as consumer collections are composed of heterogeneous entities (e.g., consumers, categories, and items), a single hypergraph is incapable of representing the intricate web of interactions. To address this, we interlock three hypergraphs to model the higher-order network structure.

Interlocked Hypergraphs of Consumers, Categories, and Items

Figure 3 illustrates the structure of these hypergraphs based on the small example from Figure 1, with the three hypergraphs shown at the bottom. The first hypergraph (left) consists of six items (vertices) and two consumers (hyperedges), capturing consumer preferences for their collected items irrespective of the consumption contexts. The second hypergraph (middle) also has items as vertices, but here the four categories form its hyperedges, explicitly characterizing how a common consumption context establishes simultaneous relationships among items grouped in a category. The third hypergraph (right) has categories as vertices and consumers as hyperedges, representing consumer preferences for categories or consumption contexts. Note that vertices in the third hypergraph are composite entities (categories), each composed of items and serving simultaneously as hyperedges in the second hypergraph. Thus, the three hypergraphs are clearly interconnected through shared entities and together constitute a web of relational and hierarchical relationships.

Generalizing from the small example, we formally describe the three hypergraphs as follows. Each consumer $i = 1, \ldots, n_u$ is characterized using a vector of features $\mathbf{x}_i^u$ of size f_u (e.g., demographics), stacked as rows into a matrix X^u of size $n_u \times f_u$. Similarly, each category $j = 1, \ldots, n_p$ is described by a feature vector $\mathbf{x}_j^p$ of size f_p (e.g., title words), collected into the matrix X^p matrix of size $n_p \times f_p$. Each item $k = 1, \ldots, n_s$ has features (e.g., genre, tags) represented by a vector $\mathbf{x}_k^s$ of size f_s, which form the rows of the matrix X^s of size $n_s \times f_s$. The three interlocked hypergraphs can thus be succinctly written as $\mathcal{H}^{us} = \{\mathcal{V}^s, \mathcal{E}^{us}, X^s, X^u\}$, $\mathcal{H}^{ps} = \{\mathcal{V}^s, \mathcal{E}^{ps}, X^s, X^p\}$, and $\mathcal{H}^{up} = \{\mathcal{V}^p, \mathcal{E}^{up}, X^p, X^u\}$.

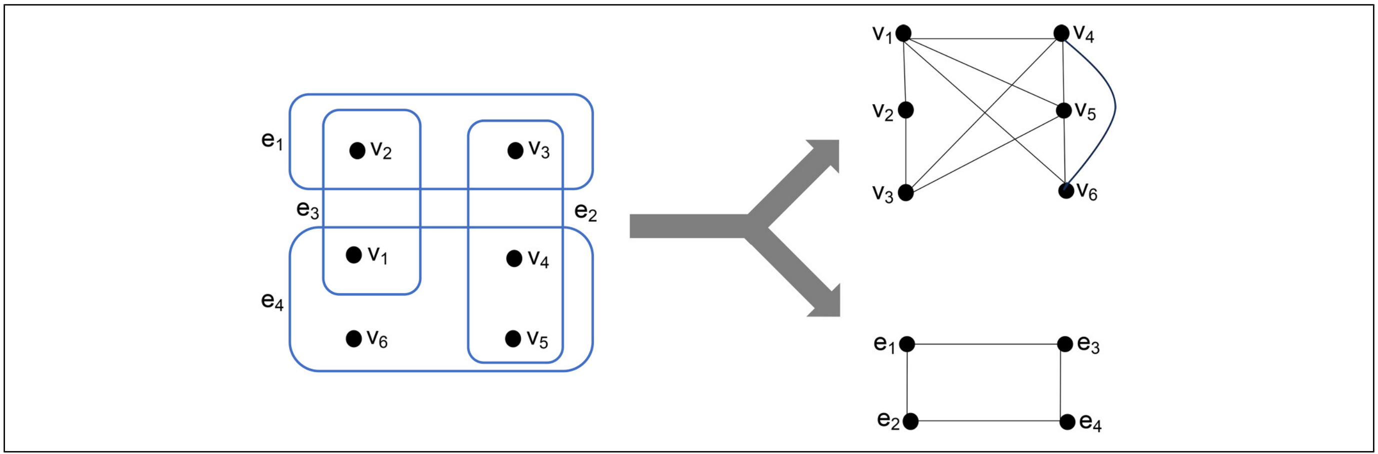


Figure 2. Hypergraph Versus Ordinary Graphs.

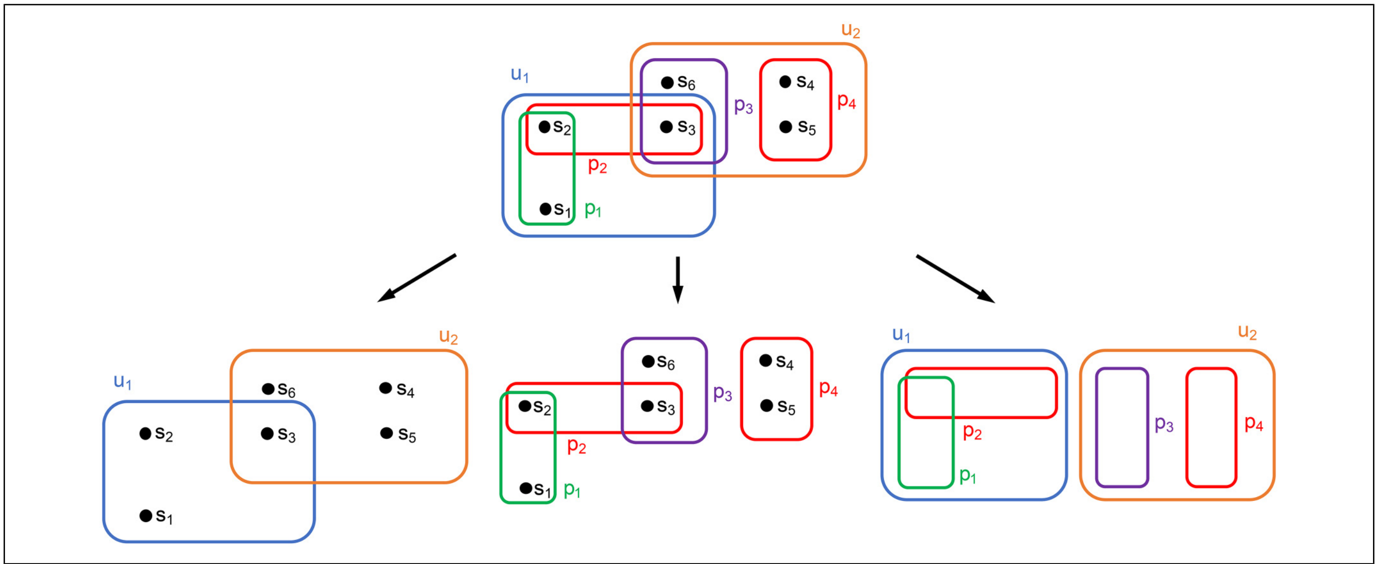


Figure 3. Three Interlocked Hypergraphs with Six Items ($s_1, \ldots, s_6$), Four Categories ($p_1, \ldots, p_4$), and Two Consumers (u_1, u_2).

As illustrated in the preceding example, the hypergraph $\mathcal{H}^{us}$ characterizes the intrinsic preferences of consumers for different items, while the hypergraph $\mathcal{H}^{ps}$ captures the composition of items within categories, and the hypergraph $\mathcal{H}^{up}$ models consumer preferences for specific categories. The relations between vertices and hyperedges can be represented using their respective incidence matrices: $\mathrm{H}^{us} \in \{0, 1\}^{n_u \times n_s}$, $\mathrm{H}^{ps} \in \{0, 1\}^{n_p \times n_s}$, and $\mathrm{H}^{up} \in \{0, 1\}^{n_u \times n_p}$, where each binary element indicates whether a vertex belongs to a given hyperedge. Importantly, these three hypergraphs are interdependent as they share entities, which enable information propagation across the hypergraphs during model training.

Variational Hypergraph Autoencoders

Having outlined the three interlocked hypergraphs, we introduce a novel generative modeling framework that represents consumer collection structures, captures heterogeneous consumer preferences and categorization behaviors, and produces latent embeddings for heterogeneous entities. These embeddings can then be leveraged for various downstream managerial tasks.

Our framework leverages the distributional hypothesis popular in linguistics (Firth 1957), which in our context implies that items frequently co-occurring across consumer collections share a similar preferential appeal, while items that co-occur within the same categories offer similar consumption experiences or fulfill comparable consumption goals. Furthermore, consumers with overlapping collections tend to have similar preferences, and categories with shared items reflect aligned consumption contexts.

To characterize the complex structural relationships in the three hypergraphs, we develop a *message-passing interlocked variational hypergraph autoencoder* to generate deep embeddings for items, categories, and consumers. Given the varying types and sizes of these entities, they need to be probabilistically embedded in a *joint* latent space; that is, embeddings for all entity types share a common dimensionality, enabling meaningful distance computations and effective representation of similarities across entities in the hypergraphs. This probabilistic approach facilitates the generation of novel and coherent item sets (e.g., consumer collections of songs and category subsets like playlists), and the embeddings are useful for downstream tasks such as item and subset recommendation, subset generation, and subset expansion.

We model the entries in the incidence matrices of the three hypergraphs using a variational interlocked hypergraph autoencoder that consists of two main components. The first is a *generative* model, also called a decoder, which probabilistically captures the connections in the hypergraphs by utilizing latent variables (embeddings) for the entities. The second is an *inference* model, or encoder, which specifies an amortized variational distribution for the latent embeddings in the generative component. The amortization is achieved using neural networks built with hypergraph convolutional layers. These layers employ a message-passing mechanism to ensure that an entity's embedding is informed by the embeddings of its neighbors within the three hypergraphs. We next describe these two components in greater detail.

Generative Model (Decoder)

Given the large size of consumer collections, we adopt a parsimonious approach to represent the hypergraph structures. Specifically, we assume entries in the hypergraph incidence matrices are generated by a probabilistic process involving low-dimensional latent embeddings for consumers, categories, and items. These embeddings reside in a space of dimensionality κ, a number much smaller than the network size. Each consumer i is represented by an embedding vector $\mathbf{z}_i^u$, stacked into the matrix $Z^u \in \mathbb{R}^{n_u \times \kappa}$. Similarly, each category j is characterized by an embedding $\mathbf{z}_j^p$, collected into $Z^p \in \mathbb{R}^{n_p \times \kappa}$, and each item k by an embedding $\mathbf{z}_k^s$, aggregated into $Z^s \in \mathbb{R}^{n_s \times \kappa}$.

Our generative model specifies the probabilities of the binary entries in the three incidence matrices using the dot products of the latent embeddings. While our framework can accommodate more complex decoders, we choose a simple dot product–based decoder because research has shown that it predictively outperforms more complex neural network–based alternatives (Rendle et al. 2020). The probabilities of the positive entries in the incidence matrices can be written as $p(H_{ik}^{us} = 1 | \mathbf{z}_i^u, \mathbf{z}_k^s) = \sigma(\mathbf{z}_i^{u\top} \mathbf{z}_k^s)$, $p(H_{ij}^{up} = 1 | \mathbf{z}_i^u, \mathbf{z}_j^p) = \sigma(\mathbf{z}_i^{u\top} \mathbf{z}_j^p)$, and $p(H_{jk}^{ps} = 1 | \mathbf{z}_j^p, \mathbf{z}_k^s) = \sigma(\mathbf{z}_j^{p\top} \mathbf{z}_k^s)$, where $\sigma(.)$ is the logistic sigmoid function. The probability of a zero entry is one minus that of a positive entry. Conditional on the latent embeddings, the entries in the different cells are mutually independent. Moreover, as in Kingma and Welling (2013), the latent embeddings are assumed to be mutually independent with isotropic Gaussian priors, $p(\mathbf{z}_i^u) = p(\mathbf{z}_j^p) = p(\mathbf{z}_k^s) = \mathcal{N}(0_\kappa, I_\kappa)$, where 0_κ is the null vector and I_κ is the identity matrix of dimension κ. The use of latent random variables yields a probabilistic generative model that enables us to generate novel item collections that reside in the space spanned by the data.

Other nonbinary outcomes. While our decoders model the binary entries in the incidence matrices of the three hypergraphs, they can be easily adapted to capture other outcomes relevant to specific applications. For example, in the music domain, beyond modeling playlist composition, data could also include song preference intensity. In such cases, the decoder can be extended with additional components that model these outcomes using exponential family distributions and appropriate link functions, resulting in generalized linear model specifications (Nelder and Wedderburn 1972).

