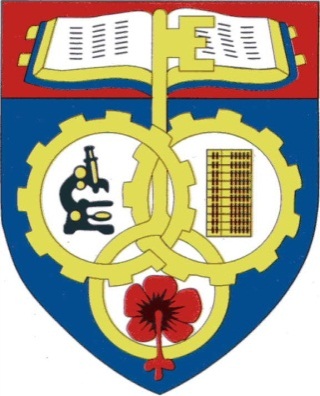
PERSONALIZED RECOMMENDER SYSTEM FOR E-COMMERCE

By

WILSON ONG KOK WENG



SCHOOL OF ARTS AND SCIENCE

TUNKU ABDUL RAHMAN COLLEGE

KUALA LUMPUR

2012/2013

Personalized Recommender System

For E-Commerce

By

Wilson Ong Kok Weng

A seminar report submitted to the School of Arts and Science

in partial fulfillment of the requirement for the

Bachelor of Science, Campbell University, U.S.A.

and

Advanced Diploma in Science,

Tunku Abdul Rahman College

2012/2013

**Table of Content**

[Problem Statement 1](#_Toc337013146)

[1. Introduction to Personalized Recommender System For E-commerce 2](#_Toc337013147)

[1.1. Content-based Filtering 2](#_Toc337013148)

[1.2. Collaborative-based Filtering 2](#_Toc337013149)

[2. Algorithms 3](#_Toc337013150)

[2.1. Manhattan distance and Euclidean distance 3](#_Toc337013151)

[2.2. Cosine similarity 6](#_Toc337013152)

[2.3. Pearson Correlation Coefficient 7](#_Toc337013153)

[3. Other Techniques 9](#_Toc337013154)

[3.1. Classification 9](#_Toc337013155)

[3.1.1. Tagging 9](#_Toc337013156)

[4. Issues to Be Considered 10](#_Toc337013157)

[4.1. Data Sparsity 10](#_Toc337013158)

[4.2. Grade-Inflation 11](#_Toc337013159)

[5. Application to Final Year Project 12](#_Toc337013160)

[5.1. Characteristics of My E-commerce Web Application 12](#_Toc337013161)

[5.2. Techniques and Features 12](#_Toc337013162)

[5.3. Algorithms 13](#_Toc337013163)

[5.4. The Implementation 13](#_Toc337013164)

[6. References 15](#_Toc337013165)

[7. Appendices 18](#_Toc337013166)

[7.1. Sample Dataset of Fashion Web Application (Rating) 18](#_Toc337013167)

[7.2. Pythagorean Theorem 19](#_Toc337013168)

[7.3. Relationship Between Users 20](#_Toc337013169)

[7.4. Distance-based Classification 21](#_Toc337013170)

[7.5. iKnowU Fashion Portal’s Recommendation Logic 22](#_Toc337013171)

# Problem Statement

A comprehensive research was carried out on personalized recommender system to assist in the implementation of such a system in my final year project, which is an electronic commerce web application about fashion. Schafer, Konstan and Riedi (1999, p. 158) states that recommender system can increase e-commerce sales in various ways. Non-personalized recommender system recommends products that are interested by majority of people to every user. However, those generally recommended products may not be precise enough to attract more transactions to happen. Therefore, a recommender system that can recommend products based on behaviors and tastes of every user is needed. Here comes the purpose of conducting this research paper to find out the ways to increase the probability where users will be interested in the products recommended by the system by personalizing the contents of the recommendation and aims to increase the profits of my e-commerce website significantly.

The research was carried out to study on some famous algorithms of recommender system and find out some useful techniques to be applied in my system.

# Introduction to Personalized Recommender System For E-commerce

Personalized recommender system can be deployed at different checkpoints in an e-commerce website (Dias, Locher, Li, El-Deredy, Lisboa, 2008). Different from non-personalized recommender system; personalized recommender system is able to custom-made recommendations for every users according to their implicit and explicit information.

Candillier, Meyer, and Boulle’s study (cited in Mohammad Khoshneshin and Street, 2010, p. 87) indicates that recommender system has two categories in common – content-based and collaborative-based.

## Content-based Filtering

Content-based filtering filters products according to the characteristics of the products while collaborative filtering recommends products to the users according to their preferences.

It builds each user a user profile and recommends the products to the user according to the preferences stated in the profile. The profiles are built by requesting the users to define their preferences or collect users’ browsing data through various machine learning techniques (Pazzani and Billsus, 2007).

## Collaborative-based Filtering

Collaborative-based filtering make recommendations by selecting top-N number of existing users from the database who have highest level of similarity to the user and recommend the in-common favorite items of the N users.

Collaborative filtering is a preferred way to implement recommender system in e-commerce website. It is considered as one of the most successful techniques used in recommender system. It predicts users’ preferences through the user’s past activities (Takacs, Pilaszy, Nemeth, and Tikk, 2009). The e-commerce websites that used this technique including Amazon.com (Zhang and Pu, 2007).

