Exploiting Interest-Based Proximity for Content Recommendation in P2P Networks

Zoltán Novák
Zoltán Pap
Department of Telecommunications and Media Informatics
Budapest University of Technology and Economics
H-1117, Magyar tudósok körútja 2, Budapest, HUNGARY
Phone: (36)1-463-2225, Fax: (36) 1-463-3107
E-mail: novak@tmit.bme.hu, pap@tmit.bme.hu

Abstract

This paper explores the feasibility of content recommendation over interest-aware unstructured peer-to-peer (P2P) systems where peers sharing similar contents are connected. We present a novel and simple general metrics, by extending the Sorgenfrei coefficient to measure content similarities among peers. We provide two simple approximations of the proposed measure, that can be calculated by aggregating only the pair wise Sorgenfrei similarities, relaying on certain assumptions of statistical independence in the input data. We conduct experiments using a massive set of P2P file-sharing data to show that our new similarity measure could be a good predictor of the recommendation quality in unstructured distributed systems. The feasibility of finding similar peers in a simple unstructured system is also examined by simulation. We conclude that in unstructured P2P networks, an efficient recommendation system can be built without relying on any centralized or structured architectural extensions.

1 Introduction

Recommendation systems have gained in both popularity and importance during recent years [16, 23]. The fundamental goal of these systems is to offer content to users that may be in the field of their interest. Most of them are based on the collaborative filtering [12,21,22] approach. Although recommendation systems are achieving widespread success on the Web, these solutions are much less prevalent in non-centralized information systems.

Several unstructured P2P architectures have been proposed in the recent years, based on the idea of exploiting the semantic or interest-based proximity of the peers [8,9,26,28]. These solutions extend Gnutella-like unstructured P2P architectures with extra links between peers based on content or semantic similarities of the shared data. The purpose of these solutions is to improve the quality and the performance of search, assuming that peers usually search for similar contents to those they already have. Following the same assumption, our main goal is to examine whether content recommendation could be possible by using only these interest-based proximity links, without

laying on any further architectural extensions.

Another important part of designing a good recommendation system in a distributed system is to find an efficient measure to select neighbouring nodes. The current paper considers the most restricted – and most realistic – scenario when no semantic meta-data is available and the similarity measurement can only be based on the number of the common files between the participating peers. By extending the Sorgenfrei set similarity measure, we present a novel scalar measure of similarity between a set and collection of other sets, which, as we show later, could be used as a good forecaster of the performance of the recommendation procedure.

Finally, a very simple unstructured network build-up is simulated, to confirm our initial assumption that these unstructured architectures give feasible ways to find nodes sharing similar contents, and to give a hint about the cost-performance trade-off of these solutions.

The rest of the paper is organized as follows. In Section 2 the related work is presented. As one of the main contribution of this paper, in Section 4 our new similarity measure is introduced. Section 6 contains our examinations of the achievable recommendation quality, and the evaluation of the influencing factors. In Section 6.5 we present the results of our architectural simulations. The results of the simulations are summarised in Section 6.6 Finally, Section 7 concludes our work.

2 Related Work

The most simple way to exploit semantic similarities among peers is to extend a simple Gnutella-like unstructured architecture with interest-based links. Sripanidkulchai et. al. [26] introduced first these extra links called interest-based shortcuts. During queries these nodes are always searched for the content, and when none of the shortcuts have it, only then is the query flooded to the entire system. This result was improved by Voulgaris et. al. [28] examining different strategies of shortcut selection.

A different approach is to design a structured peer-to-peer architecture to support certain types of distributed collaborative filtering algorithms [11, 29]. These solutions are based on the idea of building and managing a large – usually DHT based – distributed database of all shared content in the system. Although they provide certain guaranties on finding similar contents in the system, their tighter administrative requirements could make these DHT based systems more sensitive to failures or to frequent joins and disconnects than the unstructured solutions.

To design a good recommendation system – for a particular data set – empirical work is indispensable. The publicly available Netflix prize [4] solutions [14, 17, 27] are the typical examples of the usual process of selecting the best solutions by training and testing several different solutions and their combinations. In P2P networks, the problem is not only to find the best recommendation algorithm, but to operate it in a distributed manner, without the help of central computing or data storage units. There are two main types of collaborative filtering algorithms, the neighbourhood-based, and the model-based approaches [18]. Model-based algorithms work by training a predictive model using available user ratings. This model can be used later to predict ratings of users for new items. Model-based methods are not feasible in P2P systems without major simplifications, due to the difficulty of training and maintaining a predictive model without using centralized resources. Neighbourhood-based solutions work by making recommendations using the data of similar user ratings or user profiles directly, and therefore these solutions, utilizing user similarities, fit well to

P2P architectures.