Joint distribution. The joint distribution of the data and latent variables is given by

$$
\begin{aligned}
&p(H^{us}, H^{up}, H^{ps}, Z^u, Z^p, Z^s) \\
&= \prod_{i=1}^{n_u} \prod_{j=1}^{n_p} \prod_{k=1}^{n_s} p(H_{ik}^{us} | \mathbf{z}_i^u, \mathbf{z}_k^s) p\left(H_{ij}^{up} | \mathbf{z}_i^u, \mathbf{z}_j^p\right) p\left(H_{jk}^{ps} | \mathbf{z}_j^p, \mathbf{z}_k^s\right) \\
&\quad \times \prod_{i=1}^{n_u} p(\mathbf{z}_i^u) \prod_{j=1}^{n_p} p\left(\mathbf{z}_j^p\right) \prod_{k=1}^{n_s} p\left(\mathbf{z}_k^s\right).
\end{aligned}
$$

The posterior distribution of the embeddings, $p(Z^u, Z^p, Z^s, | H^{us}, H^{up}, H^{ps})$, is not available in closed form as the likelihood and the prior are nonconjugate. Markov chain Monte Carlo methods can be used to summarize the posterior and infer the latent variables. However, such simulation-based approaches are next to impossible to use for large consumer collections. Therefore, for scalability, we adopt *amortized variational inference* via an encoder to approximate the posterior distributions and efficiently obtain estimates of the latent embeddings (Blei, Kucukelbir, and McAuliffe 2017; Rezende, Mohamed, and Wierstra 2014).

Inference Model (Encoder)

The variational approximation for the posterior distribution of the embeddings is represented as a product of three variational factors. We use hypergraph neural networks to obtain *deep embeddings* that are informed by both the network structure of the collections and the entity covariates. Deep embeddings differ from *shallow embeddings* obtained via simpler matrix factorization or Word2vec-based methods on several aspects. Shallow embedding methods (Qiu et al. 2018) rely on encoding functions that do not share parameters across entities and therefore are statistically and computationally inefficient. These approaches also fail to leverage entity features or exploit information from distant neighborhoods within the hypergraphs. Moreover, shallow methods are transductive; that is, they cannot produce embeddings for entities not encountered in the training data.

Typically, variational factors are assumed to be conditionally independent and normally distributed (Kipf and Welling 2016b), as specified:

$$q(Z^u|H^{us}, H^{up}, X^u) = \prod_{i=1}^{n_u} q(\mathbf{z}_i^u|H^{us}, H^{up}, X^u) = \prod_{i=1}^{n_u} \mathcal{N}(\mathbf{z}_i^u|\boldsymbol{\mu}_i^u, \boldsymbol{\Sigma}_i^u),$$

$$q(Z^p|H^{up}, H^{ps}, X^p) = \prod_{j=1}^{n_p} q\left(\mathbf{z}_j^p|H^{up}, H^{ps}, X^p\right) = \prod_{j=1}^{n_p} \mathcal{N}\left(\mathbf{z}_j^p|\boldsymbol{\mu}_j^p, \boldsymbol{\Sigma}_j^p\right),$$

$$q(Z^s|H^{us}, H^{ps}, X^s) = \prod_{k=1}^{n_s} q(\mathbf{z}_k^s|H^{us}, H^{ps}, X^s) = \prod_{k=1}^{n_s} \mathcal{N}(\mathbf{z}_k^s|\boldsymbol{\mu}_k^s, \boldsymbol{\Sigma}_k^s).$$

Here, $\boldsymbol{\mu}_i^u$, $\boldsymbol{\mu}_j^p$, and $\boldsymbol{\mu}_k^s$ represent the means of the variational distributions, while $\boldsymbol{\Sigma}_i^u$, $\boldsymbol{\Sigma}_j^p$, and $\boldsymbol{\Sigma}_k^s$ denote their covariance matrices. As is standard, we assume that the covariance matrices $\Sigma_i^u = \text{diag}(\boldsymbol{\sigma}_i^{u2})$, $\Sigma_j^p = \text{diag}(\boldsymbol{\sigma}_j^{p2})$, and $\Sigma_k^s = \text{diag}(\boldsymbol{\sigma}_k^{s2})$ are diagonal (Kingma and Welling 2013). Rather than optimizing a unique set of mean and variance parameters for each variational distribution, as in stochastic variational inference (Hoffman et al. 2013), we employ *amortized* inference that ties the variational distributions for a given entity type to a common set of parameters. For example, all user-level variational distributions are parameterized by a single neural network with shared weights and biases, enabling scalable inference when dealing with a large number of entities. In addition, the amortization allows the variational distributions to be data dependent, which is particularly beneficial for cold-start scenarios and the generation of novel collections or categories.

Specifically, we reparameterize the variational means and log-variances for the embeddings of a given entity type (e.g., $\boldsymbol{\mu}_i^u$ and $\log\boldsymbol{\sigma}_i^{u2}$ for consumers) using *message-passing hypergraph convolution neural networks*. We utilize separate encoder networks for consumer, category, and item embeddings. Each encoder network takes as input the corresponding hypergraph incidence matrices and feature matrices, and outputs the variational means and log-variances for the embeddings of that entity type. Therefore, the means are reparameterized as $\boldsymbol{\mu}_i^u = \text{NN}(H^{us}, H^{up}, X^u; \phi_\mu^u)$, $\boldsymbol{\mu}_j^p = \text{NN}(H^{up}, H^{ps}, X^p; \phi_\mu^p)$, and $\boldsymbol{\mu}_k^s = \text{NN}(H^{us}, H^{ps}, X^s; \phi_\mu^s)$, where ϕ_μ^u, ϕ_μ^p, and ϕ_μ^s denote the neural network parameters (weights and biases) for each entity type. Similarly, the log-variances are reparameterized as $\mathbf{l}\boldsymbol{\sigma}_i^u = \text{NN}(H^{us}, H^{up}, X^u; \phi_{l\sigma}^u)$, $\mathbf{l}\boldsymbol{\sigma}_j^p = \text{NN}(H^{up}, H^{ps}, X^p; \phi_{l\sigma}^p)$, and $\mathbf{l}\boldsymbol{\sigma}_k^s = \text{NN}(H^{us}, H^{ps}, X^s; \phi_{l\sigma}^s)$. We next describe the architecture of these neural networks.

Message-passing hypergraph convolution networks. Message-passing hypergraph convolution networks are multilayer neural networks composed of convolutional layers where computations at each layer are guided by a message-passing mechanism, with layers corresponding to iterations of the message-passing procedure. During message passing, the latent representations for an entity type (e.g., items) are iteratively updated by integrating information from neighboring entities along with learnable neural network parameters, as detailed in Algorithm W1 in Web Appendix A. The initial latent representations are set directly using observable entity features. Subsequently, the message-passing iterations progressively update these representations. The final outputs of the neural network are the means and log-variances for the variational distributions of the embeddings. The specific update steps in Algorithm W1 employ hypergraph convolutional layers, which we next discuss briefly. A detailed description is provided in Web Appendix B.

Hypergraph convolutional layers. We use the hypergraph convolutional layer from Bai, Zhang, and Torr (2021) that transforms the latent representations of an entity at each layer by aggregating information from its local neighborhood within the hypergraph. Consider the t^{th} message-passing convolutional layer that updates the intermediate representation associated with the variational mean $\boldsymbol{\mu}_k^s$ of an arbitrary item k.[1] The update depends on an extended dual-incidence matrix $H^s = (H^{us}, H^{ps})^\top$, where H^{ps} is stacked below H^{us}. Thus, the columns of this combined incidence matrix correspond to users and categories (hyperedges), while the rows represent items (vertices).

The t^{th} message-passing update for item k can be expressed as

$$\mathbf{g}_{\mu,k}^{s,(t)} = \psi\left(\sum_{j\in\mathcal{C}_k^\epsilon}\sum_{r\in\mathcal{C}_j^v}\frac{W_\mu^{s,(t)}\mathbf{g}_{\mu,r}^{s,(t-1)}}{|\mathcal{C}_j^v|\sqrt{|\mathcal{C}_k^\epsilon||\mathcal{C}_r^\epsilon|}} + \mathbf{a}_\mu^{s,(t)}\right),$$

where $\mathbf{g}_{\mu,k}^{s,(t)}$ is the k^{th} row of the matrix $G_\mu^{s,(t)}$, which contains the latent representations for all the items. Here, $\mathcal{C}_k^\epsilon$ denotes the edge neighborhood of item k, consisting of all the hyperedges (consumers and categories) that include item k. Similarly, $\mathcal{C}_j^v$ denotes the vertex neighborhood of hyperedge j, containing all the vertices connected by that hyperedge. For each item k, this layer *aggregates* messages (latent representations) from the previous layer across all the items connected via the hyperedges in k's edge neighborhood. These messages are weighted by the weight matrix $W_\mu^{s,(t)}$ of the layer, which is estimated from the data. The denominator applies symmetric normalization (Kipf and Welling 2016a), which down-weights messages from highly connected hyperedges and vertices, stabilizes numerical computations, and mitigates issues such as vanishing or exploding gradients during optimization. A bias term (intercept) $\mathbf{a}_\mu^{s,(t)}$ is added to the aggregated, normalized, and weighted representations. The resulting message is transformed through a nonlinear

[1] The analogous update steps for consumer and category embeddings are detailed in Web Appendix B.

activation function $\psi(.)$ yielding the output message for the next layer.

As illustrated in Algorithm W1 of Web Appendix A, at the first layer ($t = 1$), the input $G_{\mu}^{s,(0)}$, which is the matrix of initial representations for the items, is set equal to the vertex feature matrix X^s. The first message-passing step aggregates messages from the immediate (one-hop) neighborhood of each item. Subsequent hypergraph convolutional layers incorporate information from increasingly distant neighborhoods (two-hop and beyond). Consequently, the embedding of a given entity is informed not only by its own features but also by those of entities from both its immediate and *extended* neighborhoods, capturing richer structural information.

While the preceding discussion focused on the hypergraph neural network for variational means, a similar message-passing mechanism can be applied to update the representations of variational log-variances. This mechanism is especially beneficial when entities have missing attribute data, as it leverages the hypergraph structure to inform the variational parameters of their embeddings. Through message-passing, entities within the same neighborhood exchange information, allowing the encoder to aggregate data from neighbors with nonmissing attributes. Consequently, embeddings for entities with missing data are still informed by the feature and structural information of their neighboring entities.

In summary, our model flexibly integrates entity features with the interlocked hypergraph structures using hypergraph convolution, which aggregates information from connected entities within the hypergraphs. These operations enable inference of the latent embeddings for consumers, categories, and items. The model parameters, that is, the weights and biases of the neural network layers, are optimized by maximizing the evidence lower bound (ELBO), as detailed in Web Appendix C. The complete model training procedure is described in Algorithm W2 of Web Appendix D. Furthermore, we perform simulation studies to demonstrate our model's capability of accurately recovering the topology of the joint latent space, with details provided in Web Appendix E.

Empirical Illustration: Consumer Music Collections

We apply our framework to two datasets, focusing on digital music collections in the main article and food recipes in Web Appendix F. Music collections provide an ideal setting to illustrate our model, as listeners not only curate collections of tracks but also categorize them into multiple playlists. These playlists are typically organized around coherent themes that reflect consumer preferences for specific genres, artists, or listening contexts. Next, we describe the music collection dataset, which we compiled from multiple sources.

Data

Our dataset comprises consumer-curated song collections organized into playlists, crowd-sourced textual tags that reflect how internet users describe the songs within these playlists, and acoustic fingerprints that summarize the phonic features of these songs.

Song categories: playlists. The playlists are sourced from a dataset described and made available by Pichl, Zangerle, and Specht (2015), containing 265,575 playlists curated by 15,910 Spotify users and comprising 2,773,081 songs, along with their respective performing artists. Consumers categorized the songs in their collections into multiple playlists based on different listening contexts. Each playlist is titled, and both playlist sizes and the number of playlists per consumer vary, reflecting the diverse ways users structure their music collections.

We exclude single-song playlists, as they are indistinguishable from individual songs, and remove consumers with only one playlist, since their collections are identical to the playlists they contain. This filtering results in a dataset of 13,384 unique consumers. Due to computational constraints, we randomly sample 2,000 consumers, retaining their entire collections of curated playlists. These consumers collectively created 33,692 playlists, encompassing 769,294 unique songs by 94,729 unique artists.[2]

We next highlight key patterns of variation in our dataset. Figure 4 presents histograms illustrating the number of songs per playlist and the number of playlists per consumer. The median playlist contains 16 songs, with 75% of playlists having fewer than 39 songs. On average, a song appears in two playlists, and 75% of songs are included in at most two playlists. The median number of playlists per consumer is 11, while 75% of consumers have fewer than 26 playlists. These playlists serve different listening contexts, which can be inferred from both their titles and the songs they contain.

