

Bryant University

## Bryant Digital Repository

---

Information Systems and Analytics Faculty  
Conference Proceedings

Information Systems and Analytics Faculty  
Publications and Research

---

7-10-2023

### On Teaching Multi-Criteria Decision Making with a Robot Assistant

Chen Zhang

Hakan Saraoglu

David A. Louton

Follow this and additional works at: <https://digitalcommons.bryant.edu/cis-conf-proceedings>



Part of the [Artificial Intelligence and Robotics Commons](#), and the [Finance and Financial Management Commons](#)

---

# On Teaching Multi-Criteria Decision Making with a Robot Assistant

Chen Zhang  
Information Systems and Analytics  
Bryant University  
1150 Douglas Pike  
Smithfield, RI 02917  
[czhang@bryant.edu](mailto:czhang@bryant.edu)  
ORCID ID: 0000-0002-7403-4866

Hakan Saraoglu  
Finance Department  
Bryant University  
1150 Douglas Pike  
Smithfield, RI 02917  
[saraoglu@bryant.edu](mailto:saraoglu@bryant.edu)  
ORCID ID: 0000-0002-0419-3669

David Louton  
Finance Department  
Bryant University  
1150 Douglas Pike  
Smithfield, RI 02917  
[dlouton@bryant.edu](mailto:dlouton@bryant.edu)  
ORCID ID: 0000-0002-4908-4853

## Abstract

We propose a system and method for a robot assistant for teaching multi-criteria decision making (MCDM). Through questions and answers in natural language, the robot assistant learns the user's preferences on multiple criteria involving a selection decision and makes recommendations using data on each criterion and the learned user preferences. It will include a use-case demonstration where NAO the robot will assist a human in forming a simple portfolio of mutual funds. Presenters will illustrate the architecture of the robot assisted MCDM and describe a method that is extensively used to structure complex decision problems and has been applied to a variety of problems in a diverse set of disciplines, such as selecting a project, selecting a life insurance contract, selecting public relations firms, deciding on library acquisitions, hostage negotiations, selecting sites for wildlife management, and selecting a nonprofit for donation.

**Keywords:** robot assistant, emerging technology for learning, multi-criteria decision making

## Introduction

While the use of speech enabled smart devices that serve as intelligent assistants is gaining acceptance, it is becoming important to find ways to advance these devices beyond the task of question answering and information discovery, like ChatGPT. An important next level in this endeavor is to make intelligent assistants that aid human users in making selections from a set of competing alternatives based on a complicated set of selection criteria.

We present a system for natural language interactive specifications of the user's preferences and a method of checking their consistency, determining the relative importance weights of attribute preferences that can be used by the system to search a dataset of the attribute scores of competing alternatives, and returning a recommendation based on the user's stated preferences. The natural language question-and-answer set and the attribute values for the competing alternatives can be generated from existing datasets by a human(s) or by using machine learning methods, or by a combination thereof. This is an innovative way of teaching MCDM in business schools.

### Design of the Robot Assistant

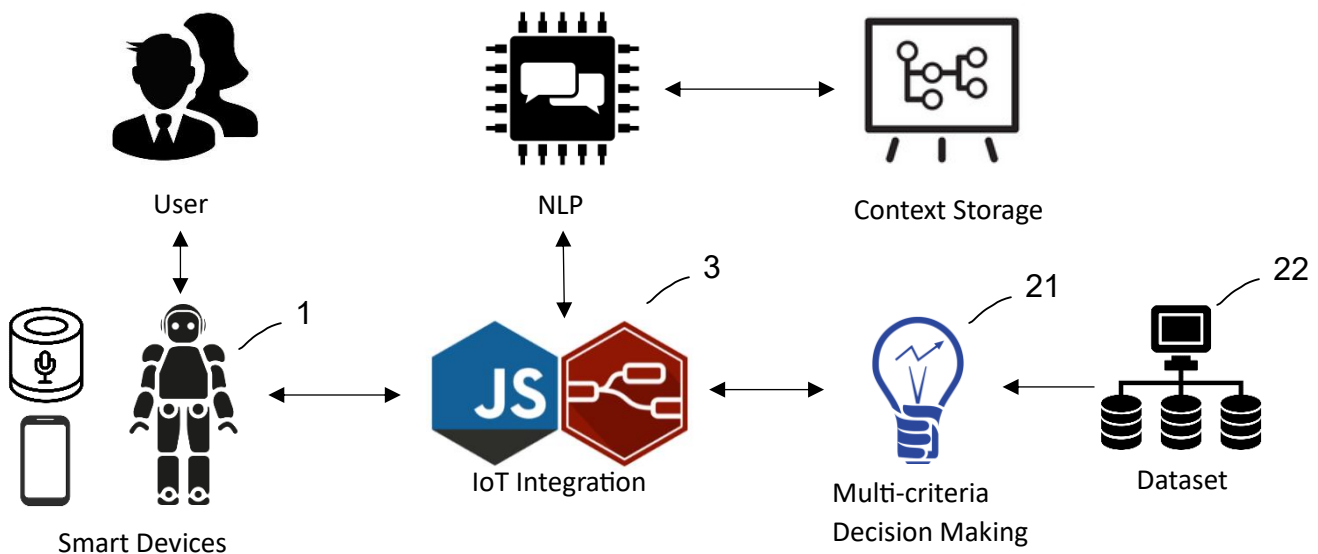


Figure 1. Robot Assistant System Architecture  
Components of the system architecture with the numerical labels are explained in detail below.

