Predicting Content Consumption from Content-to-Content Relationships

Jinyoung Han[a], Daejin Choi[b], Taejoong Chung[c], Chen-Nee Chuah[d], Hyun-chul Kim[e,*], Ted "Taekyoung" Kwon[f,*]

[a]*Hanyang University, Korea*
[b]*Georgia Institute of Technology, USA*
[c]*Rochester Institute of Technology, USA*
[d]*University of California, Davis, USA*
[e]*Sangmyung University, Korea*
[f]*Seoul National University, Korea*

Abstract

As the majority of Internet traffic today is attributed to content-centric applications, there has been ever-increasing demand for highly scalable and efficient content delivery. An accurate prediction on future content consumption is essential for such demand. To address such an issue, this paper introduces a new computational approach, *Content Network (CN)* that can capture the relations among contents, and its potential applications. We conduct a measurement study to investigate how contents are inter-related from the viewpoint of content spreading on one of the popular BitTorrent portals: The Pirate Bay. Based on the large-scale dataset that contains 18 K torrents and 9 M users, we construct the CN and investigate its structural properties. Our key finding is that contents in the same community in the CN (i) belong to the same content category with 94% probability, (ii) are uploaded by the same content publisher with 76% probability, and (iii) have the similar titles with 51% probability, which implies that contents in the same community collectively contain common (shared) interests of users. Our trace-driven study demonstrates that the proposed CN model is useful

[*] Corresponding Authors, Tel: +82-2-880-9105, Fax: +82-2-872-2045

Email addresses: jinyounghan@hanyang.ac.kr (Jinyoung Han), soyenze@gmail.com (Daejin Choi), tjc@cs.rit.edu (Taejoong Chung), chuah@ucdavis.edu (Chen-Nee Chuah), hkim@smu.ac.kr (Hyun-chul Kim), tkkwon@snu.ac.kr (Ted "Taekyoung" Kwon)

in (i) content recommendation for increasing sales and (ii) content caching for networking efficiency. We believe our work can provide an important insight for content stakeholders, e.g., content providers for efficient publishing strategies, network engineers for networking efficiency, or content marketers for accurate recommendation.

Keywords:
Content Sharing, Peer-to-Peer, BitTorrent

1. Introduction

The majority of Internet traffic today is attributed to content-centric applications such as BitTorrent, YouTube, or Netflix. According to the recent Cisco's report in 2016, the Internet video content traffic will be 82 percent of all consumer traffic by 2020 [1], which confirms the prevalence of content-centric Internet usage. This in turn has led to the ever-increasing demands for highly scalable and efficient content distribution and delivery [2, 3, 4, 5, 6]. An accurate prediction on future content consumption is one of the important building blocks for such demand.

To address this issue, we introduce a new computational approach, *Content Network (CN)* that can capture the relations among contents, and its potential applications. To this end, we build a bipartite network consisting of two different types of nodes: (i) contents and (ii) users downloading those contents (See Figure 1). Projecting [7, 8] the contents-users bipartite network into the contents space, we obtain the notion of a *Content Network (CN)*, whose nodes are contents, which are linked if the contents are downloaded by common users (Figure 1). Those linked contents in the proposed CN implies common interests of the users. While the literatures of the Web graph (whose vertices are webpages) [9, 10] assume two webpages to be related if there exists a (physical) hyperlink between them, the CN (whose vertices are various types of contents such as movie or music) characterizes a relation between two contents from a viewpoint of users' requests or common interests.

As an initial attempt to evaluate our methodology, we conduct a measurement study using the BitTorrent dataset consisting of 18,776 torrents[1] and 9,043,054 users. As one of the widely-used applications for sharing various types (e.g., videos, audios, e-book, software, etc.) of contents on the

[1] In this paper, we regard a torrent as a content.

Internet, BitTorrent generates 5-30% of all the Internet traffic, according to the Sandvine's recent report [11]. Using the collected dataset, we construct a BitTorrent-based CN, and analyze its community structure. Note that a community in the CN is the set of contents that are shared by a set of users who share similar interests. Based on the lessons learned, we explore how to predict future content consumption using the CN, which provides an important implication on content applications such as content recommendation [12] or content caching [13], to increase sales or improve system performance.

We highlight the main contributions of this paper as follows:

1. **Measurement:** This measurement study investigates the relations of various types (e.g., movie, music, or e-book) of contents shared by a large number of users (9 M) on the Internet. We introduce and analyze a content network (CN), which captures the relations among contents, based on the collected large-scale data.
2. **Key Findings:** We find that contents in the same community in the CN (i) belong to the same content category with 94% probability, (ii) are uploaded by the same content publisher [14, 15] with 76% probability, and (iii) have the similar titles with 51% probability. This implies that contents in the same community collectively contain common (shared) interests of users.
3. **Implications - Predictions on Content Consumption:** We explore the implication of our findings for predicting future content consumption with two popular real world applications: content recommendation and content caching. Our trace-driven study demonstrates that the proposed CN model is useful in content recommendation and content caching. We show the CN model can achieve more than 5-times higher accuracy in content recommendation, compared to other widely-used content recommendation algorithms. We also show that the CN model is useful for improving caching performance, which leads to an efficient content delivery. We believe our work can provide an important insight for many content stakeholders, e.g., content providers for their publishing/bundling strategies [14, 6], network operators for their content caching strategies, and marketers for efficient advertisement or recommendation [12].

