PAIRWISE COMPARISON AS A RANKING IN THE SEARCH PORTALS FOR DECISION-SUPPORT

Sergio Silva Ribeiro[1]
[1]Pontifícia Universidade Católica - PUCPR, 80215-901, Curitita, PR, Brasil.
ssr.informatica@gmail.com

Abstract: *Embedded systems are present in a lot of solutions making our lifestyle safe, easier and comfortable. Pairwise comparison theory was a milestone in decision making science because the modern society has a lot of situations where decision making is mandatory by boards of Governors, committees, city councils, panels of experts, etc. A Python implementation of pairwise comparison was presented having Airbnb as a case study base representing a search portal for decision support situation.*

Keywords*: Analytical Hierarchical Process, Decision-making, Information. Python.*

INTRODUCTION

The contemporary city is characterized by the informational phenomenon, with the propagation in its environment of electronic devices, digital networks of communication and information, as well as the internet. Terms such as Digital City, Virtual City, Telecity, Smart City, among others are used to refer to this new city that appears in the informational wave and the network society [14], [1].

Embedded systems are present in a lot of solutions making our lifestyle safe, easier and comfortable. The embedded system is a computer system designed to control a lot of different devices like cellular, washing machine, microwave, television, lift, rocket, life support machine, and much more [13].

A decision-making methodology was introduced by Thurstone in 1927, his methodology was based on the pairwise comparison (PC). It was a milestone in decision making science because the modern society has a lot of situations where decision making is mandatory by boards of Governors, committees, city councils, panels of experts, etc [6].

In the minor or major level embedded systems are used to decision making having been used since to control room temperature as well to control airplane flying. In all of these cases are used a combination of hardware and software [7].

Python is a modern program language used to a large number of purposes, also is multi-platform and can be embedded in small or minimal hardware devices, depending on how limiting the devices actually are [8]. Therefore, the main purpose of this work was to present a PC algorithm developed in Python that could be used as an embedded code in any solution as a PC module.

BACKGROUND

The PC is described by [4] "as a powerful inference tool and knowledge acquisition technique in knowledge-based systems". The Analytical Hierarchical Process (AHP) can be employed integrated with PC to classify and organize decision options by importance or preference order [12], [3]. Preferences of n stimuli are established to be estimated its

priorities. The values of stimuli called relative weights are distributed into a reciprocal square matrix A = [aij]

$$A = \begin{bmatrix} 1 & a_{12} & \dots & a_{1n} \\ 1/a_{12} & 1 & \dots & a_{2n} \\ \vdots & \vdots & \vdots & \vdots \\ 1/a_{1n} & 1/a_{2n} & \dots & 1 \end{bmatrix}$$

where $a_{ij} > 0$ expresses the relative importance of two stimuli i and j. Also, each a_{ij} has the reciprocal value $a_{ij} = 1 / a_{ij}$. A PC matrix A is called consistent if $a_{ij} \cdot a_{jk} = a_{ik}$ for every i, j, k = 1, . . . , n. While every consistent matrix is reciprocal, the converse is false in general. There has been an ongoing discussion regarding the better method for finding solutions to a pairwise comparison matrix, but because simplicity and mathematical elegance will be used Geometric Means (GM), and the Eigenvector (EV) methods [2], [11].

The GM is given by

$$v_i = \left(\prod_{j=1}^{n} a_{ij} \right)^{1/n}$$

and the EV is given by

$$w_i = \frac{\sum_{i=1}^{n} a_{ij}}{n}$$

Considering an example with three stimuli where a expresses stimulus 1, b expresses stimulus 2, and c the stimulus 3 still considering that a expresses preference over b, and b expresses preference over c this example would be represented by reciprocal matrix like that:

$$A = \begin{bmatrix} 1 & a & b \\ 1/a & 1 & c \\ 1/b & 1/c & 1 \end{bmatrix}$$

After defined the reciprocal matrix is needed to verify matrix consistency. [5] proposed a way to do the verification using what was called a new consistency measure (CM).

PROPOSED METHOD

For this work were used bibliographic research, and case study methodology. In the bibliographic research were studied papers referent AHP theory and its practical application to make decision support, also were studied others topics related to this work

theme. In addition to bibliographic research was used a case study for help to comprehension regarding AHP practical application [15].