Since Jaccard proposed [13] his set similarity index in 1901 to measure the diversity of plant populations, numerous new numerical coefficients were proposed to measure similarities between binary – presence-absence – data sets. Many of them were also proposed by biologists or botanists for taxonomic, biogeographic, ecologic or paleoecologic purposes for example, the Dice [10], the Sørensen [24] or the Sorgenfrei [25] index. Since their initial introduction in ecology, they become widespread as a standard tool in classification tasks and have been used or for various purposes such as handwritten character [31] recognition. In recent comparative studies [6, 30], more than seventy binary similarity indices have been collected, compared and analysed.

3 Data Set

3.1 Technical Details

The simulations presented later on are all based on real-world file-sharing data that have been collected from approximately 50000 different Direct Connect [1] (DC) network nodes. Direct Connect is a P2P file sharing network. Clients connect to central hubs that provide information of other clients as well as file searching capabilities. Hubs are not interconnected, but a single peer can connect to multiple hubs. File transfers are done directly between clients.

```
<Directory Name="Judas Priest">
<File Name="Freewheel burning.mp3" Size="4232672"
TTH="EW4I3CT57QOGO4PEHZRAAFBX7CDBN2HL6T5NOGY"/>
<File Name="Hell Bent For.mp3" Size="2556656"
TTH="TYNRPZMGPSKQLLRXKAWFMNPZPRSMD2FJNR67FVA"/>
<File Name="Breaking the law.mp3" Size="2487275"
TTH="SMHXNSXA4M55VJ6AMJ4YD3EUPKEQPAGTYCL3CNY"/>
<File Name="Deceiver.mp3" Size="2623529"
TTH="PKWHG2O76WFNKIZRVJPSUYM7D6NNSB47AAV4YIA"/>
<File Name="Before the dawn.mp3" Size="3223719"
TTH="7I7RVZUJFHQ5S5ZBKI5LNM3MJC6BXNEHY5YYBKA"/>
</Directory>
```

Figure 1: Part of a DC++ XML file list

DC clients also can serve an XML-based list of all their shared files to the other clients. In Figure 1 you can see part of an example XML file list. The file list contains the name, the size and a content based hash code of all shared files of a given client. Directory information is also available.

A modified Direct Connect client program has been used – based on the StrongDC++ [2] sources – to automate the collection of file lists and to collect more than 50000 lists from peers on different hubs. The main parameters of the collected data are presented in Table 1.

File lists downloaded	50106
Total number of shared files	257942017
Number of different shared files	90707624
Sum of the size of shared files	1595.5 TB

Table 1: Main parameters of the collected data set

3.2 Pre-processing

The data set has been restricted to mp3 files. This filtering can be viewed as a fast and straightforward dimensionality reduction: User similarities are considered only in musical taste. As Table 2 shows, nearly half of the distinct files are mp3s in the collected data set.

Nodes containing mp3	34756
Total number of shared mp3 files	68567712
Number of different mp3 files	41102406
Sum of the size of shared mp3 files	368.8 TB

Table 2: Main parameters of the shared mp3 files

4 Set Similarity Measurement

The rationale behind basing our examination on using set similarity measures is their simplicity and usability even in restricted scenarios where only binary – presence-absence – data are available.

To measure similarity between nodes, the Sorgenfrei [25] set similarity coefficient was used.

Definition 1. *Given two sets A and B, the Sorgenfrei similarity coefficient is the cardinality of the intersection squared and divided by the product of the cardinalities of the two sets:*

$$\mathrm{S}(A,B) = \frac{|A \cap B|^2}{|A||B|}$$

In our case, the sets correspond to nodes of the P2P network, and members of a set are the shared files of the given node.

According to a comparison study of 76 different binary similarity coefficients [6], the behaviour of the Sorgenfrei coefficient is very similar to the more prevalent Jaccard similarity index. Sorgenfrei has been chosen over Jaccard because of its linearity, which made possible to give a simple probabilistic interpretation, and to extend it to measure similarity between a set and a collection of sets based on this interpretation (see Section 4.1).

The Sorgenfrei distance: $\mathrm{D}(A,B) = 1 - \mathrm{S}(A,B)$ is a semi-metric [30]. Semi-metric is a distance function $\mathrm{D} : X \times X \to \mathbb{R}$ where:

$$\mathrm{D}(A,B) \geq 0$$

$$\mathrm{D}(A,B) = 0 \Leftrightarrow A = B$$

$$\mathrm{D}(A,B) = \mathrm{D}(B,A)$$

is true for $\forall A, B \in X$

Contrary to other metrics, the Sorgenfrei distance does not fulfil the triangle inequality:

$$\exists A, B, C \in X \rightarrow \mathrm{D}(A,C) \nleq \mathrm{D}(A,B) + \mathrm{D}(B,C)$$

4.1 Combining Coefficients

A simple probabilistic interpretation of the binary Sorgenfrei similarity coefficient is provided, that can be used to extend the similarity index to measure similarity between a given set and a collection of sets.