We organize this paper as follows. We first present the definition of the CN in Section 2 and the dataset in Section 3. In Section 4, we investigate the structural properties of the CN, which reveals how contents are connected

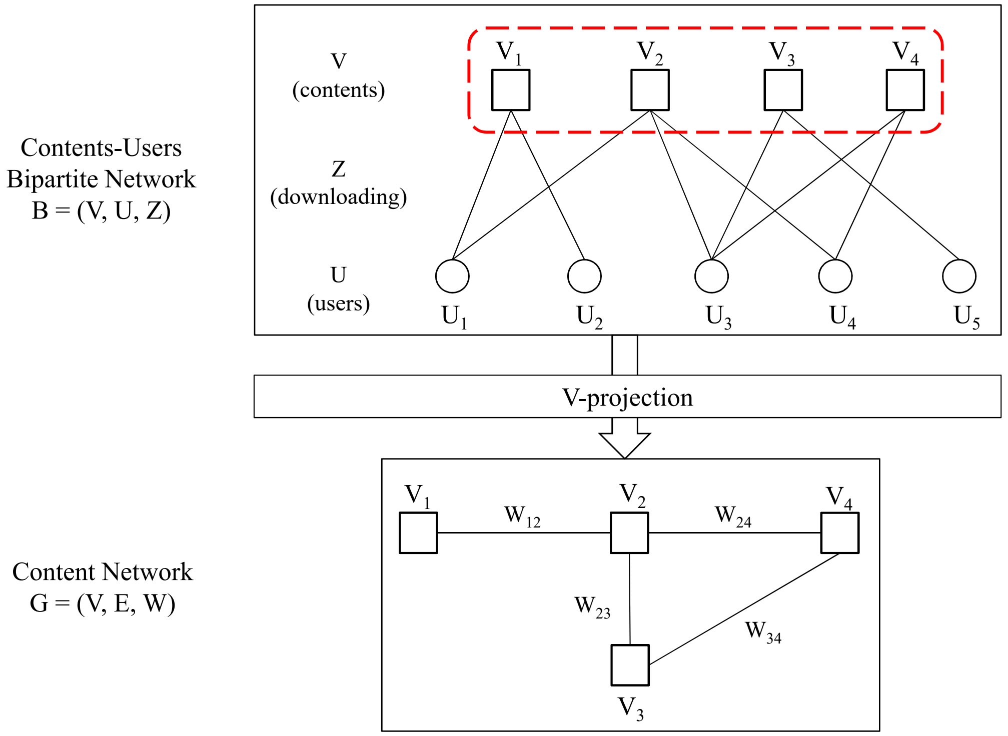


Figure 1: An illustrative example of a bipartite network $B = (V, U, Z)$, as well as its V projection. V represents the set of content and U is the set of users who downloaded the content. An undirected weighted graph $G = (V, E, W)$ represents a CN where V is the set of contents, E is the set of edges between two contents, and W is the set of weights of the corresponding edges.

and grouped. We introduce two use cases for the CN in Section 5: (i) content recommendation and (ii) content caching. After reviewing related work in Section 6, we conclude the paper in Section 7.

2. Content Network Model

In this section, we introduce the notion of a content network (CN) that represents relations among contents. To this end, we first consider a contents-users bipartite network $B = (V, U, Z)$ whose nodes are divided into two disjoint sets V and U, such that every edge in Z connects a node in V to one in U, i.e., V and U are independent sets [16, 7]. V represents the set of contents and U is the set of users who have downloaded contents.

To show the relations among a particular set of nodes (e.g., V or U), a bipartite network can be compressed by the one-mode projection [7, 8]. That is, the one-mode projection onto V (V projection for short) results in a graph that consists of nodes only in V where nodes in V are connected if they have at least one common nodes in U. Similarly, U projection results in a graph that consists of nodes only in U where nodes in U are connected if they have common nodes in V. There have been studies that focus on *relations among people* such as a movie actor network where two actors are connected by movie(s) they played together [17], scientific collaboration networks where two authors are linked based on their co-authorships [18], and peer-to-peer networks where two users share same content(s) [19], by U projection. Instead, we focus on *relations among contents* by V projection, which results in a CN.

Figure 1 illustrates an example of a contents-users bipartite network and its V projection. We assume that an undirected weighted graph $G = (V, E, W)$ represents a CN where V is the set of contents and E is the set of (undirected) edges between two contents. That is, an edge $E_{i,j}$ exists between two contents V_i and V_j in a CN if there is at least one user who has downloaded both of V_i and V_j. In transforming (i.e., projection) from a bipartite network into an one-mode projection, information can be lost. The weighted projection is a way to address this problem [7, 8]. Therefore, we define the weight $W_{i,j}$ of a given edge $E_{i,j}$ as the Jaccard similarity between two contents V_i and V_j as follows:

$$W_{i,j} = \frac{V_i \cdot V_j}{\|V_i\| \|V_j\|} \tag{1}$$

where each vertex is associated with a set of flags that indicate whether each user has downloaded the corresponding content or not, i.e., $V_i = \{V_i^1$, V_i^2, ..., $V_i^m\}$, where $V_i^u = 1$ if user u has downloaded the content i and $V_i^u = 0$ otherwise. For example, in Figure 1, the edge $E_{2,4}$ in the CN has the highest weight, which implies a strong relation between two contents V_2 and V_4 in terms of common user base.

Given n contents and m users, calculating the Jaccard similarity values for the ${}_nC_2$ relations (i.e., all pairs of n nodes) takes $O(n^2)$. Also, calculating a Jaccard similarity between two nodes, each of which is a vector that consists of m users, takes $O(m)$. Therefore, the computaional compelxity in constructing the proposed CN model is $O(n^2m)$.

3. Dataset

As a case study, this paper chooses to conduct a data-driven measurement study based on BitTorrent, one of the most popular applications for sharing various types (e.g., movie, music, e-book) of contents over the Internet. In this section, we describe our dataset to empirically investigate the relations among contents in the CN.

We conducted a measurement study [14] on one of the most popular BitTorrent portal, The Pirate Bay (TPB) [20]. For the purpose of data collection, we developed a BitTorrent monitoring software to keep track of each swarm by modifying Azureus [21], a widely used BitTorrent open source software. To monitor each swarm from its beginning, we leveraged the rich site summary (RSS) notification of a new torrent to immediately retrieve its publisher [15, 14] information (e.g., its username) and '.torrent' file. Note that we kept track of swarms of all the torrents published on TPB during our measurement period. By analyzing the '.torrent' file, we could connect multiple trackers to retrieve the lists of peers. We further leveraged the Peer EXchange (PEX) protocol to discover more peers not found via the trackers; the PEX protocol allows us to discover new peers via the known peers without contacting trackers [22]. After finding the peers in each swarm, our monitoring software (operated by 14 machines) began to monitor each swarm periodically (once every two to four hours). Finally, we obtained the information of each torrent such as content category (e.g., movie, music, etc.) given by TPB and publisher's username, as well as the information of peers who download the torrent from the corresponding swarms.