Depending on the purpose are defined attributes and its weights to use into AHP calculation to find the percentage of the attributes to decision support. For the purpose of this work were used the attributes and its respective weights shown in Table 1. These attributes were extracted from Airbnb property information result [16].

Table 1 – Attributes and its weights of decision

	Extreme favors	Very strong favor	Strongly favors	Slightly favors	Equal	Slightly favors	Strongly favors	Very strong favor	Extreme favors	
	9	7	5	3	1	3	5	7	9	
Price		X								Cleanliness
Price			X							Location
Cleanliness						X				Location

Table 1 show that were used price, cleanliness, and location how attributes which were enough for this work.

Defined attributes and its weights of the decision, the next step was to process the calculation based on AHP methodology. For it was developed a computer program in Python language where was inputted all data to processing and calculation (Algorithm 1) [9], [10].

Algorithm 1 - Python program develped for this work

```
1:      from scipy import stats as st
2:      import numpy as np
3:
4:      # Array columns: sequence, Price, Cleanliness, Location, index value
5:      options = np.array([       [1,  42, 0, 0, 0.0],
6:                                 [2,  50, 4, 5, 0.0],
7:                                 [3,  50, 5, 5, 0.0],
8:                                 [4,  41, 4, 3, 0.0],
9:                                 [5,  48, 5, 4, 0.0],
10:                                [6,  48, 0, 0, 0.0],
11:                                [7,  41, 5, 5, 0.0],
12:                                [8,  45, 0, 0, 0.0],
13:                                [9,  48, 3, 4, 0.0],
14:                                [10, 41, 0, 0, 0.0],
15:                                [11, 50, 4, 4, 0.0],
16:                                [12, 48, 4, 5, 0.0],
17:                                [13, 47, 4, 4, 0.0],
18:                                [14, 48, 0, 0, 0.0],
19:                                [15, 35, 0, 0, 0.0],
20:                                [16, 50, 0, 0, 0.0],
21:                                [17, 45, 0, 0, 0.0],
22:                                [18, 45, 0, 0, 0.0],
23:                                [19, 48, 0, 0, 0.0]])
24:
25:     a = np.array([[1    , 7, 5    ],
26:                   [0.143, 1, 0.333],
```

```
27:                     [0.200, 3, 1    ]])
28:
29:       a_sum = np.sum(a,axis=0)
30:
31:       gmean = np.array( [st.gmean(a[0]),
32:                          st.gmean(a[1]),
33:                          st.gmean(a[2])])
34:
35:       priority = np.array( [ gmean[0]/np.sum(gmean),
36:                              gmean[1]/np.sum(gmean),
37:                                 gmean[2]/np.sum(gmean)] )
38:
39:       lambdamax = np.sum( np.multiply( a_sum,priority ) )
40:
41:       CR = ((lambdamax-3)/2)/0.58
42:
43:    options[:,4] = (options[:,1] * priority[0]) + ((5-options[:,2]) * priority[1]) + ( (5-
       options[:,3]) * priority[2])
44:
45:       index_value = np.argsort(options[:,4])
46:
47:       print a
48:       print a_sum
49:       print gmean
50:       print priority
51:       print lambdamax)
52:       print CR
53:       print options[index_value]
54:
```

In the Algorithm 1 was presented the algorithm responsible for processing, calculating and showing results of information inputted. In line 1 and 2 was included Scipy and Numpy libraries to be used in the program. In the lines from 5 to 27 was defined the dataset, attributes, and parameters used by the program. The calculation was done in the commands found between the lines from 28 to 45. The results are performed and classified regarding weights of each option and are printed on the screen by commands from the line 47 to 53.

RESULTS AND DISCUSSION

The Airbnb was founded in 2008 in San Francisco, California by Brian Chesky and Joe Gebbia. Currently, Airbnb is present in more than 34.000 cities in 191 countries with more than 60.000 guests, and connecting people around the world offering service to help to find an accommodation [16].

Entering the desired destination, dates, and additional filters a guest can seek a room or property to rent. After submit desired principal data of filter is returned a result list with all found rooms presenting other property features and characteristics like price, general location, pictures, rate, and availability. The guest chooses the room and book using Airbnb's request and payment systems [17].