# Algorithms

There are over hundreds of algorithms developed for recommender system. Majority of the algorithms are used to calculate the similarity or distance of users and items. The algorithms might not be specific for any category of recommender system; for example, Pearson Correlation Coefficient (To be discussed in section 2.2) can be used for both item-based and user-based collaborative filtering.

## Manhattan distance and Euclidean distance

Manhattan Distance and Euclidean Distance are simple, fast yet useful algorithms to calculate similarity between two variables. The formulae (Dunham, 2003, p. 58) are:-

*Manhattan Distance:*

*Euclidean Distance:* **2**

The results of them are always zero or non-negative rational number. Visualizing the calculations would be clearer for understanding. By using the following ratings:-

|  |  |  |  |
| --- | --- | --- | --- |
| Item\User | Amy | Julian | Willie |
| Green Summer Pants | 2 | 2 | 5 |
| White Wedding Gown | 5 | 4 | 1 |

Table ‑ sample rating dataset

\* The rating scaled from ‘1’ to ‘5’, indicating ‘dislike the most’ to ‘like the most’.

“Item × Item” graph is plotted:-

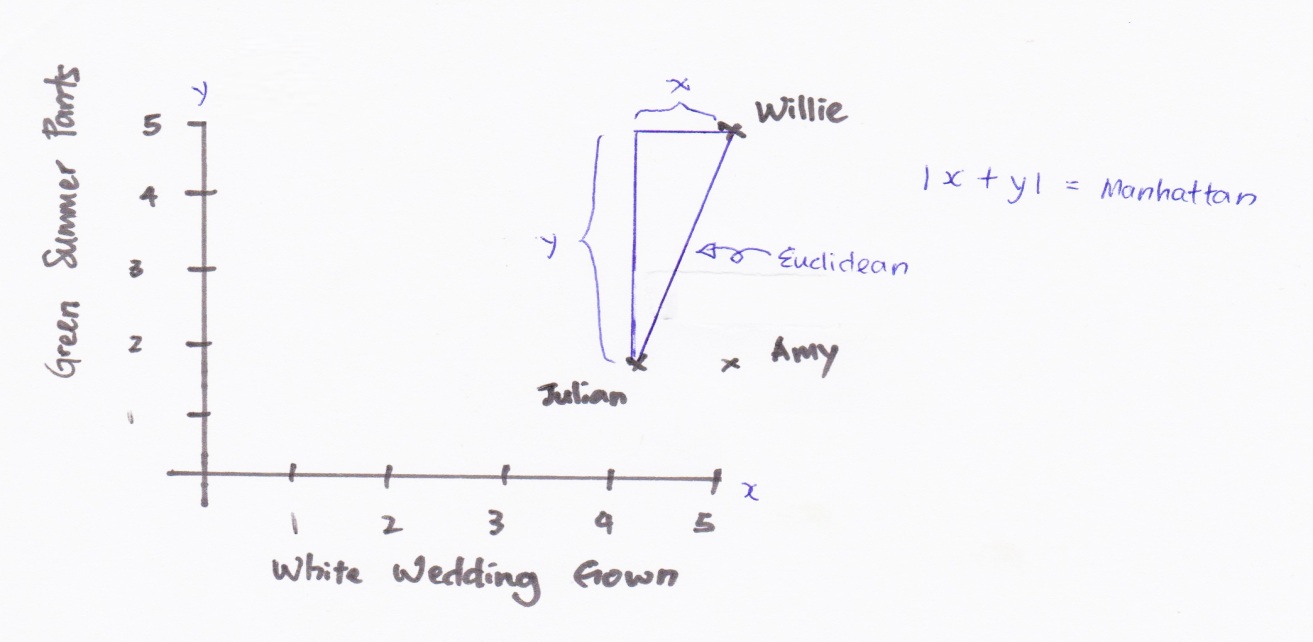


Figure ‑ Manhattan and euclidean graph (adopted from (Zacharski, R., 2010 p.2-4))

The sum of x and y is the level of similarity of Julian and Willie in Manhattan Distance Metric while Euclidean Distance Metric calculates the distance by applying Pythagorean Theorem (refers to appendix 7.2).

The smaller the value of the result, the more similar the two variables are. In this example, Amy is chosen over Willie as the nearest neighbor (most similar user) to Julian. Recommender system will recommend other products which are highly rated by Amy and are not yet rated by Julian.