In Figure 1, the smart device front-end and the back-end components are integrated on Internet of Things (IoT) platforms such as Node.js and IBM Node-red. Back-end components include:

1. The Natural Language Processing (NLP) based chat-bot module with context information storage in cloud or edge-devices.
2. Multi-criteria decision-making module accessing dataset, taking input and returning decisions.
3. Internet of Things (IoT) integration module forwarding messages between modules.

In an embodiment,

1. The user device component interacts with the user via a microphone and speaker. Examples include humanoid such as Softbank Robotics NAO v6 running Choregraphe 2.8 (with Python 2.7), cellular phones, smart speakers such as Amazon Alexa™ Echo or a personal computer with microphone and speaker;
11. The NLP component is implemented by a Speech to Text (STT) engine. Examples include IBM Watson™ Assistant, Google Dialogflow™, Amazon Alexa™ Voice Service, Microsoft Azure™ STT or other open-source STT engines;

12. Context Storage is implemented via a relational or non-relational database. Examples include IBM Cloudant™ non-relational database and Oracle database;
21. The Multi-Criteria Decision-Making component is implemented using a programming language tool with web sockets capacity. Examples include Jupyter Notebook with Python 3.5 kernel running on IBM Watson Studio, Java or C++;
22. The dataset is stored in a file, such as CSV file in Watson Studio, or in local or network accessible databases.
3. The IoT integration module runs on web integration tools, such as IBM Node-Red (based on Node.js) or Amazon Web Service™ (AWS) IoT;

Figure 2 provides an illustrative flow-chart of an implementation of the MCDM module. One possible embodiment uses the Python programming language in Jupyter Notebook running from IBM Watson Studio. Documentation of IBM Watson Studio can be found on the IBM website<sup>1</sup>. The embodiment first loads the dataset from a data table and creates the Pandas data frame in memory, including all the related categories and attributes, or dialog related data fields. The embodiment uses the intent of the user as an input and invokes a set of handlers implemented as functions in Python. First, the MCDM function is invoked. The MCDM function checks the consistency of the user's pairwise comparisons of selection criteria with the method proposed by Saaty (1977, 1980), then, if necessary, it corrects the consistency of the user's preferences, using the algorithm proposed by Xu and Wei (1999), then it sets up a decision matrix consisting of competing alternatives and selection criteria, normalizes the matrix, and weights the elements of the matrix based on the user's preferences and processes the resulting normalized weighted matrix to produce a ranking of the competing alternatives as described in the corresponding steps of the AHP-TOPSIS algorithm below. Finally, the handler function formulates and returns a verbal reply, which includes an answer to the user's request.

### **A Summary of the AHP-TOPSIS Algorithm**

- Step 1. Determine decision criteria and the competing alternatives from which a selection will be made.
- Step 2. Identify the set of criteria that are considered as a benefit and the set of criteria that are considered as a cost.
- Step 3. Obtain pairwise comparisons of the decision criteria from the decision-maker (the user) in the form of a square matrix.
- Step 4. Check the pairwise comparisons of the user for consistency and, if necessary, adjust them until they are consistent.
- Step 5. Calculate relative importance weights of the decision criteria from the matrix of pairwise comparisons.
- Step 6 Construct a decision matrix, with rows representing the competing alternatives and columns representing the decision criteria.
- Step 7. Normalize the decision matrix.
- Step 8. Apply the weight vector obtained in Step 5 to the normalized decision matrix.
- Step 9. Determine the worst possible condition and the best possible condition for each competing alternative under each decision criterion using the decision matrix.
- Step 10. Obtain the separation measure from the best possible and the worst possible conditions.
- Step 11. Calculate the relative closeness to the worst condition for each competing alternative.
- Step 12. Rank the competing alternatives based on their relative closeness scores.
- Step 13. Communicate to the user the highest ranked alternative(s) as the solution to the selection problem.

