Mr. Sakshi Siva Ramakrishna, 2025, 13:6
ISSN (Online): 2348-4098
ISSN (Print): 2395-4752

International Journal of Science, Engineering and Technology
An Open Access Journal

Tripartite Graph-guided Analysis to Build Recommender Systems

[1]Mr. Sakshi Siva Ramakrishna, and [2]Dr. T Anuradha
[1]Department of Cse,Acharya Nagarjuna University, India
[2]Department of CSBS,RVR&JC College of Engineering, INDIA

Abstract - Business transactions generate a large amount of data that requires thorough analysis to provide vital insights for businesses' decision-making.A transaction is linked with a good number of record attributes. Traditional transactional data analysis treated all attributes equally, but a subset of attributes can distinguish fine-grained transactions. This discrimination is achievable through attribute and transaction weighting. User surveys or empirical data are the sources for weighting data attributes. If user views are unavailable or the empirical study is missing, weighting becomes tough. Some effective methods exist for weighting. When attributes are binary-valued, the weighting process should rely on transaction-item relationships. The HITS (Hypertext Induced Topic Search) algorithm can perform this weighting. A new algorithm called "Bijective HITS" is proposed, capable of weighing transactions and items by mapping transaction-item relations to item-feature relations. This two-level processcan identify important transactions and items. A new distance measure named "W-distance" is derived from this weighting process. Additionally, a link and density-based hierarchical clustering method is proposed to cluster transaction data using only binary information. Experiments conducted with real and hypothetical datasets compare the results of this approach with those of existing well-known methods. The findings indicate that the proposed models outperform the compared processes, offering better tools for implementing recommendation systems.

Keywords - Tripartite graph, Link based weighting, Bijective HITS, W-distance, Recommender systems.

I. INTRODUCTION

Analysis of transaction datasets provides rich knowledge in terms of association rules, transaction patterns and grouping, models from training, and so on. This analysis reveals implicit interactions and similarities in data. It has turned into a prospering research topic in data mining and analytics, which has abundant practical applications, including classification, recommendation systems and clustering. The classical models of data mining treat every transaction equally. But different transactions may have different weights in existence. For example, market basket transactions with different combinations of items should be considered with varied importance. Online business customers need to be differentiated or classified by their buying practices and patterns. An e-health provider should classify the patient data based on physical, demographic, and medical observations. Most of the current methods ignore the transaction-wise importance while applying the mining and analytical procedures, as the majority of the data types have no preassigned importance. For numerical attributes, there are some ways to get feature weights. User preferences can be collected through various surveys and observations and the same can be converted to weights.

But a customer or user may deviate while giving his/her votes or preferences. Some user responses have a chance of bias. For categorical and binary attributes, preassigned or derivedweights are either hard to get or missing. Some notion of importance in those data ought to be considered. For example, transactions with several items and high quantity should be thought more precious than transactions with items less in quantity and number. An item with more desirable features should be treated as more

important. Current methods are not able to get such estimates by emphasizing the important transactions and features.

In this paper, a three-way cyclic procedure to measure the weights of transactions, items, and features as well in datasets with only binary attributes. The basic idea of this three-way weighting is that a transaction has more weight when it has more weighted items and an item has a good weight when it has more desired features. A feature is precious when it is desirable in more items. Such weights are totally derived from the internal relationships of datasets based on the assumption that good transactions consist of good items and good items generally have more good features. Kleinberg's HITS model and algorithm [11][22]to bipartite graphs is extended to tripartite graphs to exploit the present idea of three-way weighting.

Therefore, three-way weighting is distinct from preassigned weighted models and is useful when user defined weights are not available. Furthermore, a new measurement framework of data clustering from derived weights is proposed. Experimental results show that weights can be derived without much overhead, and interesting clusters may be discovered through this new measurement to implement better recommender systems. The organizationof the paper is further as follows: First, the foundations of weighted clusteringare discussed. Next, it is presented about the evaluation of transactions and items with Bijective HITS is presented, followed by the definition of w-distance and the corresponding clustering algorithm. A popular real-life example is exercised to explain the proposed algorithms and to explore experimental results on different types of data. The last section comes with concluding remarks.

II. WEIGHTED CLUSTERING

The assertion that clustering algorithms primarily analyze unweighted data is generally true, as many fundamental algorithms operate under the assumption that all data points have equal importance, or "weight". However, the field of weighted clustering is an area of active research that has emerged to handle situations where data points have varying levels of importance or influence. [1]. Feature selection and feature weighting lifted the concept of weighted clustering [19].A weight function assigns a weight w(x) Ɛ R+ for a feature or attribute x Ɛ X, a feature set. Many real-life datasets generally have features with preferences and these preferences are the weights of those features [13]. Weighted clustering deals with weights of features while clustering dataset. This weight is incorporated with distance metric to find dissimilarities between objects. The weights of features are easy to use when they are pre-assigned. This pre-assignment is explicit [17] or implicit. The weight decision can be made by surveying among the product or service users. There are variety of methods to get weights through surveys like rank order weighting [9], decision-tree based attributeweighting [7], Analytical Hierarchy Process (AHP)[16] [20],SMART[18].