Playlist titles are available for 95% of the playlists and comprise a vocabulary of 17,686 unique words. Figure 5 presents a word cloud illustrating the most common terms. These titles reveal that playlists cater to diverse listening contexts, ranging from musical genres (e.g., "rock," "jazz") to situational themes such as weather or seasons. For instance, playlists with "summer" in the title might be intended for pool parties, while others reference specific months, such as "July" or "December." Some playlists are designed for social settings, like "party" or "birthday," whereas others cater to individual experiences, such as "workout" or "run." Many playlists also reflect emotional states, including "love," "chill," and "inspiration." Importantly, playlists without titles do not pose a challenge for our analysis, as our model is capable of handling missing data through its message-passing mechanism.

Although playlists appear specialized around distinct consumption themes, they also exhibit complex structures characterized by varying degrees of overlap. Figure 6 illustrates the

[2] In the data, 7% of the playlists share the same title, and 4% contain the same songs. However, only 3% share the same title and songs. Since we cannot ascertain whether these playlists are identical by chance or are shared on the platform, we assume that these playlists are different and treat each as a unique entity in our dataset.

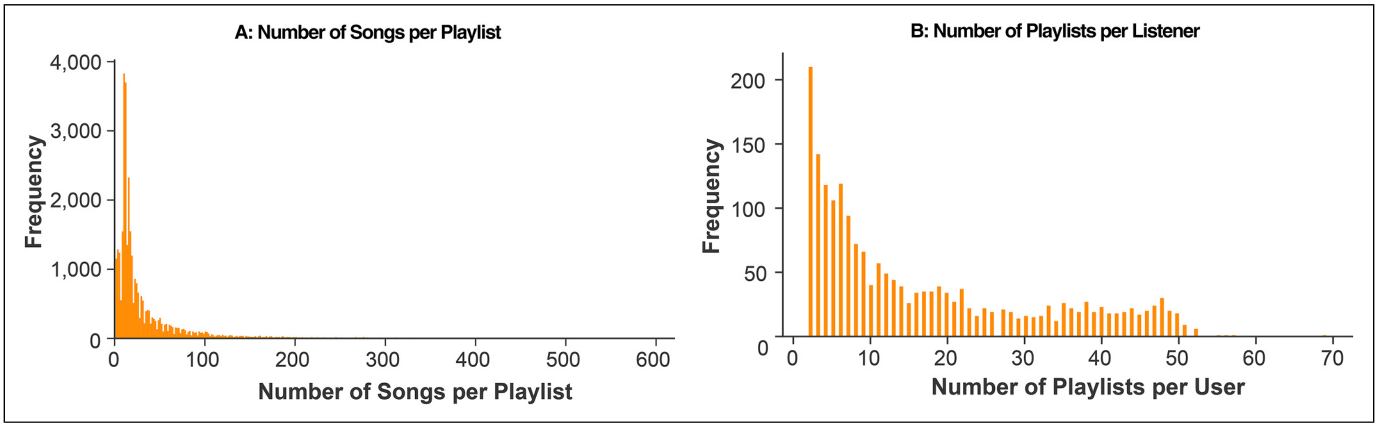


Figure 4. Distribution of Songs Across Playlists and Playlists Across Listeners.

Figure 5. Word Cloud of Playlist Title Vocabulary.

song overlap between different playlists curated by a consumer in our dataset. While some playlists, such as "Classical Music," are highly specialized, others share nonempty intersections. For example, the playlists "Rock," "Light Listening," and "Nostalgia" contain five common songs, suggesting that consumers seek similar musical experiences across these contexts. Both playlist titles and the structure of song overlaps provide valuable insights into consumers' contextual preferences. However, they alone do not fully capture the musical experience associated with each song. To enrich this understanding, we augment the playlist data with textual tags and acoustic fingerprints, providing a more comprehensive representation of song characteristics.

User-generated tags. User-generated tags reflect how online listeners describe songs. We use the Last.fm API to download tags for our tracks. Last.fm, an online music platform launched in 2002, allows users to apply tags to describe songs. However, it is important to note that Last.fm members are not necessarily the same consumers in our data. Thus, the tags represent how a broader cross-section of online users categorize the songs (see Web Appendix G for an example). These tags capture both technical information, such as genres, and contextual details related to listening experiences. Our data collection efforts yield tags for 371,430 songs, covering 48% of all tracks in our dataset.

To retain important tags used by most online listeners, we remove the bottom 1% of tags based on their frequency across songs. This results in a vocabulary of 4,959 unique tags applied to 352,134 songs (46% of all songs in our dataset), with a total of 6,131,019 tag applications—an average of 17 tags per song for tracks with available tags. Songs without tags do not pose a challenge in our analysis, as our modeling framework effectively handles missing data. Last.fm assigns a weight to each tag, denoting the fraction of online listeners who used that tag to describe a song. For instance, 80% of users might label a song as "happy," while 50% might use "love." These weights do not sum to one, as they reflect independent tagging behaviors rather than proportions of a fixed total. The weighted tags capture how online listeners characterize a song, allowing for meaningful comparisons between songs based on categorization and perception. Figure 7 presents a word cloud of all unique tags in our final dataset.

Acoustic fingerprints. Acoustic fingerprints capture the phonic features of a song. We use the Spotify API to collect 13 acoustic attributes, which are detailed in Web Appendix H. These fingerprints are produced by The Echo Nest utilizing machine learning approaches (Ellis et al. 2010). Certain acoustical features, such as duration, key, loudness, mode, tempo, and time signature, represent the physical properties of music, while others, including acousticness, energy, instrumentalness, danceability, liveness, valence, and speechiness, describe the listening experience. Together, they capture both subjective and objective dimensions of a song. We also include track duration as an additional acoustic attribute. Our data collection efforts yield acoustic fingerprints for 755,257 songs, covering 98% of all tracks in our dataset.

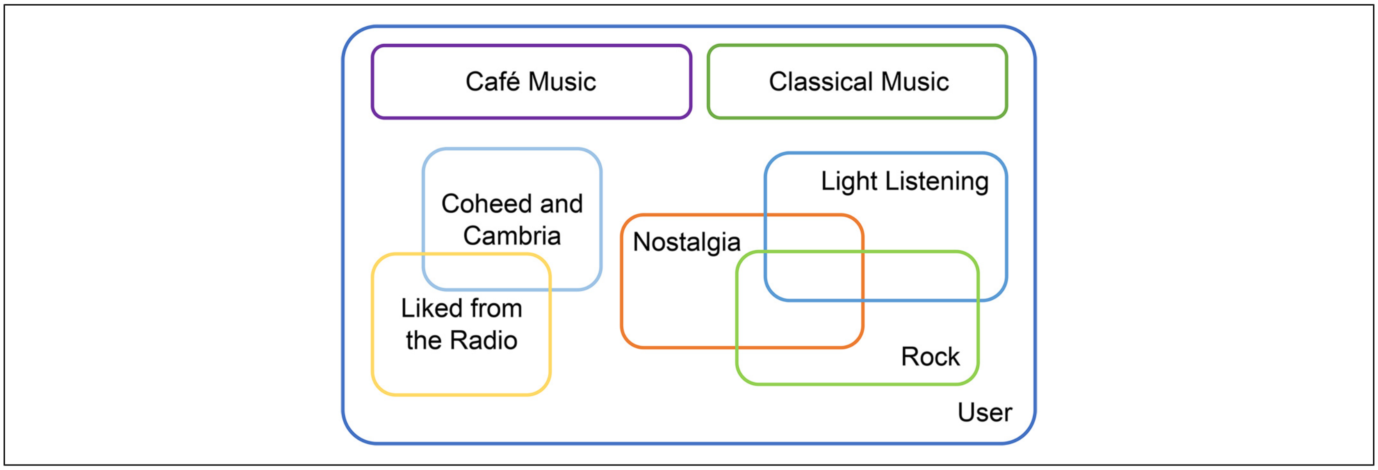


Figure 6. Overlapping Playlist Structures for a Consumer.

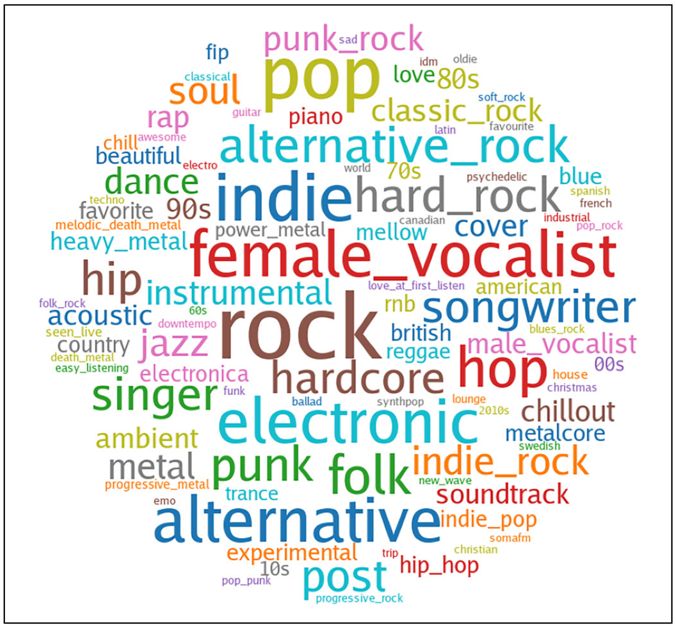


Figure 7. Word Cloud of Song Tags.

Feature Specifications

We next specify how we incorporate observed features into our model. In our application, consumers are music listeners, categories are playlists, and items are songs. As consumer demographics are not available, we use a one-hot (binary) indicator for each consumer, that is, $f_u = n_u$. Playlist features are derived from the vocabulary of words in playlist titles. To account for untitled playlists, we include the word "missing" in the vocabulary. Let $\mathcal{M}_p$ of size f_p denote the set of unique words in the titles including the word "missing." Each playlist $j = 1, \ldots, n_p$ is then described by a feature vector $\mathbf{x}_j^p$ of size f_p, where the w^{th} element $\mathbf{x}_{jw}^p$ indicates the frequency of the w^{th} word from $\mathcal{M}_p$ in playlist j's title. Last, song features are constructed from both textual tags and acoustic fingerprints. Let $\mathcal{M}_s$ represent the unique set of song tags across all songs, and let f_s^t represent the vocabulary size. This vocabulary also includes a tag "missing" to account for songs lacking tags. Similarly, let f_s^a be the number of acoustic variables including a "missing" dummy to represent songs without acoustic data. Then, each song $k = 1, \ldots, n_s$ is described by a vector $\mathbf{x}_k^s = (\mathbf{x}_k^{ts}, \mathbf{x}_k^{as})$ of size $f_s^t + f_s^a$. The w^{th} element $\mathbf{x}_{kw}^{ts}$ represents the weight of the w^{th} tag in $\mathcal{M}_s$ for song k, where absent tags have zero weight. The v^{th} entry in $\mathbf{x}_{kw}^{as}$ indicates the value of the v^{th} acoustic fingerprint for song k.

Having described our data, we next report the results from applying our modeling framework to the dataset. We then discuss several managerial applications of our approach.

Model Estimation and Results

In this section we report the estimation details, describe the benchmark models, and report our results pertaining to the joint latent space.

The three interlocked hypergraphs applied to our dataset contain a total of 27,525,025,448 binary entries across the respective incidence matrices. Of these, only 1,846,592 entries signify positive links among consumers, playlists, and songs. We randomly allocate 80% of these positive entries for training and set aside two datasets, each comprising 10% of the positive observations, for model validation and testing. Given the large number of negative observations (zero entries), we follow Kipf and Welling (2016b) and adopt negative sampling to tackle the computational challenges. In particular, for model training, we use a dataset containing all the positive entries and an equal number of negative entries, randomly drawn from the pool of negative links.[3] We apply the same negative sampling strategy for the validation and test datasets, while ensuring that there is no overlap among the training, validation, and test data. We implement the

[3] Model performance remains robust across different negative sampling ratios, as demonstrated in Web Appendix I.

Table 1. Predictive Performance of Proposed Model Across Varying Latent Dimensions.

Latent Dimensions	Hit Rate (%)	Precision (%)	Recall (%)	F1 Score (%)	AUC
A: Training					
5	89.54	86.68	93.43	89.93	.96
10	93.18	91.61	95.07	93.31	.98
15	93.65	92.42	95.11	93.74	.98
20	93.49	91.92	95.37	93.61	.98
30	93.73	92.20	95.55	93.84	.98
40	93.60	92.01	95.49	93.72	.98
50	94.07	92.70	95.66	94.16	.98
B: Validation					
5	82.06	83.58	79.79	81.64	.89
10	85.38	88.65	81.15	84.74	.90
15	86.03	89.17	82.04	85.45	.90
20	85.65	89.09	81.24	84.99	.90
30	85.91	89.61	81.23	85.22	.90
40	85.77	89.40	81.17	85.09	.90
50	86.24	89.94	81.61	85.57	.91

estimation algorithm using PyTorch and PyTorch-geometric and leverage the Adam optimizer to maximize the ELBO via stochastic gradient descent (Kingma and Ba 2014). All computations are performed on an NVIDIA Quadro RTX A6000 GPU.

