Spotting Out Emerging Artists Using Geo-Aware Analysis of P2P Query Strings

Noam Koenigstein Yuval Shavitt Tomer Tankel
School of Electrical Engineering
Tel Aviv University, Israel
{noamk, shavitt, tankel}@eng.tau.ac.il

ABSTRACT

Record label companies would like to identify potential artists as early as possible in their careers, before other companies approach the artists with competing contracts. The vast number of candidates makes the process of identifying the ones with high success potential time consuming and laborious. This paper demonstrates how datamining of P2P query strings can be used in order to mechanize most of this detection process. Using a unique intercepting system over the Gnutella network, we were able to capture an unprecedented amount of geographically identified (geo-aware) queries, allowing us to investigate the diffusion of music related queries in time and space. Our solution is based on the observation that emerging artists, especially rappers, have a discernible stronghold of fans in their hometown area, where they are able to perform and market their music. In a file sharing network, this is reflected as a delta function spatial distribution of content queries. Using this observation, we devised a detection algorithm for emerging artists, that looks for performers with sharp increase in popularity in a small geographic region though still unnoticable nation wide. The algorithm can suggest a short list of artists with breakthrough potential, from which we showed that about 30% translate the potential to national success.

Categories and Subject Descriptors

H.2.8 [**Database Management**]: Database Applications-Data Mining; I.6.5 [**Computing Methodologies**]: Simulation and Modeling Model Development

General Terms

Algorithms, Experimentation

Keywords

P2P Queries, Emerging Artists

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.
KDD'08, August 24–27, 2008, Las Vegas, Nevada, USA.
Copyright 2008 ACM 978-1-60558-193-4/08/08 ...$5.00.

1. INTRODUCTION

For a record label, investing in an unknown artist is a high risk business. Most of the artists will fail to make the anticipated "breakthrough", but when one does succeed, the return on investment far exceeds the initial cost. Locating artists with high success potential is thus important for the recording industry. In this study we suggest an algorithm to automatically identify promising artists at very early stages of their career, based on geo-aware query strings collected from a file sharing network. We used here data collected from the Gnutella network but the same analysis technique is applicable for other file sharing networks (e.g., BitTorrent, eDonkey and Kad) and even for Web 2.0 websites (e.g., YouTube, Myspace, Metacafe).

A KDD process is the nontrivial process of identifying valid, novel, potentially useful, and ultimately understandable patterns in data [11]. Many KDD studies are done on databases of customers received from commercial companies. Much emphasis is put in the process of detecting potential buyers in a data set of known customers [7, 3]. This paper suggests a database of geographically identified (geo-aware) P2P query strings as a target for KDD processes. Previous work has shown that P2P information can be used in prediction of top albums life-cycle in the *Billboard Top 200* chart [2]. In our study we took a different approach and looked at fairly rare queries, those that are not even on the top 2000 list, trying to detect emerging artists with higher potential to make a national level breakthrough. Our main idea is that artists that are extremely popular in their hometown area, have high potential to make a breakthrough in the national level. Local popularity comes as a result of performances in the artist's hometown that create a word of mouth ripple that sends people to download the artist's songs from a file-sharing application or to listen to the artists clips on YouTube. By looking for emerging artists that show specific patterns of local popularity increase, our detection algorithm was able to generate a short list of artists, with a 15-30% prediction success. Without our algorithm, ones needs to spot the successful artists from a list thousands times longer, and with a success probability of less than 0.1%.

The rest of this paper is organized as follows: In Section 2, we explain how the database of Gnutella geo-aware query strings is created. We then describe our data set statistics and some pre-processing procedures we performed. In Section 3, we describe the small world model for the spatial and temporal diffusion of new products. In Section 4, we specifically discuss the diffusion of songs created by new

artists. In Section 5, we describe the detection algorithm and its prediction results. Finally in Section 6, we present our conclusions.

2. DATA SET

Shaked Gish *et al.* [5] used the Skyrider systems[1] to collect geo-aware queries over a period of three and a half months, the largest data collection effort in scale and accuracy until this paper. Our study uses the same Skyrider data collection system, but over longer time period, nine and a half months, in order to demonstrate how valuable information regarding emerging artists can be extracted from P2P query strings.

2.1 Methodology

While it is possible to capture a large quantity of queries by deploying several hundred ultrapeer nodes[2], it will not be possible to tell the origin of most of these captured queries. The basic problem in identifying the origin of captured queries is that queries do not in general carry information regarding their origin. What they do usually carry is an "Out Of Band" (OOB) return IP address. This address allows clients that have content matching a query to respond to a location close to the origin of the query, without having to backtrack the path taken by the query message. However, as most queries come from firewalled clients, in most cases the OOB address will belong to the ultrapeer connected to the query origin, acting as a proxy on behalf of the query originator.

It was previously shown that Gnutella clients show a tendency for inter region clustering [9]. It thus might be concluded that, when the location of the leaf originating a query is not known, its ultrapeer's geographic location could be used instead. To verify this, we performed 24 hourly crawls of the Gnutella network in one day in November 2006. Each crawl reached between 3.5 million leafs (at US night time) to 7 million (at US evening time) to obtain a total of 20-22 million distinct leaves. Comparing the geographic location of ultrapeers to their leaves reveled that the probability of a leaf being from the same country as its ultrapeer is between 55% and 69% for ultrapeers in the USA, between 24% and 44% for Japanese ultrapeers, and as low as 3% to 10% for ultrapeers in the UK. Thus, an ultrapeer's geography is not a good enough predictor of the geography of its leaves, contradicting the results reported by Rasti *et al.* [9], at least for 2006.