Our goal is to capture the total amount of information contained by a collection of nodes with respect to a given node. This can be achieved by a function that is monotone in the sense, that by extending the set of similar nodes with a new node the aggregated similarity coefficient cannot decrease. To elaborate such a function the following probabilistic interpretation of the binary Sorgenfrei similarity coefficient was used:

If a uniform random sample from the set B of size $|B \cap A|$ is taken, thus each item of set B will be in that sample with probability $|B \cap A|/|B|$, and then a single item from set A is selected randomly, then that selected item will be part of the sample from B with probability:

$$\frac{|B \cap A|}{|B|}\frac{|A \cap B|}{|A|} = \frac{|A \cap B|^2}{|A||B|} = \mathrm{S}(A,B)$$

This is obviously true for two sets because the event that the selected item from set A is in the intersection, and the event that this particular item is in the random sample from set B, are independent events.

To ensure our requirement of monotonicity, the aggregated Sorgenfrei similarity index is then defined the following way:

Definition 2. *Let* $\mathrm{S}(A, \{B_1, B_2, \ldots, B_k\})$ *be the similarity coefficient of the set* A *and the collection of sets* $\{B_1, B_2, \ldots, B_k\}$*, where* $\mathrm{S}(A, \{B_1, B_2, \ldots, B_k\})$ *equals the probability, that an element chosen randomly from set* A *occurs at least in one of the uniform samples form sets* $B_1, \ldots, B_k$ *of size* $|B_1 \cap A|, \ldots, |B_k \cap A|$

The combined Sorgenfrei similarity index can be calculated by calculating and summing probabilities for each item from set A separately.

Theorem 1. *For every item* X*, let* $P(B_i, X) = 1/|B_i|$ *if* $X \in B_i$*, and* $P(B_i, X) = 0$ *if* $X \notin B_i$*, then the aggregated similarity index can be calculated by:*

$$\mathrm{S}(A, \{B_1, B_2, \ldots, B_k\}) = 1 - \frac{\sum_{\forall X, X \in A} \prod_{i=1}^{k} (1 - |A \cap B_i| P(B_i, X))}{|A|} \tag{1}$$

This calculation is based on the fact that the occurrences of a given single element from set A in the k different samples are independent, therefore the probability of an element X not being contained in any of the samples, simply equals the product of the individual probabilities: $1 - |A \cap B_i| P(B_i, X)$. Selecting a random element from set A is an independent event, therefore each term in the sum has to be multiplied by $1/|A|$.

We present two simple approximations, capable of estimating the aggregated similarities using only the following pairwise Sorgenfrei similarities:

A randomly selected item of A occurs in the random sample from set

- B_1 with probability $\mathrm{S}(A, B_1)$
- B_2 with probability $\mathrm{S}(A, B_2)$

$\vdots$

- B_k with probability $\mathrm{S}(A, B_k)$

Combining these k events under the assumption of independence, gives our first approximation:

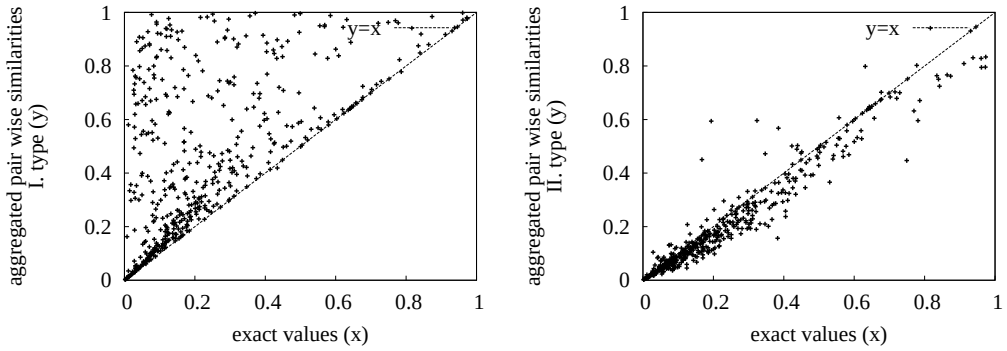
$$\mathrm{S}(A, \{B_1, B_2, \ldots, B_k\}) \approx 1 - \prod_{i=1}^{k} (1 - \mathrm{S}(A, B_i)) \tag{2}$$

Unfortunately, these k events are not necessary independent. Only a smaller part of set A is likely to appear in the sets B_i: the most popular files. Therefore, we propose another approach for estimation. This time we assume that intersections of set A and sets B_i are highly dependent: Each intersection is a subset of the largest intersection denoted by AB_{max}