Our dataset had been collected for 16 days from April 5 to April 20, 2011, which contains the information on 9,043,054 peers (users) and 18,176 torrents published by 4,050 BitTorrent publishers. Note that only anonymized user information is used for this research, and no personally identifiable information is used. Throughout this paper, we investigate the CN for the seven major (98% in terms of the number of torrents) content categories given by TPB: Movie, TV, Porn, Music, Application, Game, and E-book. Figure 2 shows the Percentages of the number of published torrents and number of users who download the corresponding torrents in each content category, respectively. As shown in Figure 2, the Movie is the most popular category in terms of both number of published torrents and number of users. The Porn category shows an interesting pattern; while it is the third popular category out of seven categories in terms of number of published torrents, it is the second popular one in terms of number of users.

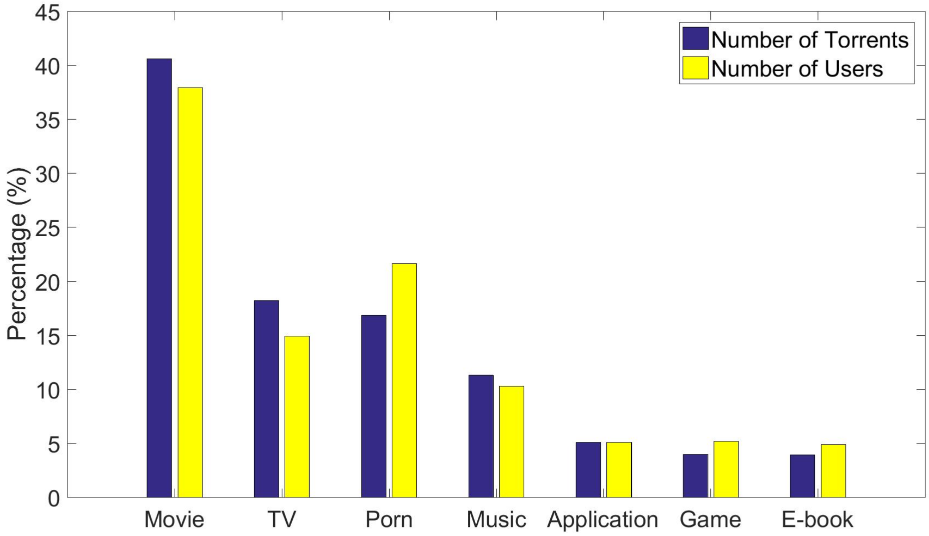
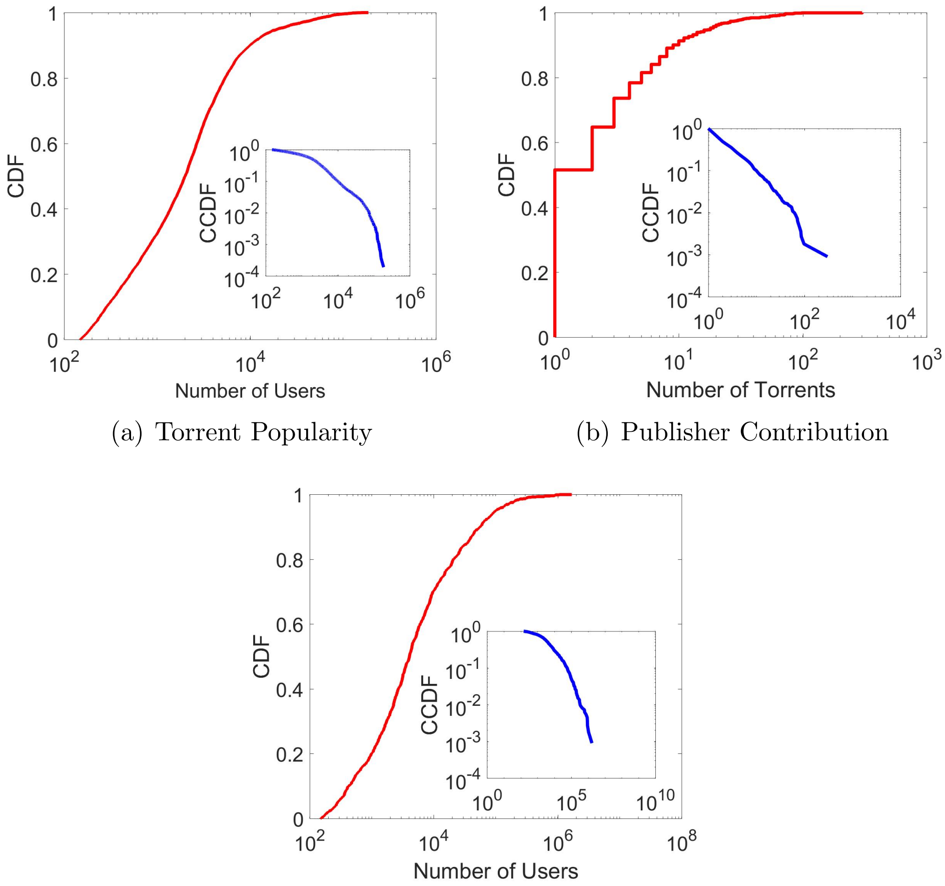


Figure 2: Percentages of the number of published torrents and number of users who download the corresponding torrents in each content category, respectively.

Figure 3(a) illustrates the distributions of torrent popularity, which is calculated as the number of users who download the corresponding torrent. As shown in Figure 3(a), while most torrents (90%) are downloaded by less than 1 K users, a small portion of torrents (top 0.1%) are substantially popular, i.e., downloaded by more than 20 K users. Figure 3(b) next shows the distributions of the number of published torrents by each publisher, and reveals that a small portion of publishers publish a large number of torrents; while 90% of publishers publish less than 10 torrents, top 0.1% publishers publish more than 500 torrents. Note that the number of published torrents by each publisher follows a power law distribution. When we look at the publisher popularity in terms of the number of users who downloaded the torrents published by each publisher in Figure 3(c), we find that a small portion of publishers are significantly popular; the top 0.1% publishers (in terms of publisher popularity) have more than 100 K audiences (or downloaders).