We were able to overcome the above difficulties by deducting the missing origin IP address. Let us briefly explain how this can be achieved in the Gnutella network. Figure 1 depicts a small network segment containing an intercepting Skyrider node, along with other ultrapeers and leafs. Ultrapeer B is directly connected to the Skyrider node. Thus any query that traversed only a single hop must have come from it, and we thus know its IP address. Leaf A, leaf C (firewalled), and ultrapeer D are at a distance of two hops away. We cannot easily distinguish between queries coming from A, C, and D. Furthermore queries originating at C will contain B's OOB return address as C is firewalled or otherwise unable to accept incoming connections. However as we are

[1]Skyrider is a startup company dedicated to providing enhanced services to users of peer-to-peer (P2P) networks and to make these networks more useful to the consumer and business communities.

[2]Ultrapeer nodes are special nodes that route search queries and responses for users connected to them

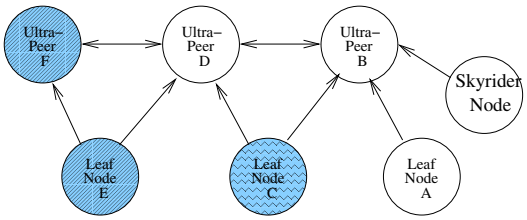


Figure 1: Geo-Aware Query Measurement in a Two-Tier Overlay

directly connected to ultrapeer B, we can simply compare the query's OOB address with B's address. If they are not identical, the query must have come from A or D, and the address is guarantied to be the origin's address. If the query contain's B's address but passed two hops, it must be acting as a proxy for C. In this case C's address is not available, and the query is not recorded. Ultrapeer F and leaf E are at a distance of 3 hops away. When we intercept their queries we cannot know whether the OOB IP address belongs to them, or perhaps to ultrapeer D acting as a proxy for E. Thus any query that traversed 3 hops or more is discarded. As a result, a Skyrider node records traffic originating from its immediate neighborhood only (having a hop count ≤ 2), thus requiring a massive deployment of such nodes. The described setting eliminates most of the bias against popular queries which travel only short distances before being satisfied. Discarding queries that traveled more than two hops cancels the advantage of "rare" queries that stay in the network longer. However, this setting does introduce a bias against queries from firewalled clients, as only queries that can receive incoming connections are recorded.

According to [9] the vast majority of the Gnutella network is comprised of Limewire clients (80%-85%) and Bearshare clients (6%-10%). The Limewire client does not allow users to perform any kind of automatic or robotic queries. It does not allow queries with the SHA1 extension [3], nor does it allow the automatic re-sending of queries. When it does send duplicate queries, it uses a constant Message ID which enables a simple removal of any duplication. The presence of duplicate records is an important data quality problem in many KDD applications [8]. By recording only queries originating from Limewire clients, we were able to significantly reduce the amount of duplications and automatic (non-human) queries, without loosing too much of the traffic. Capturing only Limewire queries is an easy task as Limewire "signs" the message ID associated with each message it sends. This signature can be easily verified by the intercepting node, allowing it to ignore queries from all other clients.

2.2 Data Set Statistics

After the removal of queries which traveled more than two hops, non Limrewire clients, firewalled quires and non OOB enabled queries we remain with approximately 25% of the intercepted queries. A daily log file typically contains 25-40 million record lines, each line consists of the query string, a

[3]SHA1 queries are queries in which only the hash key of a known file is sent without a string. This is useful when a client already started downloading and needs more sources.

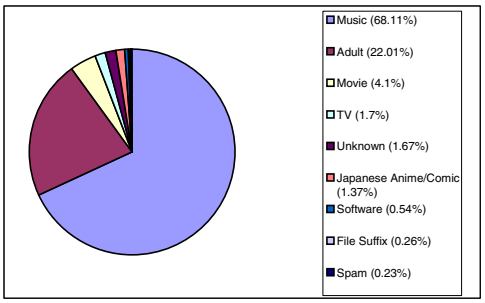


Figure 2: Classifying Gnutella Queries

date/time field, and the IP address of the node issuing the query. Using IPligence database to resolve the geographical location of IP addresses, we bound country, state, city, latitude and longitude to each query string, allowing us to pin point the source of each query string to the level of cities and sometimes even smaller areas like the boroughs of NYC. Since we concentrate our study on American artists, we removed all the non US queries reducing an additional 55%-60% of the data records.

Our data-set comprised of query strings collected over a period of nine and a half months from mid October 2006 until July 2007. The activity on the Gnutella networks increases by 20%-25% over the weekend [5]. We thus used weekly samples taken on a Saturday or a Sunday of every week of that period. The sample from the 51st week of 2006 and the samples from weeks 24 and 25 of 2007 were not recorded as a result of technical difficulties. We thus remained with 38 samples instead of 41. The total number of geo-aware query strings processed in this study is **310,380,190** making it unprecedented in scale among P2P queries studies.

Using the geo-aware query strings we generated weekly global and local popularity charts. For each string, its global and local popularity was calculated by aggregating the number of appearances intercepted. The chart position was calculated by sorting the queries according to popularity. This means that more than a single string can be ranked in a given position. However, since we dealt with millions of queries this rarely ever happened among the popular strings.

2.2.1 *Classifying The Queries*

Manually classifying the top 500 most popular queries, we found that 68.11% of the files were music related while 22.01% were adult content. These two categories dominate the Gnutella network accounting together for 90.12% of the queries. Figure 2 depicts the top 500 queries classification.

3. SPATIAL DISTRIBUTION IN EARLY PREDICTION OF A NEW PRODUCT SUCCESS

Before discussing emerging artists diffusion in P2P networks, let us review the small world model for the spatial and temporal diffusion of new products. Small-world modeling assumes that the main driver behind a product growth is communication between individuals. A successful product is noticeable by the formation of adopter-clusters around early adopters. These clusters represents areas where the new product is spreading as a result of the *Word Of Mouth* (WOM) effect and can be used in order to predict a product's future success.

