Design of Single User Decision Support System
Model Based on Fuzzy Simple Additive Weighting
Algorithm to Reduce Consumer Confusion

Problems in Smartphone Purchases

Robertus Nugroho Perwiro Atmojo
School of Information Systems, Bina Nusantara University

Anggita Dian Cahyani
Department of Psychology, Bina Nusantara University

Bahtiar Saleh Abbas
Department of Industrial Engineering, Bina Nusantara University

Bens Pardamean
Graduate School of Information Systems, Bina Nusantara University

Anindito
School of Information Systems, Bina Nusantara University

Imanuel Didimus Manulang
Research & Technology Transfer, Bina Nusantara University

Copyright © 2014 Robertus Nugroho Perwiro Atmojo, Anggita Dian Cahyani, Bahtiar Saleh Abbas, Bens Pardamean, Anindito, and Imanuel Didimus Manulang. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.
Abstract
Nowadays, choosing the most suitable smartphone product is not an easy matter. A large number of product brands, technology features are offered, and unobjective product recommendation from others could make customers more difficult to choose. Therefore, this study proposed a single-user model of Decision Support System application based on Fuzzy Simple Additive Weighting algorithm. It is purposed to facilitate customers to narrow down the set of smartphone product alternatives into several recommendations.

Keywords: Single-User Decision Support System, Fuzzy Simple Additive Weighting, consumer confusion, smartphone selection problem

Introduction
In every aspect when people should purchase something like products or services, people definitely face various confusion problems to consider what is the most suitable product or service for them, before deciding to buy it. Those feelings are such as: feeling of regrets or disappointment may come if people fail to address what they need or what they expect. Scientifically, in this case experts from many disciplines mapped those confusion problems as consumer confusion [8, 9, 12, 13].

Naturally to avoid or to reduce the confusion, people may have strategies to deal with it. The strategy may be vary, like the traditional method oftenly used by people to reduce the confusion, such as: doing the several product/service check by seeking additional information on the internet, discussing about products’ after sales services with close relatives (family, friends, colleagues, etc), or asking to the salesperson to help consider what is the most worth product to be purchased. Experts name those strategies as confusion reduction strategy (abbreviated as CRS) [8, 9, 12, 13].

In some traditional CRS methods, those methods may cause inaccurate decision making process, because there were many assumptions from many sources. In order to reach more accurate, more effective and more objective decision results, several characteristics of Multi Criteria Decision Making (MCDM) could be considered as another alternative of traditional or conventional CRS method. As a subset of Decision Support Systems (DSS), there were several reliable MCDM algorithms could fit to solve consumer confusion problems, such as: Analytical Hierarchy Process (AHP), ELECTRE (Elimination and Choice Expressing Reality), TOPSIS (Technique for Order of Preference by Similarity to Ideal Solution), Simple Addition Weighting (SAW), PROMETHEE and so on.
This study was trying to reduce consumer confusion problem in smartphone product purchases using single user DSS approach. Researchers proposed a single user model of DSS using Fuzzy SAW algorithm because of its simplicity to develop and implement in computer program. The confusion problem in smartphone product purchases is an interesting issue to investigate, because of the proliferation of smartphone brands, the more advanced smartphone technology offered, further the more competitive smartphone prices with a huge amount of its attractive features and many other aspects which are definitely causing a confusion for its potential consumer to choose the right one.

The main focus of this study is to propose a single user DSS application model based on Fuzzy SAW algorithm, using a computer program to reduce the confusion issues specifically in consumer confusion problem. This static desktop application was built using Microsoft Visual Basic .NET SDK without database management system capability. The software development in this study refers to linear sequential model developed by Pressman [6].

