Proceedings of the International MultiConference of Engineers and Computer Scientists 2009 Vol I
IMECS 2009, March 18 - 20, 2009, Hong Kong

Combining Multiple Criteria and Multidimension for Movie Recommender System

Keittima Chapphannarungsri * and Saranya Maneeroj †

Abstract—**Most of current Recommender Systems based on Content-Based Filtering, Collaborative Filtering, Demographic Filtering and Hybrid Filtering which are concentrated on user and item entities. Many research papers are improved by pointing out either Multiple Criteria Rating approach or Multidimensional approach for Recommender System. This paper proposes an advanced Recommender System to provide higher quality of recommendations by combining the Multiple Criteria rating and the Multidimensional approaches. For the Multiple Criteria approach, this paper proposed a method that changes the way of weighting to be more suitable and also concern about the frequency of the selection movie features. To do Multidimensional approach, the Multiple Linear Regression is applied to analyze the contextual information of user characteristics. According to the experimental evaluation, the combining of Multiple Criteria Rating and Multidimensional approaches provide more accurate recommendation results than the current Hybrid Recommender Systems.**

Keywords: Recommender System, Content-Based Filtering, Collaborative Filtering, Multiple Criteria, Multidimensional

1 Introduction

Recommender Systems[1] are widely used in the Internet and help user to get the interesting information easily. For example, Amazon.com[2] recommends on many kinds of items. Many Recommender Systems based on Collaborative Filtering, Content-Based Filtering, Demographic Filtering and Hybrid Filtering[3].

Collaborative Recommender Systems predict the rating of items for target user by recognizing commonalities between users on the basis of their ratings. Content-Based Recommender Systems recommend items to a user based upon a description of the items and a profile of the user's interests[4]. Demographic Recommender Systems make recommendations based on feedback of similar demographic user's characteristic[5]. Hybrid Recommender Systems combine two or more recommendation techniques to gain better performance with fewer of the drawbacks of any individual one[4].

However, the current systems use the rating value on the items for evaluating user's preference opinions. The rating value represents the overall preference of the user. A user might express his/her opinion based on some specific features of the item. For more accurate recommendations, the users' interest in more detailed features should be considered. This problem is called *without distinction of interest problem*. However, there is a Recommender System that solves this problem by using the Multiple Criteria. The Multiple Criteria approach generates the most relevant items to a user based on the giving criteria set by the user[5].

Moreover, many Recommender Systems have covered the *without distinction of interest problem* by using Multiple Criteria[6] but it is still missing the weight of features that affect to the user's preference opinion which is called *without weight feature problem*. After that, the Recommender System, called Thai-Music was proposed to provide Multiple Criteria with weight of features. However, it weights only the highest priority component of each feature. It is going to lose the some components that affect the user's opinion[7] which is not suitable. This problem is called *unsuitable weight feature problem*.

Furthermore, the traditional Recommender Systems deal with two types of entities; users and items and do not concern about other dimensions which affect to the user preference on each item. In order to do Multidimensional Recommender System, the contextual information about the user characteristics such as where he saw the movie, when the movie was seen and with whom, are needed. There is a research paper that concentrated on Multidimensional by using Reduction-Based approach[8]. The Reduction-Based approach uses the intersection of Multidimensional data which is going to lose a lot of rating data in the *training set*. This problem is called *losing a huge of rating data problem*.

The major purpose of this paper is to propose an advanced movie Recommender System which provides

*Department of Mathematics, Faculty of Science, Chulalongkorn University, Bangkok, 10330 Thailand. Email: keittima.c@student.chula.ac.th. The author wishes to express my special thanks to the Advanced Virtual and Intelligent Computing Center, and everyone in the department of mathematics, faculty of science, Chulalongkorn University for any support.

†Department of Mathematics, Faculty of Science, Chulalongkorn University, Bangkok, 10330 Thailand. Email: saranya.m@chula.ac.th.

higher quality of recommendation results by combining Multiple Criteria and Multidimensional approaches. For the Multiple Criteria approach, this paper proposed a method that changes the way of weighting to be more suitable by weighting all the component of each feature and also concerns about the frequency of the selection movie features. To do Multidimensional approach, the Multiple Linear Regression[9] is applied to analyze the contextual information of user.