3.1 The Small-World Model and The WOM Effect

In his groundbreaking work from 1973 "The Strength of Weak Ties" [6], sociologist Granovetter suggested an explanation of how micro-level interactions between individuals affects macro level phenomena. Relationships between individuals can be modeled as a network, where persons are nodes connected through ties to other individuals. This modeling can be used to understand the way information is spread in a community; a phenomenon known as the WOM effect. Small world networks were introduced to mathematically model Granovetter's ideas of strong and weak ties and the WOM effect [12]. Adopting a consumer research orientation, Garber et al. [4] used it to investigate a new product spatial diffusion after it was introduced to the market [4].

The small-world model depicts the market as a binary matrix, the elements of which represent individuals in different locations. A '1' represents a consumer who bought the product (adopted) and a '0' is a consumer who hasn't (a non-adopter). Each consumer can interact with his acquaintances and influence them to purchase the new product. A persons's group of acquaintances consists mostly of people in his close vicinity (neighbors) and a small group of people outside his vicinity. Similarly each cell in the matrix can interact with its neighboring cells up to some specified range and a small number of random cells outside the cell's vicinity. In a classical small-world model the proportion of distant links is 5% at most. Beyond this level, the social system becomes similar to a random network [1].

According to [4] there are two types of events that cause a non-adopter to buy the product:

- *Internal Factors:* An interaction with an acquaintance adopter that influence the consumer to buy the product (The WOM effect). Such an event happens with probability q due to either the person's strong (close) or weak connections.

- *External Factors*: An individual decides to adopt because of external influence such as advertising. This event happens with probability p.

Therefore the probability that a non-adopter will adopt at a time slot t is:

$$prob(t) = 1 - (1-p)(1-q)^{v(t)+r(t)} \quad (1)$$

Where $v(t)$ is the number of previous adopters with whom the non-adopter have connections in his vicinity (strong ties) and $r(t)$ is the number of adopters with whom he has ties outside his vicinity (weak ties).

The adoption pattern of the new product depends on the numeric values of p and q. High values of p mean effective advertisement which cause a uniformly distributed increase in new adopters. However a product's success depends mostly on the value of q, which models the WOM effect. High values of q is an indication that people "like" the new product, and so adopters effect non-adopters to purchase it. Low values of q mean adopters are not satisfied with the product and the product is likely to fail.

3.2 Divergence Measurements in The Prediction of a New Product Success

High values of q causes the formation of adopter clusters at the early stages of a new product spatial diffusion. These adopter clusters represents areas where the new product is spreading as a result of the WOM effect. When q is low, the product diffuses uniformly in space due to the "external" marketing efforts. Therefore looking at a new product spatial sales distribution, we should distinguish between the uniform distribution and the presence of adopter clusters.

Garber et al. [4] suggested the Kullback-Leibler Divergence Measurement to predict products success probability. The Kullback-Leibler Divergence for the difference between two probability vectors P and Q is defined as follows:

$$D_{KL}(P||Q) = \sum_i P(i) \log \frac{P(i)}{Q(i)} \quad (2)$$

If we take P to be the product's sales distribution vector, where component $P(i)$ is the relative amount of sales occurred in region i, and Q is the uniform distribution, then we receive a numeric value that measures how much the sales distribution differs from the uniform distribution. The minimum value, zero, is received when the sales distribution is uniform.

The Kullback-Leibler Divergence is non negative but asymmetric and thus it can not serve as a true distance metric. Instead the Jensen-Shannon measurement was suggested:

$$D_{JS}(P||Q) = \frac{1}{2} D_{KL}(P||M) + \frac{1}{2} D_{KL}(Q||M) \quad (3)$$

where $M = \frac{1}{2}(P + Q)$

In [4] the two measurements were compared and shown to poses similar predictive capabilities. Therefore we limit our discussion from here on to the Kullback-Leibler divergence only.

When the Kullback-Liebler Divergence is used, it is advisable to use P as the product sales distribution and Q as the uniform distribution. Using the limit: $\lim_{x \to 0} x \log x = 0$, will avoid division by zero, in the case that in one of the regions there were no sales at all.

According to [4], after the new product becomes popular and adopted by many consumers, the spatial distribution of buyers becomes more similar to the uniform distribution resulting in a decline of divergence measurement. Eventually there are adopters everywhere, and no remains of the initial clusters. At that stage there is no difference between the distribution vector of a successful product and that of a failure. They are both uniformly distributed. The difference is shown in the number of adopters, which is actually what determines if the product succeeded or failed.

3.2.1 Maximum Divergence

We want to stress out two points regarding the divergence measurement of actual sales distribution from the uniform distribution:

The furthest distribution from the uniform distribution is the δ (single peak) distribution. This can be mathematically proved: If Q is uniform with N vector components then

$$D(P||Q) = \log N - H(P)$$

where $H(P)$ is the entropy of P. Since the entropy function achieves its minimum, zero, only when the probability vector is δ, it follows that the maximum value of $D(P||Q)$ is $\log N$ and it is achieved when P is a δ function. Therefore the divergence measurements suggested by [4] will give a higher value to a distribution in which all the sales occurred in the same region, than to a distribution with the anticipated adopter clusters. This may mean that the use of the divergence measurement in [4] is not optimal. Thus, we suggest to look into pattern recognition methods to detect future product success. This, however, is beyond the scope of our work.