(a) Torrent Popularity

(b) Publisher Contribution

(c) Publisher Popularity

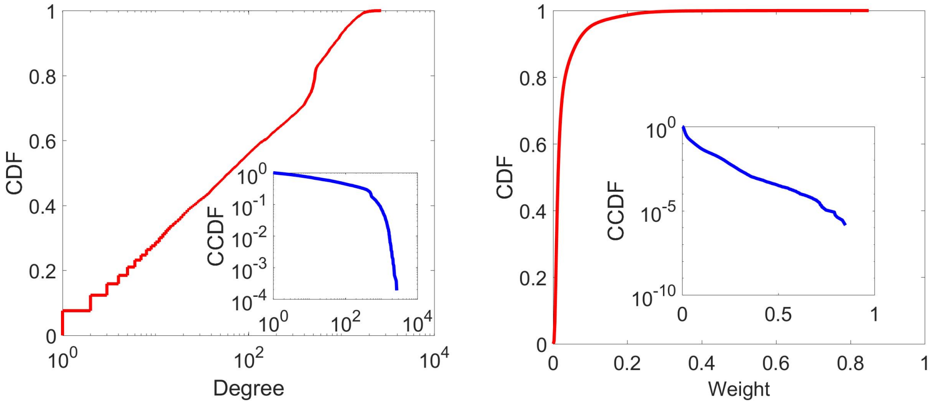
Figure 3: Distributions of (a) torrent popularity in terms of number of users who downloaded the corresponding torrent, (b) publisher contribution in terms of number of torrents published by each publisher, and (c) publish popularity in terms of number of users who downloaded the torrents published by each publisher.

4. Case Study: BitTorrent Content Network

In this section, we investigate a BitTorrent-based content network (CN) where a vertex is a content and an edge is a relation between two contents.

4.1. Network Structure Analysis

We first analyze the structural properties of the CN. The numbers of nodes and edges in the CN are 5,290 and 729,520, respectively, which signi-



(a) Degree

(b) Weight

Figure 4: Degree and weight distributions of the CN.

fies that the CN is substantially dense. Figure 4 illustrates the degree and weight distributions of the CN. As shown in Figure 4(a), the degree distribution shows a quite proportional pattern. About 50% of nodes show lower than 100 degree while top 10% nodes have more than 1 K degree in the CN. When we look at the weight distribution in Figure 4(b), it shows a substantially skewed pattern; about 90% of weights are less than 0.1 while top 0.1% weights are over 0.4. This indicates that a small portion of content-to-content relationships are strong. Note that the average degree and weights of the CN are 275.8 and 0.027, respectively. We also calculate the clustering coefficient and the average path length [23] of the CN. We find that the clustering coefficient of the CN is substantially high (0.7) while the average path length is small (2.54), which implies a *'small-world'* property [23] of the CN.

4.2. Community Analysis

We now investigate how contents form groups (or communities) in the CN. We first examine whether and how contents form communities by calculating the modularity [24, 25]. Here, a community is a group of contents, within which edges are denser, but between which edges are sparser. We use a well-known definition of the modularity [25] as follows:

$$Q = \frac{1}{2m} \sum_{i,j} \left[1 - \frac{k_i k_j}{2m} \right] \delta(c_i, c_j) \qquad (2)$$

where k_i is the degree of node i, m is the summation of node degrees for all nodes, c_i is the community where node i belongs, and δ is the Kronecker's delta ($\delta(i,j)=1$ if $i=j$, and $\delta(i,j)=0$ otherwise). The modularity value above 0.3 is known to be a strong presence of community structures [24]. We find that the modularity of the CN is 0.54, which means contents tend to substantially form communities.

We identify communities of the CN using the Louvain method [25], a well-known fast community detection algorithm that maximizes the ratio of the number of edges within communities to that of edges between communities. We use the weighted version of Louvain method. Note that the computational complexity of the Louvain method is $O(nlogn)$. The number of identified communities in the CN is 140, and their average number of members is 37.8.

Based on the identified communities, we examine what makes contents belong to the same community. To this end, we devise a metric, similarity index SI, which quantifies how much similarity exists among members in the same community [26]. That is, SI indicates the probability that randomly selected two contents *within the same community* have the same property such as content category or publisher. Let $I(v)$ denotes the indicator of property of content v. Suppose we have c communities in the given network, and community k consists of nodes $V^k=\{v_1^k,\cdots,v_{n_k}^k\}$. Then, the similarity index SI is

$$SI=\frac{2}{\sum_{k=1}^{c}n_k\left(n_k-1\right)}\sum_{k=1}^{c}\sum_{i=1}^{n_k-1}\sum_{j=i+1}^{n_k}\delta\left(I\left(v_i^k\right),I\left(v_j^k\right)\right) \tag{3}$$

where n_k is the number of nodes in community k and δ is the Kronecker's delta.

We conjecture that contents in the same community may contain common interests of users as they are downloaded by common users. To validate this conjecture, we consider three content properties that may be linked to shared interests of users: (i) category (e.g., movie, music, or e-book), (ii) publisher, and (iii) title. That is, we examine whether the contents in the same community belong to the same category or publisher, or have similar titles. For the first two properties (i.e., category and publisher), we calculate the SIs by Equation 3. To calculate the SI for the content title, we use the Levenshtein distance[2] [27] to quantify the similarity between titles of two

[2] Levenshtein distance [27] between two titles is defined as the minimum number of

contents instead of using δ in Equation 3.

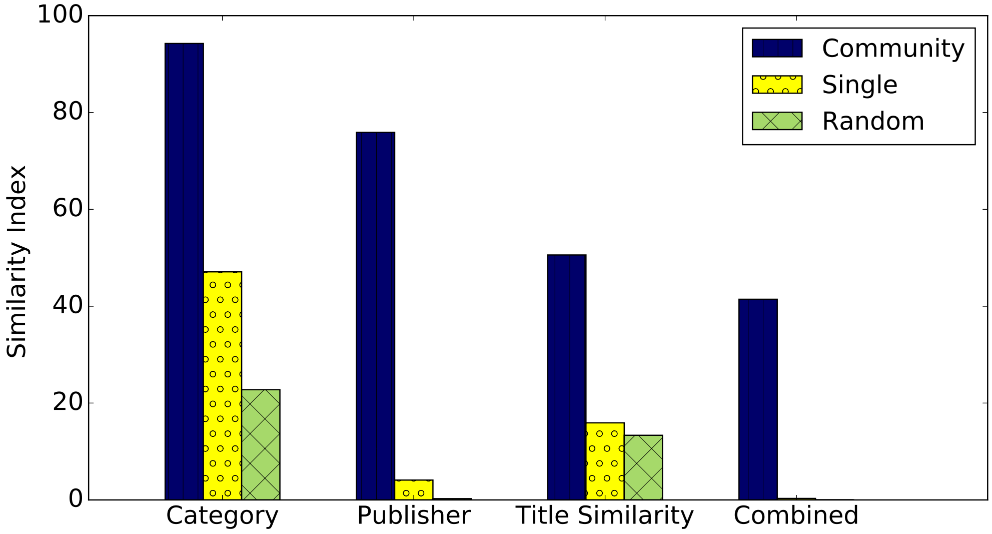


Figure 5: The similarity indices of categories, publishers, and titles of contents in the same community, respectively.