In the next section, the current Hybrid Recommender Systems are discussed. Then, the proposed method is applied to get better recommendation in Section 3. In Section 4, the implementation of the prototype system, called MoviePlanet, is described. Then the results of its evaluation are presented in Section 5. In Section 6, the derived evaluation results are discussed. Finally, the conclusion is given in the last section.

2 Related Work

In Recommender System, two basic approaches: Content-Based Filtering and Collaborative Filtering, have emerged for making recommendations. Particularly, many Recommender Systems combine two or more recommendation techniques to gain better performance with fewer of the drawbacks of any individual one.

In Hybrid Recommender System, user's preference varies and always has Multiple Criteria[5]. For example, if user (A and B) rate the same score 3 for the movie "The Matrix" , but A likes its actor and B likes its genre, so the systems conclude that they have the same tastes. Therefore, the neighbor from these systems tends to be low quality of Recommender Systems. This is called *without distinction of interest problem.* Fortunately, many Recommender Systems that solve this problem were proposed by using the Multiple Criteria.

Moreover, the current systems based on Multiple Criteria do not concern the weight of features that affects to the user's preference. For example, if two users (A and B) like the same movie feature, i.e. same genre (Action), same actor (Brad Pitt), current systems conclude that both of them will be the good neighbors for each other. However, this conclusion may not be true. If A usually selects movies based on genre but not select from the actor. The weight of genre has a higher priority than weight of actor in A's opinion. On the other hand, if B select movie based on actor then the weight of actor has a higher priority than genre in B's opinion. It can be concluded that, although each couple of users that like the same movie features, they may select the different movie. This called, *without weight feature problem.* Thai-Music[7] Recommender System figures the *without weight feature problem* out by weighting only the biggest component of each feature but the way of this weight is not suitable because other component of each feature might be lost. This problem is called unsuitable weight feature problem.

Additionally, the contextual information of user characteristics also affect to the user preference on each selected movie such as where he saw the movie, when the movie was seen and with whom. Gediminas[8] concentrated on Multidimensional by using Reduction-Based approach. The Reduction-Based approach uses the intersection of Multidimensional data. For example, if the third dimensional is a Day dimension, they use Day = "Weekday" to be intersected value for filtering the data in the database as Figure 1. It would eliminate the Day dimension by selecting only the "Weekday" rating from the set of all ratings. Then, Gediminas[8] used any of basis recommendation technique such as Collaborative Filtering to generate the recommendation. This called *losing a huge of rating data problem.*

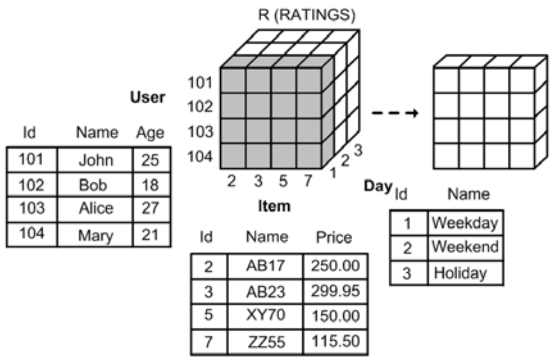

Figure 1. Current Multidimensional model

3 Proposed Method

This paper focuses on providing the higher quality of recommendations. User Profile should be represented on various necessary features and the components of each feature should be weighted in more suitably way to avoid *unsuitable weight feature problem.* Instead of weight only the biggest component of feature, this proposed method weights all the component of each features and it also take the frequency of the selection movie features into account. Moreover, to create user profile, the contextual information should be taken into consideration and use an appropriate approach to avoid *losing a huge of rating data problem.* This paper uses the Multiple Linear Regression analysis to perform the Multidimensional instead of using Reduction-Based approach.

3.1 Characteristic of Movie

3.1.1 Movie Feature Vector (MFV)

Movie data in this case are stored in a database with characteristic data for each item. The movie character-

Movie Features	*Movie Component*
Feature(1): Genre	Action(1), Adventure(2), Animation(3), Children(4), Comedy(5), Crime(6), Documentary(7), Drama(8), Fantasy(9), Film-Noir(10), Horror(11), Musical(12), Mystery(13), Romance(14), Sci-Fi(15), Thriller(16), War(17) and Western(18)
Feature(2): Release Period	2009-2005(1), 2004-2000(2), 1999-1995 and before(3)
Feature(3): Movie Award	Oscars(1), Golden Globe(2), No award(3)