For the purpose of this case study were used the following filters: Entire home/apt; 1 Guest; Price Range = $35CAD to $50CAD; Neighborhoods = Downtown Toronto; Amenities = Wireless Internet; and Property Type = Apartment. The searching was made by June of 2016. After submitting filters to Airbnb were returned data shown in Table 2.

Table 2 – Airbnb search results

Option	Price	Cleanliness	Location
1	42,00	0	0
2	50,00	4	5
3	50,00	5	5
4	41,00	4	3
5	48,00	5	4
6	48,00	0	0
7	41,00	5	5
8	45,00	0	0
9	48,00	3	4
10	41,00	0	0
11	50,00	4	4
12	48,00	4	5
13	47,00	4	4
14	48,00	0	0
15	35,00	0	0
16	50,00	0	0
17	45,00	0	0
18	45,00	0	0
19	48,00	0	0

Criteria of a decision were processed by own program developed in Python (Figure 1) to calculate rates of priorities based on AHP methodology.

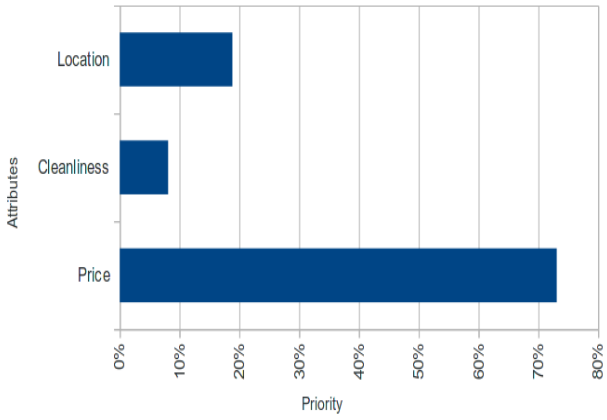
Figure 2 shows partial results of processing. In the Figure is shown Priority column with a rate of criterion based on the matrix of weights, supported by sum and geometric mean calculation, as well is shown lambda maximum calculation and consistency ratio to check the consistency of criteria.

Figure 2 – (a) AHP results demonstration, and (b) percentage of attributes Priorities

	Price	Cleanliness	Location	G-Mean	Priority
Price	1	7	5	**3,2711**	73,06%
Cleanliness	0,143	1	0,333	**0,3625**	8,10%
Location	0,2	3	1	**0,8434**	18,84%
Σ	**1,343**	**11**	**6,333**	**4,4770**	**100,00%**

λ max	3,06
CR	5,60%

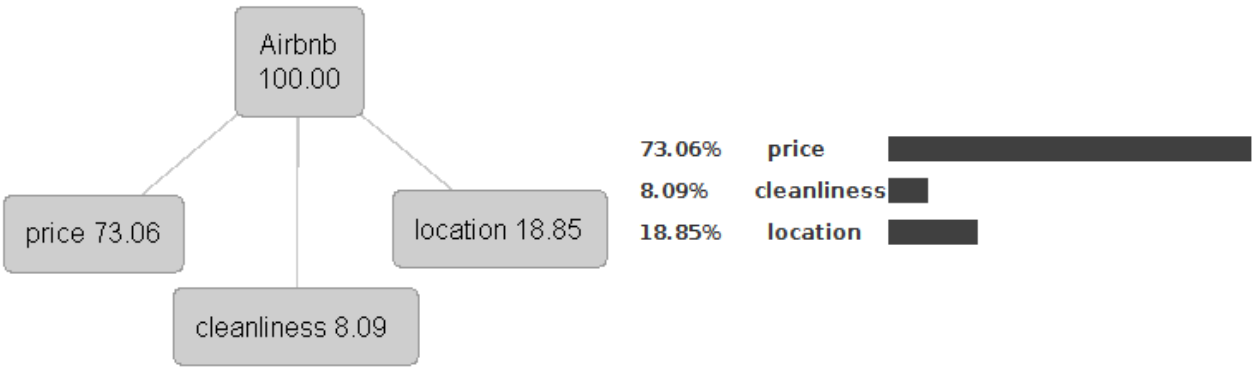
(a)



(b)

Results of the Python program (Figure 2) were compared with Concluder program results (Figure 3) to check consistency and results.

Figure 3 – Concluder AHP results, (a) percentage and (b) chart representation



(a) (b)

After that rates of priorities were applied over data results (Table 2) from Airbnb to find out the better option to decision support. The results are shown in Table 3.