In small-world modeling, a δ spatial distribution of sales can be explained by the following conditions: First there is no national level advertisement, namely, no external influence; second, virtually no random links (no weak ties) between individuals in different geographic locations; and finally, a single initial location where some sales do occur (due to local exposure). If the product is successful, the WOM effect will take place in the vicinity of the first adopters. Sales will increase only in the region where the first sales were made and all the other regions will have no sales at all. In fact, this is the typical case with emerging artists. For example a new rapper usually starts by performing in his local neighborhood. If he is successful his initial audience will spread the word, and the artist will become increasingly popular in the region of his hometown. However, since this rapper does not have the means for a marketing campaigns in the national level, it is very unlikely he will break out of his original region.

3.2.2 Divergence Measurement in P2P Queries

In practice, sales data is not extracted from uniform sized regions. For example US sales may be aggregated on the level of states. More sales in California than in North Dakota, may not necessarily mean that the product is received better in California since the population size difference between the two states is huge. Therefore if the distribution of customers is not uniform, we need to adjust the divergence measurement by letting the vector Q reflect the spatial distribution of potential customers at each region.

This study investigates the popularity of new artists according to their local popularity as reflected in the Gnutella network. The spatial regions we classified differed in the amount of total query strings originating from them. In our divergence measurement we set the vector Q to represent the distribution of query strings in the classified regions. Component $Q(i)$ is the fraction of total query strings originated from region i of the number of total query strings in all the regions:

$$Q(i) = \frac{R(i)}{\sum_{i=0}^{N} R(i)} \quad (4)$$

where $R(i)$ is the number of query strings in region i and N is the number of regions. In the reminder of the paper we will use Q as the distribution vector of all the intercepted queries (like we explained above), and P for the distribution of queries specific to some artist or song, namely

$$P(i) = \frac{R_s(i)}{\sum_{i=0}^{N} R_s(i)} \quad (5)$$

where $R_s(i)$ is the number of queries in region i that are in the subset of strings s relating to a ceratin artists, and N is the number of regions.

Position	String	Appearances
1	adult	882
2	akon	583
3	lil wayne	345
4	this is why im hot	290
5	justin timberlake	270
6	fergie	233
7	beyonce	230
8	dont matter	229
9	mims	224
10	pretty ricky	203
11	party like a rockstar	198
12	ciara	195
13	porn	186
14	party like a rock star	185
. . .		. . .
37	shop boyz	132

Table 1: Atlanta's local popularity chart on February 18th (week 8)

4. THE SPATIAL DIFFUSION OF EMERGING ARTISTS

The small-world model is a general model that is not product specific. In this section, we specifically investigate the diffusion of songs created by new artists in time and space. By presenting typical real world showcases, we will explain the general trends for most artists. We examine the local popularity and the distribution of audience before a breakthrough at the national level occurs. Using marketing terminology we can say that a good "product" means a catchy hit single, and that the lack of a nation wide advertising campaign results in a delta like distribution of adopters. This delta distribution is reflected in high divergence values. We also show that after a commercial breakthrough, the artist reaches a larger audience and the spatial distribution of audience becomes close to uniform resulting in a decreased divergence.

4.1 Party Like A Rockstar

Emerging from the Bankhead area of Atlanta, the *Shop Boyz* are a typical example of locally popular artists rise to nation wide success. Their hit single *Party Like a Rockstar* entered the *Billboard Hot 100* on the chart issued on May 5th 2007 (week 18) at the 80th position. On the chart issued on June 9th (week 23) it already reached the second position.

On February 18th 2007 (week 8), very few people outside Atlanta knew who the *Shop Boys* were. The string "party like a rockstar" ranked 10156 in the global queries chart. However among the Hip-Hop fans in Atlanta the group was already highly popular. Table 1 depicts the local popularity chart in Atlanta that week. The strings "party like a rockstar" and "party like a rock star" are ranked 11 and 14 respectively. The string "shop boyz" was ranked 37 in Atlanta that week. This is especially impressive considering the fact that, as we describe later, the song entered the bottom of the Billboard charts only a month and a half afterwards. Also note that the query chart contains the strings "adult" and "porn" which are not music related. By removing the non music related queries, we get even higher ranks for the song.

Table 2 depicts the total number of queries intercepted by our system in eleven major US cities, the number of *Shop Boyz* related queries and the corresponding distribution vectors Q and P (*Shop Boyz* related queries are the queries that include the substrings "shop boyz" or "party like a rockstar").

City	All Queries	Shop Boyz	Q	P
Atlanta	778,960	1,046	0.123	0.906
Chicago	761,124	6	0.12	0.005
Dallas	875,189	11	0.138	0.01
Detroit	262,193	2	0.041	0.002
Houston	644,150	10	0.102	0.009
Los Angeles	446,822	55	0.07	0.048
New York	859,056	5	0.135	0.004
Philadelphia	284,607	7	0.045	0.006
Phoenix	518,578	4	0.082	0.003
San Antonio	454,705	3	0.072	0.003
San Diego	455,547	6	0.072	0.005

Table 2: Number of total queries, number of Shop Boyz related queries and the corresponding distribution vectors Q and P for different US cities. This information reflects the data sampled on the weekends of February of 2007.