In conclusion, the methodology of weighted clustering is to assign weights to features, invent new measures (weighted distance) based on these weights, and develop the consequent algorithms to analyse data. When the survey is hard to perform or when the user preferences are not available for analysis the above-mentioned methods are not applicable. For datasets with only binary attributes, some weighting procedures need to be developed. Alinking process on item ranking can be found in [24]. This link-based model based on HITS process,is applied on the graph representing transactional data to rank the items, where all nodes and links can have weights.

Ranking Transactions and Items with Bijective HITS

Tripartite graph: Let m≥0 and m≥0 be integers, a finite simple graph G (V, E) with m vertices V={v1,v2,….,vn} and m edges E={ e1,e2,…..,em}. If the vertex set V is decomposed into three disjoint sets such that no two graph vertices within the same set are adjacent, then G is called a Tripartite graph. If every vertex vof a set is adjacent to every vertex in the other two sets then G is called a complete tripartite graph. Assume a dataset D of m transactions(T), n possible items (I)and o possible features(F) where T= {t1,t2,….,tn},I ={i1,i2,….,in},and F

={f1,f2,….,fn}.Then D can be visualized as a tripartite graph G=(D,I,F,E),whereE={(t, i ,f):t Ɛ T, i Ɛ I , f Ɛ F }.Here, E is a composite mapping edge of two simple mappings (t, i) and (i, f).

Example 1. Consider the dataset shown in Table 1. It can be equivalently represented as a tripartite graph, as shown in Fig. 1.

Transaction ID	Transaction(s)
101	{1,3,5}
102	{1,2,3,5}
103	{4}
104	{3,4,5}
105	{3}

Table 1a: A dataset of Transactions with items.

Item	Feature(s)
1	{3}
2	{1,4}
3	{4}
4	{1,4,5}
5	{2,3,5}

Table 1b: A dataset of items with features.

Figure1: The tripartite graph representation of a dataset.

The representation of the transaction dataset is graphical format is motivating. This composite mapping gives an idea of applying ranking transactions, items and features based on links of the mappings. In this tripartite graph, the degree associated with a transaction/item/feature, shows the differentiation of one transaction/item/feature with other. A classical analysis of data mining does not considersuchdifferencesbetween transactions/items/features. Still, it is important to have different weights for different transactions /items/features tosignify their uniquestatus. These weights are used to derive further weights to analyse the dataset. At this point a significant question is how to acquire weights in a dataset with only binary attributes. Innately, a transaction has good weight, when it contains good items; at the same time, a good item is a part of many good transactions. This reinforcing relationship does also exist between items and features. These relationships are just like the relationship between hubs and authorities in the HITS model. Relateto the transactions as first level hubs and the items as authorities as well as second level hubs, and the featuresas pureauthorities,Bijective HITS(two level HITS) can be applied to this tripartite graph.

The iterations of Bijective HITS make use of the following equations:

Authority(f)=∑_(I:fϵI)▒Hub1(I)

Hub1(I)=∑_(f:fϵI)▒Authority(f)

Hub2(T)=∑_(i:iϵT)▒Hub1(i)

Here Hub1 and Hub2 are two functions at item and transaction levels, respectively. Hub2 is a composite mapping composed by two functions Hub1() and Authority (). When thisBijective HITS model eventually converges, the hub level wise weights of all transactions and items are obtained. The potential of transactions and items to contain high-value items and high-value features can be assessed through these weights. A transaction (item) witha small number of items(features) may however be a good hub if all component items (features)are better ranked. Equally, a transaction (item)with many normal items(features) may have a low hub weight. Based on these properties and observations, we can derive weights for transactions/items/features in a level wise manner, as the mapped-HITS algorithm demonstrated in Algorithm 1.

Algorithm 1: Bijective HITS

*1)*Initialize auth(f) to 1 for each feature f
2)**for** (it=0; it < num_it; it++) **do begin**
3) authd(f)=0 for each feature f

4) hub1(i)=0 for each item i
5) hub2(t)=0 for each transaction t
6) **for all** items$i \in D$ **do begin**
7) hub1(i)=$\sum_{f:f\in i} auth(f)$
8) authd(f)+=hub1(f) for each feature f
9) **end**
10) **for all** transactions $t \mathcal{E}$ D **do begin**
11) hub2(t)=$\sum_{i:i\in t} auth(i)$
12) **end**
12) auth(f)= authd(f) for each feature f, normalize auth
*13)***end**

This section has demonstrated the application of the Bijective HITS algorithm to the ranking of the transactions and items with the information fed back form features of items. When the iterative procedure converges the significant transactions and items can be identified based on the ranking (weights)outputs. If a transaction database, when converted to a tripartite graph representation an interesting observation can be found as mentioned in the following lemma.