Related Works

The use of DSS application to solve several decision making problems has many success records on its best practices. Most of scholars or practitioners employs powerful characteristic of MCDM algorithms to simplify several complex input criteria into one, two, or three important recommendation(s) that could affect to expected decision results. Previous study by Nativade-Jesus, Coutinho-Rodrigues, & Antunes [1] tells about the important contribution of DSS application to solve the housing selection problem in Portugal. The development of DSS application in their study has several purposes, such as: (1) the assessment of property value, (2) the evaluation of property quality, and (3) for supporting consumer needs about the “for sale” property. The combination of three MCDM algorithms, consisting of: SAW, TOPSIS, and ELECTRE in tandem with Geographic Information System (GIS) tool was used in Nativade-Jesus, Coutinho-Rodrigues, & Antunes [1] study. Another important contribution of DSS application study by Coutinho-Rodrigues, Simao, & Antunes [3] which was still located in Portugal, has several strategic impacts for the urban infrastructure investment problem. Coutinho-Rodrigues, Simao, & Antunes [3] named their GIS-Based DSS application as MCPUIS (Multi Criteria Planning of Urban Infrastructure Systems) which has capability to evaluate and compare four investment options on water supply network project in Coimbra city. Their MCPUIS was employed together with three MCDM algorithms, consisting of: SAW, TOPSIS, and ELECTRE I; used for evaluate and compare the alternative of course actions. In this study the development of DSS application aimed to reduce the confusion problem in smartphone product selection. Fuzzy SAW algorithm was used because of its simplicity to develop and implement.
Method

The design of DSS application software in this study used Pressman sequential linear model on software development and the decision weighting process of the DSS was based on Fuzzy SAW algorithm. The illustration of Pressman framework could be seen in Figure 1.

The sequential linear model of Pressman framework [6] consists of four phases. Those phases are: (1) Analysis phase, (2) Design phase, (3) Coding phase, and (4) Testing phase.

Result

Below is brief explanation of several implementation processes on Figure 1.

Phase one: the analysis steps (A)

First step
Using linguistic variable (LV) grouping for fuzzy numbers as used in previous study by Aghajani, Sedaghat, Dargahi, & Pourhossein [2]. The scale of fuzzy numbers in LV was ranging from “very low” to “very high” (see Table 1). Then the scale of fuzzy numbers was converted into crisp numbers using the representation of linear triangulation [7] (see Figure 2).
Figure 2. Linear triangulation representation model by Kusumadewi [7]

Table 1. The linguistic variable for alternative and criteria weighting of Fuzzy SAW algorithm (adapted from Aghajani, Sedaghat, Dargahi, & Pourhossein [2])

<table>
<thead>
<tr>
<th>Fuzzy linguistic variable</th>
<th>Conversion from fuzzy linguistic variable into crisp numbers</th>
</tr>
</thead>
<tbody>
<tr>
<td>Very Low (VL)</td>
<td>{(0.01), (0.01), (0.25)}</td>
</tr>
<tr>
<td>Low (L)</td>
<td>{(0.01), (0.25), (0.50)}</td>
</tr>
<tr>
<td>Normal (N)</td>
<td>{(0.25), (0.50), (0.75)}</td>
</tr>
<tr>
<td>High (H)</td>
<td>{(0.50), (0.75), (1.00)}</td>
</tr>
<tr>
<td>Very High (VH)</td>
<td>{(0.75), (1.00), (1.00)}</td>
</tr>
</tbody>
</table>

Second step
After getting the total sum of alternative and criteria weighting values, the second step was creating a comparison matrix between alternative towards the criteria. For example, the data results from smartphone experienced user, gathered four main criteria about smartphone product valuation. The criteria are: (1) product price, (2) battery lifetime, (3) processor speed, and (4) screen size. After all alternative weighting values were gathered, the sample values can be written like in Table 2.
Table 2. Comparison matrix between alternative towards criteria

<table>
<thead>
<tr>
<th>Criteria/Alternative</th>
<th>Criteria</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>product price</td>
</tr>
<tr>
<td>BlackBerry 9330 CDMA</td>
<td>0.9</td>
</tr>
<tr>
<td>Samsung Galaxy S4</td>
<td>0.3</td>
</tr>
<tr>
<td>LG L9</td>
<td>0.5</td>
</tr>
<tr>
<td>HTC One</td>
<td>0.4</td>
</tr>
</tbody>
</table>

