Published by :
http://www.ijert.org

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 9 Issue 07, July-2020

A CMiner Algorithm based Mining Technique to Extract Competitors for Kaggle Dataset

Dr. G. Malini Devi
Assistant Professor,
Department of CSE,
G. Narayanamma Institute of Technology & Science For Women,
RangaReddy, Telangana, India.

Marlapudi Apurupa
M. Tech Student,
Department of CSE,
G. Narayanamma Institute of Technology & Science For Women,
RangaReddy, Telangana, India.

***Abstract*:- In any competitiveness business, achievement depends on the capacity to make a thing more speaking to clients than the challenge. Various inquiries emerge with regards to this assignment: how would we formalize and measure the intensity between two things? Who are the principle contenders of a given thing? What are the highlights of a thing that most influence its competition? In spite of the effect and importance of this issue to numerous areas, just a constrained measure of work has been given toward a powerful arrangement. In this paper, we present a formal definition of the aggressiveness between two things, in light of the market fragments that they can both spread. The assessment of intensity uses client audits, a plenteous wellspring of data that is accessible in a wide scope of spaces.**
To address these challenges, a highly scalable framework is used for finding the top-k competitors of a given item which includes an efficient evaluation algorithm and an appropriate index. This framework is efficient and applicable on real datasets with very large populations of items from different domains.

Keywords: Data mining, Web mining, Information Search and Retrieval, Electronic trade.

1. INTRODUCTION

A deep research has shown the key significance of distinguishing and observing a firm's contenders. Inspired by this issue, many advertisements and the broad networks have concentrated on experimental strategies for contender identification [8], by using the techniques for breaking down known contenders. Surviving examination on the previous has concentrated on mining relative articulations (for example "Thing A is superior to Thing B") from the Web or other printed sources [12]. Even though such expressions can indeed be indicators of competitiveness, they are missing in numerous areas. For example, when brand names are compared at the firm level, almost certainly, relative examples can be found by just questioning the web. Nonetheless, it is anything but difficult to recognize standard spaces where such proof is incredibly rare, for example, shoes, jewelery, inns, eateries, and furniture. Inspired by these inadequacies, another formalization of the competition between two things, in light of the market sections that they can both spread is proposed.

Formal definition 1. Let U be the number of inhabitants in every single imaginable client in a given market. We think that a thing I covers a client u ∈ U on the off chance that it can cover the majority of the client's necessities. At that point, the intensity between two things i, j is relative to the quantity of clients that they can both spread.
The competitiveness depends on the accompanying perception [3]: the competition between two things depends on whether they go after the consideration and business of similar gatherings of clients (for example a similar market fragments). For instance, two eateries that exist in various nations are clearly not their target groups. Consider the example shown in Figure1.

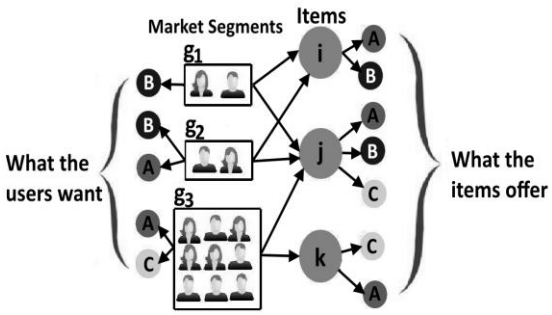

Fig. 1: An example of the competitiveness

The figure outlines the competition between three things i, j and k. There are many types of customers with some features such as A,
B, C as their priority. There are also different items such as i, j, k with different features. Therefore users are divided into different groups based on their mutual priorities. The most prioritized features can be known based on the number of customers in a group and the items that provides those features can be considered as the competitive items in the market.

This strategy enables to operationalize the definition of competitiveness and address the issue of finding the top-k contenders of a thing in some random market. This also presents significant computational difficulties, particularly within the sight of huge datasets with hundreds or thousands of things with an efficient assessment calculation and a suitable file.