Lemma: A transaction database is equivalent to a complete tripartite graph if and only if all the transactions /items/features are equally ranked with weight 1.

Proof: Let the tripartite graph representing the transaction database is complete.

- All hub counts as well as authority counts show an equal and complete number
- The convergence procedure ends with unit weights for transactions /items/features.
- All the ranks are equal with weight 1.
- Conversely assume that, all transactions /items/features are equally ranked with weight 1
- All hub counts as well as authority counts show an equal and complete number
- Two sets of nodes of the tripartite graph have complete inter set links.
- The tripartite graph representing the transaction database is complete.

This observation intuitively says that a transaction/item weight is a weighted sum of its belonging parts(items/features). That is a good transaction/item is having good number of better weight items/features. Conversely better items/features are associated with good transactions/items.

When a transaction database is binary valued, it can be equivalently interpreted as a tripartite graph. Applying Bijective HITS to this tripartite graph provides weights to the database elements(transactions/items).

Example 2. Consider the dataset shown in Table 1 again. The Bijective HITS iteration gives the weights of each transaction /item /feature, as shown in Table 2. It is interesting to point out that the highest weighted transaction is 104 [3,4,5]is not the one with the best item number, and the most significant item is item 4 which is not the one with the largest support of transactions. This shows the significance of link-based measurement. Feature 4 got highest ranking as it is a part of two better ranked items [2,3,4]. The existence of cross relationships among transactions, items, and features reinforce the weights.

Transaction:	1	2	3	4	5
Weight	0.27	0.08	0.09	0.33	0.23
Item:	1.00	2.00	3.00	4.00	5.00
Weight	0.04	0.27	0.15	0.37	0.18
Feature:	1.00	2.00	3.00	4.00	5.00
Weight	0.17	0.29	0.17	0.31	0.07

Table2 Weights of transactions /items/features of the Example Database

3 W-distance: A new distance measure

Classical data clustering algorithms find dissimilarity between two patterns using a distance measure defined on the feature space[8].Euclidean distance is a popular measure to find such dissimilarities. The dominance of some features generally rules the dissimilarity between the objects/patterns. To normalize such domination standardized Euclidean distance [23] come into picture. When feature variables are on different measurement scales, some balance mechanism is needed. The Euclidean distance computed on standardized variables is called the standardized Euclidean distance. This standardization in the distance calculations can be interpreted as weighting of variables.

A distance measure with any choice of weights is called weighted distance. Chi-square distance [4] is one such weighted distance for count variables. For binary attributed data more than 70 distance measures are used over the last century. Jaccard, dice, Hamming, cosine, and their variants are popular [3]. In this paper a new distance measure named W-distance if proposed for binary attributed data. Thisdistance measure makes use of weights derived from link-based weighting introduced in the previous section. W-distance is a count and weight-based measure,and it is used to formulate weighted clustering in terms of this new concept. Based on the weights derived from mapped-HITS, acount and weight-based distance called W-distance between to objects Aand Bis formulated as:

W-distance(A, B) = ∑_i⟦ (w"i" * m"i")⟧ ,

where wiis the weight ofithattribute and mi =0 or 1 based on matched binary value of attribute pair at position ifrom objects A and B.

mi={(0,&binary values of i th position are equal@1,&binary values of i th position are unequal)┤

Example 3. Consider two objects A (1,0,1,0,1) and B (1,1,1,0,1) with weighted vector= (0.0416948, 0.26531243, 0.14638256, 0.36823552, 0.17837460).

W-distance(A,B)=1-0.0416948*0+0.26531243*1+0.14638256*0+0.36823552*0+0.17837460*0=0.26531243,and similarity (A, B) =1-0.26531243=0.73468757.

Using the weights, the distance matrix based on W-distance is given in Table 3.

Transaction	101	102	103	104	105
101	0.00	0.27	0.73	0.41	0.22
102	0.27	0.00	1.00	0.68	0.49
103	0.73	1.00	0.00	0.32	0.51
104	0.41	0.68	0.32	0.00	0.55
105	0.22	0.49	0.51	0.55	0.00

Table3 W-distance matrix for transactions

Now the similarity matrix for the example transactions is given in Table 4.

Transaction	101	102	103	104	105
101	1.00	0.73	0.27	0.59	0.78
102	0.73	1.00	0.00	0.32	0.51
103	0.27	0.00	1.00	0.68	0.49
104	0.59	0.32	0.68	1.00	0.45
105	0.78	0.51	0.49	0.45	1.00

Table4 similarity matrix for the example dataset transactions.