**Multi-dimensional Calculation**

Using dataset from appendix 7.1, assume that recommender system wants to recommend new products to Boyce, the following table is generated by comparing Boyce to Ann:-

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Item\User | Boyce, *x* | Ann, *y* | Difference, |x-y| | Difference2,  (x-y)2 |
| Knit Combi Dress | 4 | 2 | *2* | *4* |
| Fern by Heart Dress | 4 | 3 | *1* | *1* |
| Wow Shift Dress | - | 1 | *-* | *-* |
| Stretch skinny jean | - | - | *-* | *-* |
| Flared Chino Jeans | - | - | *-* | *-* |
| Tiptoe Glow Heel | 5 | 4 | *1* | *1* |
| Fetish Sandals | 5 | 5 | *0* | *0* |
| **Manhattan Distance, ∑ |x-y|** |  |  | **4** |  |
| **Euclidean Distance, 2** |  |  |  | **2.4495** |

Table ‑ manhattan and euclidean distances for boyce and ann

The distances from Boyce to Catherine and to Daniel are calculated respectively on a later time. Recommend the favorite items of the user who has smallest distance value against Boyce to Boyce.

Zacharski (2010, p. 2-11) indicates that these two algorithms can be generalized using Minkowski Distance Metric.

## Cosine similarity

Zacharski (2010, p. 2-28) states that Cosine Similarity is a very popular in text mining technique and can be used in collaborative filtering. The concept of it is that it will convert the degree of similarity between two vectors into a value ranged from ‘0’ to ‘1’, indicating ‘share no same attribute’ to ‘share exactly same attributes’. It overcomes the problem of shared-zero in distance-based metrics when the data is sparse (refer to section 4.1).

Cosine Similarity’s formula (Zacharski, 2010, p. 2-30) is defined as:-

*, where*

Using the same scenario from previous section (section 2.2), the degree of similarity between Ann and Boyce is going to be calculated using Cosine Similarity algorithm.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Item\User | Ann, *x* | Boyce, *y* | ***xy*** |  |  |
| Knit Combi Dress | 2 | 4 | ***8*** | ***4*** | ***16*** |
| Fern by Heart Dress | 3 | 4 | ***12*** | ***9*** | ***16*** |
| Tiptoe Glow Heel | 4 | 5 | ***20*** | ***16*** | ***25*** |
| Fetish Sandals | 5 | 5 | ***25*** | ***25*** | ***25*** |
| **∑** | **14** | **18** | ***65*** | ***54*** | ***82*** |

Table ‑ TABLE USED FOR cosine similarity CALCULATION

By substituting the values into Cosine Similarity algorithm:-

The value of 0.9768 indicates that Ann and Boyce have about 97.68% similarity.

## Pearson Correlation Coefficient

Pearson Correlation Coefficient is used to measures how strong the relationship between two users or items is. Pearson Correlation Coefficient is a commonly used correlation coefficient method for recommender system. It is known as Adjusted Cosine Similarity sometimes.

**Agreement**

Referring to appendix 7.3, the four graphs are plotted according to the sample dataset. When Adam is plotted against his own ratings, the graph shows a perfect agreement where a straight line is able to pass through every single dot. This shows that the two variables used to compare are exactly similar.

When Adam is plotted against Susan, it is impossible to use a straight line to connect every dot. However, the dots are closely populated; this is a pretty good agreement indicates both users have a certain level of similarity.

When Adam is plotted against Julie and James, the dots are sparse. This indicates a poor agreement between the users, indicates that they are not similar and is not recommended to use for recommendation purpose.

**The Algorithm**

The result of Pearson Correlation Coefficient is always between -1 and 1, where -1 indicates perfect disagreement while 1 indicates perfect agreement (Marmanis and Babenko, 2009, p.111). The formula (Zacharski, 2010, p. 2-23) of Pearson Correlation Coefficient, *r* is as below:-

*, where n = size of dataset.*

Let’s use the sample dataset from appendix 7.1 again to demonstrate the algorithm:-

Pearson Correlation Coefficient, *r* for Catherine to Daniel is calculated as below (only items that are rated by both of them are able to be considered in the calculation).

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Item\User | Ann, *x* | Boyce, *y* | ***xy*** |  |  |
| Knit Combi Dress | 2 | 4 | ***8*** | ***4*** | ***16*** |
| Fern by Heart Dress | 3 | 4 | ***12*** | ***9*** | ***16*** |
| Tiptoe Glow Heel | 4 | 5 | ***20*** | ***16*** | ***25*** |
| Fetish Sandals | 5 | 5 | ***25*** | ***25*** | ***25*** |
| **∑** | **14** | **18** | ***65*** | ***54*** | ***82*** |

Table ‑ table used for pearson correlation coefficient calculation

By substituting the values into the equation:-

The result 0.8944 or 89.44% indicates that the two users are quite similar.

# Other Techniques

Algorithms to calculate similarity alone is insufficient to create a good recommender system; it is a combination of many other techniques.

## Classification

Classification is important technique used to determine the concepts of an instance. Generally speaking, it is a technique used to automatically put the items into the categories it should be at. The concept of an item can be determined by using various methods, and one of the simplest methods is to define the classes, and then put items to the class that are nearest to them. This is called distance-based classification; refer to appendix 7.4 for the logical concept of distance-based classification. The distance is calculated using distance-based algorithms (Dunham, 2003).