We optimize the objective function on the training data for 10,000 epochs, iterating through gradient evaluation and updates.[4] Training is performed for several choices of hyperparameter values, and we retain the most parsimonious network architectures that yield the best results on the validation dataset. These experiments result in an architecture consisting of two hypergraph convolution layers ($L = 2$) with $f^{(\ell)} = 25$ hidden units within each layer.[5] As shown in Table 1, we evaluate the predictive performance using several key metrics: hit rate, the proportion of correctly predicted entries; precision, the proportion of true positive predictions out of all positive predictions made by the model; recall, the proportion of actual positive cases that are correctly identified by the model; and F1 score, the harmonic mean of precision and recall. In addition, we compute the area under the curve (AUC) statistic based on the model's receiver operating characteristic curve (Swets 2014) to assess the predictive performance across different probability thresholds used for classification. As Table 1 illustrates, the validation results suggest that a latent space of $\kappa = 30$ dimensions is sufficient, as model performance does not significantly improve beyond this threshold. Next, we present the results of our model comparison.

[4] A visual inspection of the objective function, shown in Figure W5 in the Web Appendix, confirms the model convergence.

[5] Table W13 in the Web Appendix provides further details on the selection of hidden units.

Benchmark Models

We next compare the predictive performance of our model against several benchmark models. While all these models yield latent representations of the entities, a basic requirement for handling the complex structures of consumer collections, they differ in the type of representations they produce and, consequently, in their effectiveness for various managerial tasks. As outlined in the literature review, we classify these models into shallow embedding and deep embedding approaches. Among the deep embedding models, we further distinguish between those that incorporate message-passing mechanisms to explore graph structures and those that do not. Next, we briefly discuss each of these competing models.

Shallow representation models. We estimate three models that yield shallow embeddings:

- **Matrix factorization (MF) and neural matrix factorization (NMF; Rendle et al. 2020).** These models directly estimate the latent embeddings of entities without leveraging entity features or graph structures. While MF models use dot-product decoders, NMF employs multilayered perceptrons to specify the probabilities of link formation in the data. As these models do not use encoders, they are incapable of handling new entities that are not encountered in the training data.
- **Hypergraph embeddings for music recommendation (HEMR; La Gatta et al. 2022).** This model represents playlist data using three hypergraphs, where consumers act as hyperedges over playlists and songs, playlists serve as hyperedges over songs, and tags function as hyperedges over songs. It then applies random walks on these hypergraphs to generate sequences of consumers, playlists, songs, and tags. A Word2vec model is trained on these sequences to infer entity embeddings, which are subsequently used to predict links. Unlike our approach, this model follows a two-stage process and ultimately produces shallow embeddings, limiting its ability to capture the complex relationships in consumer collections.

Non-message-passing deep representation models. These models employ neural networks to infer entity representations by leveraging their individual features. However, they do not incorporate the graph/hypergraph structure of consumer collections, meaning embeddings are not influenced by relationships with neighboring entities. We evaluate two neural network models of this type:

- **VAE (Kingma and Welling 2013).** This model only inputs entity features to the encoders to infer their latent embeddings. Specifically, it incorporates three deep nonlinear encoders for the three types of entities in our data to specify the means and standard deviations of the variational distributions.

Table 2. Conceptual Differences Between Proposed and Benchmark Models.

	Message Passing			Representation	
Model	**Interlocked**	**Disconnected**	**None**	**Deep**	**Shallow**
HEMR			x		x
MF			x		x
NMF			x		x
HCNF			x	x	x
VAE			x	x	
GCN		x		x	
HeteHG-VAE		x		x	
NIHCNN		x		x	
Proposed model	x			x	

- **Hybrid collaborative neural filtering (HCNF; Sarker and Matin 2021).** This model combines collaborative and content filtering approaches popular in the recommendation system literature. It differs from the VAE in two ways. First, it is not a generative model as it lacks priors over the representations. Second, it assigns two low-dimensional vectors for each entity in the data. The first embedding vector $\mathbf{z}$ represents a deep embedding, learned using a neural network that compresses entity features into a low-dimensional space. The second vector $\mathbf{z}'$ is a shallow, feature-free representation, similar to those used in matrix factorization. Links in the hypergraph are predicted using the sum of dot products involving both embedding types. For instance, the probability of a connection between a song and a playlist is given by $p\left(H^{ps}_{jk}=1|\mathbf{z}^p_j,\mathbf{z}^s_k,\mathbf{z}'^p_j,\mathbf{z}'^s_k\right)=\sigma\left(\mathbf{z}^p_j(\mathbf{x}^p_j)^\top\mathbf{z}^s_k(\mathbf{x}^s_k)+\mathbf{z}'^{p\top}_j\mathbf{z}'^s_k\right)$.

Message-passing deep representation models. The final set of benchmark models integrates entity features along with graph/hypergraph structures to generate latent representations. We estimate five variants within this group:

- **Graph convolutional neural networks (GCNs; Kipf and Welling 2016b).** GCNs differ from our approach as they are designed for traditional graphs and model dyadic connections between entities of the same type. In our setting, this model constructs three separate graphs for listeners, playlists, and songs. Consumers who share a common song are connected in the consumer graph. Similarly, playlists (songs) that share common consumers or songs (playlists) are connected in the playlist (song) graph. The model then applies graph convolution to propagate and update entity features based on the adjacency matrices of their respective graphs. However, unlike our approach, GCNs do not capture the higher-order relationships inherent in hypergraph structures.
- **Heterogeneous hypergraph variational autoencoder (HeteHG-VAE; Fan et al. 2021).** This model represents consumers as hyperedges over songs and playlists. The consumer encoder utilizes two hypergraphs, user–song and users–playlists, to exchange information among consumers who share playlists and songs. An attention mechanism is then employed to merge the two learned embeddings. The representations of songs and playlists are inferred separately using their features, which are propagated through their respective hypergraphs via convolutional operations. This model differs from ours in two ways. First, its message-passing mechanism is constrained within specific hypergraphs. While we allow songs to exchange information with other songs through shared playlists and users, HeteHG-VAE restricts songs to pass messages only within the user–song hypergraph. The second distinction lies in the decoder structure. Our model decodes entries across three heterogeneous hypergraphs, whereas HeteHG-VAE only decodes two hypergraphs. We test two variants of this model. The first, HeteHG-VAE 1, retains the original specification. In the second variant, HeteHG-VAE 2, we additionally decode the playlist–song hypergraph.
- **Noninterlocked hypergraph convolutional neural network (NIHCNN).** This model highlights the importance of interlocked hypergraphs and the representational limitations that arise when the interlocked structure is ignored. NIHCNN applies two separate hypergraphs, consumer–playlist and playlist–song, with message-passing restricted within each hypergraph. Thus, songs share information only through playlists but not through consumers. We consider two variants of this model differing in their decoders: NIHCNN 1 reconstructs only the original two hypergraphs, whereas NIHCNN 2 decodes all three hypergraphs but without integrating them into a unified interlocked structure.

Table 2 summarizes the main conceptual differences between our proposed approach and the benchmark models.

Model Comparison

We use the validation dataset to identify the optimal set of hyperparameters for each benchmark model (see Web Appendix J), and compare their predictive performance using

Table 3. Predictive Performance of Proposed and Benchmark Models.

Model	Hit Rate (%)	Precision (%)	Recall (%)	F1 Score (%)	AUC
Message-Passing Deep Representations					
Proposed model	**85.98**	**89.67**	**81.34**	**85.30**	**.90**
NIHCNN 1	61.25	60.74	63.66	62.16	.65
NIHCNN 2	79.28	84.81	71.33	77.49	.81
HeteHG-VAE 1	59.89	60.11	58.80	59.45	.61
HeteHG-VAE 2	76.21	83.90	64.86	73.16	.83
GCN	73.06	72.14	75.15	73.61	.82
Non-Message-Passing Deep Representations					
HCNF	**80.71**	79.70	**82.42**	**81.03**	**.88**
VAE	79.12	**80.43**	76.98	78.66	.87
Shallow Representations					
HEMR	73.33	66.14	**95.57**	78.18	.78
MF	**77.42**	**74.28**	83.89	**78.79**	**.84**
NMF	66.88	67.99	63.77	65.81	.72

Notes: Bold digits indicate the best metrics in each model category.

the best-selected architectures. Table 3 reports the hit rate, precision, recall, F1 score, and AUC statistics for the test dataset. Overall, our proposed model achieves the highest hit rate of 85.98%, precision of 89.67%, F1 score of 85.30, and AUC of .90. In addition, Table W16 in the Web Appendix presents model performance under different data modalities. Together, these results confirm that the superior performance of our model is primarily driven by its interlocked hypergraph structure, which utilizes a sophisticated message-passing mechanism to effectively harness the rich information in the features of the different entities, along with the simultaneous reconstruction of these hypergraphs by the decoder.

Message-passing deep representations. Our model outperforms all others in this category. While NIHCNN 2 comes closest in accuracy, its reliance on uncoupled hypergraphs limits message propagation, reducing overall performance. NIHCNN 1 underperforms further, as it reconstructs only two of the three hypergraphs. HeteHG-VAE 2 struggles due to restricted message exchange among consumers and the confinement of playlists and songs within separate hypergraphs. Although GCN enables message dissemination, it fails to model hyperedges, treating all entities as vertices and relying solely on dyadic connections, which weakens its ability to capture overlapping relationships among entities. In contrast, our model leverages stacked incidence matrices to effectively integrate higher-order relationships into its computations.

Non-message-passing deep representations. Both models in this class perform notably worse than ours. HCNF outperforms VAE across most metrics, except for precision, due to its additional embeddings, whereas VAE relies solely on embeddings generated from entity features. However, neither approach leverages message-passing to incorporate the higher-order structural information in graphs/hypergraphs, which limits their predictive performance.

Shallow representations. The HEMR, MF, and NMF models exhibit inferior performance mainly because they cannot harness the rich information embedded in the textual and acoustic features. MF consistently outperforms NMF, confirming findings from Rendle et al. (2020) that dot-product decoders often deliver better results than more complex alternatives. HEMR, which relies on shallow embeddings without feature convolution, characterizes entity co-occurrences using random walk–generated sequences rather than reconstructing hypergraphs, reducing its predictive accuracy.

Having demonstrated the predictive superiority of our model over several state-of-the-art alternatives, we next examine the latent embedding space it generates.

Latent Embedding Space

We first describe the learned joint latent space for songs, playlists, and consumers, illustrating how our model provides insights into the musical preferences of listeners and the listening contexts of songs and playlists. We then discuss managerial applications of our model for song recommendation and playlist curation.

Figure 8 displays two-dimensional t-distributed stochastic neighbor embedding (t-SNE) projections (Van der Maaten and Hinton 2008) of the variational means of the latent embeddings for a subset of songs and playlists. The t-SNE method visualizes high-dimensional data by projecting it into a two-dimensional space. Figure 8, Panel A, depicts songs of ten randomly selected playlists, with each song color-coded according to its playlist. Songs within the same playlist clearly cluster together, highlighting that the embeddings coherently capture playlist-based song relationships and effectively reflect consumer categorization behaviors. Similarly, Figure 8, Panel B, illustrates playlists from ten randomly selected consumers, with each consumer's playlists shown with a distinct symbol. Playlists belonging to the same consumer also form tight clusters. These clustering patterns suggest that songs and playlists positioned close together in the latent embedding space are likely to deliver similar musical experiences and that the estimated embeddings can be subsequently used to satisfy heterogeneous consumer musical tastes.

Our model is able to capture similarities among consumers and efficiently discover items and categories with similar musical themes. Next, we illustrate these similarity patterns.

Consumer similarity. We use the cosine distance to measure the similarity between the variational mean embeddings $\boldsymbol{\mu}_i^u$ and $\boldsymbol{\mu}_{i'}^u$ of two consumers i and i′. Figure 9 displays word clouds of tags from the song collection of a focal consumer and those of the two nearest consumers. Consumers with similar latent embeddings tend to like similar songs and playlists. For instance, Figure 9, Panel A, shows that the focal consumer is interested in genres such as electronic, pop, indie, and dance, and it is evident that their closest neighbors also highlight tags associated with these genres. Music platforms can capitalize on these similarities to cross-recommend playlists and songs to these consumers, potentially enhancing user experience and engagement.