From the above table it can be observed that, transactions 102 and 103 have zero similarity as one is the compliment of the other with respect to the items they contained.

Here we can observe that:

i)W-distance (A, A) =0

ii) W-distance (A, B) ≥0

iii) W-distance (A, B) = W-distance (B, A), and

iv) W-distance (A, B) +W-distance (B, C) ≥W-distance (A, C)

The W-distance introduced here follows the properties of a metric space [6]and hence, it is a distance metric.

A link and density-based clustering algorithm

Data clustering algorithms generally follow two types of approaches, namely partitioning algorithms, and hierarchical algorithms[2]. As hierarchical algorithms are mostly suitable for arbitrary data, in this paper a hybrid clustering is proposed which follows mixed properties of hierarchical type clustering algorithms and density-based algorithms [5] [12] as well. Here a clustering link is started with highest similar pair ofobjects. This pair extends with adding more objects with equal similarity (if exist), and with a pre-defined similarity. This pre-defined similarity decides the density of a cluster. The kink(links) timed out when there are no more elements to extend the current cluster. Next a new pair of objects with next largest possible similarity starts a new cluster, and the process continues to form another cluster. In this way the initial level clusters are formed. Now the clusters formed in the first level clustering act as objects and more larger clusters are formed using these sets of clusters. Single linkage hierarchical clustering [14]is followed

to merge the clusters into next level clusters. This hierarchical and density-based clustering ends when there are no more objects to merge with the pre-defined threshold value.

Example 3. Consider the similarity matrix presented in Table 3, of the example dataset. The average similarity is 0.5858712(pre-defined threshold). The pair(101,105) is one with highest similarity of 0. 7799305(>pre-defined threshold). Therefore, a first level cluster object is formed with these two objects. The second row have next highest similarity with already clustered object 101, and so the process moved to third row. The next highest similarity is held with the third row and the similarity pair is (103,104) forming another cluster object at the same level.

The next level similarity matrix is as shown in Table 5.

	[101,105]	[103,104]	[102]
[101,105]	1.0000000	0.2653124	0.7346876
[103,104]	0.2653124	1.0000000	0.0000000
[102]	0.7346876	0.0000000	1.0000000

Table5 Second level similarity matrix for the example dataset transactions.

The average value of this similarity matrix is 0. 5555.Setting this as second level threshold and continuing the same process, the next level clustering ends the process with the following clusters: Cluster1: [101,105,102] and Cluster2: [103,104]. This proposed process named "A link and density-based clustering algorithm" is presented inAlgorithm 2.

Algorithm 2: A link and density-based clustering algorithm

Input: A similarity matrix M for n objects.

Output: Clusters of the objects

Process:

level L=1.
Find the average of M and set it as level L threshold th(L).

Locate the row r with highest similar pair (r, c) , r! =c (with similarity value s) in M and form this pair as one of the level L cluster. Search the row r for same similarity value s, and if exist add the corresponding column(s) object(s) to the current cluster. For each added column object find the th(L)-density reachable object to the current cluster. Recursively search the rows with added object indexes to add more density reachable objects. This linking end when no more reachability is possible.

Locate the next highest unlinked row index to form more clusters at the same level. This level of process ends when all rows of the matrix are either linked to a cluster or stand as a single element cluster.

The sets of clusters are now the candidates for further level clustering. Find the similarity matrix CM for clustered objects. Here each row represents a cluster formed at previous level.
L=L+1
Find the average of CM and set it as level L threshold th(L).
Use this threshold to follow the same procedure as in steps 2,3, and 4 to get the further level clustering. The process can be terminated by posing a limit on the size of the level L similarity matrix.

A real-lifeexample dataset processing

In this section, the usefulness of derived weights is demonstrated through an intuitive example. Good number of popular movie recommendation system s are studied to understand the nature of datasets.The dataset of MovieLens is used, which consists of 105339 ratings applied over 10325 movies with 668 users. Each movie has some features. Movies are rated with a rating scale up to 5. Some movies are not rated. A movie has zero or more ratings given by multiple users [15].

The data is pre-processed into two binary tables. One is about user versus ratings, and the other is about movie versus features. The first table records whether a user rated a movie or not. The second table records whether a movie has a feature or not. This data is processed with the Bijective-HITS algorithm. Samples of the results of the Algorithm 1.