Then, the comparison matrix between alternative towards the criteria can be written as in matrix $M_1$:

$$M_1 = \begin{bmatrix} 0.9 & 0.3 & 0.2 & 0.3 \\ 0.3 & 0.4 & 0.9 & 1.00 \\ 0.5 & 0.7 & 0.7 & 0.7 \\ 0.4 & 0.6 & 0.8 & 0.8 \end{bmatrix}$$

After the $M_1$ values were obtained, the next step was consumers turn to do the criteria weighting process using the scale range like in Table 1, the scale starts from 0.01 to 1.00. The reason of using 0.01 as the smallest value in the scale range is to avoid division by zero which may cause the calculation error.

In example: The consumer wants a smartphone which has several criteria of weighting values like in Table 3.

Table 3. Matrix with criteria weighting

<table>
<thead>
<tr>
<th>Criteria</th>
<th>Weight</th>
</tr>
</thead>
<tbody>
<tr>
<td>Product price (-)</td>
<td>0.60</td>
</tr>
<tr>
<td>Battery lifetime (+)</td>
<td>0.75</td>
</tr>
<tr>
<td>Processor speed (+)</td>
<td>0.82</td>
</tr>
<tr>
<td>Screen size (+)</td>
<td>0.47</td>
</tr>
</tbody>
</table>

*Legend
Negative (-) = i.e. cheaper product price is better
Positive (+) = i.e. bigger screen size is better

Then, from Table 3 it can be formed a matrix called as $(N_1)$ which consists of:

$$N_1 = \begin{bmatrix} 0.60 \\ 0.75 \\ 0.82 \\ 0.47 \end{bmatrix}$$
Third step
Do the normalization for matrix $M_1$, with the following rules (Theorem 1 & Theorem 2):

For the criteria which have negative weight

$$n_{ij} = \frac{r_{ij}^{MIN}}{r_{ij}}$$

**Theorem 1.** The normalization formula for negative criteria

For the criteria which have positive weight

$$n_{ij} = \frac{r_{ij}}{r_{j}^{MAX}}$$

**Theorem 2.** The normalization formula for positive criteria

The results should be:

<table>
<thead>
<tr>
<th>Row</th>
<th>(Product price)</th>
<th>(Battery lifetime)</th>
<th>(Processor speed)</th>
<th>(Screen size)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1</td>
<td>0.33</td>
<td>0.43</td>
<td>0.22</td>
<td>0.30</td>
</tr>
<tr>
<td>1.2</td>
<td>1.00</td>
<td>0.57</td>
<td>1.00</td>
<td>1.00</td>
</tr>
<tr>
<td>1.3</td>
<td>1.00</td>
<td>1.00</td>
<td>0.77</td>
<td>0.70</td>
</tr>
<tr>
<td>1.4</td>
<td>1.00</td>
<td>1.00</td>
<td>0.75</td>
<td></td>
</tr>
<tr>
<td>2.1</td>
<td>0.33</td>
<td>1.00</td>
<td>0.90</td>
<td>0.43</td>
</tr>
<tr>
<td>2.2</td>
<td>0.43</td>
<td>0.57</td>
<td>1.00</td>
<td>0.60</td>
</tr>
<tr>
<td>2.3</td>
<td>0.22</td>
<td>0.77</td>
<td>0.70</td>
<td></td>
</tr>
<tr>
<td>2.4</td>
<td>0.30</td>
<td>0.70</td>
<td>0.75</td>
<td></td>
</tr>
<tr>
<td>3.1</td>
<td>0.30</td>
<td>1.00</td>
<td>0.77</td>
<td>0.60</td>
</tr>
<tr>
<td>3.2</td>
<td>0.43</td>
<td>1.00</td>
<td>0.77</td>
<td></td>
</tr>
<tr>
<td>3.3</td>
<td>0.22</td>
<td>0.77</td>
<td>0.70</td>
<td></td>
</tr>
<tr>
<td>3.4</td>
<td>0.30</td>
<td>0.70</td>
<td>0.75</td>
<td></td>
</tr>
<tr>
<td>4.1</td>
<td>0.75</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Row 4.2 (Battery lifetime) = \frac{0.6}{\text{Max}(0.3;0.4;0.7;0.6)} = \frac{0.6}{0.7} = 0.85  
Row 4.3 (Processor speed) = \frac{0.8}{\text{Max}(0.2;0.9;0.7;0.8)} = \frac{0.8}{0.9} = 0.88  
Row 4.4 (Screen size) = \frac{0.8}{\text{Max}(0.3;1.00;0.7;0.8)} = \frac{0.8}{1.00} = 0.80