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 9 Issue 07, July-2020

2. LITERATURE SURVEY

Author name	Title of research paper	Advantages	Limitations
K.Xu, S.S.Liao J.Li, Y.Song	Mining Comparative Opinions From Customer Reviews For Competitive Intelligence [6] (2011)	The graphical model, CRF[5] is used.	Can only model fixed dependencies
TN.Doan, F.C.T.Chua, EP.Lim	Mining Business Competitiveness from User Visitation Data [13] (2015)	Uses PageRank [7] model	It favors older pages
S.B.orzs¨onyi, D. Kossmann, K. Stocker	The skyline operator [10] (2001)	It filters out main points from large set of points [4]	Inefficient performance on some datasets.
Dr. P. Banumathi	Finding top -k frequent Itemset on Online Shopping [2] (2017)	top-k HUIs(TKU) [14] is used	Tedious process for user
Rui Li, S. Bao, J. Wang, Y. Yu	CoMiner: An Effective Algorithm for Mining Competitors from the Web [9] (2006)	Cominer algorithm is used. [11]	Computational time is more.

3. EXISTING SYSTEM

Management literature is rich with works so that human intervention is necessary to identify competitors. Distinguish key serious measures indicated how a firm can induce the estimations of these measures for its rivals by mining (I) its own point by point client exchange information and (ii) total information for every contender.

DRAWBACKS:

These days the information is duplicating each day. Identifying competitors manually is impossible with vast data resources. To recognize contenders, it is to be done physically which is outlandish with tremendous information assets. Existing methodology isn't proper for assessing intensity in the middle of any two distinct things or firms in a given market.

4. PROPOSED SYSTEM

The formalization of the competitiveness is a method for processing all the fragments in a given market depended on mining huge audit datasets is portrayed. This technique presents noteworthy computational difficulties, particularly within the sight of huge datasets containing hundreds or thousands of things, for example, those which are frequently found in standard areas. The top-k calculation includes a proficient assessment calculation and a suitable file.

ADVANTAGES:

The proposed system works effectively to find top-k rivals of an item from huge datasets.

5. METHODOLOGY

The main purpose of the project is to find top-k competitors from large datasets. The following methodology provides the process of finding competitors from the given datasets.

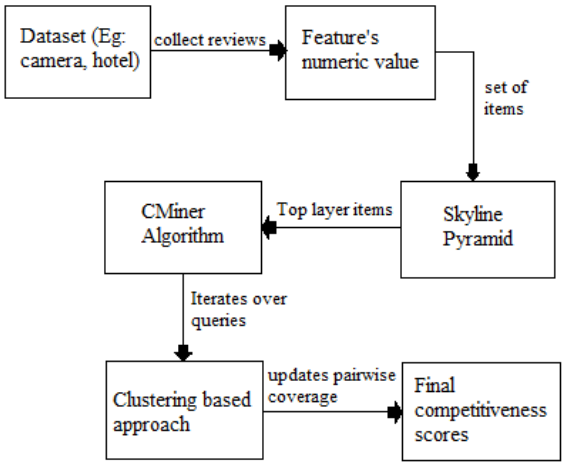

Fig. 2: Methodology for mining top-k competitors.

The figure shows the description about the modules/steps followed to find top-k competitors:

- Load the required datasets.
- Find all the required features for competitiveness using pairwise coverage.
- Submit the query and retrieve matching items.
- Process the reviews of returned items and make purchase decisions.
- Finding Top-k competitors.

5.1 SKYLINE PYRAMID

Skyline pyramid is a structure that greatly reduces the number of items that need to be considered. It represents the subset of points in a population that are not dominated by any other point. An item dominates another if it has better or equal values across features.

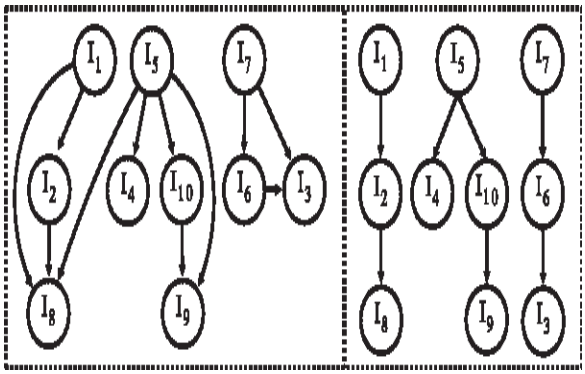

Figure 3: Skyline Pyramid

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 9 Issue 07, July-2020

5.2 PAIRWISE COVERAGE

The pairwise coverage is denoted as $V_{i,j}^{f}$ where i,j are two items and f is a feature, gives the percentage of all possible values of f that can be covered by both i and j. $V_{i,j}^{f}$ can be computed in different ways for different types of features.

5.2.1 Binary and categorical features :

Categorical features contain one or more values from a finite space. Each binary feature indicating the coverage of the original feature's possible values. Here brand of a digital camera is an example of single value and the amenities offered by a hotel is an example of multi-value features.