Definition 3. *Let* $AB_{max} = \max_{1 \le i \le k} |A \cap B_i|$ *be the size of the largest intersection between* A *and the sets* B_i. *Then* $\mathrm{S}(A, \{B_1, B_2, \ldots, B_k\})$ *can be approximated by:*

$$\begin{aligned} \mathrm{S}(A, \{B_1, B_2, \ldots, B_k\}) &\approx \frac{AB_{max}}{|A|} \left[1 - \prod_{i=1}^{k} \left(1 - \frac{|A \cap B_i|}{AB_{max}\, |B_i|} \right) \right] = \\ &= \frac{AB_{max}}{|A|} \left[1 - \prod_{i=1}^{k} \left(1 - \frac{|A|\, \mathrm{S}(A, B_i)}{AB_{max}} \right) \right] \end{aligned} \tag{3}$$

To put it in another way, we compute the simple aggregated similarity index as if the size of set A was AB_{max}, which is the smallest imaginable size of set A considering the intersection sizes, then rescale the result using $AB_{max}/|A|$.



(a) Comparing formulae (2) and (1)

(b) Comparing formulae (3) and (1)

Figure 2: Evaluation of the aggregated similarity measures

As depicted in Figure 2, which compares the exact similarity indices with our two simple approximations, in about half of the cases, our simplest estimation gives very good results – points lying on the identity line, especially in the $y < 0.1$ domain. In the other cases, this assumption of independence tends to lead to overestimation. The second, slightly more complicated aggregation mechanism gives much better results, however in many cases it slightly underestimates the real values.

Correlation coefficients have been computed between real values and approximations from our two formulae, using more than 5000 randomly selected nodes with various nearest neighbour numbers between 10 and 200. The results are: 0.8527 for the first, and a very high correlation of 0.9843 for our second approximation.

The benefit of using these approximations lies in their simplicity while being nearly as good indicators of the recommendation performance (see Section 6) as the exact value.

5 Recommendation System

To suggest files to nodes an association rule learning approach has been used [3]. Association rules have been designed primary for market basket analysis, e.g., to identify customers purchasing habits, but the idea of using them to make recommendations comes naturally.

As Weiyang Lin et. al. have shown [15], association rule based recommendation can have significantly better performance than the traditional correlation based approaches.

Definition 4. *Let $D = B_1, B_2, B_3, \ldots, B_k$ be a collection of sets which are called transactions, and $I = i_1, i_2, i_3, \ldots, i_n$ be a set of binary attributes called items. Each transaction in D has a unique transaction ID and contains a subset of the items in I. An association rule is then defined as:*

$$X \Rightarrow Y$$

where $X, Y \subseteq I$ and $X \cap Y = \emptyset$. X is called the antecedent, while Y is called the consequent of the rule.

The rule expresses a connection between X and Y regarding their joint appearances in the sets of D.

There are several measures of significance to select relevant rules from the set of all possible rules. The two most widely used constraints are minimum threshold on support and confidence.

Definition 5. *The support* $\mathrm{supp}(X)$ *of an item set X is defined as the proportion of transactions in D that contains the item set X. The confidence of a rule $X \Rightarrow Y$ is defined as:*

$$\mathrm{conf}(X \Rightarrow Y) = \frac{\mathrm{supp}(X \cup Y)}{\mathrm{supp}(X)}$$

For example, $\mathrm{conf}(X \Rightarrow Y) = 0.9$ means that 90% of all transactions containing X also contain Y.

Confidence is not a good measure when the support of the consequent set is large. Imagine the case when:

$$\mathrm{supp}(Y) \gtrsim \mathrm{conf}(X \Rightarrow Y)$$

Then the appearances of Y in the transactions can actually be independent or in worst case, negatively correlated with the appearances of X, even when the confidence of the rule $X \Rightarrow Y$ is high.

Therefore, we have used a third measure – based on Pearson's chi-square (χ^2) test of independence – to select rules where the appearances of the antecedent set are truly positively correlated with the appearances of the consequent set. For mining association rules, C. Borgelt's implementation of the apriori frequent item set mining algorithm has been used [5].

The consequent part of an association rule can be used as a recommendation to a node if the node's file list contains the antecedent.

It is important to note that this method uses only information about item similarities by exploiting their co-occurrences, and does not rely directly on user-based similarities.

6 Simulation Results

6.1 Measurement Details

6.1.1 Environment and Methodology

For every measurement, we have used our preprocessed file-sharing database (Section 3.2). The k nearest neighbours of each peer – according to the Sorgenfrei set similarity measure – have been selected, where k represents the maximum number of semantic proximity links a peer can have. A local content recommendation to the peers was simulated by searching significant association rules (Section 5), using only the file sharing data of the k most similar nodes. An important advantage of the simulation-based approach was that comparable global recommendations could also be made by using the same recommendation approach that uses not only the most similar nodes but the whole database.