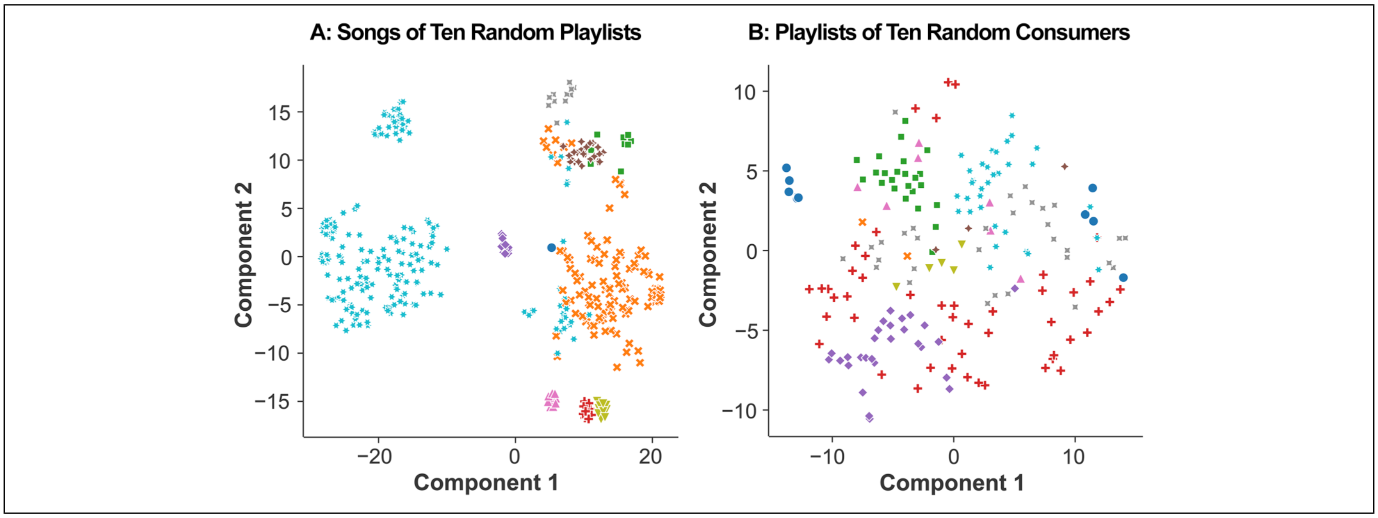


Figure 8. Examples of t-SNE Graphs of Random Songs and Playlists.

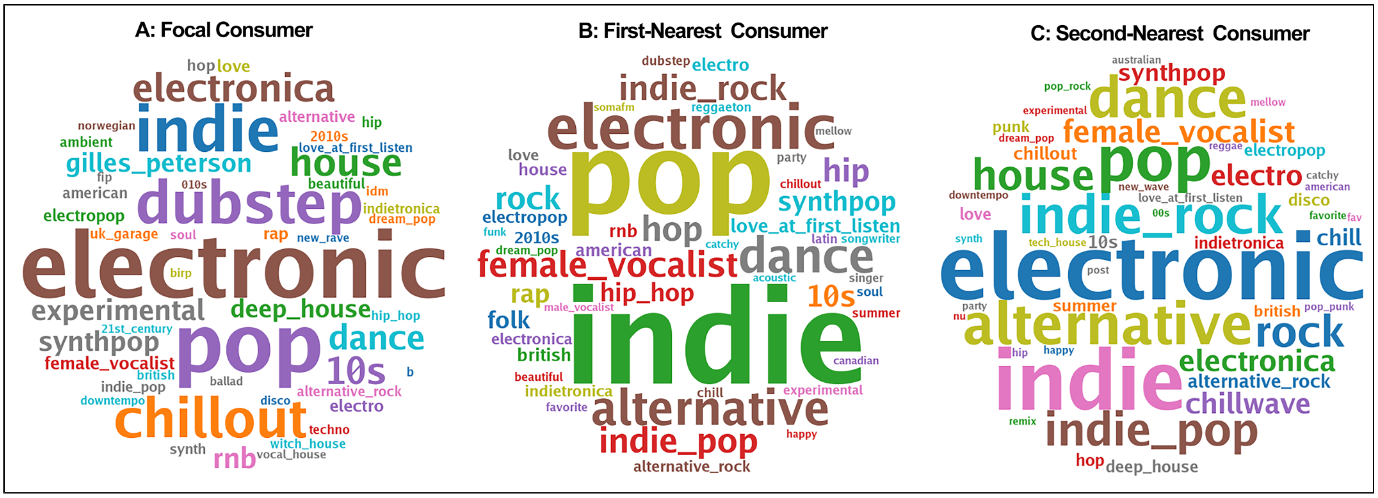


Figure 9. Word Clouds of Tags from Songs in Collections of Focal Consumer and Nearest Neighbors in Latent Space.

Similarity among consumer collections can also be assessed through the categorization of songs into playlists. Figure 10 displays word clouds for two playlists of the focal consumer ("Liked from Radio" and "Starred") and two playlists of this consumer's nearest neighbor ("Hermitude" and "cfcf – Outsiders"). It is noteworthy that all the playlists prominently feature electronic music, as indicated by the common tag "electronic." This highlights the alignment of contextual musical preferences recovered by the latent embeddings. Additional examples illustrating category and item similarity are provided in Web Appendix L.

Missing tags. We next illustrate how our model leverages the message-passing mechanism to infer item representations in the presence of missing features. For this task, we consider a song with missing tag data, retaining only its acoustic features. As an illustrative example, we use "Abduction from Seraglio, K. 384: Overture" by Mozart.

Table 4 displays the top ten songs most similar to the Mozart song, as inferred by our model and by the VAE. Notably, despite the absence of tags, our model successfully identifies songs primarily from the classical genre, with the two closest neighbors being compositions by Mozart. The word cloud of the top ten closest songs highlights tags typical of classical music, such as "classical," "piano," and "instrumental." In contrast, the VAE suggests a more varied and less coherent mix of songs. For instance, its top recommendation is "Stay in Spring Field" by Thanksgiving, who is a rock artist. Furthermore, none of the VAE's top ten recommendations have associated tags, indicating that it relies heavily on the *missing* tag indicator. These differences underscore the strengths of our model, which, unlike the VAE, whose encoders rely solely on song features, utilizes the message-passing mechanism within hypergraphs to learn more nuanced and meaningful song embeddings.

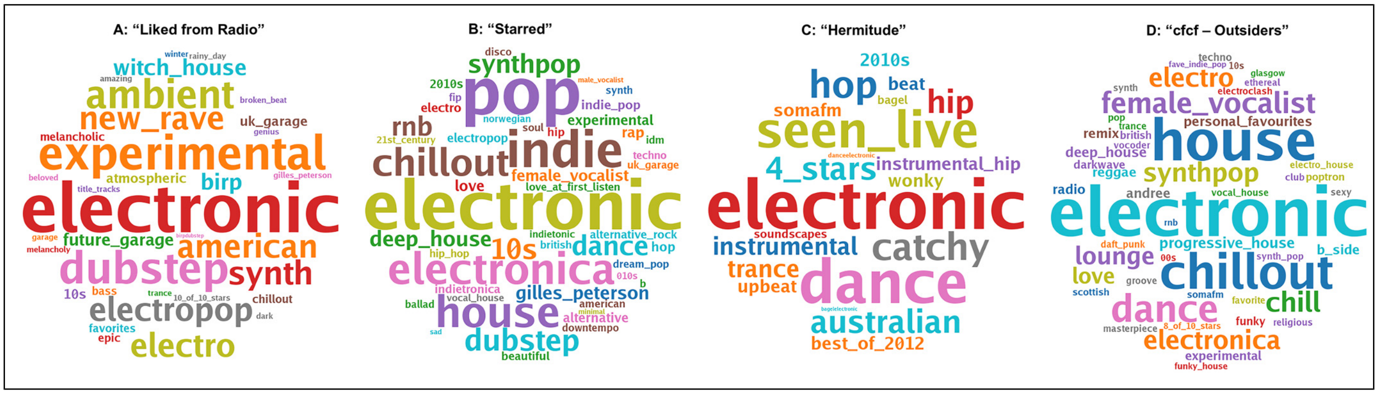


Figure 10. Word Clouds of Playlists from Focal Consumer ("Liked from Radio" and "Starred") and Nearest Neighbor ("Hermitude" and "cfcf – Outsiders").

Table 4. Top Ten Songs Most Similar to "Abduction from Seraglio, K. 384: Overture" by Mozart, as Identified by the Proposed Model and VAE.

Artist	Song	Word Cloud
Proposed Model		
Wolfgang Amadeus Mozart	"Piano Sonata #11 in A, K 331 - I. Andante Grazioso"	
Wolfgang Amadeus Mozart	"Concerto for Flute and Harp in C Major, K. 299: II. Andantino"	
Mozart Lullabies Baby Lullaby	"Carol of the Bells"	
Jack Brymer	"Mozart: Concerto for Clarinet and Orchestra in A (K. 622)"	
Radio Orchestra Berlin	"Turkish March"	
Radio Orchestra Berlin	"Overture: The Marriage of Figaro"	
Gervase De Peyer	"Clarinet Concerto in A Major, K. 622: II. Adagio"	
Wolfgang Amadeus Mozart	"Ave verum corpus K. 618"	
Mozart Lullabies Baby Lullaby	"Brahms Lullaby—Baby Sleeping Song"	
Mozart Lullabies Baby Lullaby	"Brahms—Hungarian Dances"	
VAE		
Thanksgiving	"Stay in Spring Field"	
Donato Di Stefano	"Le Nozze Di Figaro (The Marriage of Figaro), K. 492: Final Scene: Gente! Gente!"	
Sun Ra & His Astro-Infinity Arkestra	"Love in Outer Space (Alternate Take)"	
The Hi and Lo	"My Father's House"	
Ithamara Koorax	"Can't Take My Eyes Off of You"	
Ithamara Koorax	"Corcovado"	
Renée Fleming	"Still, Still, Still"	
Monica Zetterlund	"Men Tiden Går—As Time Goes By"	
Maranatha! Praise Band	"Have Faith in God"	
Maranatha! Promise Band	"I Need You/Change My Heart, O God"	

After describing the topological and probabilistic aspects of the latent space, we next turn to the marketing applications of our model.

Managerial Applications

In this section, we elaborate on how our estimated model can be leveraged for several generative applications, including (1) generating novel and personalized categories, (2) creating new categories based on titles, (3) adding new products to existing product collections, (4) recommending existing categories to users, and (5) adaptively incorporating new products into collections. Subsequently, we use our music playlist data to illustrate these applications.[6]

Generating Novel Categories

A new playlist j can be generated by first sampling the playlist embedding $\tilde{\mathbf{z}}^{p}_{j} \sim \mathcal{N}(0_{\kappa}, \mathrm{I}_{\kappa})$ and then identifying the songs in our

[6] To strengthen the managerial implications, Web Appendix M presents playlist reconstruction results, evaluated across all playlists for every user rather than selectively chosen examples. The findings are promising, highlighting the robustness and generalizability of the proposed model.