Figure 5 shows the SIs of the content category, publisher, and title, respectively (denoted by COMMUNITY). For the comparison purpose, we randomly select an edge in the CN, and calculate its SI (denoted by SINGLE); we repeat this 1,000 times. We also randomly select two nodes in the CN (1,000 times), i.e., two nodes may not be connected in the CN, and calculate its SI (denoted by RANDOM). As shown in Figure 5, 94.3% of the pairs of two contents in the same community belong to the same content category, which is significantly higher than the SINGLE (47.1%) and RANDOM (22.8%) cases. This implies that users' interests are likely to be bounded in a same content category; e.g., users who are interested in Porn contents may download another Porn content with high probability. Also, the SI of content publisher by COMMUNITY is significantly higher than those by SINGLE and RANDOM. While the SIs of content publisher by SINGLE and RANDOM are 4.1% and 0.3%, respectively, 75.9% of the pairs of two contents in the same community belong to the same publisher. This signifies that users

edits needed to transform one title into the other, with the allowable edit operations being insertion, deletion, or substitution of a single character.

tend to download multiple contents published by a same publisher. The SI of content title by COMMUNITY (50.6%) is also substantially higher than those by SINGLE (15.9%) and RANDOM (13.3%), meaning that users tend to download contents whose titles are similar.

To collectively quantify common interests of users, we calculate a combined metric by multiplying $\delta_{category}$ ($\delta_{category} = 1$ if two contents belong to a same category, and $\delta_{category} = 0$ otherwise), $\delta_{publisher}$ ($\delta_{publisher} = 1$ if two contents belong to a same publisher, and $\delta_{publisher} = 0$ otherwise), and *Levenshtein distance*. As shown in Figure 5, the SI of (combined) common interests by COMMUNITY (41.4%) is significantly higher than those by SINGLE (0.3%) and RANDOM (0.008%). This indicates that contents in the same community (i) belong to the same content category, (ii) are uploaded by the same content publisher, and (iii) have the similar titles, which implies contents in the same community collectively contain common (shared) interests of users.

5. Prediction on Content Consumption

Let us illustrate two use cases for the CN towards predicting future content consumption: (1) content recommendation for increasing sales and (2) content caching for networking efficiency. An accurate prediction on future content consumption can provide important insight for many content business stakeholders, e.g., content providers for their bundling [14, 6] or publishing strategies [15], network operators for their content caching or prefetching strategies [13, 28], and marketers for efficient advertisement or recommendation [12, 29]. To this end, we perform a trace-driven simulation study.

5.1. Trace-driven Simulation

To conduct a trace-driven simulation for each case, we collected additional dataset for 7 days from April 21 to 27, 2011, which consists of 4,614 torrents with 13,901,982 content requests from 1,870,350 users. We denote this additionally collected dataset as $dataset - test$. Based on the original dataset (denoted by $dataset - learning$, described in Section 3) collected from April 5 to April 20, 2011, we construct the CN. We then generate the content request patterns based on $dataset - test$.

5.2. Content Recommendation

To investigate which recommendation methods (described below) are efficient, we conduct a trace-driven simulation. We first select 38,285 target users who appear both in $dataset - test$ and $dataset - learning$. In our simulation, if a target user (in $dataset - test$) downloads a content, we recommend 10 contents based on the learning information on a basis of the $dataset - learning$. To validate whether the recommended contents are actually consumed by each user during the period of $dataset - test$ (i.e., from April 21 to April 27), we check all the downloaded contents by each target user. For evaluation, we measure precision, which is defined as the ratio of the number of contents that the user actually has downloaded to the total number of predicted contents.

As shown in Section 4, our analysis revealed that contents in the same community of the CN collectively contain common (shared) interests of users. By leveraging the lessons learned from our analysis, we suggest a recommendation method using the community information of the CN (denoted by $community - based$). That is, if a user downloads a content, we recommend contents in its community of the CN in $community - based$. If there are more than 10 contents in the community, we select popular ones, i.e., the most downloaded content, among them in our simulation. The basic idea of $community - based$ is to find similar contents based on the (collective) opinions of other like-minded users. Algorithm 1 describes the $community-based$ method.

Algorithm 1 Community–based method

```
procedure COMMUNITY–BASED(Target user $A$, Number of predicted contents $n$)
    $S_A \leftarrow$ contents downloaded by $A$
    $R \leftarrow$ empty table
    for each content $t$ in $S_A$ do
        $C \leftarrow$ list of contents whose community is same as t's one.
        for each content $t'$ in $C$ do
            if $t'$ is not in $R$ and $S_A$ then
                Add $t'$ into R
            end if
        end for
    end for
    $R' \leftarrow$ sorts $R$ in terms of popularity
    return first $n$ elements in $R'$
end procedure
```

For the comparison purpose, we first adopt the *(item-to-item) collaborative filtering (CF)* [30, 12, 29] technique, which is a well-known recommendation algorithm, denoted by $cf - based$. In the $cf - based$, the similarity between two contents is calculated by the cosine similarity [30]. A key difference between $community{-}based$ and $cf{-}based$ is as follows: while $cf{-}based$ considers the similarity between two contents, $community - based$ considers multiple similar contents in the same community that collectively contain shared interests of users. We further examine four baseline methods to find similar contents (like Section 4): (i) $category - based$ [31, 12, 26] that finds contents of the same category, (ii) $publisher - based$ [12, 31] that finds contents of the same publisher, (iii) $title - based$ [6, 32] that finds contents with the most similar titles using the Levenshtein distance [27], and (iv) *random* that finds contents randomly. If we cannot find similar contents (e.g., belonging to the same category or publisher) for a given content, we select popular contents in our simulation.

