Chapter 5: Content Prioritization through Social Opinion

Incorporating social opinion into designing an e-learning course increases its acceptability in society. This requires elicitation of social opinion from various stakeholders associated with the system. However, stakeholders are disparate in their perception towards the intricacies of the system, leading to generation of numerous assorted ideas. A need is, therefore, felt for assimilation of these ideas to bring congruency into the system. The process of compiling social opinion and utilizing it to prioritize the e-learning content is referred to as Externalization phase in the KMeLS model. This chapter presents an algorithm PARSeL (Prioritizing Alternatives using Recommendations of Stakeholders in e-Learning) to prioritize e-contents using stakeholder recommendations through Analytic Hierarchy Process (AHP) and Fuzzy Modeling.

5.1 Introduction

Content creation plays an important factor in acceptance of an e-learning course and in turn leveraging the e-learning system as a whole. Out of many areas of concern of this process, the process of content designing forms a significant one. There is a need to design content that make the learners informed of the social applicability of the knowledge being acquired by them (Angehrn, Nabeth and Roda, 2001). Knowledge construction is a social process of information sharing, negotiating, revising and agreement achieving based on social constructivism (Wanga, Woo and Zhaob, 2009). Capuruco and Capretz (2009) reflect that traditional methodologies are insufficient in dealing with the complications that surface due to incorporation of social presence into interactive learning systems. Research to increase adaptability of e-learning systems is in progress. However, not much work has been done in literature for inclusion of social opinion into content designing.
Society and content have direct bearing on each other, making inclusion of social opinion into the process of content creation a necessary aspect. Incorporation of expert opinion in designing a system is quite essential, yet it tends to introduce inconsistency into the decision making process due to possibility of imprecision and non-uniformity in the perceptions of various experts. The ideas gathered from the participants may be extremely valuable, yet it might not always be feasible to adopt and implement all of them together either due to lack of resources or they may be conflicting to each other while implementation. This restriction entitles the problem of prioritizing a list of alternatives on the basis of multiple criteria to be classified as a Multi Criteria Decision Making (MCDM) problem (Triantaphyllou et al, 1998; Doumpos and Zopounidis, 2002). Analytic Hierarchy Process (AHP) (Saaty, 1994) is an MCDM technique that provides solutions to complex problems by considering and prioritizing multiple conflicting objectives and bringing various stakeholders into a consensus over them (Geldermann et al, 2009). However, MCDM problems are uncertain in nature as they are associated with human perception. Hence, the best method of dealing with such kind of problems is fuzzy modeling. Fuzzy modeling is apt in handling imprecision or uncertainty arising due to vagueness in human thoughts and perceptions (Bozbura, Beskese and Kahrman, 2007).

The following section presents related research work conducted in implementing MCDM and fuzzy approaches to prioritize various characteristics involved in designing efficient e-learning systems.

5.2 Related Work

Several researchers have employed AHP and fuzzy techniques to evaluate e-learning systems against multiple factors (Colace and De Santo, 2008; Fazlollahtabar and Mahdavi, 2009; Lin, 2010; Ray and Shu-Li, 2010; Mehragan et al, 2011; Syamsuddin, 2012; Bhuasiri et al, 2012). Begićević and Divjak (2006) identified five important factors for e-learning implementation in higher education: human resources, basic ICT infrastructure for e-learning, specific ICT infrastructure for e-learning, strategic readiness for e-learning implementation and legal/formal readiness for e-
learning implementation. They concluded that the experts involved in the survey did not recognize the importance of intellectual property rights and standardization of digital educational materials. Tzeng, Chiang and Li (2007) proposed a hybrid model that addressed the independent as well as dependent relations of the evaluation criteria. The AHP and the fuzzy integral methods were used for synthetic utility in accordance with subjective perception environment.

Shee and Wang (2008) proposed a multi-criteria methodology from the perspective of learner satisfaction to support those evaluation-based activities taking place at the pre- and post-adoption phases of the web-based e-learning system (WELS) life cycle. Their study investigated learners’ perceptions of the relative importance of decision criteria. They found that learners placed great value on the learner interface and content. As a result, emphasizing the non-technical aspect of the system content was critical. They also established that a sound WELS needs a high level of participation from other non-technical experts, such as teachers, teaching material editors, and pedagogy professionals in the construction phase as well as in the subsequent operation and maintenance. Chao and Chen (2009) utilized the consistent fuzzy preference relations (CFPR) in AHP model to evaluate the factors affecting effective teaching or learning through web learning. The factors included—e-learning material, quality of web learning platform, synchronous learning, learning record and self-learning. The result demonstrated that the participants considered e-learning material as essential and the most important factor for e-learning courses.