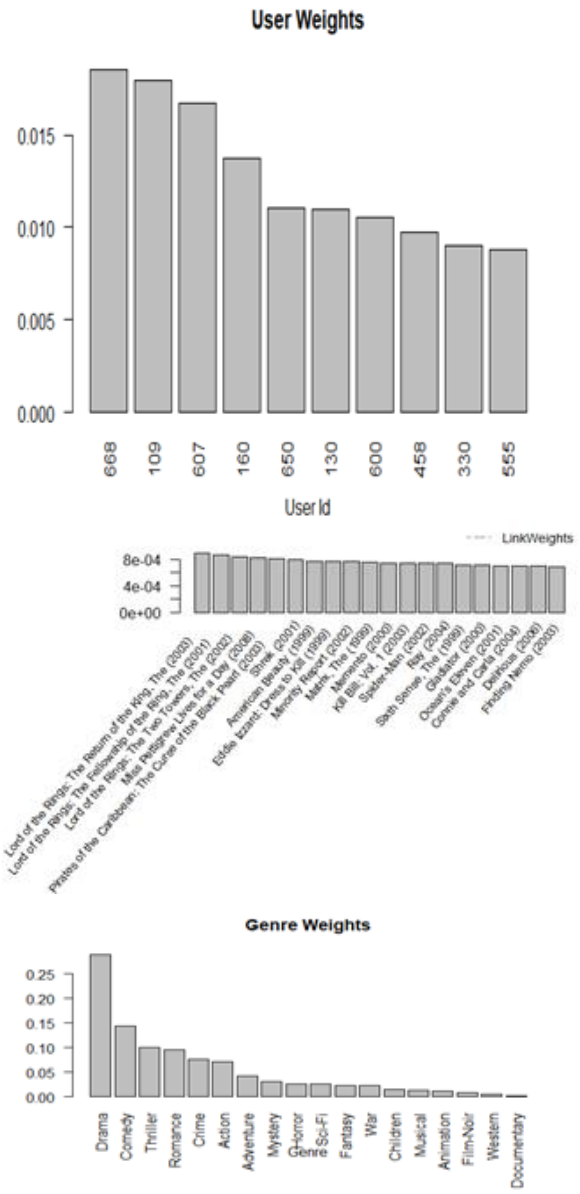

Figure 2: Results of mapped-HITS: Weights of top 20 users, top 20 movies, and 18 genres.

Here top users who rated more movies are elevated. The movie American Beauty (1999) with Movie Id 2858 got highest weightage. The dominated genre is Drama followed by comedy.

The Bijective-HITS weights (link weights) of top 20 most popular movies along with their popularity weights (count weights) are presented in Figure.3. It is assumed that the popularity of a movie is proportional to its number of ratings but not to the actual rating values. Here count weight of a movie is the ratio of number of ratings of a movie to total ratings of all movies.

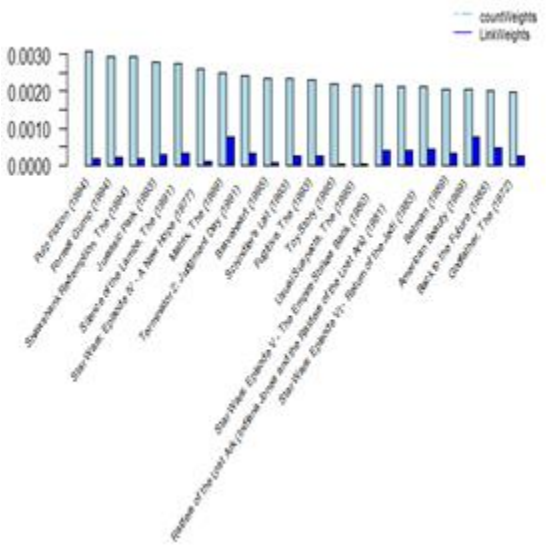

Figure 3: Count weights versus derived weights.

From the above graph it can be observed that some of the popular movies are not having good weights. That is a popular movie may or may not has a dominated feature. At the same time, some popular movies may not be rated properly. This is most probably due to the ratings given by non-active users in terms of movie rating habit. One can see that the count weights simply counts the number of ratings of each movie. In contrast, the link weight differentiates the roles of different viewers and genres of movies. These differences, pointing the need and importance of derived weights based on other means rather than the actual ratings. The is one of the motivating things behind the design and application of proposed Bijective-HITS approach.

Many of the transactions have item information and ratings of items are not available. Items have features based on which items can be evaluated and can be weighted. These item weights implicitly rate transactions. This inter relationship among transactions, items, and features pave the way for deriving weights/ratings where actual rating information is absent. Therefore, the Bijective-HITS algorithm proposed here provisions weights derivation. These weights also provide means for recommendation systems to recommend products