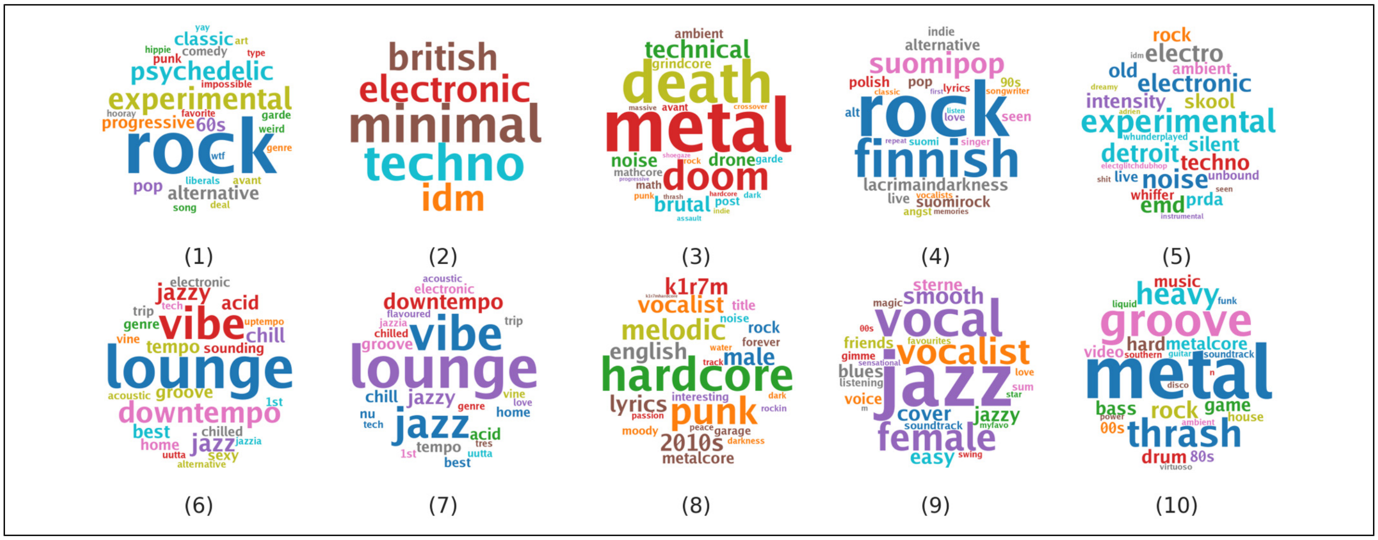
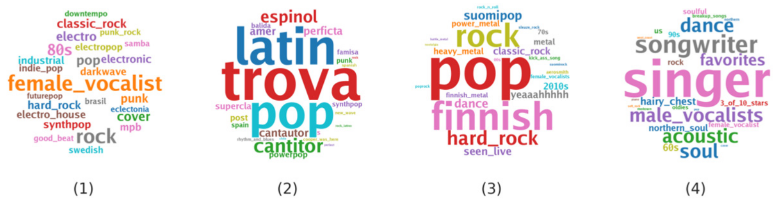
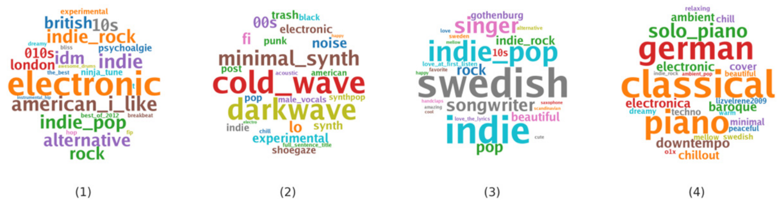


Figure 11. Word Clouds of Ten Randomly Generated Playlists.

Table 5. Two Target Consumers and Their Four Artificially Generated Playlists.

Target Consumer	Artificially Generated Playlists
Consumer 1	
	
Consumer 2	
	

data that have the highest expected probability $p\left(H_{jk}^{ps}=1\right)=\mathbb{E}_{q(\mathbf{z}_k^s)}\left[\sigma\left(\tilde{\mathbf{z}}_j^{p\top}\mathbf{z}_k^s\right)\right]$ of connecting with this novel playlist. The expectation is computed using Monte Carlo draws of $\mathbf{z}_k^s$ from its variational distribution $\mathcal{N}\left(\boldsymbol{\mu}_k^s, \Sigma_k^s\right)$. We randomly generate ten novel playlists consisting of 16 songs (the median number of songs in a playlist in our dataset) with the highest probabilities of being included in each playlist. Figure 11 shows the word clouds of tags for the songs in each playlist. We observe that the generated playlists showcase a diverse range of musical themes. For example, Playlist 1 focuses on rock and experimental music. Playlists 6 and 7 share the "lounge" tag, highlighting atmospheric and soothing music. Playlists 3 and 10 emphasize metal music, while Playlist 2 is centered around minimal electronic music. Playlist 9 features jazz and female singers.

Generating personalized categories. We can further leverage this generative mechanism in a targeted fashion to design personalized playlists for a given consumer i with a mean embedding $\boldsymbol{\mu}_i^u$. To accomplish this, we first sample random playlist embeddings $\tilde{\mathbf{z}}_{ij}^p \sim \mathcal{N}\left(\boldsymbol{\mu}_i^u, \tau \times \mathbf{I}_\kappa\right)$, where $\tau > 0$ is chosen by the firm. A small τ yields playlist embeddings that are very similar to the consumer's embedding. Then, we can connect appropriate songs in our dataset to these playlist embeddings using the expected probabilities $p\left(H_{jk}^{ps}=1\right)=\mathbb{E}_{q(\mathbf{z}_k^s)}\left[\sigma\left(\tilde{\mathbf{z}}_{ij}^{p\top}\mathbf{z}_k^s\right)\right]$.

For instance, consider two arbitrary consumers from our dataset. For each consumer, we sample four random playlist embeddings using $\tau = .5$ and populate each of these generated playlists with the 16 songs having the highest inclusion probabilities. Table 5 presents word clouds of the song collections for each consumer's four artificially generated playlists. We observe that the generated playlists consistently reflect cohesive musical

Table 6. User-Specified Title-Based Playlist Generation.

Word Clouds of Tags	Artist	Song
"Thanksgiving Dinner with Friends"		
	Jodeci	"Come Up to My Room (feat. Tha Dogg Pound)"
	Lil Wayne	"It's Good (featuring Drake & Jadakiss)"
	Jace Everett	"Bad Things"
	John Legend	"It's Over (feat. Kanye West)"
	Little River Band	"Cool Change - 2010 Digital Remaster"
	Van Morrison	"Golden Autumn Day - 2007 Remastered"
	John Legend	"King & Queen"
	Fleetwood Mac	"Landslide - Live /Fade"
	Goran Bregović	"7/8 & 11/8"
	Goran Bregović	"Death"
"Symphonies and Piano to Study and Focus"		
	101 Strings Orchestra	"The Green Leaves of Summer"
	2Cellos	"Benedictus"
	2Cellos	"Candle in the Wind"
	2Cellos	"Celloverse"
	2Cellos	"Clocks"
	2Cellos	"Every Breath You Take"
	2Cellos	"Fields of Gold (Bonus Track)"
	2Cellos	"Highway to Hell"
	2Cellos	"Hysteria"
	2Cellos	"I Will Wait"

themes aligned with each consumer's preferences. For example, Consumer 1 enjoys Spanish, dance, rock, and pop music. The four personalized playlists directly reflect these preferences, with each playlist focusing on different themes: rock (Playlists 1 and 3), Latin (Playlist 2), and dance (Playlist 4). Consumer 2, in contrast, favors electronic and ambient music. Accordingly, our model generates four playlists that embody these musical styles; that is, Playlists 1 and 2 focus on electronic and cold wave genres,[7] whereas Playlist 4 concentrates on classical and calm music. Finally, we illustrate how this mechanism of targeted playlist generation can be extended to mimic exploration–exploitation approaches for playlist recommendation in Web Appendix N.

Generating title-based categories. We next demonstrate how our model can generate on-demand playlists based on a user-specified playlist title that reflects a specific consumption goal or context. First, we extract the feature vector from the playlist title words and identify existing playlists in our dataset with similar titles. Then, we construct a hypothetical omnibus playlist containing all songs from these identified playlists. This hypothetical playlist is then added as an additional column in the incidence matrix H^{ps}, producing an updated matrix $\tilde{H}^{ps}$. Using this augmented incidence matrix, our encoder generates an embedding for the new playlist. In this step, the encoder is used specifically to infer a meaningful embedding $\mathbf{z}_j^p$ for the new playlist, rather than deterministically selecting its songs. Finally, we use this embedding to select songs with the highest expected probability of inclusion, as described previously.

We illustrate this application by generating two novel, title-based playlists. The first playlist is titled "Thanksgiving Dinner with Friends," and the second is titled "Symphonies and Piano to Study and Focus." Table 6 shows the top ten songs with the highest probabilities of inclusion in each target playlist, along with word clouds of the top 100 recommended songs. We observe that the playlist designed for the dinner context mixes rock and downtempo music by artists such as John Legend and Lil Wayne, whereas the second playlist includes a blend of classical and calm music with artists such as 101 Strings Orchestra and 2Cellos. Personalized playlists can similarly be created by first associating them with specific consumers in the interlocked hypergraph, allowing the new playlists to exchange information with existing playlists of these users. This approach addresses more personalized and context-specific needs. We demonstrate the potential of this method to generate emotion-based playlists in Web Appendix O.

Expanding Consumer Collections Through Personalized Recommendations

Having discussed playlist generation, we next demonstrate how our model can expand existing consumer collections by recommending additional items to consumers, identifying existing playlists suitable for specific contexts, and suggesting products to include in a consumer's existing playlists.

[7] Cold wave is a musical genre related to the electronic music of the 1970s.

Table 7. Two Consumers and Their Recommended Songs.

Consumer 1	Top 100 Songs	Consumer 2	Top 100 Songs
			
Artist	**Song**	**Artist**	**Song**
#1 Halloween TV	"Buffy the Vampire Slayer"	AWOLNATION	"Everybody's Got a Secret"
'N Sync	"Digital Get Down"	AWOLNATION	"Guilty Filthy Soul"
'N Sync	"It's Gonna Be Me"	AWOLNATION	"Jump on My Shoulders"
'N Sync	"This I Promise You"	AWOLNATION	"Mf"
'Nsync	"Merry Christmas Happy Holiday"	AWOLNATION	"Not Your Fault - Robert Delong Remix"
*NSYNC	"Girlfriend"	AWOLNATION	"People - Thomas from Ghostland Observatory Remix"
*NSYNC	"I'll Never Stop"	AWOLNATION	"Sail - Borgore Pop the Sweating I'm Sailing Remix"
*NSYNC	"It Makes Me Ill"	AWOLNATION	"Sail - Dan the Automator Remix"
*NSYNC	"It's Gonna Be Me"	AWOLNATION	"Sail - Feed Me Remix"
*NSYNC	"Tearin' Up My Heart"	AWOLNATION	"Sail - TDE Remix Featuring Kendrick Lamar & Ab Soul"

Recommending items for consumer collections. Our model can be leveraged to recommend new items for inclusion in a consumer's existing collection. On digital music platforms, recommending individual songs is crucial. Spotify's Discover Weekly service is an example of such personalized recommendations. We compute the probability that consumer i will connect with song k as $p\left(H^{us}_{ik}=1\right)=\mathbb{E}_{q(\mathbf{z}^u_i,\mathbf{z}^s_k)}\left[\sigma\left(\mathbf{z}_i^{u\top}\mathbf{z}^s_k\right)\right]$ to identify existing songs that may appeal to the consumer but are not currently part of their collection. Table 7 displays word clouds representing the tags of the songs in the collections of two consumers, alongside word clouds of the tags for the top 100 recommended songs for each consumer. The table also exhibits the ten songs with highest probabilities of being liked by each consumer.[8] Note that the word clouds of the consumers and their recommended songs share numerous tags. For example, Consumer 1's collection includes songs that span genres, indicated by the tags "pop," "90s," and "rnb." The model suggests that this consumer might also enjoy songs by *NSYNC, a 1990s group known for pop music. Consumer 2 prefers alternative and emo, a rock subgenre that focuses on emotions and confessional lyrics. Our model consequently recommends compatible songs of a similar nature, such as tracks by the alternative rock band AWOLNATION. In addition to rock rhythms, songs like "Everybody's Got a Secret" and "Guilty Filthy Soul" are lyrical and storytelling, aligning with the emo genre appreciated by Consumer 2.

Recommending existing categories. Apart from recommending single items, our model can also recommend entire product bundles based on the hypergraph H^{up} along with the user and category embeddings. We illustrate this process by recommending existing playlists to consumers. Table 8 presents the word clouds of the collections of two consumers and of the top two recommended playlists for each consumer. Note that Consumer 1 prefers rock and alternative rock music. The recommended playlists indeed reflect such preference. For example, the playlist "Money Mark – Push The Button" pertains to rock music of the 1990s, whereas the playlist "Morphine – Cure For Pain" focuses on alternative rock songs. Consumer 2, in contrast, enjoys downtempo, electronic, and chillout music. The top recommended playlists are centered around these musical styles. The playlist "Newest" mixes love, pop, and road trip music, and the playlist "All the Pretty Lights" relates to electronic and downtempo songs. Having shown how our model recommends existing playlists for different consumers, we next discuss the use of our proposed approach to expand existing consumer categories with additional items.

Recommending items for category expansion. We can expand existing categories by recommending additional items relevant to the same consumption context. To illustrate this, we consider expanding a playlist titled "christmas." Table 9 presents the word cloud of this playlist and the top ten extra songs that our model recommends adding to it. As reflected in the word cloud, the playlist clearly relates to Christmas festivities, with tags such as "christmas," "holiday," and "xmas." Our model recommends ten songs that are all related to Christmas. For instance, the song with the highest probability of inclusion is "Joy to the World" by Aretha Franklin, followed by "Have Yourself a Merry Little Christmas" by Dan + Shay, and "Heaven Everywhere" by Francesca Battistelli.

Adaptive expansion of collections and categories. Our modeling framework enables firms to accommodate evolving consumer

[8] We have excluded songs that are already present in the collections of each consumer.

Table 8. Two Consumers and Their Top Recommended Playlists.