6.1.2 Evaluation by Validation

In centralized systems the evaluation of recommendation systems can be based on collecting and evaluating user feedbacks or behaviour over time. In pure P2P systems – especially in unstructured architectures – building and managing user profiles to precisely monitor user behaviour or to collect regular feedbacks is technically nearly impossible.

Giancarlo Ruffo and Rossano Schifanella [19] have proposed to use cross-validation to evaluate recommendation systems in P2P environments. They presented a simple case study using K-fold cross-validation: The original data set is randomly partitioned into K equal samples. In a single validation round $K-1$ samples are used to generate recommendations, and the remaining 1 sample is used to validate the performance of the system. This process is then repeated K times, by using each of the K samples as validation data exactly once. The results from the K cross-validation steps are averaged to get the final result.

Based on this idea, we have used a simple random sampling approach. The input database has been divided into two random parts – to a 80% large training and a 20% large validation sample. We have only used the 80% part to create recommendations, and then evaluated them using the 20% part as reference data.

6.1.3 Error Function

Performance has been evaluated by calculating the Sorgenfrei similarity coefficient, that has been used to select similar nodes previously, between the recommendation and the validation set.

For example, suppose that 10 different recommendations are found, from which 3 are also presented in the validation set of size 20 of the given node, then $3^2/10/20 = 0.0045$ will be assigned to the given recommendation as a measure of quality.

6.1.4 Measurements

To examine the effect of certain system parameters on the performance of recommendation, the following measurements have been performed:

- The correlation between the number k and the performance of recommendation based on the k most similar nodes, as well as recommendations based on the whole database (Section 6.2).
- The connection between the aggregated similarity measures and the success of recommendation (Section 6.3).
- The distribution of the aggregated similarity measures between peers and their k nearest neighbours (Section 6.4).
- A simple simulated best-first approach to find similar nodes in a distributed environment (Section 6.5).

Recommendations have been generated for more than 800 randomly selected nodes, where the number of nearest neighbours has been varied between 50 and 250. To describe the relationship between certain variables and the recommendation success, the Pearson product-moment correlation coefficient was used, which describes linear relationship between two random variables.

6.2 Number of Neighbours and the Quality of Recommendation

Our first main result is that recommendation success is not correlated with the number of neighbours (Table 3). It does not mean that increasing the number of neighbours does not increase the recommendation quality: in this case, the examined file suggestions have been run on different nodes, and therefore are independent. Thus this result indicates that the performance of independent recommendations cannot be estimated based exclusively on information about the number of similar nodes used to generate the recommendations.

Table 3: Correlation between the number of neighbours and the recommendation success

Correlation:	-0.027262
95% confidence interval:	-0.0932 – 0.0389
p value:	0.419

Also, this means that to certain nodes it is possible to generate much better file suggestions than to others, and increasing the number of neighbours cannot change this situation appreciably.

A different question is that if the nearest neighbour number of a given node is increased, will it influence recommendation quality positively, or negatively.

To answer this question, recommendations based on the full data set and recommendations based on only the 200 nearest neighbours have been compared for 200 randomly selected nodes. Our expectation was that the global recommendation would outperform the local version, however – as shown in Figure 3 – the recommendations based on only the nearest neighbours are better in most cases.

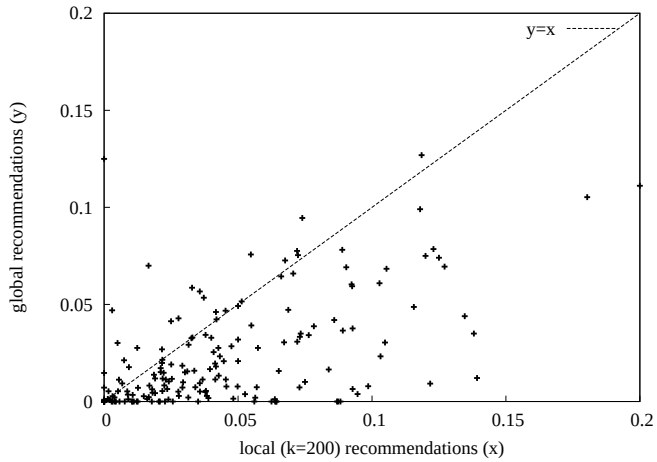


Figure 3: Comparing global and local recommendation performance

The explanation is simple: By using an association rule based recommendation algorithm in the global case – which considers information from all transactions with equal weight – we have lost user based similarity information, which seems to be crucial to the success of file suggestions.

The conclusion is that pre-filtering of input data based on user similarities – which happens naturally in certain P2P systems utilizing content similarities (Section 2) – can actually increase the performance of item based recommendation algorithms.