or services to users. The sparsity and cold start problems of recommendation systems [21][10], can be resolved to some extent based on these derived weights. To do so the transactional data must be grouped based on derived weights, and the grouping shows the way to fill the gaps in data with most probable replacements. The top 50 movies in the order of derived movie weights of MovieLensdataset are clustered using the Algorithm 2. The results are shown in Figure.4. There are five clusters formed from the process. The clustering is done based on the proposed w-distance measure, which considers the derived weights of features of the movies.

```
cluster 1
 [1] "Batman (1989)"                          [2]"Die Hard (1988)"
 [3] "Silence of the Lambs, The (1991)"       [4]"Usual Suspects, The (1995)"
 [5] "Jurassic Park (1993)"                   [6]"Independence Day (a.k.a. ID4) (1996)"
 [7] "Matrix, The (1999)"                     [8]"Terminator, The (1984)"
 [9] "Mission: Impossible (1996)"             [10]"Rock, The (1996)"
[11] "Speed (1994)"
cluster 2
 [1] "Toy Story (1995)"                                  [2] "Aladdin (1992)"
 [3] "Ace Ventura: Pet Detective (1994)"                 [4] "Back to the Future (1985)"
 [5] "Lord of the Rings: The Fellowship of the Ring, The (2001)"
 [6] "Lord of the Rings: The Two Towers, The (2002)"     [7] "Batman Forever (1995)"
 [8] "Men in Black (a.k.a. MIB) (1997)"                  [9] "Princess Bride, The (1987)"
[10] "Shrek (2001)"                                      [11] "True Lies (1994)"
[12] "Star Wars: Episode IV - A New Hope (1977)"
[13] "Star Wars: Episode V - The Empire Strikes Back (1980)"
[14] "Star Wars: Episode VI - Return of the Jedi (1983)"
[15] "Stargate (1994)"
[16] "Raiders of the Lost Ark (Indiana Jones and the Raiders of the Lost Ark) (1981)"
[17] "Indiana Jones and the Last Crusade (1989)"                  [18] "Aliens (1986)"
[19] "Terminator 2: Judgment Day (1991)"
cluster 3
[1] "Pulp Fiction (1994)"                    [2] "Fargo (1996)"
[3] "Fight Club (1999)"                      [4] "Shawshank Redemption, The (1994)"
[5] "Godfather, The (1972)"                  [6]"Forrest Gump (1994)"
[7] "Mrs. Doubtfire (1993)"                  [8] "American Beauty (1999)"
cluster 4
[1] "Dances with Wolves (1990)"              [2] "Apollo 13 (1995)"
[3] "Lion King, The (1994)"                  [4] "Sixth Sense, The (1999)"
[5] "Schindler's List (1993)"                [6] "Braveheart (1995)"
[7] "Saving Private Ryan (1998)"
[8] "Lord of the Rings: The Return of the King, The (2003)"
[9] "Gladiator (2000)"
cluster 5
[1] "Twelve Monkeys (a.k.a. 12 Monkeys) (1995)" [2]"Seven (a.k.a. Se7en) (1995)"
[3] "Fugitive, The (1993)"
```

Figure 4: Clustering result of Top 100, weighted Movies.

From the list of top 50 list of weighted movies it can be observed that, most of these movies are in the list of top-rated movies shown in Fig.6.The clustering result reveals that, cluster1and cluster3 have more top-rated movies. These observations support the strength of feature weights and the application of w-distance proposed in this paper.As this clustering is guided by features of movies, one can find a place in a cluster, for a new movie which has no user rating information yet, and then find a set of users to recommend that movie. In this way, the proposed clustering based on derived feature weights, provides means to improve recommendations.

Experiments

Performance Study

To evaluate the Bijective-HITS, W-distance, and the associated clustering framework, tests have been carried out on three hypothetical data sets named D100, D250, and D500. Each dataset provides transactional information of 8 items(brands) of a product type. Each brand differs with other with respect to existence of a feature (0 or 1). Each transaction represents the existence of an item(brand) (0 or 1). All code was compiled using R Version 4.0.0.

The proposedBijective-HITSprocedure needs a scan of data at each iteration. The convergence rate of the of the proposed process is critical to the performance. Let Hi denote the hub weight of an item after the ith iteration. Fig. 5 shows | Hi – Hi+1| as a function of i on the data sets in log scale. The figure explains convergence information with respect to three datasets. Three curves of the graph reveal the fast convergence trend of the iterations.

Now it is clear that the proposed weighting process (Bijective HITS) converges fast on transaction databases. Mostly, three or four iterations brings good convergence, which means that the proposed Bijective-HITSmethod works well with only three or four additional database scans than traditional procedures.

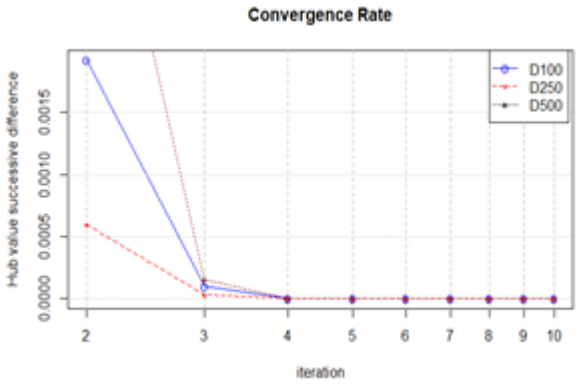

Figure5:Mapped-HITS convergence rate.