$V_{i,j}^{f}$ = f[i] x f[j] {binary features} →Eq.1

5.2.2 Numeric features:

The numeric features takes values from a range of [0,1], where the higher values indicates the higher preference of the particular feature and the least value indicates the less preference.

For example if there are two restaurants i, j with values 0.8 and 0.5 respectively for the quality of food, then the customers who go in search for higher standards such as values > 0.6 would eliminate the restaurant j.

$V_{i,j}^{f}$ = min(f[i],f[j]) {numeric features}→Eq.2

5.2.3 Ordinal features :

The ordinal features the popular five star scale to evaluate the quality of service or product. The customers that demand at least 3 stars will eliminate the product with less than 3 stars.

$V_{i,j}^{f} = \frac{\min(f[i],f[j])}{|Vf|}$ {ordinal features} →Eq.3

Therefore the competitiveness between two items i and j in a market with a feature subset F can be defined as follows:

$C_F(i, j) = \sum_{q \epsilon 2^F} p(q) \times V_{i,j}^{q}$ →Eq.4

From the above equation, let 'p(q)' be the percentage of users represented by a query 'q' and let $V_q^{i,j}$ be the pairwise coverage offered by two items i and j to the space defined by the features in q. Therefore the competitiveness $C_F(i, j)$ between i and j can be derived. For every feature f ϵ q,

$$V_{i,j}^{q} = \prod_{f \epsilon q} V_{i,j}^{f} \quad \rightarrow \text{Eq.5}$$

The above equation allows computing the pairwise coverage of any query of features.

6. PSEUDO CODE FOR CMINER

Input: set of items I, Item of interest i∈ I, feature space F, Collection Q ∈ 2^F of queries with non-zero weights, skyline pyramid D_I, int k
Output: Set of top-k competitors for i

```
1: TopK← masters(i)
2: if (k ≤ |TopK|) then
3:    return TopK
4: end if
5: k ←k - |TopK|
6: LB ← -1
7: X ← GETSLAVES (TopK, D_I ) U D_I[0]
8: while (|X| != 0) do
9:    X ←UPDATETOPK(k, LB, X)
10:  if (|X| != 0) then
11:     TopK← MERGE (TopK, X)
12:     if (|TopK| = k) then
13:        LB ← WORSTIN (TopK)
14:     end if
15:     X ← GETSLAVES (X, D_I)
16:  end if
17: end while
18: return TopK
19: Routine UPDATETOPK(k, LB, X)
20: localTopK← Ø
21: low(j) ← 0, ∀j ∈ X.
22: up(j)← ∑ p(q) * V^q_{j,j} , ∀j ∈ X
23: for every q ∈ Q do
24:    maxV←p(q) * V^q_{i,i}
25:    for every item j ∈ X do
26:       up(j)← up(j) - maxV + p(q) * V^q_{i,j}
27:       if (up(j) < LB) then
28:          X ←X \ {j}
29:       else
30:          low(j)← low(j) + p(q) * V^q_{i,j}
31:          localTopK.update(j, low(j))
32:          if (|localTopK|≥ k) then
33:             LB ←WORSTIN (localTopK)
34:          end if
35:       end if
36:    end for
37:    if (|X|≤ k) then
38:       break
39:    end if
40: end for
41: for every item j ∈ X do
42:    for every remaining q ∈ Q do
43:       low(j)← low(j) + p(q) * V^q_{i, j}
44:    end for
45:    localTopK.update(j, low(j))
46: end for
47: return TOPK(localTopK)
```

6.1 Finding Competitive Product

The dataset was deliberately chosen from various spaces to depict the cross-area materialness of the methodology. Notwithstanding the full data on everything in the dataset, the full arrangement of audits that were accessible on the source site was gathered likewise. These surveys were utilized to (1) gauge questions probabilities, as depicted and (2) separate the suppositions of commentators on specific features. The exceedingly referred to strategy is utilized to change over each audit to a vector of sentiments, where every assessment is defined as a component extremity blend. The level of audits on a thing that express a positive supposition on a specific highlight is utilized as the element's numeric incentive for that thing. These are considered as assessment highlights.

7. RESULTS & DISCUSSIONS

The CMiner algorithm is compared to naïve and GMiner [1] algorithms for better understanding the performance and computational analysis of the CMiner algorithm.