Table 3 – Options sorted by index value

Option	Price	Cleanliness	Location	Index value
15	35,00	0	0	26,15
7	41,00	5	5	29,96
4	41,00	4	3	30,41
10	41,00	0	0	31,30
1	42,00	0	0	32,34
8	45,00	0	0	34,23
18	45,00	0	0	34,23
17	45,00	0	0	34,23
13	47,00	4	4	34,61
12	48,00	4	5	35,15
5	48,00	5	4	35,26
9	48,00	3	4	35,42
6	48,00	0	0	36,42
14	48,00	0	0	36,42
19	48,00	0	0	36,42
3	50,00	5	5	36,53
2	50,00	4	5	36,61
11	50,00	4	4	36,80
16	50,00	0	0	37,88

In Table 3 the data results are presented in the index value order (5th column) that is an information based on the weights of price, cleanliness, and location previously defined.

CONCLUSION

A Python implementation of pairwise comparison was presented having Airbnb as a case study base representing a search portal for decision support situation.

This work was limited by the dataset, group of filters, and the attributes used. However, those were enough to demonstrate how PC and AHP concepts can be applied with success in real and simple situations.

It is a contribution for PC, AHP, and computer science application studies by showing how to apply theoretical concepts to the practical solving problem.

REFERENCES

[1] CASTELLS, Manuel. *The rise of the network society: The information age: Economy, society, and culture.* John Wiley & Sons, 2011.

[2] Crawford, G. The Geometric Mean Procedure for Estimating the Scale of Judgement Matrix. *Mathematical Modelling*, Vol. 9, pp. 327-334, **1987**.

[3] Gomes, L. F. A. M.; Gomes, C. F. S. *Tomada de Decisão Gerencial: Enfoque Multicritério.* 5. ed. São Paulo: Atlas, **2014**.

[4] Herman, M. W.; Koczkodaj, W. W. A Monte Carlo study of pairwise comparison. *Information Processing Letters*, v. 57, n. 1, p. 25-29, **1996**.

[5] Koczkodaj, W. W. A new definition of consistency of pairwise comparisons. *Mathematical and computer modelling*, v. 18, n. 7, p. 79-84, **1993**.

[6] Koczkodaj, W. W. CONCLUDER system for decision making*. Researchgate*, 2016. Available in: <https://www.researchgate.net/publication/301882911_intro_to_Concluder>

[7] Künzli, S.; Thiele, L.; Zitzler, E. Multi-criteria decision making in embedded system design. *System On Chip: Next Generation Electronics*, p. 3-28, 2006.

[8] Lutz, M. *Learning python.* O'Reilly Media, Inc., **2013**.

[9] Ribeiro, S. S. *Métodos computacionais aplicados à agricultura: Python & Weka.* Novaterra Editora e Distribuidora LTDA, **2016**.

[10] Ribeiro, S. S.; Ferrasa, A.; Falate, R. USING PYTHON WITH SIMPLECV TO DETECT A CORN KERNEL IN DIGITAL IMAGE. *Iberoamerican Journal of Applied Computing,* vol. 4, n.o 2, **2014**.

[11] Saaty, T. L. A Scaling Method for Priorities in Hierarchical Structure. *Journal of Mathematical Psychology*, Vol. 15, 234-281, **1977**.

[12] Saaty, T. L. Decision making with the analytic hierarchy process. *International journal of services sciences*, v. 1, n. 1, p. 83–98, **2008**.

[13] Ševek, J. Intelligent Embedded Systems. *Information Sciences & Technologies: Bulletin of the ACM Slovakia*, v. 4, n. 1, **2012**.

[14] Toffler, A. *The third wave*. New York: Bantam books, **1981**.

[15] Yin, R. K. Case Study Research: Design and Methods (Applied Social Research Methods, Vol. 5). Sage Publications, Beverly Hills, CA. *Rick Rantz Leading urban institutions of higher education in the new millennium Leadership & Organization Development Journal*, 23(8), **2002**.

[16] Airbnb. Web Site Oficial. available in <http://airbnb.com> access in <June, 2016>

[17] Edelman, B. G.; Luca, M. Digital discrimination: The case of Airbnb. com. Harvard Business School, 2014.