6.3 The Aggregated Similarity Measures and the Quality of Recommendation

The effect of neighbour similarity on the recommendation performance is significantly positive. As shown on Table 4, the correlation between the aggregated similarity coefficients and recommendation success is high. Comparing the results in the three columns, the first obtained by using the simple aggregated measure with assumption of independence (2), the second by using our second estimation formula (3), and the third obtained by using real similarity values (1), the conclusion is that our simple estimated measures are nearly as good indicators of success as the more complicated real values.

To further investigate the problem, another measurement was conducted by selecting groups of peers having aggregated distance measures in the $0.1 \cdot n \pm 0.01$ domain – where n is an integer between 0 and 9 – and by averaging the achieved quality of recommendation for each group. The results are depicted on Figure 4. This measurement does not express a mathematically solid

Table 4: Correlation between the similarity of neighbours and the recommendation success

	aggr., I. type	aggr., II. type	real
Correlation:	0.5105	0.6143	0.6242
95% confidence interval:	0.46 – 0.558	0.571 – 0.654	0.582 – 0.663
p value:	$< 2.2e\text{-}16$	$< 2.2e\text{-}16$	$< 2.2e\text{-}16$

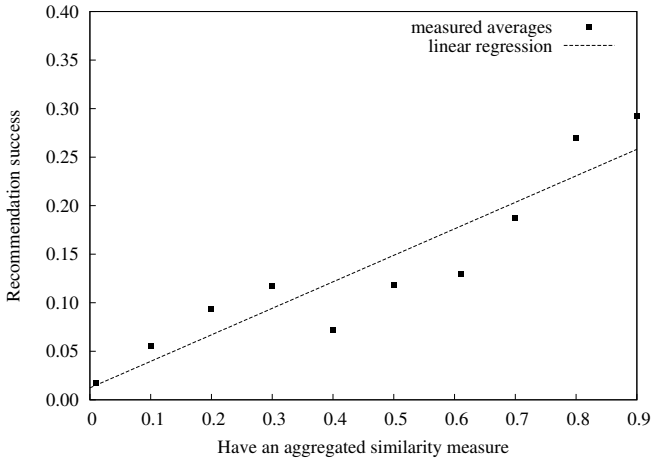


Figure 4: Averaged success of recommendation

relationship between the node similarities and the recommendation performance due to the fact that the input was made to be unbalanced artificially in an arbitrary way. We only provide it to present a simple graphical representation of the positive connection between the aggregated similarity values and the recommendation quality.

6.4 Distribution of Aggregated Similarities

The empirical complementary cumulative distribution function of aggregated similarities between nodes and their k nearest neighbours are depicted in Figure 5. The complementary CDF of the generated and evaluated recommendations is also depicted. The distributions are uneven, actually, they follow a power-law type distribution, which is a rather common feature of ranked data sets [7]. For example to the $k = 100$ case a Pareto distribution of second kind[1] can be fitted well with parameters: $\alpha = 0.73751$ and $\beta = 0.0156$

Aggregated similarities are good forecasters of recommendation quality, as we have shown in Section 6.3. From the distribution of these similarity values, and the actually measured recommendations we can draw the conclusion, that high quality recommendations can only be made to a smaller subset of nodes in the system. However, only a small 4.32% part of the nodes did not get any usable suggestion in our measurements.

[1]It is simply a standard Pareto distribution but shifted along the x-axis so that it starts at $x = 0$. Its cumulative distribution function is: $F(x) = 1 - \left(\frac{\beta}{x+\beta}\right)^{\alpha}$

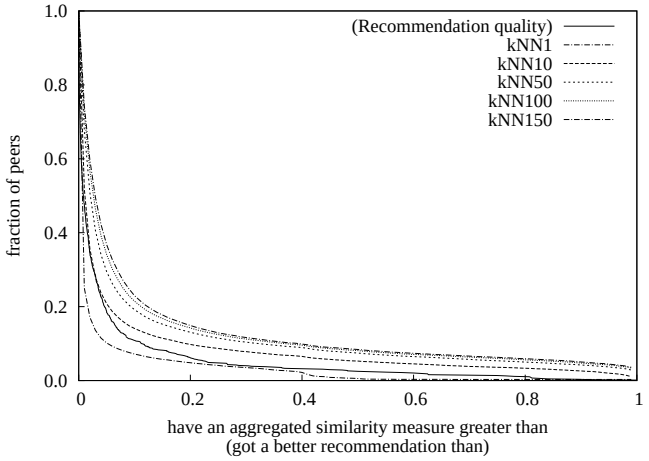


Figure 5: Distribution of aggregated similarity

6.5 Architectural Simulation

A simple simulation has been conducted – using the file lists of our database of 34756 nodes (see Table 2) – to show the feasibility of the assumption that a node can find most of its nearest neighbours even in a simple unstructured P2P architecture. The main properties of our set-up were the following:

- Nodes kept track of only the 100 most similar peers that they have met so far during the simulation.
- Nodes connected in random order by getting the address of a randomly selected peer being already in the system.
- During the simulation, each node – in random order – has initiated a neighbour list exchange with the most similar node from among those that had not participated in an exchange with this node before. During this exchange, the two nodes have sent their neighbour lists to each other and then each of them has merged the received list into its neighbour list keeping only the most similar one hundred nodes. This step was repeated 75 times. This procedure could be repeated in real systems from time to time in a completely distributed manner in order to keep the system up to date, however, in this case – for simplicity reasons – we considered and examined only this static scenario.
- To show the feasibility of this best-first approach, another simulation has been executed in which, peers have also kept track of the 100 most similar nodes, but the initiated neighbour list exchanges have always been conducted between two randomly chosen nodes.

In Figure 6, the main results from the simulation are shown. More than 40% of the nodes have found their absolute nearest neighbour even after only 10 initiated neighbour list exchanges per peer. The best-first approach has given surpassingly better results compared to the random procedure. This proves that the neighbours of similar nodes tend to be similar too, and that the

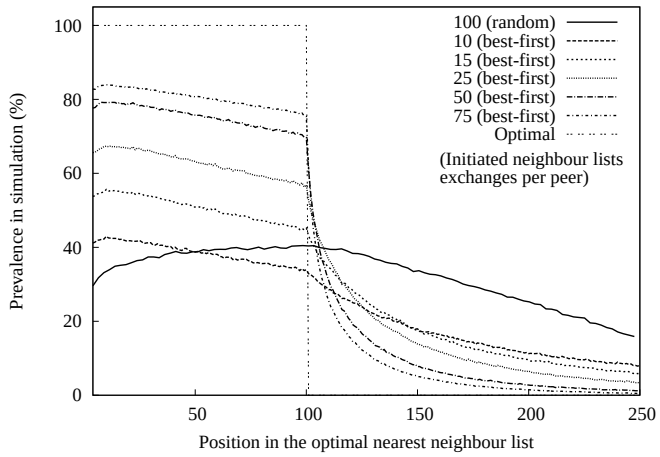


Figure 6: Success of finding the 100 nearest neighbour in a simple unstructured system

good results of the best-first approach are not only the consequence of the sheer number of the executed neighbour list exchanges.

We believe that these results could be improved even more by using more refined algorithms. For example, neighbour lists are changing and improving so rapidly that a relaxed approach about the exclusion of nodes which have formerly participated in list exchange could result in much better performance.

6.6 Summary

We showed that content recommendation is possible with a local procedure by using only nearest neighbours in distributed systems. The results indicate that the number of neighbours does not correlate with the quality of recommendation. Actually, increasing excessively the number of neighbours results in a decreased performance. Results presented in Section 6.3 show that the overall similarity of neighbours could be the main factor influencing the achievable recommendation quality. The statistical measurement of the file sharing data, showed that similarities – and therefore the achievable recommendation success – follows a heavy-tailed distribution. Usable recommendations can be given to a large percentage of the nodes – actually, seldom was the measured performance zero – and to a smaller group of nodes excellent recommendations could be offered. Finally, in Section 6.5 we showed that every node can find its nearest neighbours efficiently in unstructured networks, even by using a very simple best-first searching algorithm.

7 Conclusions

In this paper we have investigated the feasibility of content recommendation over unstructured P2P systems. Our measurements conducted on a large file-sharing database clearly showed that content recommendation is possible in distributed P2P architectures, without strict requirements concerning the structure of the overlay network.

Additionally, as a contribution to the field of presence-absence data analysis, we have introduced a novel aggregated measure of similarity between a given set and a collection of sets, and

provided two simple formulae to estimate this overall similarity based on pair-wise similarities only. According to our results, these measures can be good forecasters of recommendation performance.

We believe that a further investigation of the properties and variations of similarity measures could lead to a better understanding of the influencing factors of the recommendation quality.

References

[1] Direct connect protocol. `http://www.teamfair.info/wiki/index.php?title=Main_Page`.

[2] StrongDC++. `http://strongdc.sourceforge.net/index.php?lang=eng`.

[3] R. Agrawal, T. Imieliński, and A. Swami. Mining association rules between sets of items in large databases. *ACM SIGMOD Record*, 22(2):207–216, 1993.

[4] J. Bennett and S. Lanning. The netflix prize. In *Proceedings of KDD Cup and Workshop*, volume 2007, 2007.

[5] C. Borgelt and R. Kruse. Induction of association rules: Apriori implementation. In *Compstat: Proceedings in Computational Statistics: 15th Symposium Held in Berlin, Germany, 2002*, page 395. Physica Verlag, 2002.