5.2 Comparison of countWeight and LinkWeight
Using the Bijective-HITS procedure, link weights are derived for the dataset D500.These weights are compared with the weights of each item counted as the size of item membership in transactions (count weight). This comparison is shown in Figure.6.

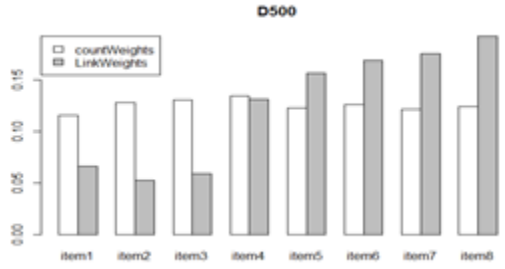

Figure 6: Comparison of Count weights, derived weights for the dataset D500.

The derived weights of some items are significantly higher than their counts, and some items have more counts than their weights. This difference is due to reinforcing relationship between transactions and items. Therefor the proposed weighting algorithm can differentiate importance items form the set of items and important transactions from the set of transactions. This differentiation picks weight vectors, that can be used to evaluate new items or transactions.

W-distance and associated clustering
The w-distance proposed in this paper is used in link-based clustering proposal. Three datasets are processed with the proposed algorithm. The results are furnished in Table.6. Each cluster is evaluated by testing the membership of each cluster element with the cluster neighborhood (μ-2σ,μ+2σ),where μ and σ are mean and standard deviations of distances from center to each element of a cluster. Purity of a cluster is defined as the percentage of elements of a cluster that fall in the specified neighborhood.

Dataset	Number of clusters	Size of cluster	Share of cluster	Cluster purity
D500	3	77	0.154	0.96104
		190	0.38	0.91579
		233	0.466	0.94421
D250	4	95	0.38	0.94737
		67	0.268	0.95455
		51	0.204	0.96078
		39	0.156	0.94737
D100	3	27	0.27	1
		40	0.4	0.975
		33	0.33	1

Table6. Evaluation of clustering results

The clustering result reveals that, the purity of clustering is near to 95% which is a sigh of good performance of proposed clustering with significant intra cluster homogeneity. This homogeneity is influenced by the size of individual clusters and number of clusters. The proposed clustering is compared with state-of-the-art existing clustering method. The comparison of results is presented throw a graph in Figure.7.

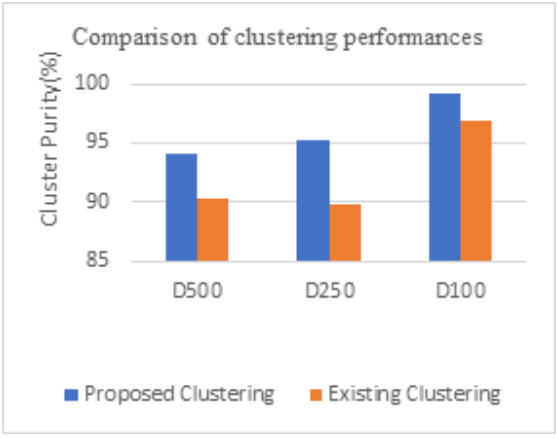

Figure7: Proposed versus Existing clustering.

Proposed method of clustering performed well than existing method. One interesting observation is that the purity of clustering is inversely proportionalto the size of data(number of transactions). But in all data cases the proposed method provided better performance in terms of cluster purity.

III. CONCLUSIONS AND DISCUSSION

A new way of weighted data clustering is presented in this paper. First, a Bijective HITS model and algorithm based on tripartite graph structure are used to weight transactions, items, and item features from a dataset with only binary attributes. These weights are used to upraise the respective objects (transactions/items/features). These weights are different from traditional counts of items or transactions or features. Some significant items/features that are not so high in counts can be found in the model. Based on the significance of items/features a new way of distance measure W-distance is derived to group objects.

A density and link-based clustering is proposed to group data objects based on the derived distance measure. A real-life data example is exercised to test the proposed algorithms. Three hypothetical datasets representing the relationships among transactions, items, and features are also exercised to compare the performance of the proposed algorithms with state-of-the-art existing procedures and better results are found with the proposed algorithms. Through comparison, it is found that the proposed processes and methods highlighting high-quality transactions and items. The transaction and item weighting methodologiesare precious when datasets only provides binary information.This paper also suggests means to solve the familiar problems of recommendation systems.