The related research suggests that most of the researchers have assessed learner and instructor satisfaction in online environments and have considered various factors to evaluate the performance of an e-learning system as a whole. However, much work could not be found that employed social context in depth for designing of learning content. The present study considers various criteria and sub-criteria for selecting appropriate content by utilizing the perspectives of different stakeholders from diverse domains of society in choosing the same. The next section introduces AHP and Fuzzy Modeling techniques in detail and discusses their relevance in prioritizing content using social opinion.
5.3 Techniques used for Content Prioritization

The following sub sections present the technical details of AHP and Fuzzy modeling techniques implemented in the present work for prioritizing of learning content by utilizing opinion of various stakeholders.

5.3.1 Analytic Hierarchy Process (AHP)

AHP is an MCDM technique that provides solutions to complex problems by prioritizing available alternatives against multiple objectives. The three principles of guidance in this technique are decomposition, comparative judgement and synthesis of priorities (Saaty, 1990). It involves estimation of priorities from pairwise comparison matrices. Different techniques are available to extract priorities vectors from the comparison matrices. By deriving priority vectors for all matrices in the hierarchy created for a given decision problem, it is possible to perform standard AHP aggregation and obtain the final (composite) vector of priorities for alternatives at bottom level of the hierarchy (Srdjevic, 2005). The following subsections explain the steps involved in AHP in detail.

A. Decision weights calculation by Eigenvalue method

Decision hierarchy formulation: The first step towards solution to the AHP problem is the formulation of a decision hierarchy. The goal defines level I of the hierarchy, followed by criteria at level II. Further detailing of criteria into subcriteria at various levels may be done to bring robustness into the system though it adds to computational complexity. The alternatives form the lowest level of the decision hierarchy. A typical decision hierarchy is shown in Figure 5.1.

Pairwise comparison: The participants from various concerned fields (as discussed in former section) are requested to rank the multiple criteria according to the ranking order given in Saaty’s fundamental scale (Appendix A-I, Section I, Table A-1). The criteria are judged against each other in pairs and a comparison matrix is constructed. On similar lines, the alternatives are compared with each other with respect to each criterion. Each participant builds the comparison matrices individually. The
mathematical formulation of pairwise ranking in AHP is as follows.

Let \( C = \{C_1, C_2, ..., C_m\} \) and \( A = \{A_1, A_2, ..., A_n\} \) be the sets of \( m \) criteria and \( n \) alternatives respectively. A participant needs to perform pairwise ranking for criteria \((C_i, C_j)\) (similarly for alternatives) using Saaty’s Fundamental Scale (Appendix A-I, Section I, Table A-1). An \( m \times m \) pairwise comparison matrix \( M = (C_{ij}) \) is formed when all criteria are ranked against each other.

\( C_{ij} \) denotes the rank assigned to criterion \( C_i \) in comparison to \( C_j \). A ranking of \( C_{ij} \) for the pair \((C_i, C_j)\) implies that \( C_i \) is strongly preferred over \( C_j \) and the rank of \( C_j \) over \( C_i \) is reciprocal of \( C_{ij} \), that is, \( 1/C_{ij} \).

\( C_{ii} \) refers to the criterion \( C_i \) being compared to itself and equals 1 as a criterion always ranks equally in comparison to itself.

**B. Decision weights calculation by Eigenvalue method**

*Determining priority weight vector:* The eigenvalue method refers to the process of deriving a priority weight vector, \( w = \{w_1, w_2, ..., w_n\}^T \) where \( w_i \geq 0 \) and
Adaptive Content Sequencing Incorporating Social Opinion in an e-Learning Environment

\[ \sum_{i=1}^{n} w_i = 1 \] from the comparison matrix \( M \). The principal eigenvector of \( M \) represents the desired priority vector \( w \), obtained by solving the following system of linear equations (Srdjevic, 2005):

\[
Mw = \lambda w; \quad e^T = 1.
\]

where, \( \lambda \) is the principal eigenvalue of \( M \). \( w \) represents priority ordering that shows the decision maker’s preference among alternatives and \( \lambda \) represents the measure of consistency.