Then, the normalization results from matrix \textbf{M}_1 can be written as in matrix \textbf{M}_N 

\[ \textbf{M}_N = \begin{bmatrix} 0.33 & 0.43 & 0.22 & 0.30 \\ 1.00 & 0.57 & 1.00 & 1.00 \\ 0.60 & 1.00 & 0.77 & 0.70 \\ 0.75 & 0.85 & 0.88 & 0.80 \end{bmatrix} \]

Fourth step: 
Do the calculation for \textbf{M}_N and \textbf{N}_1 using SAW algorithm (Theorem 3) 

\[ A_i = \sum N_1 \cdot M_N \]

Theorem 3. The SAW algorithm 

\[ \textbf{N}_1 = \begin{bmatrix} x_1 & \cdots & (x_1,n) \\ \vdots & \ddots & \vdots \\ (x_3,1) & \cdots & (x_3,n) \end{bmatrix}; \textbf{M}_N = \begin{bmatrix} y_1 & \cdots & y_1, n \\ \vdots & \ddots & \vdots \\ (y_3,1) & \cdots & (y_3,n) \end{bmatrix} \]

\[ \textbf{N}_1 \textbf{M}_N = \begin{bmatrix} x_1 y_1 & \cdots & (x_1,n)(y_3,1) \\ \vdots & \ddots & \vdots \\ (x_3,1)y_1 & \cdots & (x_3,n)(y_3,n) \end{bmatrix} \]

\[ A_1 = \{(0.60 \times 0.33) + (0.75 \times 0.43) + (0.82 \times 0.22) + (0.47 \times 0.30)\} = 0.84 \]
\[ A_2 = \{(0.60 \times 1.00) + (0.75 \times 0.57) + (0.82 \times 1.00) + (0.47 \times 1.00)\} = 2.32 \]
\[ A_3 = \{(0.60 \times 0.60) + (0.75 \times 1.00) + (0.82 \times 0.77) + (0.47 \times 0.70)\} = 2.07 \]
\[ A_4 = \{(0.60 \times 0.75) + (0.75 \times 0.85) + (0.82 \times 0.88) + (0.47 \times 0.80)\} = 2.18 \]

Finally, the ranking of smartphone recommendations based on Fuzzy SAW algorithm will be (Table 4):

<table>
<thead>
<tr>
<th>( A_i )</th>
<th>Ranking</th>
<th>Alternatives</th>
</tr>
</thead>
<tbody>
<tr>
<td>( A_2 )</td>
<td>1 (2.32)</td>
<td>Samsung Galaxy S4</td>
</tr>
<tr>
<td>( A_4 )</td>
<td>2 (2.18)</td>
<td>HTC One</td>
</tr>
<tr>
<td>( A_3 )</td>
<td>3 (2.07)</td>
<td>LG L9</td>
</tr>
<tr>
<td>( A_1 )</td>
<td>4 (0.84)</td>
<td>BlackBerry 9330 CDMA</td>
</tr>
</tbody>
</table>
Phase one: the analysis steps (B)

In the smartphone selection criteria, there were many aspects can be considered as a measurement indicator on criteria weighting process. The aspects are such as: innovative feature, brand image, product price, product recommendation among consumers (like product review), durability, and product portability [11]; advertising and after sales service [10]; product design (quality aspect) [4]. In this study choose ten criteria as a fundamental weighting process; those criteria are based on the combination of literature review and direct interview (survey) from ten smartphone experienced users. From direct interview, the data which were generated qualitatively can be deducted that innovative features, product price, battery lifetime, and camera quality were most chosen criteria by the consumer. Meanwhile from literature review the other six criteria were display/resolution, weight, dimension, processor speed, built in memory, and built in storage Mokhlis & Yaakop [11], Mokhlis [10].