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 9 Issue 07, July-2020

7.1 Computational analysis of algorithms:

NAÏVE ALGORITHM:

The Table 7.1 contains the values of time taken to compute top-k competitors for camera dataset.

Table 7.1: Computational time of naïve algorithm for camera dataset.

No of competitors	Time (sec)
10	1.2
50	1.3
100	1.4
200	1.62
300	1.82

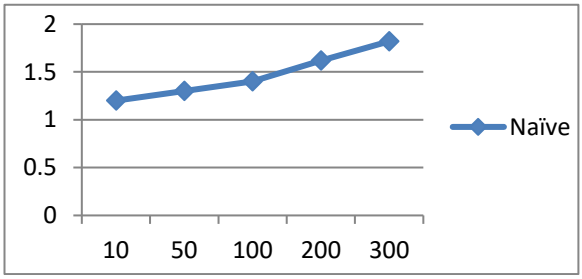

Fig 4: Computational graph of naïve algorithm for camera dataset

The graph in fig 4 shows the computational time for the naïve algorithm to mine top-k competitors for different sets of camera dataset.

GMINER ALGORITHM:

The Table 7.2 contains the values of time taken to compute top-k competitors for different number of sets of camera dataset.

Table 7.2: Computational time of GMiner algorithm for camera dataset

No of competitors	Time(sec)
10	1.1
50	1.3
100	1.32
200	1.35
300	1.8

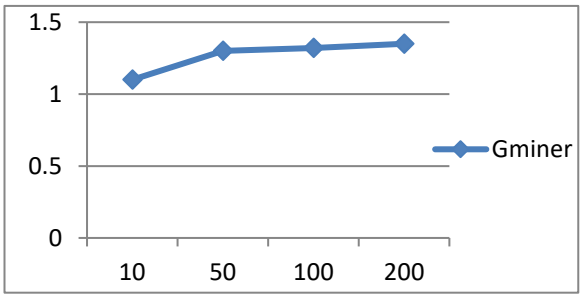

Fig 5: Computational graph of GMiner algorithm for camera dataset

The graph in fig 5 shows the computational time for the GMiner algorithm to mine top-k competitors for different sets of camera dataset.

CMINER ALGORITHM:

The Table 7.3 contains the values of time taken to compute top-k competitors for different number of sets of camera dataset. CMiner computational time is less when compared to both GMiner and Naïve algorithms.

Table 7.3: Computational time of CMiner algorithm for camera dataset

No of competitors	Time (sec)
10	0.6
50	0.64
100	1
200	1.2
300	1.7

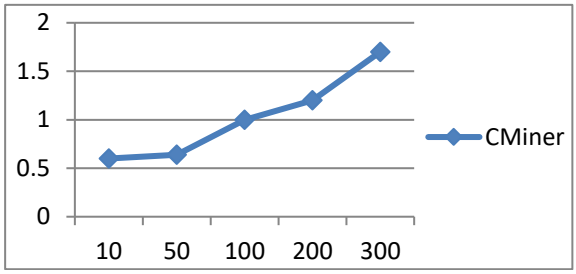

Fig 6: Computational graph of CMiner algorithm for camera dataset

The graph in fig 6 shows the computational time for the CMiner algorithm to mine top-k competitors for different sets of camera dataset.

In the below fig 7, it clearly shows the time taken by each algorithm for a set of competitors such as 10, 50 etc. The time taken by each algorithm is clearly mentioned in the above Table 7.4. Therefore it shows that CMiner algorithm takes lesser time when compared to naïve and GMiner algorithms.

Table 7.4: Comparative table of time taken for computation for camera dataset

	Time (sec)	**Time (sec)**	**Time (sec)**
Number of competitors	**Naive**	**GMiner**	**CMiner**
10	1.2	1.1	0.6
50	1.3	1.3	0.64
100	1.4	1.33	1
200	1.62	1.35	1.2
300	1.82	1.8	1.7

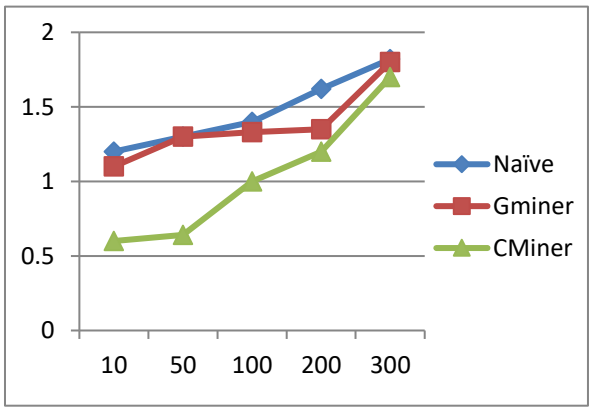

Fig 7: Computational delay graph of different algorithms for camera dataset

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 9 Issue 07, July-2020

The CMiner algorithm allows large data sets and the computational delay time to mine top-k competitors is less compared to the other algorithms.