**Consistency Ratio**: AHP deals with consistency explicitly due to unintended human inconsistencies crept in while making paired comparisons (Saaty, 1994). The comparison matrix, \( M \), is said to be consistent if its elements satisfy \( C_{ij} C_{jk} = C_{ik} \) for \( i, j, k = 1,2, ..., m \). The closer \( \lambda \) is to the number of criteria (or alternatives), the more consistent the matrix of comparisons. \( \frac{(\lambda - m)}{(m-1)} \) is called the Consistency Index (CI). This index shows the amount of deviation from consistency. To determine the goodness of CI, AHP compares it by Random Index (RI) (Appendix A-I, Section I, Table A-I), and the result is called Consistency Ratio (CR), calculated as \( CR = \frac{CI}{RI} \).

**C. Synthesis of the model to obtain the best alternative**

The priority weight vectors of alternatives for each criterion and the priority weight vectors of the respective criterion are multiplied and the products are summed up to calculate the composite weights of the alternatives. The composite weights of the alternatives for each stakeholder are calculated similarly.

**D. Aggregation of individual priorities**

Let there be \( p \) participants involved in the decision making process and \( V = \{v_1, v_2, ..., v_p\} \) be the weight vector for \( p \) participants, where \( v_k > 0 \) for \( k = 1, ..., p \) and \( \sum_{k=1}^{p} v_k = 1 \). Let \( w^{(k)} = (w_1^{(k)}, w_2^{(k)}, ..., w_n^{(k)})^T \) be the individual priority vector for \( n \) alternatives obtained from the comparison matrix \( M^{(k)} \) of the \( k^{th} \) participant. When a group of participants behaves simply as a collection of individuals and not as a cohesive unit coming to a consensus, the method of
aggregation of individual priorities (AIP) is used in group decision making (Forman and Peniwati, 1998; Dong et al, 2010). AIP uses the weighted geometric mean method (WGMM) to calculate the collective priority vector, \( w^{(c)} = \{w_1^{(c)}, w_2^{(c)}, ..., w_n^{(c)}\}^T \), where,

\[
    w_i^{(c)} = \frac{\prod_{k=1}^{P} (w_i^{(k)})^\nu_k}{\sum_{i=1}^{n} \prod_{k=1}^{P} (w_i^{(k)})^\nu_k}.
\]

The content is thus prioritized using social opinion through AHP technique. However, uncertainty is introduced into the system due to inherent though unintended human errors and conflicting perceptions of the experts. Fuzzy modeling of the recommendations tends to diminish the same. The next subsection introduces and discusses fuzzy modeling in detail.

### 5.3.2 Fuzzy Modeling

Fuzzy Inference System (FIS), based on fuzzy logic (Zadeh, 1965), is a framework based on fuzzy set theory that integrates the processes of fuzzifying the crisp input data, applying the fuzzy operators on the fuzzified data, analyzing the fuzzified data taking into account a list of fuzzy rules, aggregating the resultant fuzzy sets and defuzzifying the output to get the crisp data back (Figure 5.2).

![Fuzzy Inference System](image-url)
Fuzzification is the process of transforming crisp values into membership degrees for linguistic terms of fuzzy sets such as low, medium and high (de Vasconcelos, de Oliveira Lira and Teixeira, 2010). Each input variable is defined in linguistic terms over a universe of discourse representing all the permissible values. The linguistic terms are usually represented using a variety of available fuzzy sets with different shapes like-triangular, trapezoidal, gaussian and so forth (Zimmermann, 2001). This study has used s-shaped and z-shaped membership functions. These spline-based curves are the respective mappings on the vector x, and are so named because of their respective shapes. The parameters $a$ and $b$ locate the extremes of the sloped portion of the curve, as given by equations (5-1) and (5-2) for s-shaped and z-shaped functions respectively.

\[
f(x; a, b) = \begin{cases} 
0, & x \leq a \\
2 \left( \frac{x-a}{b-a} \right)^2, & a \leq x \leq \frac{a+b}{2} \\
1 - 2 \left( \frac{x-b}{b-a} \right)^2, & \frac{a+b}{2} \leq x \leq b \\
1, & x \geq b 
\end{cases} \quad (5-1)
\]

\[
f(x; a, b) = \begin{cases} 
1, & x \leq a \\
1 - 2 \left( \frac{x-a}{b-a} \right)^2, & a \leq x \leq \frac{a+b}{2} \\
2 \left( \frac{x-b}{b-a} \right)^2, & \frac{a+b}{2} \leq x \leq b \\
0, & x \geq b 
\end{cases} \quad (5-2)
\]