### Tagging

Tagging is a method to achieve classification by labeling the items with appropriate tags, either manually by users, or automatically by the system. It is always used in folksonomy approach recommendation system (Manish Gupta, Li, Yin and Han, 2010). The ontology (Wikipedia, 2012a) of the tags must be predefined so that the relationships between the tags can be identified by recommender system and retrieve the items with similar tags semantically.

Alan Said, Kille, De Luca and Sahin Albayrak (2011) states that tagged folksonomy approach can reduce the impact of data sparsity and make the results more accurate. Website that is using tagging technique including Netflix.com.

# Issues to Be Considered

No single algorithm can fit all purposes. Each recommender system approach has its strengths and limitations. There are a number of issues developers must consider when selecting the most suitable approach and algorithms for their systems.

## Data Sparsity

Collaborative-based recommender system is the most successful approach for recommender system but it is unable to recommend unrated products (Cremonesi, Turrin and Airoldi, 2011); here comes the data sparsity issue.

Sparsity happens when system has insufficient information about the users. Cold start (Wikipedia, 2012b; Asanov, n.d.) is a typical example, when a user just joined the website; he has too less information for the system to personalize the recommendation. Cacheda, Carneiro, Fernandex and Formoso (2011, p. 2-4) claims that ‘cold start’ is not only difficult for system to find similarity; it also caused new items unable to be promoted.

**Workarounds**

For collaborative-based recommender system, algorithms such as Manhattan and Euclidean distance are not resist to the problem of sparsity in a high-dimensional dataset. Zacharski (2010, p.2-31) states that the solution to data sparsity is using Cosine Similarity algorithm. Cosine Similarity is able to ignore the shared-zero issue and hence is more resist to data sparsity.

The better resolution is to use hybrid approach. Forbes and Zhu (2011, p. 261) indicates that data sparsity problem can be solved by using content-based technique to predict the unrated items first.

## Grade-Inflation

Grade-inflation is subjected to different users using different rating schemas in the same range of rating scale. One may rate a ‘5’ as good and a ‘3’ as poor while another user rated ‘3’ as good and ‘1’ as poor. Using different rating scale will cause algorithms such as Manhattan to make inaccurate predictions (Zacharski, 2010).

**Workarounds**

Pearson Correlation Coefficient is the solution. It finds out the similarity between two users based on the cosine degree between the vectors of the users’ ratings which also considered the invariant of the vectors at the same time.

# Application to Final Year Project

Concluding the literature reviews from section 1 to section 4, a number of options can be spotted including the techniques and algorithms to be used in my web application for personalized recommendation purpose.

## Characteristics of My E-commerce Web Application

The characteristics of my e-commerce website are as below:-

* **New.** Since my system is new (cold start), the existing data set might not be dense but data sparsity must be an issue.
* **Categories are important.** My e-commerce website is selling fashion products. The products have to be sorted logically according to the categories such as dresses, jeans, and so forth.
* **Socialized recommendation.** Social is important in my system and products should be recommended based on social more than any kind.

## Techniques and Features

To develop a successful personalized-recommender system, collaborative-based filtering must be implemented. However, collaborative-based recommender system has a lot of challenges such as data sparsity and grade-inflation. I have coming out with a solution by using hybrid-approach instead of collaborative-approach. Below are the techniques which I am going to implement in my web application:-

* **Rating.** This feature is going to be applied in my system because it is important explicit information to make an excellent collaborative recommender system.
* **Tagging.** By allowing the users to tag the products based on the characteristics of the products, the products can be more accurately recommend to the users by using content-based approach.

## Algorithms

I shall select the appropriate algorithms based on the characteristics of my website and the techniques that I have chosen in previous sections.

There are three factors driving the way I select the algorithms for my recommender system:-

* **Grade-inflation.** ‘Rating’ in my system is going to be scaled from ‘1’ to ‘5’. Everyone will have different opinion on the indications. Hence every user will have their own rating schema and this lead to the inconsistency.
* **Data sparsity.** As it is impossible for all users to rate every items, data sparsity is always the main concern. The algorithms that I have chosen must address this problem.
* **Socialized recommendation.** As I wish to implement a social-based recommendation, the best approach would be the user-based collaborative recommendation. I have to choose algorithms that are able to do calculation for such an approach.

Based on the factors, I have decided that Pearson Correlation Coefficient fulfilled the requirements most. It resists to data sparsity and grade-inflation, meanwhile is famous for collaborative filtering.

## The Implementation

My system would require the following techniques and features to be implemented:-

The application included rating and tagging features where users can rate the items and sellers will tag the items by themselves. At the same time, the system will collect demographic data (age, gender, location and so forth) from the users to build user profiles, and records every purchase histories. The system will recommend products to users based on this information later.

Refers to appendix 7.5 for the logical flow of my recommender system:-

Three sub-modules together form the recommender system:-

**Module 1:** **Similar-user based filtering**

Use classification technique to find all users that have similar characteristics with the user (such as age, gender, and so forth). After that, sort the users according to the degree of similarity to the user by using similarity-algorithm (Pearson Correlation Coefficient). When sorting is done, get the products that rated highly by top-N users and generate the recommended contents.