Consumer	Top Recommended Playlists	
Consumer 1 	 "Money Mark – Push the Button"	 "Morphine – Cure for Pain"
Consumer 2 	 "Newest"	 "All the Pretty Lights"

Table 9. Top Recommended Additions to Playlist "Christmas."

	Top Recommended Additions	
Word Cloud	**Artist**	**Song**
	Aretha Franklin	"Joy to the World"
	Dan + Shay	"Have Yourself a Merry Little Christmas"
	Francesca Battistelli	"Heaven Everywhere"
	Luther Vandross	"At Christmas Time"
	K. Michelle	"Christmas Night"
	Frank Sinatra	"Mistletoe and Holly - 2002 - Remaster"
	Chicos de Barrio	"Bailando en navidad"
	Brett Eldredge	"What Christmas Means to Me - 2014 CMA Country Christmas Performance"
	Hunter Hayes	"Merry Christmas Baby - 2014 CMA Country Christmas Performance"
	Bing Crosby	"Here Comes Santa Claus – Right Down Santa Claus Lane"

preferences by dynamically updating playlist embeddings as songs are added or removed. Consider a target playlist initially composed of a set of songs. An alteration to this playlist results in updated hypergraph incidence matrices $\tilde{H}^{up}$ and $\tilde{H}^{ps}$, which our model leverages to update the playlist embedding. Recall that the embedding $\mathbf{z}_j^p$ of playlist j is sampled from $\mathcal{N}\left(\boldsymbol{\mu}_j^p, \Sigma_j^p\right)$, with variational parameters given by $\boldsymbol{\mu}_j^p = \boldsymbol{\mu}_j^p\left(H^{up}, H^{ps}, X^p; \phi_\mu^p\right)$ and $\mathbf{l}\boldsymbol{\sigma}_j^p = \mathbf{l}\boldsymbol{\sigma}_j^p\left(H^{up}, H^{ps}, X^p; \phi_{l\sigma}^p\right)$. After the playlist is modified, the playlist embedding is updated with new variational parameters $\tilde{\boldsymbol{\mu}}_j^p = \tilde{\boldsymbol{\mu}}_j^p\left(\tilde{H}^{up}, \tilde{H}^{ps}, X^p; \phi_\mu^p\right)$ and $\widetilde{\mathbf{l}\boldsymbol{\sigma}}_j^p = \widetilde{\mathbf{l}\boldsymbol{\sigma}}_j^p\left(\tilde{H}^{up}, \tilde{H}^{ps}, X^p; \phi_{l\sigma}^p\right)$, keeping the original neural network parameters ϕ_μ^p and $\phi_{l\sigma}^p$ fixed. The updated embedding is then sampled as $\tilde{\mathbf{z}}_j^p \sim \mathcal{N}\left(\tilde{\boldsymbol{\mu}}_j^p, \tilde{\Sigma}_j^p\right)$. Finally, songs can be allocated to this updated playlist based on the probability $p\left(\tilde{H}_{jk}^{ps} = 1\right) = \mathbb{E}_{q(\tilde{\mathbf{z}}_j^p, \tilde{\mathbf{z}}_k^s)}\left[\sigma\left(\tilde{\mathbf{z}}_j^{pT} \tilde{\mathbf{z}}_k^s\right)\right]$.

We illustrate this adaptation through an example that also demonstrates our approach's ability to handle missing features. We begin with an initial *untitled* playlist containing three Mozart songs. This playlist is not associated with any consumer, allowing its embedding to be inferred solely based on its existing songs. We then employ our model to suggest additional songs for this playlist. To illustrate adaptation, we next add three tracks by Tiësto, a Dutch DJ and electronic music producer, to the playlist. After updating its embedding based on this alteration, we again utilize our model to generate new song recommendations for the modified playlist.

Table 10 presents the songs from the initial and altered playlists, their corresponding word clouds, the top ten recommended additions for each playlist, and word clouds generated from the top 100 recommended songs. The initial playlist does not have a word cloud since its three Mozart songs lack available tags. Nevertheless, our model appropriately recommends classical music tracks for this playlist, as reflected by the word cloud from the top 100 recommendations, which highlights Mozart's

Table 10. Song Recommendations for Initial and Altered Playlists.

Initial Playlist		
Artist	**Song**	**Playlist Word Cloud**
Wolfgang Amadeus Mozart	"Abduction from Seraglio, K. 384: Overture"	
Wolfgang Amadeus Mozart	"Adagio and Fugue in C Minor, K.546: 1. Adagio"	
Wolfgang Amadeus Mozart	"Ave verum corpus K. 618"	
Top 10 Recommended Songs for Initial Playlist		
Artist	**Song**	**Top 100 Songs' Word Cloud**
Ludwig van Beethoven	"Symphony No. 3 in E-Flat Major Op. 55"	
London Symphony Orchestra	"Symphony No. 3 in E-Flat Major Op. 55"	
Franz Schubert	"String Quartet No. 14 in D Minor D. 810"	
Franz Schubert	"Symphony No. 9 in C Major D. 944"	
Edvard Grieg	"Concerto in A Minor for Piano and Orchestra, Op. 16: II. Adagio"	
Edvard Grieg	"Peer Gynt Op. 23 XI. Solveigs Sang"	
Giuseppe Verdi	"La Traviata Overture"	
Johann Strauss II	"Geschichten aus dem Wienerwald (Tales from the Vienna Woods), Op. 325 Waltz"	
Ludwig van Beethoven	"Symphony No. 4 in B-Flat Major Op. 60 II. Adagio"	
Nikolai Rimsky-Korsakov	"Scheherazade Symphonic Suite Op. 35"	
Altered Playlist		
Artist	**Song**	**Playlist Word Cloud**
Wolfgang Amadeus Mozart	"Abduction from Seraglio, K. 384: Overture"	
Wolfgang Amadeus Mozart	"Adagio and Fugue in C minor, K.546: 1. Adagio"	
Wolfgang Amadeus Mozart	"Ave verum corpus K. 618"	
Tiësto	"Adagio for Strings"	
Tiësto	"Flight 643"	
Tiësto	"Suburban Train"	
Top 10 Recommended Songs for Altered Playlist		
Artist	**Song**	**Top 100 Songs' Word Cloud**
Ludwig van Beethoven	"Symphony No. 3 in E-Flat Major Op. 55"	
Anton Bruckner	"Symphony No. 3 in D Minor"	
The New York Philomusica Winds & A. Robert Johnson	"Serenade No. 10 in B-Flat Major K. 361"	
The New York Philomusica Winds	"Serenade No. 10 in B-Flat Major K. 361"	
Anton Bruckner	"Symphony No. 9 in D Minor"	
Antonio Vivaldi	"Le quattro stagioni (The Four Seasons), Op. 8 - Concerto No. 2 in G Minor, RV 315"	
Ludwig van Beethoven	"Symphony No. 5 in C Minor Op. 67"	
Maor Levi & Raul Siberdi	"Infatuation (Nitrous Oxide Remix)"	
Maor Levi & Raul Siberdi	"Infatuation (Original Mix)"	
www.breakfastexclusive.com	"Summer Showcase MixTape"	

musical style with tags such as "classical," "symphony," and "instrumental." Notably, after we altered the playlist by adding three songs from Tiësto, the recommendations shift significantly, even though the Mozart songs remain tagless. The updated word cloud now emphasizes tags like "classical," "ambient," and "new age." Remarkably, the top ten recommendations include an intriguing blend of classical and contemporary electronic genres, featuring compositions by Beethoven alongside tracks from electronic music DJs such as Maor Levi and Raul Siberdi. In Web Appendix P, we further compare our model's results with those from a VAE to illustrate the importance of the message-passing mechanism in our proposed modeling framework.

Conclusion

We propose a novel modeling framework to represent the structure and content of consumer collections and demonstrate how higher-order networks involving multiple interlocked hypergraphs can capture intricate linkages in such collections. We then develop a neural network model consisting of message-passing hypergraph convolutional layers to generate probabilistic deep embeddings of consumers, categories, and items. We apply this framework to model music collections in the main text and recipe collections in the Web Appendix.

In our primary application, we model music collections categorized into playlists to capture consumer preferences for songs in diverse listening contexts. We illustrate our modeling framework using a rich dataset of music playlists augmented with multimodal song features, including user-generated tags and acoustic fingerprints. The textual tags capture information on genres and listening contexts, while the acoustic features represent the technical aspects of the music. Our results show that the model predicts better than several sophisticated benchmarks and yields embeddings that reveal meaningful similarities among consumers, categories, and items. We then use our model for multiple generative and recommendation tasks of managerial importance, including the curation of novel personalized playlists, recommendation of songs or song collections, and adaptive playlist continuation, which allows for dynamic playlist expansion.

While we showcase our model's capabilities in the context of digital platforms for music and recipes, our framework is readily and broadly applicable to other consumer collections, particularly in settings where contextual preferences can be inferred from categorization structures and where commercial platforms can recommend both individual products and curated collections to consumers. A promising application is grocery shopping baskets, where interlocked hypergraph structures can be used to model relationships among various entities, generating latent embeddings that enhance preference modeling, product bundling, and shopping basket recommendations.

From a managerial perspective, this modeling framework offers significant potential to enhance decision-making, particularly in the realm of personalized recommendations and product bundling. By leveraging deep embeddings that capture nuanced consumer preferences, managers can develop more targeted marketing strategies, such as personalized category or product suggestions that align closely with user tastes and consumption contexts. Moreover, the model's capacity for adaptive category expansion and category-specific recommendations can drive user engagement, ultimately boosting retention and revenue. For organizations, particularly digital platforms with diverse consumer bases, this model provides a scalable approach to understanding complex consumer preferences, enabling more informed and data-driven marketing and product development decisions.

While our model offers several advantages over competing frameworks, it is important to acknowledge certain limitations and opportunities for future research. First, we use hypergraph convolutional neural network layers; however, layers based on the attention mechanisms (Vaswani et al. 2017; Veličković et al. 2017), which assign differential weights to messages from various neighbors, could be explored. We employ one-hot encoding for consumer feature vectors as we do not have demographic data. Our framework, however, can readily integrate observed user heterogeneity, similar to song and playlist features. Another limitation stems from the variational inference used to approximate posterior distributions, which may face challenges in accurately capturing the complex structure of hypergraphs and could underestimate posterior variances due to a potentially misspecified variational distribution. Future research could explore more expressive variational families or alternative inference methods to mitigate these issues.

In addition, future researchers could extend this work by leveraging large language models to generate embeddings for textual tags and playlist titles. Field implementations and real-world testing of our model would provide valuable validation of its generative outputs and offer insights into its impact on key business metrics such as customer retention and lifetime value. Our study demonstrates that hypergraphs offer a powerful framework for capturing higher-order relationships that ordinary graphs cannot adequately represent, and using ordinary graphs in such contexts often results in the loss of critical structural information. We encourage researchers to apply our hypergraph-based framework to other marketing problems, such as modeling grocery shopping baskets, organizing Pinterest boards and pins, capturing collaborative content creation by multiple influencers, and analyzing multichannel consumer behavior.

Coeditor

Rebecca Hamilton

Associate Editor

Eric T. Bradlow

Declaration of Conflicting Interests

The author(s) declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

Funding

The author(s) received no financial support for the research, authorship, and/or publication of this article.

ORCID iDs

Khaled Boughanmi https://orcid.org/0000-0002-4701-910X
Asim Ansari https://orcid.org/0000-0001-6964-6297
Yang Li https://orcid.org/0000-0002-7380-1883

References

Ansari, Asim, Skander Essegaier, and Rajeev Kohli (2000), "Internet Recommendation Systems," *Journal of Marketing Research*, 37 (3), 363–75.

Ansari, Asim, Yang Li, and Jonathan Z. Zhang (2018), "Probabilistic Topic Model for Hybrid Recommender Systems: A Stochastic Variational Bayesian Approach," *Marketing Science*, 37 (6), 987–1008.

Bai, Song, Feihu Zhang, and Philip H.S. Torr (2021), "Hypergraph Convolution and Hypergraph Attention," *Pattern Recognition*, 110, 107637.

Belk, Russell W. (1995), "Collecting as Luxury Consumption: Effects on Individuals and Households," *Journal of Economic Psychology*, 16 (3), 477–90.

Belk, Russell W. (2013), *Collecting in a Consumer Society*. Routledge.

Bianconi, Ginestra (2021), *Higher-Order Networks*. Cambridge University Press.

Blei, David M., Alp Kucukelbir, and Jon D. McAuliffe (2017), "Variational Inference: A Review for Statisticians," *Journal of the American Statistical Association*, 112 (518), 859–77.

Bodapati, Anand V. (2008), "Recommendation Systems with Purchase Data," *Journal of Marketing Research*, 45 (1), 77–93.