The membership functions defined for these fuzzy sets are used to map each input variable into membership values called the degrees of membership. The degree of membership defines the extent to which a variable lies in the given fuzzy region. A fuzzy rule base is developed using fuzzy IF-THEN rules to determine a mapping from fuzzy inputs to fuzzy outputs. A rule consists of an antecedent and a consequent of the form:

\[
\text{if } (x_1 \text{ is } Y_1) \land (x_2 \text{ is } Y_2) \land \ldots \land (x_n \text{ is } Y_n) \text{ then } z \text{ is } L
\]

where, $x_1, x_2, \ldots, x_n$ are the input variables and $z$ is the output variable. $Y_1, Y_2, \ldots, Y_n$ and $L$ represent the linguistic variables defined by fuzzy sets on $x_1, x_2, \ldots, x_n$ and $z$ respectively. The antecedent (if clause) may consist of several
fragments joined together using \( \Theta \). The symbol \( \Theta \) denotes any of the logical operators— AND and OR. The conclusions are drawn from the set of rules using an inference procedure and the truth of the consequent is inferred from the degree of truth of the antecedent (Al-Jarrah and Abu-Qdais, 2006). The antecedent is evaluated to a single number using the logical operators. The number of rules is determined by the number of linguistic terms in each fuzzy set. Avoiding unachievable or unnecessary combinations of rules and including the rules that cover only parts of the whole state is sufficient while forming the rulebase (Barin, Canha, Magnago, da Rosa Abaide and Wottrich, 2009).

Defuzzification aims at transforming a fuzzy value into a crisp value. Several defuzzification methods such as Centroid method, Mean of Maximum (MOM), Smallest of Maximum (SOM) and so forth are available. The present study utilizes the centroid method (Pfluger, Yen and Langari, 1992) for defuzzification. The defuzzified value, \( z \), is calculated as:

\[
  z = \frac{\sum_{j=1}^{in} \mu(v_j) \cdot v_j}{\sum_{j=1}^{in} \mu(v_j)}
\]

where, \( in \) is the number of crisp commands in the set, \( v \) is the value of each of these crisp commands, and \( \mu(v_j) \) is the membership value of the fuzzy control command at \( v_j \).

Mamdani FIS and Sugeno FIS are the two well known models for carrying out fuzzy inferences. Though both the models work similarly in fuzzifying the inputs and applying the fuzzy operators, the main difference is that Sugeno output membership functions are either linear or constant (Jang and Ned, 1997). As Mamdani model takes linguistic variables into consideration while encoding rules as per the stakeholder opinion (Al-Jarrah and Abu-Qdais, 2006; Kilinc, 2010) and linguistic interpretation of this information is difficult using Sugeno approach (Setnes, Naute-Lemke and Kaymak, 1998; Al-Najjar and Alsyouf, 2003), the presented work follows the Mamdani approach.
The next section presents an algorithm PARSeL that integrates and utilizes the robustness of AHP and fuzzy techniques to prioritize learning content using stakeholder recommendations.

5.4 PARSeL: The Proposed Algorithm for Content Prioritization

A two-tier algorithm, PARSeL (Sharma, Banati and Bedi, 2012) has been developed to implement the externalization phase of KMeLS. PARSeL employs AHP in Tier I, followed by combination of fuzzy modeling and AHP in Tier II (Figure 5.3).

![Figure 5.3 Representation of data flow in PARSeL](image-url)
Tier I begins with the formulation of decision hierarchy comprising three steps. The first step in decision hierarchy is setting up of a goal to address the problem. The next step involves identifying criteria and sub criteria associated with the problem environment. The third element of the decision hierarchy is identification of the alternatives available for solving the problem. The next step in Tier I is to elicit the preferences of various stakeholders and formation of comparison matrices for criteria and alternatives. Then priority weight vector (pwv) for each criterion (and sub-criterion) is obtained using Eigenvector method. The elements of comparison matrices are verified for consistency by calculating CI values and are adjusted if not found consistent. The ranks of criteria and sub-criteria are then evaluated. This step ends Tier I of the PARSeL algorithm.

The initial phase of Tier II performs fuzzy modeling. An FIS is designed to improve upon the ranks of criteria calculated in Tier I. Thus, pwvs of criteria obtained in Tier I serve as input to the FIS. A rule base is constructed for criteria by utilizing the information about relationships and behaviours of sub-criteria associated with respective criteria. Fuzzy modeling of the input values using this rule base provides final ranks of the criteria. In the latter phase of Tier II, pwvs of alternatives are calculated. Aggregation of individual pwvs gives final rank to each alternative. Alternatives are rated on the basis of rank vector and a list of prioritized alternatives is subsequently obtained.