**Module 2: Purchasing history based filtering**

Get products by using classification technique and find top-N nearest items to recommend. This module filter the items based on the purchase histories of the user.

**Module 3:** **Trends based filtering**

Get products by using classification technique by selecting top-N items based on most purchased item types.

Module 1 can get ideal results with the assist of users’ rating while module 2 and 3 can recommend based on the sellers’ tagging.

There will be more than one algorithms involved during the implementation. For example, classification will make use of Manhattan Distance algorithm; sort users based on similarity will use Pearson Correlation Coefficient and so forth. Various techniques and algorithms are required to work along to ensure the quality of the personalized recommender system.

# References

Alan Said, Kille, B., De Luca, E.W. and Sahin Albayrak. 2011.

‘Personalizing Tags:A Folksonomy-like Approach for Recommending Movies’. *Proceedings of the 2nd International Workshop on Information Heterogeneity and Fusion in Recommender Systems,* pp. 53-56.Viewed on 1 September 2012. Available from < http://dl.acm.org/citation.cfm?id=2039320.2039328&coll=DL&dl=GUIDE&CFID=148308915&CFTOKEN=39262959>.

Asanov, D. n.d. Algorithms and Methods in Recommender Systems.

Viewed on 4 September 2012. Available from <http://www.snet.tu-berlin.de/fileadmin/fg220/courses/WS1011/snet-project/recommender-systems\_asanov.pdf >.

Cacheda, F., Carneiro, V., Fernandex, D. and Formoso, V. 2011.

‘Comparison of Collaborative Filtering Algorithms: Limitations of Current Techniques and Proposals for Scalable, High-Performance Recommender Systems’. *ACM Transactions on the Web(TWEB)*. Vol. 5 (1). Viewed on 1 September 2012. Available from < http://dl.acm.org/citation.cfm?id=1921591.1921593&coll=DL&dl=GUIDE&CFID=148308915&CFTOKEN=39262959>.

Cremonesi, P., Turrin, R. and Airoldi, F. 2011.

‘Hybrid algorithms for recommending new items’. *Proceedings of the 2nd International Workshop on Information Heterogeneity and Fusion in Recommender Systems,* pp. 33-40. Viewed on 1 September 2012. Available from <http://dl.acm.org/citation.cfm?id=2039320.2039325&coll=DL&dl=GUIDE&CFID=148308915&CFTOKEN=39262959>.

Dias, M.B., Locher, D., Li, M., El-Deredy, W., Lisboa, P.J.G. 2008.

‘The value of personalised recommender systems to e-business: a case study’. *Proceedings of the 2008 ACM conference on Recommender systems*, pp. 291-294. Viewed on 27 August 2012. Available from < http://dl.acm.org/citation.cfm?id=1454008.1454054&coll=DL&dl=GUIDE&CFID=148308915&CFTOKEN=39262959>.

Dunham, M.H. 2003. *Data Mining: Introductory and Advanced Topics*. New Jersey:

Pearson Education.

Forbes, P. and Zhu, M. 2011. ‘Content-boosted matrix factorization for

recommender systems: experiments with recipe recommendation’. *Proceedings of the fifth ACM conference on Recommender systems,* pp. 261-264. Viewed on 1 September 2012. Available from < http://dl.acm.org/citation.cfm?id=2043979 >.

Manish Gupta, Li, R., Yin, Z.J. and Han, J.W. 2010.

‘Survey on Social Tagging Techniques’. *SIGKDD Explorations*. Vol. 12 (1), pp. 58-72. Viewed on 1 September 2012. Available from < http://dl.acm.org/citation.cfm?id=1882480>.

Marmanis, H. and Babenko, D. 2009. *Algorithms of the Intelligent Web*. Greenwich,

CT: Manning.

Mohammad Khoshneshin and Street, W.N. 2010.

‘Collaborative filtering via euclidean embedding’. *Proceedings of the fourth ACM conference on Recommender systems*, pp. 87–94. Viewed on 1 September 2012. Available from < http://dl.acm.org/citation.cfm?id=1864708.1864728&coll=DL&dl=GUIDE&CFID=148308915&CFTOKEN=39262959>.

Pazzani, M.J. and Billsus, D. 2007. *Content-based recommendation systems*.

Viewed on 27 August 2012. Available from <http://www.google.com.my/url?sa=t&rct=j&q=&esrc=s&source=web&cd=3&ved=0CDkQFjAC&url=http%3A%2F%2Fciteseerx.ist.psu.edu%2Fviewdoc%2Fdownload%3Fdoi%3D10.1.1.130.8327%26rep%3Drep1%26type%3Dpdf&ei=-IZFUKmWF4bNmAW62YCIDw&usg=AFQjCNEyj7egvDIpSU6qvtDXFxgHap-QuA>.