Boughanmi, Khaled and Asim Ansari (2021), "Dynamics of Musical Success: A Machine Learning Approach for Multimedia Data Fusion," *Journal of Marketing Research*, 58 (6), 1034–57.

Bradlow, Eric T. and Vithala R. Rao (2000), "A Hierarchical Bayes Model for Assortment Choice," *Journal of Marketing Research*, 37 (2), 259–68.

Burnap, Alex, John R. Hauser, and Artem Timoshenko (2023), "Product Aesthetic Design: A Machine Learning Augmentation," *Marketing Science*, 42 (6), 1029–56.

Burton, Benjamin J. and Joyce P. Jacobsen (1999), "Measuring Returns on Investments in Collectibles," *Journal of Economic Perspectives*, 13 (4), 193–212.

Carey, Catherine (2008), "Modeling Collecting Behavior: The Role of Set Completion," *Journal of Economic Psychology*, 29 (3), 336–47.

Carroll, J. Douglas and Paul E. Green (1997), "Psychometric Methods in Marketing Research: Part II, Multidimensional Scaling," *Journal of Marketing Research*, 34 (2), 193–204.

Chakraborty, Ishita, Minkyung Kim, and K. Sudhir (2022), "Attribute Sentiment Scoring with Online Text Reviews: Accounting for Language Structure and Missing Attributes," *Journal of Marketing Research*, 59 (3), 600–622.

Chami, Ines, Sami Abu-El-Haija, Bryan Perozzi, Christopher Ré, and Kevin Murphy (2022), "Machine Learning on Graphs: A Model and Comprehensive Taxonomy," *Journal of Machine Learning Research*, 23 (89), 1–64.

Chung, Jaihak and Vithala R. Rao (2012), "A General Consumer Preference Model for Experience Products: Application to Internet Recommendation Services," *Journal of Marketing Research*, 49 (3), 289–305.

Chung, Tuck Siong, Roland T. Rust, and Michel Wedel (2009), "My Mobile Music: An Adaptive Personalization System for Digital Audio Players," *Marketing Science*, 28 (1), 52–68.

Datta, Hannes, George Knox, and Bart J. Bronnenberg (2018), "Changing Their Tune: How Consumers' Adoption of Online Streaming Affects Music Consumption and Discovery," *Marketing Science*, 37 (1), 5–21.

Dew, Ryan, Asim Ansari, and Olivier Toubia (2022), "Letting Logos Speak: Leveraging Multiview Representation Learning for Data-Driven Branding and Logo Design," *Marketing Science*, 41 (2), 401–25.

Dhillon, Paramveer S and Sinan Aral (2021), "Modeling Dynamic User Interests: A Neural Matrix Factorization Approach," *Marketing Science*, 40 (6), 1059–80.

Ellis, Daniel P.W., Brian Whitman, Tristan Jehan, and Paul Lamere (2010), "The Echo Nest Musical Fingerprint," https://academiccommons.columbia.edu/doi/10.7916/D8SQ98QJ/download.

Fan, Haoyi, Fengbin Zhang, Yuxuan Wei, Zuoyong Li, Changqing Zou, Yue Gao, and Dai Qionghai (2021), "Heterogeneous Hypergraph Variational Autoencoder for Link Prediction," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44 (8), 4125–38.

Feng, Yifan, Haoxuan You, Zizhao Zhang, Rongrong Ji, and Yue Gao (2019), "Hypergraph Neural Networks," *Proceedings of the AAAI Conference on Artificial Intelligence*, 33, 3558–65.

Firth, John R. (1957), "A Synopsis of Linguistic Theory 1930–1955," in *Studies in Linguistic Analysis*. Oxford, 1–32.

Fong, Hortense, Vineet Kumar, and K. Sudhir (2024), "A Theory-Based Explainable Deep Learning Architecture for Music Emotion," SSRN, https://doi.org/10.2139/ssrn.4025386.

Gabel, Sebastian and Artem Timoshenko (2022), "Product Choice with Large Assortments: A Scalable Deep-Learning Model," *Management Science*, 68 (3), 1808–27.

Hajij, Mustafa, Ghada Zamzmi, Theodore Papamarkou, Nina Miolane, Aldo Guzmán-Sáenz, Karthikeyan Natesan Ramamurthy, Tolga Birdal, Tamal K. Dey,, Soham Mukherjee, et al. (2022), "Topological Deep Learning: Going Beyond Graph Data," arXiv, https://doi.org/10.48550/arXiv.2206.00606.

Hamilton, William L. (2020), *Graph Representation Learning*. Morgan & Claypool.

Hamilton, Rebecca W., Stefano Puntoni, and Nader T. Tavassoli (2010), "Categorization by Groups and Individuals," *Organizational Behavior and Human Decision Processes*, 112 (1), 70–81.

He, Xiangnan, Lizi Liao, Hanwang Zhang, Liqiang Nie, Xia Hu, and Tat-Seng Chua (2017), "Neural Collaborative Filtering," in *Proceedings of the 26th International Conference on World Wide Web*. International World Wide Web Conferences Steering Committee, 173–82.

Hoffman, Matthew D., David M. Blei, Chong Wang, and John Paisley (2013), "Stochastic Variational Inference," *Journal of Machine Learning Research*, 14, 1303–47.

Hong, Jiyeon and Paul R. Hoban (2022), "Writing More Compelling Creative Appeals: A Deep Learning-Based Approach," *Marketing Science*, 41 (5), 941–65.

IBISWorld (2025), "Album Sales," (accessed August 1, 2025), https://www.ibisworld.com/united-states/bed/album-sales/5187/.

Ijams Spaid, Brian (2018), "Exploring Consumer Collecting Behavior: A Conceptual Model and Research Agenda," *Journal of Consumer Marketing*, 35 (6), 653–62.

Jain, Anish and Eric T. Bradlow (2021), "A Hierarchical Bayesian Approach for Tagged Playlist Generation," working paper, Wharton Business School.

Jakobsen, Laurie (2016), "Playlists Overtake Albums in Listenership, Says Loop Study," Music Business Association (September 22), https://musicbiz.org/news/playlists-overtake-albums-listenership-says-loop-study.

Jedidi, Kamel and Wayne S. DeSarbo (1991), "A Stochastic Multidimensional Scaling Procedure for the Spatial Representation of Three-Mode, Three-Way Binary Data," *Psychometrika*, 56, 471–94.

Kingma, Diederik P. and Jimmy Ba (2014), "Adam: A Method for Stochastic Optimization," arXiv, https://doi.org/10.48550/arXiv.1412.6980.

Kingma, Diederik P. and Max Welling (2013), "Auto-Encoding Variational Bayes," arXiv, https://doi.org/10.48550/arXiv.1312.6114.

Kipf, Thomas N. and Max Welling (2016a), "Semi-Supervised Classification with Graph Convolutional Networks," arXiv, https://doi.org/10.48550/arXiv.1609.02907.

Kipf, Thomas N. and Max Welling (2016b), "Variational Graph Auto-Encoders," arXiv, https://doi.org/10.48550/arXiv.1611.07308.

La Gatta, Valerio, Vincenzo Moscato, Mirko Pennone, Marco Postiglione, and Giancarlo Sperlí (2022), "Music Recommendation via Hypergraph Embedding," *IEEE Transactions on Neural Networks and Learning Systems*, 34 (10), 7887–99.

Liu, Liu, Daria Dzyabura, and Natalie Mizik (2020), "Visual Listening In: Extracting Brand Image Portrayed on Social Media," *Marketing Science*, 39 (4), 669–86.

Liu, Xiao, Dokyun Lee, and Kannan Srinivasan (2019), "Large-Scale Cross-Category Analysis of Consumer Review Content on Sales Conversion Leveraging Deep Learning," *Journal of Marketing Research*, 56 (6), 918–43.

Loken, Barbara, Lawrence W. Barsalou, and Christopher Joiner (2008), "Categorization Theory and Research in Consumer Psychology," *Handbook of Consumer Psychology*, Curtis P. Haugtvedt, Paul M. Herr, and Frank R. Kardes, eds. Taylor & Francis Group/Lawrence Erlbaum Associates, 133–63.

Manchanda, Puneet, Asim Ansari, and Sunil Gupta (1999), "The 'Shopping Basket': A Model for Multicategory Purchase Incidence Decisions," *Marketing Science*, 18 (2), 95–114.

Mikolov, Tomas, Kai Chen, Greg Corrado, and Jeffrey Dean (2013), "Efficient Estimation of Word Representations in Vector Space," arXiv, https://doi.org/10.48550/arXiv.1301.3781.

Nelder, John Ashworth and Robert W.M. Wedderburn (1972), "Generalized Linear Models," *Journal of the Royal Statistical Society Series A: Statistics in Society*, 135 (3), 370–84.

Papillon, Mathilde, Sophia Sanborn, Mustafa Hajij, and Nina Miolane (2023), "Architectures of Topological Deep Learning: A Survey of Message-Passing Topological Neural Networks," arXiv, https://doi.org/10.48550/arXiv.2304.10031.

Pichl, Martin, Eva Zangerle, and Günther Specht (2015), "Towards a Context-Aware Music Recommendation Approach: What Is Hidden in the Playlist Name?" in *2015 IEEE International Conference on Data Mining Workshop (ICDMW)*. IEEE, 1360–65.

Prebil, Miha (2023), "How Many Playlists Are on Spotify? Viberate's Deep Dive," Medium (October 26), https://medium.com/viberate-blog/how-many-playlists-are-on-spotify-viberates-deep-dive-b04499a3fcbc.

Puranam, Dinesh, Vrinda Kadiyali, and Vishal Narayan (2021), "The Impact of Increase in Minimum Wages on Consumer Perceptions of Service: A Transformer Model of Online Restaurant Reviews," *Marketing Science*, 40 (5), 985–1004.

Qiu, Jiezhong, Yuxiao Dong, Hao Ma, Jian Li, Kuansan Wang, and Jie Tang (2018), "Network Embedding as Matrix Factorization: Unifying Deepwalk, Line, PTE, and Node2vec," in *Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining*. Association for Computing Machinery, 459–67.

Ratneshwar, Srinivasan, Lawrence W. Barsalou, Cornelia Pechmann, and Melissa Moore (2001), "Goal-Derived Categories: The Role of Personal and Situational Goals in Category Representations," *Journal of Consumer Psychology*, 10 (3), 147–57.

Rendle, Steffen, Walid Krichene, Li Zhang, and John Anderson (2020), "Neural Collaborative Filtering vs. Matrix Factorization Revisited," in *Fourteenth ACM Conference on Recommender Systems*. Association for Computing Machinery, 240–48.

Rezende, Danilo Jimenez, Shakir Mohamed, and Daan Wierstra (2014), "Stochastic Backpropagation and Approximate Inference in Deep Generative Models," in *Proceedings of the 31st International Conference on Machine Learning*. *PMLR*, 32 (2), 1278–86.

Sarker, Md Rafidul Islam and Abdul Matin (2021), "A Hybrid Collaborative Recommendation System Based on Matrix Factorization and Deep Neural Network," in *2021 International Conference on Information and Communication Technology for Sustainable Development (ICICT4SD)*. IEEE, 371–74.

Swets, John A. (2014), *Signal Detection Theory and ROC Analysis in Psychology and Diagnostics: Collected Papers*. Psychology Press.

Van der Maaten, Laurens and Geoffrey Hinton (2008), "Visualizing Data Using t-SNE," *Journal of Machine Learning Research*, 9 (86), 2579–605.

Vaswani, Ashish, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin (2017), "Attention Is All You Need," *Advances in Neural Information Processing Systems*, 30.

Veličković, Petar, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Lio, and Yoshua Bengio (2017), "Graph Attention Networks," arXiv, https://doi.org/10.48550/arXiv.1710.10903.

Yang, Yi, Kunpeng Zhang, and P.K. Kannan (2022), "Identifying Market Structure: A Deep Network Representation Learning of Social Engagement," *Journal of Marketing*, 86 (4), 37–56.

Yang, Jeremy, Juanjuan Zhang, and Yuhan Zhang (2024), "Engagement That Sells: Influencer Video Advertising on TikTok," *Marketing Science*, 44 (2), 247–67.

Yu, Junliang, Hongzhi Yin, Jundong Li, Qinyong Wang, Nguyen Quoc Viet Hung, and Xiangliang Zhang (2021), "Self-Supervised Multi-Channel Hypergraph Convolutional Network for Social Recommendation," in *Proceedings of the Web Conference 2021*. Association for Computing Machinery, 413–24.

Zhou, Mi, George H. Chen, Pedro Ferreira, and Michael D. Smith (2021), "Consumer Behavior in the Online Classroom: Using Video Analytics and Machine Learning to Understand the Consumption of Video Courseware," *Journal of Marketing Research*, 58 (6), 1079–100.