Figure 6 shows the average precision of each method. The $community - based$ outperforms others (more than 5 times), which indicates that predicting based on the community information of the CN is much more accurate than others. This implies that community information of the CN is a strong predictor for predicting future content consumption. The $publisher - based$ performs better than others except the $community - based$, meaning that people tend to consume contents from the same publisher. In summary, the CN can be used in predicting content consumption patterns; this can be an important implication on designing recommendation systems or personalized services in content delivery services. For example, considering the community information of the CN can help to improve the accuracy of the content recommendation system, which can result in increasing sale.

5.3. Content Caching

Content caching is an important strategy for networking efficiency [13], where how to select a victim item to be replaced from a cache is critical. The LRU (Least Recently Used), LFU (Least Frequently Used), and LRFU (Least Recently/Frequently Used) are the most popular replacement algorithms which reflect the recency, frequency, and both of them, respectively [33]. We suggest to add another factor for the replacement process: *'community membership'* in the CN. That is, we give a penalty κ to a cached item in the cache storage if it does not belong to the community of the incoming item. With this factor, we propose a new replacement algorithm which

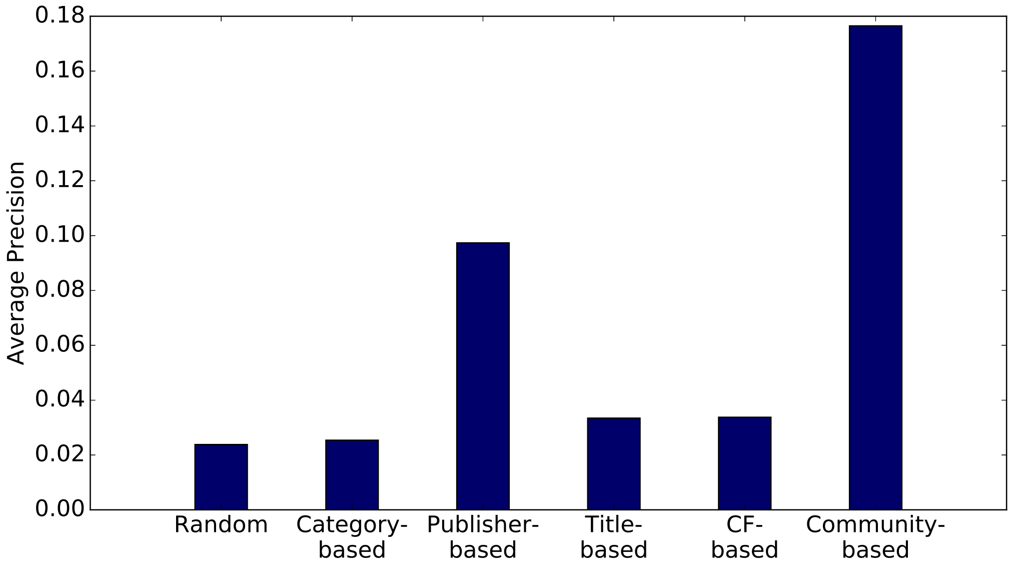


Figure 6: Average precision of each method for content recommendation.

collectively considers both frequency and community membership, dubbed by CLFU (i.e., Community-aware LFU). In the CLFU, the reference (or frequency) count of each cached item is penalized by multiplying κ if it is not a member of the community of the incoming item.

We conduct a trace-driven simulation based on the $dataset-learning$ and $dataset-test$ to evaluate the $CLFU$ by comparing other well-known algorithms (LRU, LFU, and $LRFU$ (with aging factor [33] 0.9)). We assume a simple scenario that there is only one global cache server that can store 5% of all the items in our datasets. The community membership is calculated based on the CN constructed by the $dataset-learning$. Note that we set κ = 0.999 for the $CLFU$. The content request patterns are generated base on the $dataset-test$. Since we collected swarm dynamics once in two to four hours as we described in Section 3, content requests in a range of every four hours are randomly generated.

Figure 7 shows that the $CLFU$ outperforms the other algorithms, which means that the *'community membership'* is an important factor for the efficient cache replacement. Note that the hit ratio of the $CLFU$ is around 4 times higher than that of the LFU; the hit ratio of the $CLFU$ is even 1.2 times higher than that of the $LRFU$. This suggests that the CN can be used for improving caching performance, which leads to an efficient content

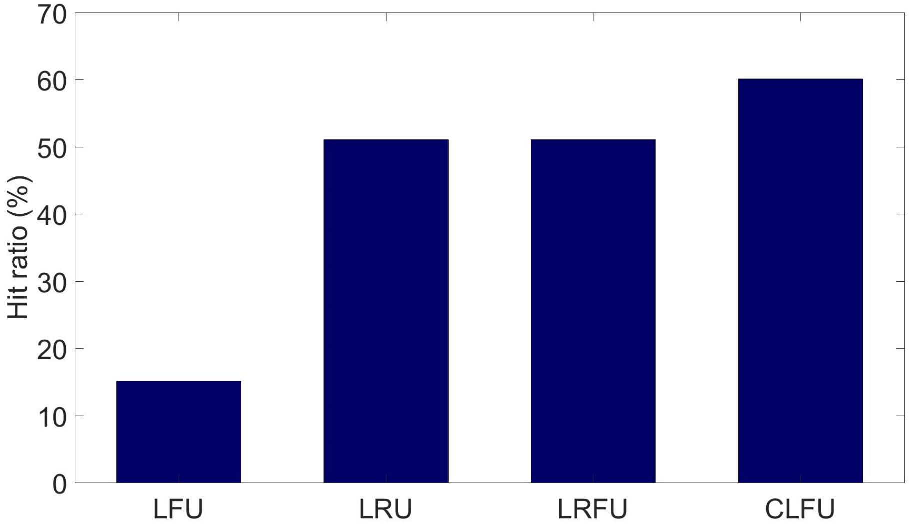


Figure 7: Cache hit ratio of each algorithm is plotted. The *CLFU* outperforms others, which signifies that the CN can be effectively used to improve caching performance.

delivery.