Schafer, J.B, Konstan, J and Riedi, J. 1999. ‘Recommender systems in e-commerce’.

*Proceedings of the 1st ACM conference on Electronic commerce,* pp. 158-166. Viewed on 27 August 2012. Available from < http://dl.acm.org/citation.cfm?id=337035 >.

Takacs, G., Pilaszy, I., Nemeth, B. and Tikk, D. 2009.

‘Scalable Collaborative Filtering Approaches for Large Recommender Systems’. *The Journal of Machine Learning Research*. Vol. 10, pp. 623-656. Viewed on 1 September 2012. Available from < http://dl.acm.org/citation.cfm?id=1577069.1577091&coll=DL&dl=GUIDE&CFID=148308915&CFTOKEN=39262959>.

Wikipedia. 2012a. ‘Ontology’. Viewed on 4 September 2012.

Available from: < http://en.wikipedia.org/wiki/Ontology>.

Wikipedia. 2012b. ‘Cold start’. Viewed on 4 September 2012.

Available from: < http://en.wikipedia.org/wiki/Cold\_start>.

Zacharski, R. 2010. *A Programmer's Guide to Data Mining.*

Viewed 1 September 2012. Available from < http://guidetodatamining.com/>.

Zhang, J.Y. and Pu, P. 2007. ‘A Recursive Prediction Algorithm for Collaborative

Filtering Recommender Systems’. *Proceedings of the 2007 ACM conference on Recommender systems*, pp. 57-64. Viewed on 1 September 2012. Available from < http://dl.acm.org/citation.cfm?id=1297231.1297241&coll=DL&dl=GUIDE&CFID=148308915&CFTOKEN=39262959>.

# Appendices

## Sample Dataset of Fashion Web Application (Rating)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Item\User | Ann | Boyce | Catherine | Daniel |
| Knit Combi Dress | 2 | 4 | - | 4 |
| Fern by Heart Dress | 3 | 4 | 2 | 2 |
| Wow Shift Dress | 1 | - | 2 | 4 |
| Stretch skinny jean | - | - | 4 | 5 |
| Flared Chino Jeans | - | - | 5 | 2 |
| Tiptoe Glow Heel | 4 | 5 | - | 4 |
| Fetish Sandals | 5 | 5 | - | 1 |

Table ‑ Sample data set

\* The rating scaled from ‘1’ to ‘5’, indicating ‘dislike the most’ to ‘like the most’.