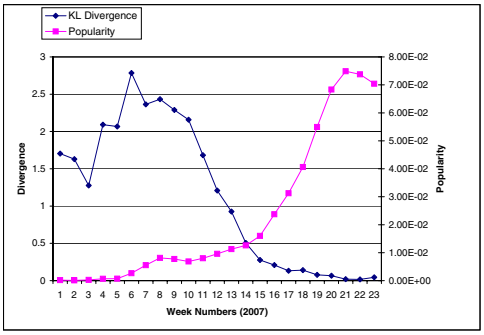


Figure 3: *Shop Boyz* Popularity and Divergence

This information reflects the data sampled on the weekends of February of 2007. Note the absolute dominance of Atlanta in the *Shop Boyz* related queries. Over 90% of the queries originated from there. The Kullback-Liebler divergence between P and Q, is 2.4 which is relatively high. For example the Kullback-Leibler divergence of the popular string "adult" was only 0.02 and the divergence of the string "Avril Lavigne" (a well establish artists) was 0.0123 during that time. When mapping the geographical location of the *Shop Boyz* audience, the popularity in Atlanta is reflected as an almost delta shaped distribution vector. This delta function is an indication of a large base of local audience. A cluster of early adopters passing the word of the hottest new band in town. Since the *Shop Boyz* signed with Universal Republic only a few weeks later[4], there was very little advertisement outside Atlanta that could have affected potential audience.

Figure 3 depicts the *Shop Boyz* popularity and the Kullback-Liebler divergence measurement. The popularity was measured by calculating the percentage of "*Shop Boyz*" related query strings of the total number of queries sampled on that week. The divergence measurement was calculated for every week as described above. The fluctuations in the first few weeks (January) are due to the limited number of *Shop Boyz* related queries intercepted in these weeks: a total of 132 in the first four weeks (compared with the 1046 intercepted in the four weeks after). The two obvious trends are the increase in the band's popularity and the decrease of the divergence. The band started to gain popularity in February, and on the 11th week of 2007 the divergence measurement

[4]Universal Republic is part of Universal Music Group. The deal was announced April the 10th, 2007

started to decline significantly. Again, this is 7 weeks before it even entered the *The Billboard Hot 100*. If you lived the Hip-Hop scene in Bankhead Atlanta, you might have known the *Shop Boyz* were hot already in early 2007, but for the rest of us, this information was not available. Using our potential detection algorithm described in In Section 5 with the right parameters (local threshold $T_l = 500$ and the second detection pattern), one can spot out the string "party like a rockstar" in the local P2P popularity chart of Atlanta in the 6th week of 2007 (February 4th). The rest of the *Shop Boyz* related strings mentioned here are detected in the three following weeks.

4.1.1 Additional Examples

Figure 4 depicts the popularity and divergence of four artists. In Figure 4(a) we see an already known, well established artists: *Madonna*. As expected, the divergence values are low, the popularity values are high, and both graphs maintain relatively constant values. Figure 4(b) and Figure 4(c) are two more examples of emerging artists: *Yung Berg* from Los-Angeles and *Soulja Boy* from Atlanta. In both figures we see the increase in popularity is simultaneous to the decrease in the divergence (in Figure 4(c) these trends are evident after week 20). The detection algorithm suggested in the next section spots both of these artists many weeks before they enter the bottom of the Billboard's charts. Figure 4(d) depicts an additional locally popular artists: *Mistah F.A.B.* from the Bay area. At least up to the time of writing, this artists remains only locally popular, as indicated by the low popularity values and the high divergence. According to different hip-hop websites, his success was stemmed after he faced serious media criticism due to his controversial lyrics [10]. This example demonstrates that local popularity alone, is not a sufficient predictor for artists future success. We shall discuss this more in the next section.

5. DETECTION ALGORITHM FOR EMERGING ARTISTS

Following the observations from the previous sections, we devised an algorithm for detecting query strings belonging to emerging artists. The algorithm is designed to be executed at fixed time intervals. Since our data collection was done on a weekly basis, we have chosen to work on weekly intervals. This is also in accordance with the *Billboard* charts, where most charts are issued once a week. The algorithm's inputs for week i are all the geo-aware queries collected since the previous execution; local popularity charts from previous iterations; and an *All Times Popular List* from previous week, $ATPL_{i-1}$, a list of known already popular query strings from the previous weeks. The outputs of the algorithm are a list of query strings with high probability to belong to emerging artists; local popularity charts for week i; and $ATPL_i$, an updated list of already popular query strings to be used by the algorithm in the next iteration. The concept of *All Times Popular List*, as well as the flow description are explained below.

5.1 Algorithm Description

5.1.1 Local Popularity

In order to mathematically model *local popularity* of queries, T_g and T_l, global and local popularity thresholds are defined. Suppose $r_g(i)$ and $r_l(i)$ are a query's global and local charts ranking at week i, respectively. In our study, $r_g(i)$, is the ranking of a query in the US queries popularity chart, and, $r_l(i)$ is the ranking in a city's popularity chart. Queries of emerging artists will hold that:

$$r_l(i) \leq T_l \tag{6}$$

$$r_g(i) \geq T_g \tag{7}$$

and

$$r_l(i) \leq r_g(i) \tag{8}$$

The first condition (6) assures a minimum level of local popularity, meaning the artist has a stronghold of hometown audience. The second condition (7) assures that the artist is not globally popular, and the third condition (8) requires that the local popularity ranking will be higher (lower in value) than the global one. It thus follows that choosing T_g and T_l such that:

$$T_g \geq T_l \tag{9}$$

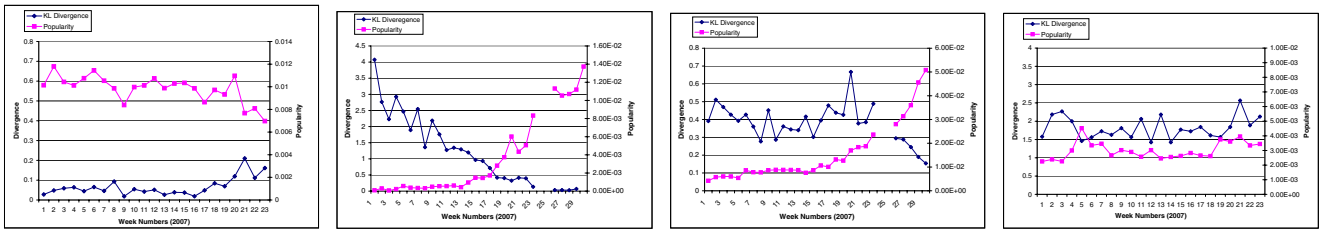
and maintaining (6) and (7) assures that a query is *locally popular*.