Table 1: The characteristic of Movie

istics are represented in the form of Movie Features Vector (MFV) which contains 24 elements (18 elements of movie genre feature, 3 elements of year feature and 3 elements of award feature). The MFV is constructed when a new item is introduced into the system. Its characteristic is $MFV = ((f_{11}, f_{12}, ..., f_{1m_1}), (f_{21}, ..., f_{2m_2}), (f_{N1}, ..., f_{Nm_N}))$; where f_{ij} is the value that represents movie characteristics component j of feature i, m is the number of component in each feature and N is number of features. The value in the vector is presented in 0 or 1. For example, MoviePlanet Web site notified that movie name "The Matrix" has component of each feature as $Genre = Action(1)$, $Adventure(2)$, $Sci-Fi(15)$ and $Thriller(16)$, $ReleasePeriod = 1999-1995(3)$, and $Award = Oscar(1)$. It has characteristic $MFV_{Matrix} = ((1, 1, 0, ..., 1, 1, 0, 0), (0, 0, 1), (1, 0, 0))$.

3.2 Characteristic of User

3.2.1 User Preference Vector (UPV)

This vector represents a user's opinion on feature or show how much each user feels towards what features affect the selection of each movie. The UPV will automatically create for each movie every time when each user gives opinion for that movie.

To construct the UPV, the MFV is needed to transform by multiplying normalized rating value in range 0-1 toward each movie. For example, if user gives the rating value 2 (1 is dislike, 2 is neutral and 3 is like) for the movie "The Matrix", then the rating value is normalized to 0.67. After that, the transformed $MFV_{Matrix} = ((0.67, 0.67, 0, ..., 0.67, 0.67, 0, 0), (0, 0, 0.67), (0.67, 0, 0))$.

The $UPV(i)$ is the direct sum of the $transformed\ MFV$ of rated movies and divided by the number of rated movies by user (i). As shown in Figure 2.

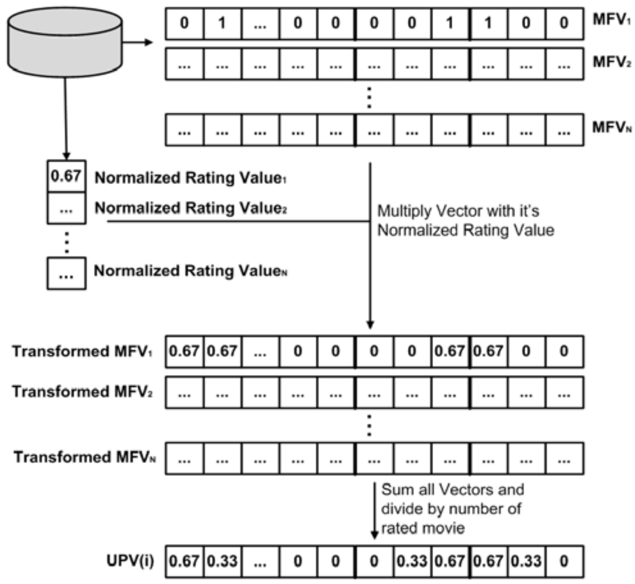

Figure 2. Construction of User Preference Vector (UPV)

3.2.2 Selection on Movie Features Vector (SMV)

In the real life, people always select the movie by style of movie, it implies that genre should have more weight than other features.

To increase weight of user's preference opinions, the frequency of feature selection is considered. This vector contains 7 elements of selection features which are title, genre, release period, actor, actress, director and award. Accordingly, this vector constructs automatically after the user give the opinion. It's characteristic is $SMV = (s_1, s_2, ..., s_N)$; where s_i is frequency of selection toward feature (i) and N is number of component.

For example, if the user (i) searches the movie by genre and give the rate of that movie is 2 (normalized rating value = 0.67) then user search the second movie from genre, release period and give score 1 (normalized rating value = 0.33). Therefore, $S_2 = \frac{0.67+0.33}{2} = 0.5$ and $S_3 = \frac{0.33}{2} = 0.17$. Accordingly, $SMV(i) = (0, 0.5, 0.17, 0, 0, 0, 0)$.