8. CONCLUSION

A formal definition of competitiveness between two things which was approved both quantitatively and subjectively was defined. The formalization is pertinent crosswise over areas, overcoming the issues of past methodologies. The proposed system is efficient and pertinent to spaces with extremely expansive populaces of things. The efficiency of the system was verified by means of a test assessment on genuine datasets from various spaces. The tests likewise uncovered that just few surveys is sufficient to confidently assess the diverse sorts of clients in a given market,as well the quantity of clients that have a place with each kind.

REFERENCES

[1] C. H. Hsieh, C. H. Yan, C. H. Mao, C. P. Lai, and J. S. Leu, "GMiner: Rule-Based Fuzzy Clustering for Google Drive Behavioral Type Mining", 2016 International Computer Symposium (ICS), Volume: 1, pp. 98-103, 2016.

[2] Dr. P. Banumathi, "Finding Top -K Frequent Item Set On Online Shopping", Volume 03, Issue 03; March - 2017 [ISSN: 2455-1457], 2017.

[3] G. Valkanas, T. Lappas, and D. Gunopulos, "Mining Competitors from Large Unstructured Datasets", IEEE Transactions on Knowledge and Data Engineering, Volume: 29, Issue: 9 , pp. 1971-1984, 2017.

[4] H. T. Kung, E. Luccio, and E. P. Preparata,"On finding the maxima of a set of vectors. Journal of the ACM", Volume: 22, Issue: 4, pp. 469- 476, 1975.

[5] J. D. Lafferty, A. McCallum, F. C. N. Pereira, "Conditional random fields: probabilistic models for segmenting and labeling sequence data", in: C. E. Brodley, A. P. Danyluk (Eds.), Proceedings of the Eighteenth International Conference on Machine Learning, San Francisco, CA, USA, 2001, pp. 282–289, 2001.

[6] K. Xu, S. S. Liao, J. Li, and Y. Song, "Mining comparative opinions from customer reviews for Competitive Intelligence", Decision Support Systems, Volume: 50, Issue: 4, pp. 743–754., 2011.

[7] L. Page, S. Brin, R. Motwani, T. Winograd, "The pagerank citation ranking: bringing order to the web", In: WWW (1998), 1998.

[8] M. Bergen, and M. A. Peteraf, "Competitor identification and competitor analysis: a broad-based managerial approach. Managerial and Decision Economics", Conversations on the dynamics, context, and consequences of strategy, Volume: 23, Issue: 4-5, pp. 157-169, 2002.

[9] R. Li, S. Bao, J. Wang, Y. Yu, and Y. Cao, "CoMiner: An Effective Algorithm for Mining Competitors from the Web", Sixth International Conference on Data Mining ICDM-2006, pp. 948-952, 2006.

[10] S. B. orzs¨onyi, D. Kossmann, and K. Stocker, "The skyline operator", in ICDE, Proceedings 17th international conference on data engineering, Volume: 1, pp. 0421, 2001.

[11] S. Morinaga, K. Yamanishi, K. Tateishi, and T. Fukushinna, "Mining product reputations on the web", In KDD-02: proceedings of the eighth ACM SIGKDD international conference on knowledge discovery and data mining, pp. 341-349, 2002.

[12] Shenghua Bao, Rui Li, Yong Yu, and Yunbo Cao, "Competitor Mining with the Web", IEEE Transactions on Knowledge and Data Engineering, Volume: 20, Issue: 10, pp. 1297-1310, 2008.

[13] TN. Doan, F. C. T. Chua, EP. Lim,"Mining Business Competitiveness from User Visitation Data", Social Computing, Behavioral-Cultural Modeling, and Prediction. Volume: 9021, pp. 283-289, 2015.

[14] Y. Liu, W. Liao and A. Choudhary ,"A two-phase algorithm for fast discovery of high utility of item sets" PAKDD-2005, Advances in knowledge discovery and data mining, pp. 689-695, 2005.