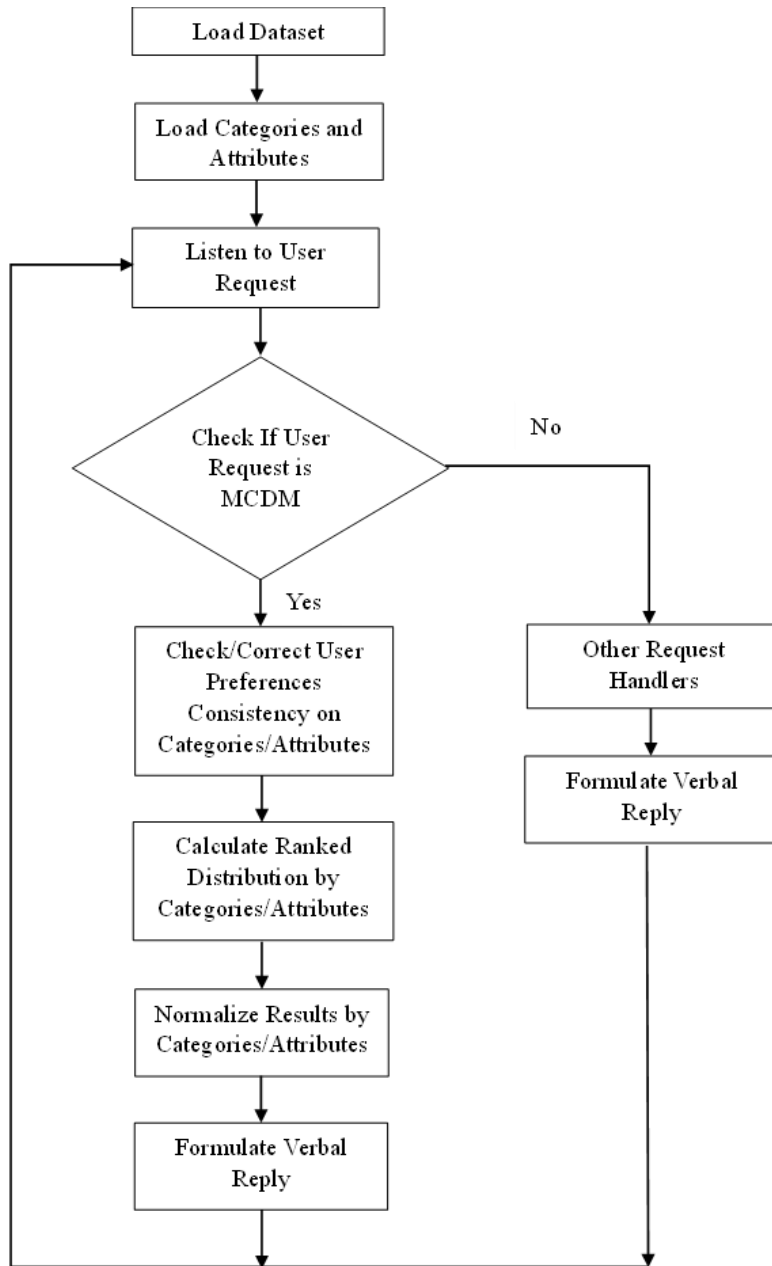


Figure 2. The Multi-Criteria Decision Making with Auto-correction and Optimization Module

A sample dialog is illustrated below:

### **A Sample Dialog to Select a Category of Mutual Funds with Multiple Criteria**

You : Mutual funds please

Bot : "What is your mutual fund objective?"

You : income funds

Bot : "How many income funds to select?"

You : 2

Bot : "Great. Let us now go over your preferences on some mutual fund metrics. Which is more important: Expense Ratio versus Manager Tenure?"

You : manager tenure

Bot : "How much more important on a scale of 1 to 9, with 9 being most important? "

You : 8

Bot : "Which is more important: Expense Ratio versus Sharpe Ratio? "

You : expense ratio

Bot : "How much more important on a scale of 1 to 9, with 9 being most important? "

You : 5

Bot : "Which is more important: Manager Tenure versus Sharpe Ratio? "

You : manager tenure

Bot : "How much more important on a scale of 1 to 9, with 9 being most important? "

You : 8

Bot : "Your fund criteria preferences were inconsistent but were automatically adjusted. Here are your importance weights for fund metrics. 16.61 percent weight on Expense Ratio, 77.71 percent weight on Manager Tenure, 5.68 percent weight on Sharpe Ratio. Here is the selection of mutual funds. 43.98 percent OCCIX, 56.02 percent GIPCX. "

### **Conclusion**

We designed a robot assistant system to teach MCDM. The system can be used for demonstration of the method and different applications in decision making in business school courses. This approach enhances the media used in classrooms to become interactive and dynamic. It helps to inspire students to apply theories they learned to practical purposes. In the new age of ChatGPT based information retrieval, an innovative decision-making pedagogy through chatbot design improves students' creativity and competency.

### **References:**

Saaty, T. L. 1977. A scaling method for priorities in hierarchical structures. *Journal of Mathematical Psychology*. 15: 234-281.

Saaty, T.L. 1980. *The Analytic Hierarchy Process*. McGraw-Hill, New York.

Xu, Zeshui, Wei, Cuiping, "A consistency improving method in the analytics hierarchy process." *European Journal of Operational Research*. 116 (1999) pp. 443-449.

---

<sup>i</sup> <https://dataplatform.cloud.ibm.com/docs/content/wsj/getting-started/overview-ws.html>, last accessed April 3, 2020.