## Pythagorean Theorem

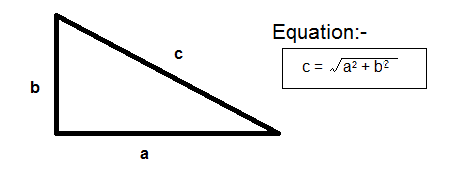


Figure ‑ Equation of pythagorean theorem

## Relationship Between Users

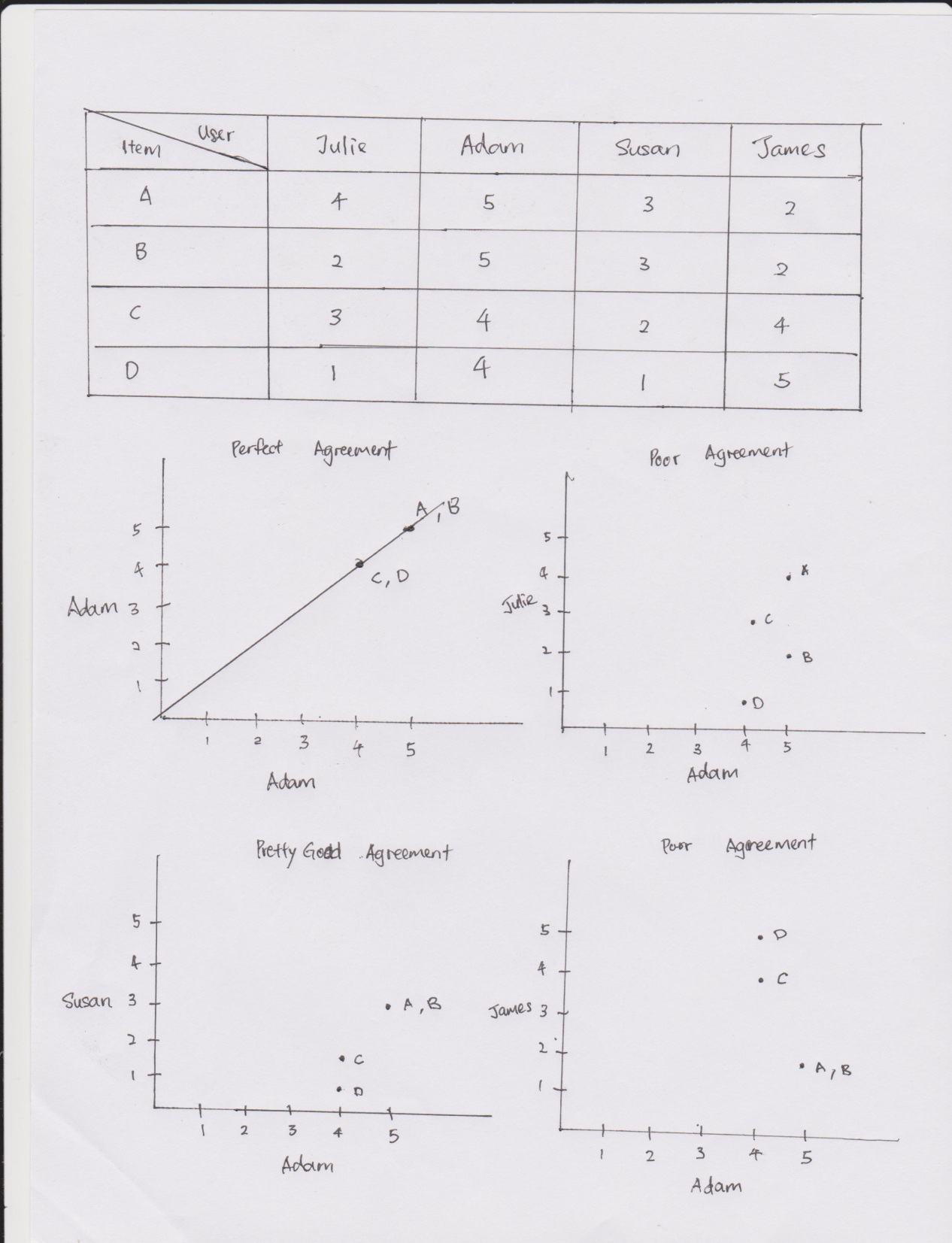


Figure ‑ Level of agreement

## Distance-based Classification

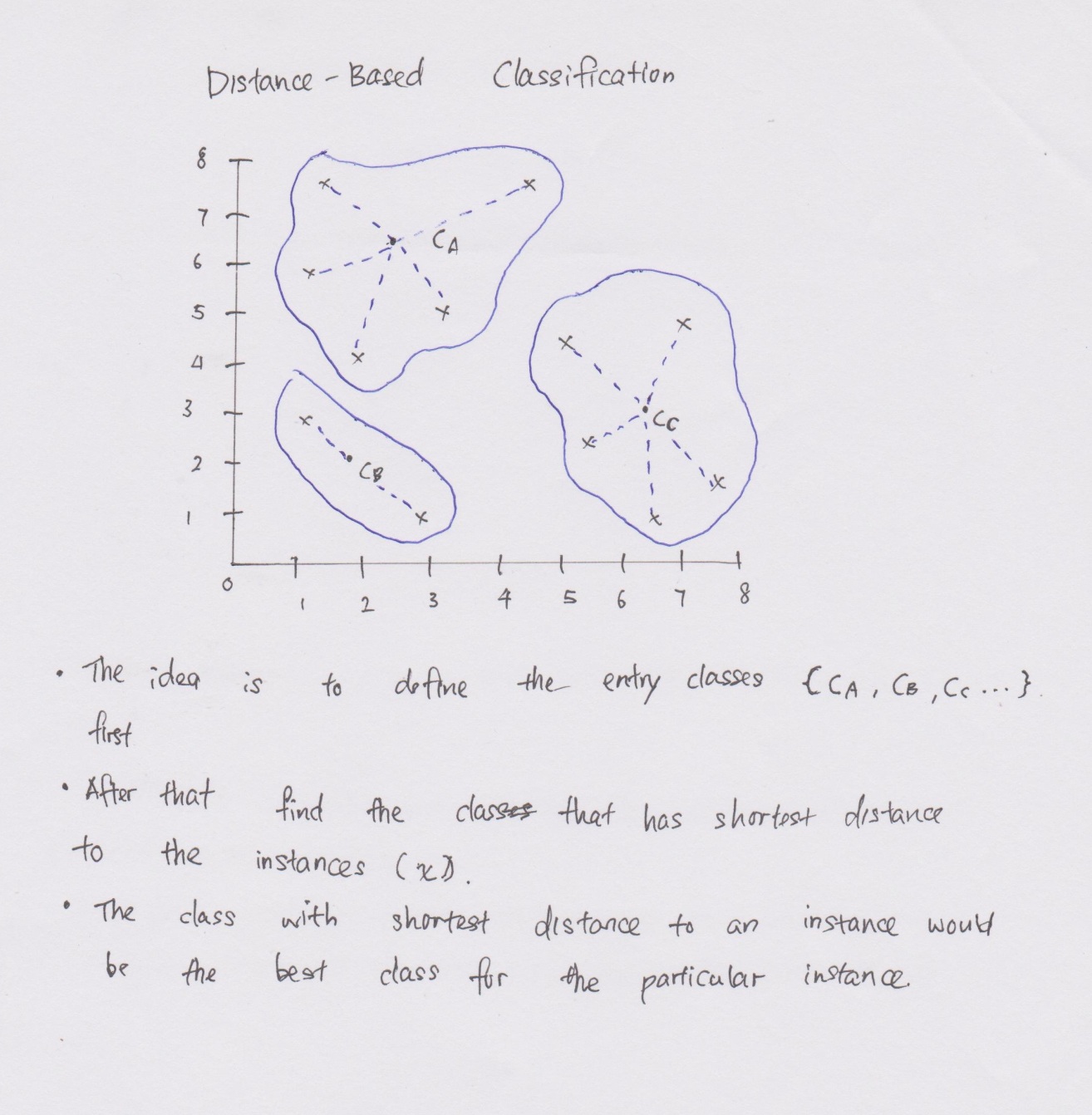