3.2.3 Multidimensional Vector (MDV)

Normally, Recommender Systems ask users to give the rating value for the movie but now it's not sufficiency. To do the Multidimensional, the system needs to ask users to give more information about their contextual information which is place, day, time and companion. This paper uses the contextual information about user characteristics to create Multidimensional Vector by using Multi Linear Regression. The form of Multiple Linear Regression equation is represented in

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + ... + \beta_N x_N \qquad (1)$$

where y is rating value, x_j is dimension j in contextual information and β_j is the coefficient valued of each dimensions. This paper considers four dimensions which are place, day, time and companion. It's characteristic is $MDV(i) = (\beta_0, \beta_1, ..., \beta_n)$; where β_i is the coefficient values of Multiple Linear Regression equation.

3.3 Finding Neighbor Process

Neighbor of the target user is derived from three vectors; User Preference Vector (UPV), Selection on Movie Features Vector (SMV) and Multidimensional Vector (MDV). The neighbor finding process has six steps below and shown in figure 3.

Step 1: For the target user, the updated UPV is selected. In order to reduce the *unsuitable weight problem*, the biggest component of each feature in UPV is taken to calculate the weight value as show in equation (2).

$$w_i = \frac{f_i}{\sum_{i=0}^{N} (f_i)} \qquad (2)$$

where w_i is the weight value for $feature(i)$, f_i is the biggest component of $feature(i)$ and N is number of component.

Then, multiply the weight value to all component of UPV by using its own weight value of that feature. This step is also shown in Figure 3. Repeat this step to do the other users in the system.

Step 2: To find the association of each pair of user (the target user and another user in the system), this approach use the distance of vector. Distance of vector calculates by equation (3).

$$Distance = \sqrt{\sum_{i=0}^{N} (v_{1i} - v_{2i})^2} \qquad (3)$$

where v_1 is element from target user vector, v_2 is element from other user vector and i is an index of element in the vector.

In this step, the distance of UPV between each pair of user (target user and another user for all the rest of users in the system) is calculated by equation (3). The distance value of each pair of UPVs is called D_{UPV}.

Step 3: In order to reduce the *unsuitable weight problem*, this paper also use the frequency of the selection movie features to improve the recommendation by calculating the distance of SMV between each pair of users (target user and another user for all the rest of users in the system) via equation (3). The distance value of each pair of SMVs is called D_{SMV}.

Step 4: To do the Multidimensional and reduce the *losing a huge of rating data problem*, the Multidimensional Vector (MDV) should be used to find the association between each pair of users (target user and another user for all the rest of users in the system) by calculating the distance of MDV which uses equation (3). The distance value of each pair of MDVs is called D_{MDV}.

Step 5: To consider the neighbor, the $Total\ Distance$ Value between target user and other user in the system is calculated as the following:

$$TotalDistance = \frac{D_{UPV} + D_{SMV} + D_{MDV}}{3} \qquad (4)$$

Step 6: Neighbor is produced by selecting the user who has the smallest value of $TotalDistance$ Value.

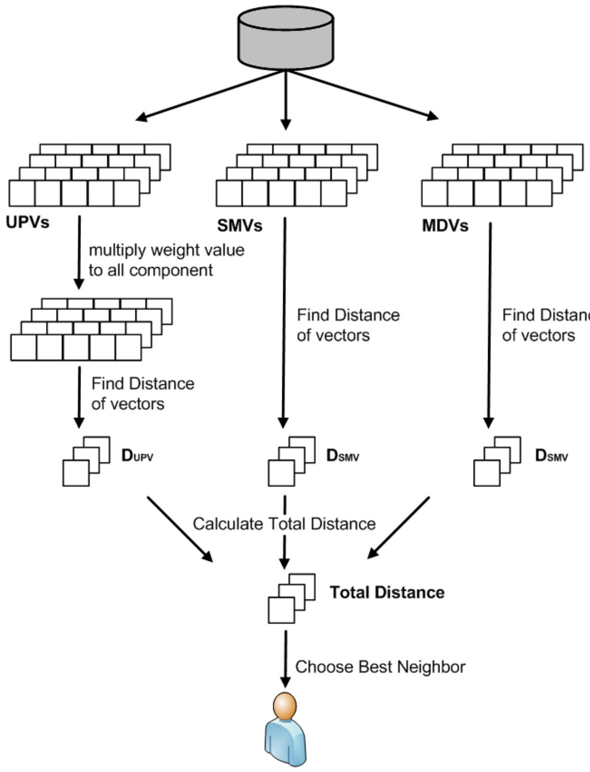

Figure 3. Finding Neighbor Process

4 MoviePlanet System

The prototype of Recommender System called MoviePlanet which is implemented to evaluate the proposed method. MoviePlanet is implemented by Microsoft Visual Studio .Net on Microsoft XP Professional and acts as WWW server. It uses Microsoft SQL Server to be data storage. The information of the available movies obtained from the Internet Movies' Database Site and the Thai-movies obtained from the www.pantip.com. The process of the system is separated in three parts: Entering user's opinion, Finding Neighbor Process and Generating the Recommendations.