6. Related Work

The structural properties (e.g., degree distribution and average path length) of various social networks such as movie actor networks [17], scientific collaboration networks [18], and human sexual networks [34] have received great attention. As online social networks have become popular, many researchers seek to examine user behaviors in and structural patterns of various online social networks such as Twitter [35, 36], Facebook [37, 38], Flicker [39, 40], Second Life [41], and Massively Multi-player Online Role-Playing Games (MMORPGs) [42, 43]. While these studies have mostly focused on relations of people [35, 36, 37, 38, 39, 40, 44] in online social networks (OSNs), avatars [41] in a virtual world, or players in an MMORPG [42, 43] (i.e., a vertex in its network is a person, avatar, or player, respectively), we focus on relations of contents (i.e., a vertex in its network is a content), which are not human beings but information or objects.

There have been studies to investigate relations among diverse information (or objects) such as web pages, words, flavor, movies, or topics [9, 45, 46, 47, 48, 49, 50, 6], instead of relations among people. Albert *et al.* showed that the World Wide Web (WWW), whose vertices are web pages, has a small-world property [9]. Kleinberg and Lawrence [45] also studied the structure of the Web. A well-known PageRank algorithm [10] that considers links among web pages is used by the Google web search engine. Han *et al.* [6] proposed the bundling strategies based on the relations among BitTorrent content files. Chatzopoulou *et al.* studied the related video network in YouTube whose vertices are videos and edges are constructed by the related videos provided by Youtube [47]. Google introduced the knowledge graph [51] whose vertices are *knowledge (or keywords)* for enhancing the search engine. Han *et al.* introduced the topic network whose vertices are topics such as animals, travel, or education in Pinterest [48]. Ahn *et al.* introduced the flavor network that cab capture the flavor compounds shared by ingredients [50]. Cancho and Solé [46] examined the network structure of the lexicon whose vertices are words, and showed that human language has a small-world property. Our work introduces the BitTorrent-based content network that can capture the collective opinions of like-minded users in the first place, and demonstrates that the community information of such a network can be an important predictor for accurately predicting future content consumption, which can be used for important content applications such as content recommendation or content caching.

7. Concluding Remarks

We proposed a notion of a *Content Network (CN)* that represents the relations among contents. As an initial attempt to evaluate our computational approach, we conducted a measurement study on BitTorrent. Our key finding is that contents in the same community in the CN (i) belong to the same content category with 94% probability, (ii) are uploaded by the same content publisher with 76% probability, and (iii) have the similar titles with 51% probability, which implies that content in the same community collectively contain common (shared) interests of users. Based on the lessons learned from our analyses, we proposed methods for predicting content consumption patterns based on the community information of the CN. We demonstrated that the CN model can achieve higher accuracy in content recommendation than other well-known methods. We also showed that the CN model is useful for improving caching performance, which leads to an efficient content delivery. We believe our work can provide an important insight for many content stakeholders such as content providers, operators, recommenders, and marketers.

Limitaion and Future work. This study has several limitations. First, this study used a dataset of a particular content sharing application, BitTorrent. We will apply the proposed computational approach to other popular content delivery platforms such as Youtube, Netflix, or online social media. Second, the dataset used in this paper was collected a few years ago. We plan to validate our proposed approach with the recent content sharing activities. Lastly, in the trace-driven simulation study for evaluating the performance of caching algorithms, we assumed a simple scenario where there is only one global cache server that can store 5% contents of all the contents. Our future work will consider a complex scenario where multiple cache servers with different capacities co-exist. As a future work, we will also explore whether the proposed computational approach can be applicable in other applications such as healthcare (e.g., connecting or managing health-related data [52]) or business analytics (e.g., connecting consumer interest [48] or social influencers [44]). Our ongoing work further includes (i) capturing the time-varying dynamics and evolution of a content network and (ii) modeling and predicting the lifetime of a content using the CN model.

References

[1] Cisco, White paper: Cisco vni forecast and methodology, 2015-2020, `http://www.cisco.com/c/en/us/solutions/collateral/service-provider/visual-networking-index-vni/complete-white-paper-c11-481360.html`.

[2] G. Fortino, W. Russo, M. Vaccaro, An agent-based approach for the design and analysis of content delivery networks, Journal of Network and Computer Applications 37 (2014) 127–145.

[3] C. T. Calafate, G. Fortino, S. Fritsch, J. Monteiro, J.-C. Cano, P. Manzoni, An efficient and robust content delivery solution for ieee 802.11p vehicular environments, Journal of Network and Computer Applications 35 (2) (2012) 753–762.

[4] G. Fortino, C. Mastroianni, W. Russo, A hierarchical control protocol for group-oriented playbacks supported by content distribution networks, Journal of Network and Computer Applications 32 (1) (2009) 135–157.

[5] G. Fortino, W. Russo, Using p2p, grid and agent technologies for the development of content distribution networks, Future Generation Computer Systems 24 (3) (2008) 180–190.

[6] J. Han, T. Chung, S. Kim, H.-c. Kim, J. Kangasharju, T. T. Kwon, Y. Choi, Strategic bundling for content availability and fast distribution in bittorrent, Computer Communications 43 (2014) 64–73.

[7] T. Zhou, J. Ren, M. Medo, Y. C. Zhang, Bipartite network projection and personal recommendation, Physical Review E (Statistical, Nonlinear, and Soft Matter Physics) 76 (4) (2007) 046115+.

[8] M. Latapy, C. Magnien, N. D. Vecchio, Basic notions for the analysis of large two-mode networks, Social Networks 30 (1) (2008) 31–48.

[9] R. Albert, H. Jeong, A. L. Barabasi, The diameter of the world wide web, Nature 401 (1999) 130–131.

[10] S. Brin, L. Page, The anatomy of a large-scale hypertextual web search engine, in: WWW, 1998.

[11] Sandvine, Global internet phenomena report, `https://www.sandvine.com/trends/global-internet-phenomena`.

[12] G. Linden, B. Smith, J. York, Amazon.com recommendations: Item-to-item collaborative filtering, IEEE Internet Computing 7 (1) (2003) 76–80.

[13] S. Borst, V. Gupta, A. Walid, Distributed caching algorithms for content distribution networks, in: IEEE INFOCOM, 2010.

[14] J. Han, S. Kim, T. Chung, T. T. Kwon, H.-c. Kim, Y. Choi, Bundling practice in bittorrent: what, how, and why, in: ACM SIGMETRICS, 2012.

[15] S. Kim, J. Han, T. Chung, H.-c. Kim, T. T. Kwon, Y. Choi, Content publishing and downloading practice in bittorrent, in: IFIP Networking, 2012.