Phase two: design of DSS application workflows

A. Use-Case Diagram:

![Use Case Diagram](image_url)

**Figure 3. Use Case Diagram**

Explanation of **Figure 3**: (1) Administrator performs weighting criteria using a spreadsheet application like MS. Excel and get the normalized result. (2) Administrator define the normalized results into the application program backend (3) Potential buyers choose smartphones based on the criteria that they need and the FSAW algorithm provide the recommendations. The description of DSS application workflow is more fully described in two section, such as: Sequence Diagram – Buyer Perspective and Sequence Diagram – Administrator Perspective.
B. Sequence Diagram – Buyer Perspective:

![Sequence Diagram – Buyer Perspective](image)

**Figure 4. Sequence Diagram – Buyer Perspective**

Explanation of **Figure 4**: The DSS application workflow is reviewed from the perspective of a potential buyer, then the phase is: (1) using the trackbar slider in the application interface, potential buyers can choose the smartphone category based on the most criteria they need. (2) After a potential buyer gives the weighting criteria by sliding the trackbar function and clicking the recommendations button, then it will display a series of recommendations that fit the potential buyer needs. The recommendation results were based on fuzzy SAW weighting algorithm as written in **Theorem 3**.

C. Sequence Diagram – Administrator/Seller Perspective

![Sequence Diagram – Administrator/Seller Perspective](image)

**Figure 5. Sequence Diagram – Administrator/Seller Perspective**
Design of single user decision support system model

Explanation of Figure 5: The DSS application workflow is reviewed from the perspective of an administrator, then the phase is: (1) the administrator has had a comparative values between smartphones. The values obtained were from the benchmark results of various technical reports from GSMArena as well as HWM Singapore magazine as in Table 2. Then the value obtained, normalized in MS. Excel uses Theorem 1 and Theorem 2. (2) In this step, the value obtained by administrator has been normalized. (3) Furthermore, the normalized values were defined in the DSS application back-end program as the indicator of criteria weighting.

Phase two: design of DSS application interface

To implement a computation algorithm (e.g. FSAW, AHP, etc.) actually can use spreadsheet applications such as MS. Excel, as done by Bonometti [5] when he made an application of DSS / ES (Expert System) Shell to help students learn to manage a project. However, in order to attract more attention and facilitate respondents to select the appropriate smartphone products, the prototype of DSS application interface which is simple and easy to use, needs to be developed [1].

![DSS application interface (A)](image)

Figure 6. DSS application interface (A)

Explanation of Figure 6: the property type of point (1) is text label, which serves as a marker of the weighting criteria; the property type of point (1A) is text label, which serves as a number counter whose value can be increased or decreased in accordance with the trackbar movement (point 2); the property type of point (1B) is text label which serves as measurement indicator, for example: when the indicator marked as "cheaper" means the potential buyer expects the smartphone category that have less cost than the others. Likewise, the indicator marked as
"Lighter" is when the potential buyer expects the smartphone has lighter weight than the others; the property type of point (1C) is text label which serves as a pointer of first and second recommendation options; point (1D) is a text label which has the function to showing the calculation results (recommendation options). In point (2) the “trackbar” serves as a slider to increase or decrease the value of weighting criteria. Point (3) contains three buttons, such as: (a) “CLEAR” button is used to reset the weighting result in DataGridView property (point 4), (b) “Clear Track Bar” is used to reset the trackbar position and value counter into initial value (the initial value is 0), (c) “GET Your Phone” button is used to execute the FSAW algorithm and then the calculation result will be shown in the DataGridView property (point 4) and in the text label property (1D). Point (5) is text label which has the function as a information “billboard” only.