4.1 Entering user's opinion

Each user starts by registering to the MoviePlanet system. Then, the search page is shown up for entering any keywords about title, genre, year, award, director, actor and actress to queries the movie from the database. After that, the result page displays the list of the movies search result.

On the result page, user clicks on the movie name. Then, user is asked to give the contextual information and the opinion about that movie. The contextual information is information about where he saw the movie (a movie was seen either in the theater or at home), when the movie was seen (he had seen either in the Day Time or Night Time and either on Weekday or Weekend) and with whom (Friend, Boyfriend/Girlfriend, Family or Other). In the MoviePlanet System, there are three levels of the opinion, which are 1-3(1 is dislike, 2 is neutral and 3 is like). After that, the UPV, SMV and MDV are automatically updated.

4.2 Finding Neighbor Process

This paper is considered the higher quality of recommendation by using Multiple Criteria Rating and Multidimensional. In order to find neighbor, the distance of each vector (UPV, SMV, MDV) between the target user and another user (for all the rest of users in the system) are calculated. Then, the total distance is calculated by averaging the distance of these three vectors. The selected neighbor is the person who has the smallest value of the $Total\ Distance$.

4.3 Generating the Recommendations

As the Recommender System usually show user's favorite or like most item, the MoviePlanet System presents the movie list of liked movies by the neighbor as the recommendations.

5 Experimental and Evaluation

The objective of the experimental evaluation is to show that the proposed method of MoviePlanet which combining the Mutiple Criteria Rating and Multidimensional could provide more accurate results than the current Hybrid Systems based on only the Multiple Criteria Rating. Therefore, in this case is Thai-Music system [7], is a Hybrid System that based on the Multiple Criteria Rating. This experiment changes domain from music to movie.

5.1 Data

In the experimental evaluation, the data of 1063 movies was inserted into the movie database and 49 users were willing to use the system. Each user was asked to rate at least 10 movies. The total of collected opinion from the experiments sum up to 964 rating (*training set*).

The accuracy of this recommendations generated by the system will be revealed when the users say that they like the favorite recommend movies and dislike the undesirable recommend movies.

This paper also simulated the method of Thai-Music on the same data set of MoviePlanet. There are 5 favorited movies of each user are predicted by each system using ratings in the *training set*. After that, the system asked the user to return his feels toward these 5 movies of each system. There are three levels of his feeling; "Want to see", "Neutral" and "Don't want to see". Since, there are 49 users in the system and each user have to return the answer for 5 predicted movies so there are 245 movies of each system in the *test set*.

5.2 Evaluation Criteria

There are two criteria that used for determining the accuracy and quality of the recommendations.

MAE (Mean Absolute Error)[10] is the average absolute deviation between the algorithm's recommendation value and the user's actual preference value. The lower MAE is the more accurate the results. The MAE is represented as equation (5).

$$|\overline{E}| = \frac{\sum_{i=1}^{T}(R_i - p_i)}{T} \qquad (5)$$

Where R_i is a recommendation value for each song in the test set ($R_i = 3$ if it is a like most movie recommend, $R_i = 2$ if it is a neutral movie recommend and $R_i = 1$ if it is the dislike most movie recommend). p_i is the user's actual preference value for each movie in the test set(the feeling toward the recommend movies: "Want to see"(score = 3), "Neutral"(score = 2) and "Don't want to see"(score = 1)). T is the number of movies in the test set.

F-measure[8] is the weighted harmonic mean of precision and recall. The higher F-measure is more accurate the results.

$$F - measure = \frac{2\ (Recall)\ (Precision)}{Recall + Precision} \qquad (6)$$

where Recall (or Sensitivity) is the probability that the relevant items will be accept by the system and Precision (or Positive Predictive Value) is the probability that the accepted items are relevant [10].