[16] R. Diestel, Graph Theory (Graduate Texts in Mathematics), Springer, 2005.

[17] D. J. Watts, S. H. Strogatz, Collective dynamics of ‘small-world’ networks., Nature 393 (6684) (1998) 409–10.

[18] M. E. J. Newman, Scientific collaboration networks. i. network construction and fundamental results, Physical Review E 64.

[19] M. Kryczka, R. Cuevas, C. Guerrero, A. Azcorra, Unrevealing the structure of live bittorrent swarms: Methodology and analysis, in: IEEE P2P, 2011.

[20] The pirate bay, `http://thepiratebay.org/`.

[21] Open sourced bittorrent client, vuze, `http://www.vuze.com`.

[22] Peer exchange, `http://en.wikipedia.org/wiki/Peer_exchange`.

[23] M. E. J. Newman, The structure and function of complex networks, SIAM REVIEW 45 (2003) 167–256.

[24] H. Kwak, Y. Choi, Y.-H. Eom, H. Jeong, S. Moon, Mining communities in networks: a solution for consistency and its evaluation, in: ACM IMC, 2009.

[25] V. D. Blondel, J.-L. Guillaume, R. Lambiotte, E. Lefebvre, Fast unfolding of communities in large networks, Journal of Statistical Mechanics: Theory and Experiment 2008 (10).

[26] J. Han, D. Choi, B.-G. Chun, T. T. Kwon, H.-c. Kim, Y. Choi, Collecting, organizing, and sharing pins in pinterest: Interest-driven or social-driven?, in: ACM SIGMETRICS, 2014.

[27] G. Navarro, A guided tour to approximate string matching, ACM Computing Surveys 33 (1) (2001) 31–88.

[28] J. Erman, A. Gerber, M. T. Hajiaghayi, D. Pei, O. Spatscheck, Network-aware forward caching, in: WWW, 2009.

[29] C. A. Gomez-Uribe, N. Hunt, The netflix recommender system: Algorithms, business value, and innovation, ACM Transactions on Management Information Systems 6 (4) (2015) 13:1–13:19.

[30] B. Sarwar, G. Karypis, J. Konstan, J. Riedl, Item-based collaborative filtering recommendation algorithms, in: WWW, 2001.

[31] J. B. Schafer, J. A. Konstan, J. Riedl, E-commerce recommendation applications, Data Mining and Knowledge Discovery 5 (1) (2001) 115–153.

[32] J. Han, T. Chung, H.-c. Kim, , T. T. Kwon, Y. Choi, Systematic support for content bundling in bittorrent swarming, in: IEEE Conference on Computer Communications Workshops, 2010.

[33] D. Lee, J. Choi, J. H. Kim, S. H. Noh, S. L. Min, Y. Cho, C. S. Kim, Lrfu: A spectrum of policies that subsumes the least recently used and least frequently used policies, IEEE Transactions on Computers 50 (12) (2001) 1352–1361.

[34] F. Liljeros, C. R. Edling, L. A. Amaral, E. H. Stanley, Y. Aberg, The web of human sexual contacts, Nature 411 (6840) (2001) 907–908.

[35] H. Kwak, C. Lee, H. Park, S. Moon, What is twitter, a social network or a news media?, in: WWW, 2010.

[36] B. Krishnamurthy, P. Gill, M. Arlitt, A few chirps about twitter, in: ACM SIGCOMM WOSN, 2008.

[37] A. Nazir, S. Raza, C.-N. Chuah, Unveiling facebook: a measurement study of social network based applications, in: ACM IMC, 2008.

[38] B. Viswanath, A. Mislove, M. Cha, K. P. Gummadi, On the evolution of user interaction in facebook, in: ACM SIGCOMM WOSN, 2009.

[39] A. Mislove, M. Marcon, K. P. Gummadi, P. Druschel, B. Bhattacharjee, Measurement and analysis of online social networks, in: ACM IMC, 2007.

[40] M. Cha, A. Mislove, K. P. Gummadi, A measurement-driven analysis of information propagation in the flickr social network, in: WWW, 2009.

[41] M. Varvello, S. Ferrari, E. Biersack, C. Diot, Exploring second life, IEEE/ACM Transactions on Networking 19 (1) (2011) 80–91.

[42] S. Son, A. Kang, H. Kim, T. Kwon, J. Park, H. Kim, Analysis of context dependence in social interaction networks of a massively multiplayer online role-playing game., PLoS One 7 (4) (2012) e33918.

[43] S. Chun, D. Choi, J. Han, H. K. Kim, T. Kwon, Unveiling a socio-economic system in a virtual world: A case study of an mmorpg, in: WWW, 2018.

[44] S. Kim, J. Han, S. Yoo, M. Gerla, How are social influencers connected in instagram?, in: Social Informatics, 2017.

[45] J. Kleinberg, S. Lawrence, The structure of the web, Science 294 (2001) 1849–1850.

[46] R. F. i Cancho, R. V. Solé, The small world of human language, Proceedings of The Royal Society of London. Series B, Biological Sciences 268 (2001) 2261–2266.

[47] G. Chatzopoulou, C. Sheng, M. Faloutsos, A first step towards understanding popularity in youtube, in: IEEE INFOCOM NetSciCom, 2010.

[48] J. Han, D. Choi, A.-Y. Choi, J. Choi, T. Chung, T. T. Kwon, J.-Y. Rha, C.-N. Chuah, Sharing topics in pinterest: Understanding content creation and diffusion behaviors, in: ACM Conference on Online Social Networks (COSN), 2015.

[49] M. Gori, A. Pucci, Itemrank: A random-walk based scoring algorithm for recommender engines, in: International Joint Conference on Artifical Intelligence, 2007.

[50] Y.-Y. Ahn, S. E. Ahnert, J. P. Bagrow, A.-L. Barabási, Flavor network and the principles of food pairing, Scientific Reports 1.

[51] Knowledge graph - google, https://www.google.com/intl/bn/insidesearch/features/search/knowledge.html.

[52] G. Fortino, R. Giannantonio, R. Gravina, P. Kuryloski, R. Jafari, Enabling effective programming and flexible management of efficient body sensor network applications, IEEE Transactions on Human-Machine Systems 43 (1) (2013) 115–133.