Figure ‑ distance-based classification (retrieve and Adopted from (Dunham, M.H., 2003, p. 91))

## iKnowU Fashion Portal’s Recommendation Logic

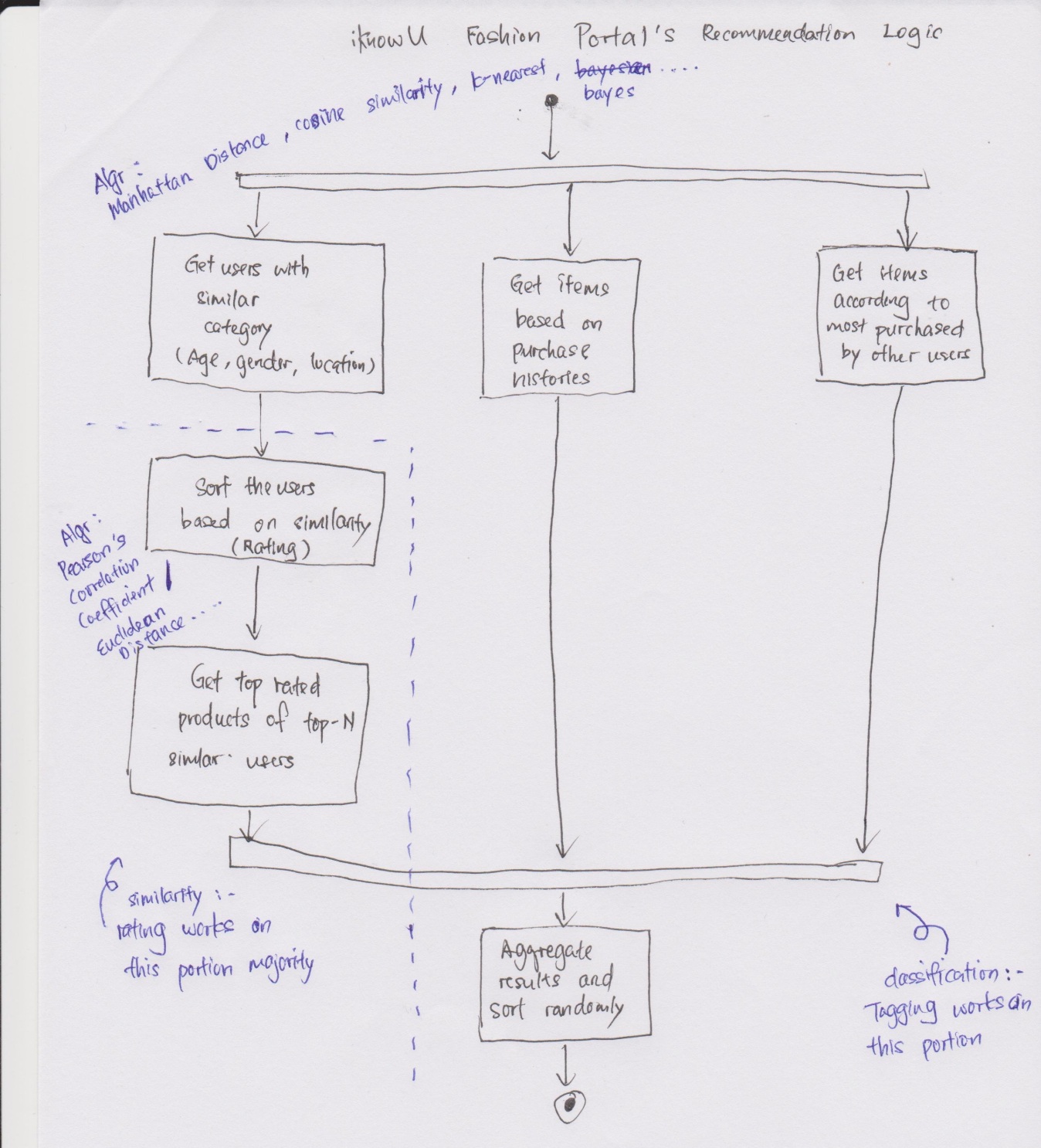


Figure ‑ iKnowu fashion portal's recommendation logic [diagram]