Recommender System	*MAE*	*F-Measure*
MoviePlanet	0.5350	0.6533
Thai-Music	0.5523	0.6447

Table 2. Evaluation result between MoviePlanet and Thai-Music

5.3 Evaluation Result

This paper is employed all criteria in Section 5.2 to compare MoviePlanet with the Thai-Music system. As the result is shown in the table 2, the MAE of MoviePlanet is lower than Thai-Music. Moreover, the F-measure of MoviePlanet is higher than Thai-Music system. Therefore, it can be concluded that MoviePlanet provides more accurate recommendation results than Thai-Music.

6 Discussion

According to the value of the evaluation results, MoviePlanet provide higher quality of recommendations than Thai-Music [7]. The reasons are, Thai-Music figures the *without weight feature problem* out by weight only the biggest component of each feature but the way of this weight is not suitable because other components of each feature might be lost and face *unsuitable weight feature problem*. Moreover, Thai-Music does not concern about contextual information. In Contrast, MoviePlanet weights all the components of each feature to reduce the *unsuitable weight feature problem* and also uses the frequency of selection movie feature to increase more weight into the movie features. In Addition, to concentrate on the contextual information and reduce the *losing a huge of rating data problem*, MoviePlanet uses the Multi Linear Regression to perform the Multidimensional. Therefore, both of Multiple Criteria Rating and Multidimensional directly affect to the user's preference on movie selection.

7 Conclusions

This paper proposed a new approach that provides higher quality of recommendations. This proposed method combines the Multi Criteria and Multidimensional to provide the recommendation results. Instead of weighting only biggest value of the features, the proposed method weights all components with its weight value and also uses the frequency of the selection movie features to increase the weight of Multi Criteria Rating. In other word, it can overcome *unsuitable weight problem*. Moreover, it can incorporate the contextual information without the *losing a huge of data problem* by using Multiple Linear Regression to perform the Multidimensional.

For evaluating the proposed method, a movie Recommender System called MoviePlanet has been created. As presented in the experimental evaluation, the combining of Multi Criteria Rating and Multidimensional system provides more accurate recommendation results than the Hybrid System based on current Multiple Criteria Rating method.

References

[1] Resnick, P., Varian, H.R., "Recommender Systems," *Communications of the ACM*, Vol.40, No.3, pp. 56-58, 1997.

[2] Linden G., Smith B., York J., "Amazon.com Recommendations: Item-to-Item Collaborative Filtering," *Proc. of February 2003 issue of IEEE Internet Computing*, 2003, http://dsonlin.computer.org/0301/d/w1lind.htm.

[3] Balabanovic, M. and Y. Shoham, "Fab: Content-based, collarative recommendation," *Communications of the ACM*, 40(3):66-72, 1997.

[4] Pazzani, M., "A framework for collaborative, content-based and demographic filtering," *Artificial Intelligence Review*, pages 393-408, 1999.

[5] Francois Le Roux, Elkunchwar Ranjeet, Vinit Ghai, Ya Gao, Jie Lu, "A Course Recommender System Using Multiple Criteria Decision Making Method," *ISKE-2007 Proceedings*, Series 'Advances in Intelligent Systems Research', 2007

[6] Saranya Maneeroj, Yuka Kato and Katsuya Hakozaki, "An Advanced Movie Recommender System Based on High-Quality Neighbors," *IPSJ Digital Courier*, Vol. 1, pp.181-192. , 2005.

[7] Saranya Maneeroj "High-quality Neighbor Formation for Music Recommender System," *IMECS*, page 805-811, 2007

[8] Gediminas Adomavicius, Ramesh Sankaranarayanan, Shahana Sen, Alexander Tuzhilin, "Incorporating contextual information in recommender systems using a multidimensional approach," *ACM Transactions on Information Systems (TOIS)*, Volume 23, Pages: 103 - 145 , 2005

[9] EFFROYMSON, M. A., "Multiple regression analysis." *In Ralston, A., and Wilf, H. S. (Eds.),Mathematical Methods for Digital Computers, Wiley, New York*, Volume 23, pp. 191-203. , 1960

[10] MedCalc Organizations, "ROC curve analysis: Introduction," *Communications of the ACM*, http://www.medcalc.be/manual/mpage06-13a.php