The global popularity threshold, T_g, is used to distinguish the top most popular queries in the US, from the rest of the queries. Each of our weekly global popularity charts for US queries, consists of nearly two million entries (on average 1.73 million). Based on our observations, we wanted the global popularity threshold to approximately discern the top one thousandth of the chart. We thus chose $T_g = 2000$. An artist that succeeds in entering the global top two thousand queries list will experience thousands of downloads a day. For example, in the sample taken on January the 14th (week 3), on the bottom of the top two thousand list are the strings "jimmy hendrix" and "ipod movies" both with 567 identified queries. Obviously these strings belong to popular queries. Remembering that approximately 75% of the queries were removed after traveling more than two hops, and taking into account the other pre-processing filtering as described in Section 2, we can assume these files are downloaded at least thousands of times a day. The global popularity threshold was used also in testing the algorithm success percentage, as well as in the construction of the *All Times Popular List*.

5.1.2 The Concept of All Times Popular List

In order to detect new artists, one needs to filter out already famous artists and non-relevant query strings e.g., sex related (22%), movies (4.1%), software (0.54%), etc. We used the data collected in 2006 in order to create the initial *All Times Popular List*, henceforth $ATPL$. This list is comprised of all the strings that reached the global top T_g queries sometime during the history, in 2006. Obviously it contained many popular sex related strings, like "sex", "adult", and "porn". Unlike emerging artists, these strings are "non-volatile", having constant popularity over time [5]. $ATPL$ also contains many artists that were already popular before 2007, for example, "jay z", "akon", "madonna", "avril lavigne" and so on. By ignoring the strings in $ATPL$, we filtered many of the sex related queries and many of the already known artists.

Naturally, as time goes by, new strings become popular, and $ATPL$ needs to be updated. As described above, our



(a) Madonna (b) Yung Berg (c) Soulja Boy (d) Mistah F.A.B.

Figure 4: Popularity and Divergence - Different showcases. Samples of weeks 24 and 25 are missing as explained earlier.

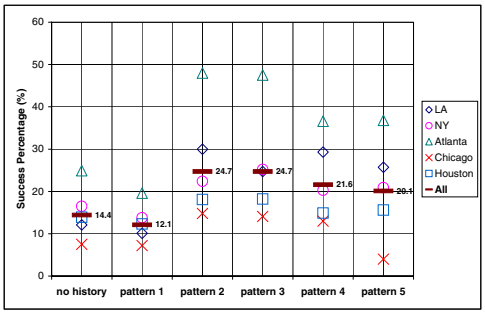


Figure 5: Success Percentage For Different Detection Patterns

Pattern 1	$r_l(1) > r_l(0)$
Pattern 2	$r_l(2) > r_l(0)$
Pattern 3	$r_l(1) > r_l(0)$ and $r_l(2) > r_l(1)$
Pattern 4	$r_l(2) > r_l(0)$ and $r_l(1) - r_l(0) > r_l(2) - r_l(1)$
Pattern 5	$r_l(1) > r_l(0)$ and $r_l(2) > r_l(1)$ and $r_l(1) - r_l(0) > r_l(2) - r_l(1)$

Table 3: Detection Patterns

algorithm is iterative, and the $ATPL$ is passed from one iteration to the next. $ATPL_0$ is the initial data collected in 2006. At iteration i, when examining new queries data, the algorithm first updates $ATPL_{i-1}$, from the previous iteration by adding all the new strings that passed the T_g threshold in the current global popularity chart. The updating is done without removing any of the strings already in the list, therefore $ATPL_i$ is an aggregation over the entire history until the examined time.

Besides removing sex related strings, and strings of already popular artists, this weekly update was effective in removing much of the other non-music related queries that has recently became popular, such as new movies, software, and TV series. This is due to the fact that content such as movies and software has a uniform distribution in the US. When a new movie or software comes out, it either becomes popular in all the cities simultaneously due to media campaigns and thus enters $ATPL$, and ignored by the detection algorithm, or remains unpopular in all the cities. Unlike emerging artists, it is very rare for such content to become locally popular.

5.1.3 *Algorithm Flow*

In the previous section we saw that emerging artists are characterized by high divergence values. Thus, one might expect high divergence to be a good indication for an emerging artist. In fact this would have been in accordance with the work of Garber *et al.* [4], where it was shown that high values of divergence in innovations, indicate higher probability to succeed. However, an attempt to detect emerging artists based directly on the divergence measurement proved to be ineffective. Where in [4], a product's geographical distribution is used in order to predict its success probability, in our case, not every query string is a "product". One example is the case of rare spelling mistakes and typos, where the distribution vector of the query string P would be a perfect delta vector. The divergence value will be maximized, but obviously this string doesn't represent an emerging artists. Therefore we took a different approach based on global and local popularity charts. Formulating the three conditions in (6), (7), and (9) allowed us to eliminate rare queries (6), while still maintaining the demand for non uniform distribution: (7) and (9).

In order to meet condition (7), again the $ATPL$ comes handy. After updating $ATPL$ at the beginning of each iteration, it contains all the globally popular strings for the current iteration. Therefore by ignoring all the strings in $ATPL$, we assert (7).