REFERENCES

1. Amin Golzari Oskouei, Negin Samadi, Shirin Khezri, Arezou Najafi Moghaddam, Hamidreza Babaei, Kiavash Hamini, Saghar Fath Nojavan, Asgarali Bouyer, Bahman Arasteh,Feature-weighted fuzzy clustering methods: An experimental review,Neurocomputing,Volume 619,2025,129176, ISSN 0925-2312, https://doi.org/10.1016/j.neucom.2024.129176.
2. Ahalya, G., & Pandey, H. M. (2015, February). Data clustering approaches survey and analysis. In 2015 International Conference on Futuristic Trends on Computational Analysis and Knowledge Management (ABLAZE) (pp. 532-537). IEEE.
3. Choi, S. S., Cha, S. H., & Tappert, C. C. (2010). A survey of binary similarity and distance measures. Journal of Systemics, Cybernetics and Informatics, 8(1), 43-48.
4. Emran, S. M., & Ye, N. (2002). Robustness of Chi-square and Canberra distance metrics for computer intrusion detection. Quality and Reliability Engineering International, 18(1), 19-28.
5. Ester, M., Kriegel, H. P., Sander, J., & Xu, X. (1996, August). A density-based algorithm for discovering clusters in large spatial databases with noise. In Kdd (Vol. 96, No. 34, pp. 226-231).
6. Gruenhage,G.(1984).Generalized metric spaces. In Handbook of set-theoretic topology (pp. 423-501). North-Holland.
7. Hall, M. (2006, December). A decision tree-based attribute weighting filter for naive Bayes. In International Conference on Innovative Techniques and Applications of Artificial Intelligence (pp. 59-70). Springer, London.
8. Jain, A. K., Murty, M. N., & Flynn, P. J. (1999). Data clustering: a review. ACM computing surveys (CSUR), 31(3), 264-323.
9. Jia, J., Fischer, G. W., & Dyer, J. S. (1998). Attribute weighting methods and decision quality in the presence of response error: a simulation study. Journal of Behavioral Decision Making, 11(2), 85-105.
10. Kenthapadi,K.,Le,B.,&Venkataraman,G. (2017, August). Personalized job recommendation system at linkedin:Practical challenges and lessons learned. In Proceedings of the Eleventh ACM Conference on Recommender Systems (pp. 346-347).
11. Kleinberg, J. M. (1999). Authoritative sources in a hyperlinked environment. Journal of the ACM (JACM), 46(5), 604-632.

12. McInnes,L.,Healy,J.,&Astels,S.(2017). hdbscan: Hierarchical density-based clustering. Journal of Open-Source Software, 2(11), 205.
13. Modha, D. S., & Spangler, W. S. (2003). Feature weighting in k-means clustering. Machine learning, 52(3), 217-237.
14. Mogotsi, I. C., Manning, C. D., Raghavan, P., & Schutze, H. (2010). Introduction to information retrieval. Information Retrieval, 13(2), 192-195.
15. MovieLens https://drive.google.com/file/d/1Dn1BZD3YxgBQJSIjbfNnmCFlDW2jdQGD/view
16. E. Bugingo, E. Leone Ndimubenshi, C. Tshimanga Kamanga, F. Xavier Rugema, O. Habimana, and J. Batamuliza, 'Application of AHP in Decision-Making: Case Studies and PracticalImplementation', Business, Management and Economics. IntechOpen, Dec. 27, 2024. doi: 10.5772/intechopen.1006966.
17. Pena, J., Nápoles, G., & Salgueiro, Y. (2020). Explicit methods for attribute weighting in multi-attribute decision-making: a review study. Artificial Intelligence Review, 53(5), 3127-3152.
18. Poyhonen, M. (1998). On attribute weighting in value trees. Helsinki University of Technology, Systems Analysis Laboratory.
19. Ramakrishna, S. S., & Anuradha, T. (2018). An Effective Framework for Data Clustering Using Improved K-Means Approach. International Journal of Advanced Research in Computer Science, 9(2).
20. Saaty, T. L. (2008). Decision making with the analytic hierarchy process Decision making with the analytic hierarchy process. International Journal of Services Sciences, 1(1), 83-98., 1(1), 83–98.
21. Sharma, L., & Gera, A. (2013). A survey of recommendation system: Research challenges. International Journal of Engineering Trends and Technology (IJETT), 4(5), 1989-1992.
22. Sun, K., & Bai, F. (2008). Mining weighted association rules without preassigned weights. IEEE transactions on knowledge and data engineering, 20(4), 489-495.
23. Tranter, G., McBratney, A. B., &Minasny, B. (2009). Using distance metrics to determine the appropriate domain of pedotransfer function predictions. Geoderma, 149(3-4), 421-425.
24. Wang, K., & Su, M. Y. T. (2002, July). Item selection by" hub-authority" profit ranking. In Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining (pp. 652-657).