Phase three: the DSS application backend coding

When running the DSS application program, a potential buyer is conditioned to choose a smartphone which is cheap and has long-life battery endurance characteristics, then after moving the TrackBar slider and clicking the “GET Your Phone” button the potential buyer will get the result as shown in Figure 7.

![Figure 7. DSS application interface (B)](image)

Explanation of Figure 7:

(a) TrackBar: if the trackbar slider is moved, then it will be an increase or a decrease value of the counter. The pseudocode to move the counter value is described in Conceptual 1.
Design of single user decision support system model

Var min, max: double
Begin
    TrackBarN setValue min=1, max=100
    TrackBarN setSliderChange=1
    LabelN.Text = TrackBarN.Value
End

Conceptual 1. The TrackBar slider pseudocode

(b) “GET Your Phone” button: if this button is clicked, then the calculation of Theorem 3 will be processed. The code of Theorem 3 and the function to find the biggest value in calculation result is described in Conceptual 2.

Pseudocode:
!-- for Theorem 3
MultiplyMatrices(X1, Y2): double
Var SigmaSum: double
Begin
    For loop Xrow and Ycolumn
    Begin
        SigmaSum += (Xrow*Ycolumn)
    End
End
Return SigmaSum
End

!-- biggest number
Arr(n): double
Var Max = Arr(0)
Var Max2 = Arr(n)
Begin
    For loop counter to Arr(n)-1
    Begin
        If (Arr(counter) > Max) Then
            Max = Arr(counter)
        Else If Max2 < Arr(counter) Then
            Max2 = Arr(counter)
        End If
    End
End

Conceptual 2. Pseudocode of Theorem 3 and the function to find the biggest number in calculation result

(c) “CLEAR” button: if the button is clicked, then the weighting result of smartphone on DataGridView (point 4) and the recommendation result on point (1D) will be reset. The pseudocode is described in Conceptual 3.

Pseudocode:

(c) "CLEAR" button: if the button is clicked, then the weighting result of smartphone on DataGridView (point 4) and the recommendation result on point (1D) will be reset. The pseudocode is described in Conceptual 3.
Conceptual 3. Pseudocode to reset the recommendation result

```
Begin
    DataGridViewPoin4.Rows.Clear()
    Label1D.Text = "******" (CharType: String)
End
```

(d) “Clear Track Bar” button: if the button is clicked, then the TrackBar slider with its counter will be reset into initial value (initial value is 1/100). The pseudocode is described in Conceptual 4.

```
Var reset=1
Begin
    TrackBarN.Value = reset
    Label1A.Text = "1" (CharType: String)
End
```

Conceptual 4. Pseudocode to reset the TrackBar and its counter value

Phase four: testing of DSS application

In this study the phase of laboratory testing application program such as alpha-beta testing was not done, the application was tested directly to the DSS research object (potential buyers). However, there are no specific rules stating that every application program must go through a series of prototype tests before testing it directly to the sample. This method refers to the previous studies of DSS application development conducted by Bonometti [5] and Coutinho-Rodrigues, Simao, & Antunes [3] which did not specifically discuss about software pre-test in their research article.

Conclusion

The result of DSS application test displayed in a separate report proves that two of 27 smartphone purchases were initiated by the DSS. The small number of purchases may be caused due to the aesthetic aspects of the Human Computer Interaction which are ignored in the user interface design phase. As well as, the criteria and alternative weighting in this DSS application are static. However, in further development, DSS application can be used as an effective marketing tool. For example, there is a DSS application for restaurant selection problems built on mobile application. Using a DSS, consumers can simplify the confusing aspects like choosing a food menu, find out the availability of food menu, or table reservations only with their “touches”. In addition, in a few seconds the DSS gives them a several recommendations of the “their choice” restaurant. The limitation in this study is the accuracy analysis of fuzzy SAW algorithm which
Design of single user decision support system model

was not conducted. Imprecision of weighting process may cause imbalance of calculation result. Therefore, the weighting process in spreadsheet application should not have any mistakes.

References


Received: December 15, 2013