Detecting locally popular artists can be performed on queries from any city, or region. In this study we focused on major US cities, since most emerging artists, especially rappers, are active in urban concentrations. The algorithm builds local popularity charts for each city, and trims the chart at T_l. Doing so asserts condition (6), and thus we are left with *locally popular* query strings only. For each such query string, the algorithm examines the local chart rank values in the past n weeks, and looks for "promising" patterns. In other words, the algorithms looks for pattern in the tuple $\overline{R_l} = \langle r_l(0), r_l(1), \ldots, r_l(n) \rangle$, where $r_l(0)$ is the local popularity of the string in this week, and $r_l(j)$ is the local popularity of the string, j weeks ago. If a desired pattern is found, the algorithm outputs the string. Table 3 describes the different detection patterns tested.

5.2 Testing the Algorithm

As described above, the data collected in 2006 was used in order to initialize $ATPL$. We then executed the iterative algorithm on the first twelve weeks of 2007. On each week,

the algorithm marked out a list of *locally popular* queries that showed a "promising" pattern of popularity increase in the local chart. For each string that was marked by the algorithm, we checked whether it reached the global top T_g in one of the following weeks, until the 30th week of 2007 (when our data collection ended). If the string reached *global popularity*, we classified it as a *success hit*. We thus defined the *success percentage* of the algorithm as the percentage of success hits of all the strings indicated by the algorithm.

In the first twelve weeks of 2007, 1612 new strings entered $ATPL$. There are approximately 1.73 million unique strings on each week. Thus, the probability of a random pick to be a success hit is $\frac{1612}{1.73M} < \frac{1}{1000}$, which is equivalent to 0.1% success percentage. The algorithm suggested in this article has an average success percentage that ranges between 15% to 30%, which is an improvement of two orders of magnitude over a random pick.

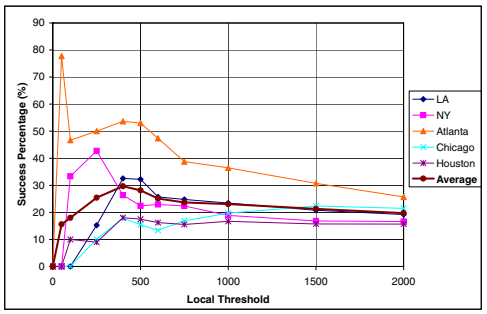
5.2.1 Results For Different Detection Patterns

Figure 5 depicts the success percentage for the detection patterns in Table 3 using queries from five cities and $T_l = 500$. We concentrated on patterns not longer than three weeks ($\overline{R_l} = \langle r_l(0), r_l(1), r_l(2)\rangle$). Results are presented for five major cities and their aggregation. All the patterns assert a recent increase in the local popularity. This means a negative local derivative in the ranking tuple $\overline{R_l}$, since higher popularity translates to a lower chart position. Also presented is the algorithm's success percentage when it uses no history at all. In this case detection is based on the current local popularity rank alone. For the five cities aggregated, this simple no history pattern already gives us a success percentage of 14.4%.

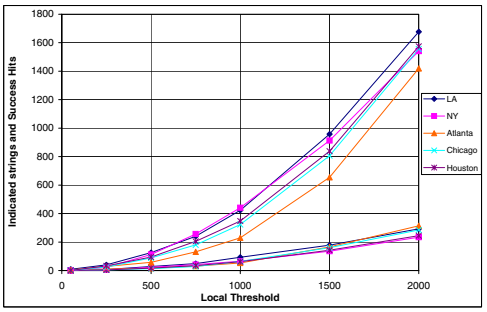
The first pattern requires a popularity improvement since the previous week; namely: $r_l(1) > r_l(0)$. In this example, pattern 1 seem to reduce the overall success percentage by 2.3%, however in measurements we performed with more cities this pattern showed a small improvement, but still insignificant when compared to the other patterns. It seems that in order to track popularity increase in emerging artists, we need to look at time periods longer than one week.

The second and third patterns require a two week popularity improvement. These two patterns are effective in increasing the success percentage by more than 10%. The second pattern is a more "relaxed" version of pattern 3. While pattern 3 requires a popularity increase of two weeks in a row, pattern 2 only requires a popularity increase since two weeks ago. When compared in other cities, pattern 2 yields slightly higher success rates than pattern 3. Again we interpret this result by suggesting that popularity changes should be measured on time periods longer than one week, and a small weekly decline in popularity should be ignored.

We attempted to improve patterns 2 and 3 by requiring also a negative second derivative. We thus tried pattern 4 and 5, which are similar to 2 and 3 with one additional requirement: $R_l(1) - R_l(0) > R_l(2) - R_l(1)$. This means that not only the chart position is higher (lower in value) from two weeks ago, but also the climbing up the chart accelerates. As seen on Figure 5 this attempt failed. Apparently, when a song is climbing up the P2P popularity chart, it usually makes bigger leaps forward when it is still in the lower part of the chart, but when it reaches the top of the chart, progress seems to come in smaller steps. This means that the leaps up the chart are getting smaller, and the second



(a) Success Rate vs. Local Threshold



(b) Identified Strings and Success Hits vs. Local Threshold

Figure 6: The effect of T_l on the success percentage

derivative is actually positive. It should be understood that in most cases of promising artists, the increase in the number of new queries per week accelerates, but as ones moves up the chart it is harder to translate this increase to an advance in ranking. For higher values of T_l pattern 2 showed the best performance, therefore we will discuss the rest of the results using this pattern.

From Figure 5 we learn that some cities have higher success rate than others. In the early nineties Seattle's music scene was considered very "hot", as many famous *Grunge* bands came from it (*Nirvana*, *Pearl Jam*, *Alice in Chains* and others). Today Atlanta is considered "hot" in the Hip-Hop scene as reflected by its high success rate. Back in the nineties, it wasn't an easy task for a record company based in New York, to follow the Grunge scene in Seattle without actually being present. Today this can be done from anywhere in world, simply by monitoring the P2P activity.