[6] S. Choi, S. Cha, and C.C. Tappert. A Survey of Binary Similarity and Distance Measures. *Journal of Systemics, Cybernetics and Informatics*, 8(1):43–48, 2010.

[7] A. Clauset, C. R. Shalizi, and M. E. J. Newman. Power-law distributions in empirical data. *SIAM Review*, 51:661–703, 2009.

[8] E. Cohen, A. Fiat, and H. Kaplan. Associative search in peer to peer networks: Harnessing latent semantics. *Computer Networks*, 51(8):1861–1881, 2007.

[9] A. Crespo and H. Garcia-Molina. Semantic overlay networks for p2p systems. *Agents and Peer-to-Peer Computing*, pages 1–13, 2005.

[10] L. R. Dice. Measures of the Amount of Ecologic Association Between Species. *Ecology*, 26(3):297–302, 1945.

[11] P. Han, B. Xie, F. Yang, and R. Shen. A scalable p2p recommender system based on distributed collaborative filtering. *Expert systems with applications*, 27(2):203–210, 2004.

[12] J.L. Herlocker, J.A. Konstan, L.G. Terveen, and J.T. Riedl. Evaluating collaborative filtering recommender systems. *ACM Transactions on Information Systems (TOIS)*, 22(1):53, 2004.

[13] Paul Jaccard. Distribution de la flore alpine dans le bassin des Dranses et dans quelques régions voisines. *Bulletin de la Société Vaudoise des Sciences Naturelles*, 37:241–272, 1901.

[14] Y. Koren. The bellkor solution to the netflix grand prize. Technical report, 2009.

[15] W. Lin, S.A. Alvarez, and C. Ruiz. Efficient adaptive-support association rule mining for recommender systems. *Data Mining and Knowledge Discovery*, 6(1):83–105, 2002.

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

[17] M. Piotte and M. Chabbert. The pragmatic theory solution to the netflix grand prize. Technical report, 2009.

[18] F. Ricci, L. Rokach, B. Shapira, and P.B. Kantor, editors. *Recommender Systems Handbook*. Springer, 1 edition, 2010.

[19] G. Ruffo and R. Schifanella. Evaluating peer-to-peer recommender systems that exploit spontaneous affinities. In *Proceedings of the 2007 ACM symposium on Applied computing*, page 1578. ACM, 2007.

[20] S. Rui-min, Y. Fan, HAN Peng, and XIE Bo. PipeCF: a DHT-based Collaborative Filtering recommendation system. *Journal of Zhejiang University-Science A*, 6(2):118–125, 2005.

[21] B. Sarwar, G. Karypis, J. Konstan, and J. Reidl. Item-based collaborative filtering recommendation algorithms. In *Proceedings of the 10th international conference on World Wide Web*, page 295. ACM, 2001.

[22] J.B. Schafer, D. Frankowski, J. Herlocker, and S. Sen. Collaborative filtering recommender systems. In *The adaptive web*, pages 291–324. Springer-Verlag, 2007.

[23] R. Sinha and K. Swearingen. Comparing recommendations made by online systems and friends. In *Proceedings of the Delos-NSF workshop on personalization and recommender systems in digital libraries*, 2001.

[24] T. Sørensen. A method of establishing groups of equal amplitude in plant sociology based on similarity of species and its application to analyses of the vegetation on Danish commons. *Biol. Skr.*, 5:1–34, 1948.

[25] T. Sorgenfrei. Molluscan Assemblages from the Marine MiddleMiocene of South Jutland and Their Environments. *Copenhagen: Reitzel*, 1958.

[26] K. Sripanidkulchai, B. Maggs, and H. Zhang. Efficient content location using interest-based locality in peer-to-peer systems. *DEF*, 3(3):0, 2002.

[27] Andreas Töscher and Michael Jahrer. The bigchaos solution to the netflix grand prize. Technical report, 2009.

[28] S. Voulgaris, A.M. Kermarrec, and L. Massoulié. Exploiting semantic proximity in peer-to-peer content searching. In *10th IEEE International Workshop on Future Trends of Distributed Computing Systems, 2004. FTDCS 2004. Proceedings*, pages 238–243, 2004.

[29] J. Wang, J. Pouwelse, R.L. Lagendijk, and M.J.T. Reinders. Distributed collaborative filtering for peer-to-peer file sharing systems. In *Proceedings of the 2006 ACM symposium on Applied computing*, page 1030. ACM, 2006.

[30] M.J. Warrens. *Similarity coefficients for binary data : properties of coefficients, coefficient matrices, multi-way metrics and multivariate coefficients*. PhD thesis, 2008.

[31] B. Zhang and S.N. Srihari. Binary vector dissimilarity measures for handwriting identification. In *Proc. of SPIE Vol*, volume 5010, page 29.