5.2.2 Choosing Local Threshold

Figure 6(a) depicts the weekly average success percentage for different values of T_l in five cities using detection pattern 2. Obviously, when $T_l = 0$, the algorithm detects no emerging artists, since it doesn't consider any of the strings. As T_l increases, the algorithm starts looking for emerging artists in the top T_l strings of the local chart. When $T_l < 50$, the success percentage is relatively low. Since locally popular emerging artists usually don't reach the very top of the local chart, the algorithm misses much of them. However, in most cities when $T_l > 50$, the success percentage increases

dramatically, and a maximum is reached in the range of $50 < T_l < 500$. The actual peak changes from city to city. The average curve peaks when $T_l = 400$ with a success percentage of almost 30%. For $T_l > 500$, condition (6) is relaxed and the success percentage decreases. Higher values of T_l will cause the algorithm to detect the artists earlier, but with less certainty. Chicago is an exception since its success percentage increases in this range and reaches a maximum at $T_l = 1500$. We explain this by suggesting that the local artists in this city do not manage to gain enough popularity to reach higher values in the local chart, and thus languish at the lower positions.

Figure 6(b) depicts the aggregated number of unique strings that the algorithm indicated during the first 12 months of 2007 (upper curves), and the aggregated number of actual success hits (lower curves). Obviously the number of indicated strings will always be higher or equal to the number of actual success hits. As the local threshold increases, the algorithm reviews more strings, and thus all curves are monotonically increasing. However, the number of indicated strings increases at super-linear rate, while the number of actual success hits only increases linearly. The latter curves show over 200 different success hits during that time period. However, the number of identified artists in this time period is smaller since most artists have more than one success hit related to them. For example when $T_l = 1000$ the *Shop Boyz* have 5 different success hits in Atlanta (The strings: "shop boyz", "shop boys", "party like a rockstar", "party like a rock star" and "like a rock star"). Finding the optimal local threshold for each city, as well as the best detection pattern can be good subjects for future research.

5.3 Suggestions for improvements

The algorithm presented above can be further improved in many ways. However, we believe it does make the case for the feasibility of early detection of emerging artists using P2P query strings. Some of the possible improvements may be:

1. The algorithm can give a confidence measurement to each string it indicates (as suggested by Ling and Li [7]).
2. A clustering algorithm can be applied to bind together different strings belonging to the same artist.
3. If additional information is available, it can be used by the algorithm for better decision making. For example, if the algorithm is implemented over a file sharing network (as the one described here), sometimes the content a user's shared directory is also available. This can be used to bind information such as file types and even musical genres to each query string. Doing so will make it much easier to remove non music related query string, and allow genre specific analysis.

6. CONCLUSIONS

In this paper, we demonstrated the use of geo-aware Gnutella query strings in the detection of emerging artists. We defined local and global thresholds in order to mathematically model the concept of *local popularity*, and devised an algorithm for spotting out locally popular artists on the rise. We tested the algorithm on real data and showed it can reach a 15-30% average success percentage. The techniques and algorithm used in this study are not limited to P2P queries and can be used with other sources of information such as websites like YouTube and Myspace. We believe our ideas can be utilized by record companies to identify unsigned artists or by other media companies for detecting emerging musical trends.

7. REFERENCES

[1] L. A. N. Amaral, A. Scala, M. Barthelemy, and H. E. Stanley. Classes of small-world networks. *PNAS*, 97(21):11149–11152, Sept. 2000.

[2] S. Bhattacharjee, R. D. Gopal, K. Lertwachara, and J. R. Marsden. Using P2P sharing activity to improve business decision making: proof of concept for estimating product life-cycle. *Electronic Commerce Research and Applications*, 4(1):14–20, 2005.

[3] P. Domingos and M. Richardson. Mining the network value of customers. In *KDD '01: Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 57–66. ACM, 2001.

[4] T. Garber, J. Goldenberg, B. Libai, and E. Muller. From density to destiny: Using spatial dimension of sales data for early prediction of new product success. *Marketing Science*, 23(3):419–428, 2004.

[5] A. S. Gish, Y. Shavitt, and T. Tankel. Geographical statistics and characteristics of p2p query strings. In *The 6th International Workshop on Peer-to-Peer Systems (IPTPS'07)*, Feb. 2007.

[6] M. S. Granovetter. The strength of weak ties. *The American Journal of Sociology*, 78(6):1360–1380, 1973.

[7] C. X. Ling and C. Li. Data mining for direct marketing: Problems and solutions. In *Knowledge Discovery and Data Mining*, pages 73–79, 1998.

[8] G. N. Noren, R. Orre, and A. Bate. A hit-miss model for duplicate detection in the who drug safety database. In *KDD*, pages 459–468. ACM, 2005.

[9] A. H. Rasti, D. Stutzbach, and R. Rejaie. On the long-term evolution of the two-tier gnutella overlay. In *IEEE Global Internet Symposium*, Barcelona, Spain, Apr. 2006.

[10] J. Shepherd. Ghost rider fallout haunts Mistah F.A.B., Mar. 2007. Featured on The VIBE Magazine website. Last Accessed December 2007.

[11] F. Usama, P.-S. Gregory, and S. Padhraic. The kdd process for extracting useful knowledge from volumes of data. In *Communication of the ACM*, volume 29, pages 27–34, Nov. 1996.

[12] D. J. Watts and S. H. Strogatz. Collective dynamics of small-world networks. *Nature*, 393:440–442, June 1998.