The next section presents an experimental study that illustrates the implementation of PARSeL algorithm for prioritizing e-contents using stakeholder recommendations using Analytic Hierarchy Process (AHP) and Fuzzy Modeling. The study considers various criteria and sub-criteria for selecting appropriate programming language to be taught at the introductory level of an e-learning course. It takes into account the perspectives of different stakeholders (experts) from various strata of the society in choosing the language.

5.5 Experimental Study

An experimental study was conducted to evaluate the PARSeL algorithm with a goal
Adaptive Content Sequencing Incorporating Social Opinion in an e-Learning Environment

to prioritize a set of computing languages for an introductory online computing course. The syllabi of BCA/B.Sc. (H) Computer Science courses offered in thirteen central and state universities of India were scanned for the various programming languages being taught there in various semesters over a period of three years. Table 5-1 presents a summarized view of the same.

Table 5-1  Semester wise/Year wise view of the programming languages taught at various Indian Universities

<table>
<thead>
<tr>
<th>Univ.</th>
<th>Sem.</th>
<th>I Year</th>
<th>II Year</th>
<th>III Year</th>
</tr>
</thead>
<tbody>
<tr>
<td>BHU</td>
<td></td>
<td>C</td>
<td></td>
<td></td>
</tr>
<tr>
<td>DU</td>
<td></td>
<td>Java, HTML</td>
<td>C++</td>
<td>JavaScript</td>
</tr>
<tr>
<td>GGSIPU</td>
<td></td>
<td>C</td>
<td>C++</td>
<td>JavaScript</td>
</tr>
<tr>
<td>GU</td>
<td></td>
<td>C, HTML, Javascript</td>
<td>C++, VB.Net</td>
<td>Java</td>
</tr>
<tr>
<td>HPU</td>
<td></td>
<td>C</td>
<td>C++</td>
<td>VB</td>
</tr>
<tr>
<td>IGNOU</td>
<td></td>
<td>C</td>
<td></td>
<td></td>
</tr>
<tr>
<td>JMI</td>
<td></td>
<td>C</td>
<td>C++</td>
<td></td>
</tr>
<tr>
<td>KU</td>
<td></td>
<td>C</td>
<td>HTML</td>
<td>C++</td>
</tr>
<tr>
<td>MDU</td>
<td></td>
<td>C</td>
<td></td>
<td>OOP</td>
</tr>
<tr>
<td>PTU</td>
<td></td>
<td>C</td>
<td>C++</td>
<td>VB</td>
</tr>
<tr>
<td>RU</td>
<td></td>
<td>C, C++</td>
<td>Java, VB</td>
<td>HTML</td>
</tr>
<tr>
<td>WBUT</td>
<td></td>
<td>C</td>
<td></td>
<td>C++</td>
</tr>
</tbody>
</table>

As illustrated, nine of the thirteen universities follow semester-based curriculum and four (shaded) follow annual curriculum. Twelve of the thirteen universities offer C language in the first year as introductory programming course. One university offers Java along with HTML. On the other hand C++ is offered in second year by seven of thirteen universities and in third year by three universities. Java is offered by two universities in second and five universities in the third year. Three universities each offer VB in second and third years respectively. The conclusion drawn from the above mentioned facts is that C, C++, Java and VB are popular languages in the field
of academics and C seems to be the most popular candidate for introductory course in programming at undergraduate level.

The most important learning outcome of an introductory programming course in computing is to inculcate problem solving skills in novice programmers. To achieve this objective, it is imperative that a programming language must possess certain features beneficial for the novice learners, like the language should have an easily graspable syntax and easy debugging functionality. At the same time, having an intuitive editing interface is also desirable. Industrial demand and freely available compiler are some other looked-for features. As there are a number of criteria that may be counted in to select a computing language, so the problem becomes a multi-criteria problem and the need arises for construction of a decision hierarchy as discussed in the following subsection.

5.5.1 Description of criteria for language selection

Ekuobase, Akwukwuma and Egbokhare (2009) have laid out various criteria for selection of a programming language to be taught at the introductory level of the course. The following criteria and sub-criteria have been shortlisted, in this study, after discussions with various experts (stakeholders) from the field:

A. Programming Aspects (PA)
   a) Object Oriented Approach (P1): Objects are instances of generalized modules called classes. Object-oriented approach is data-centric and makes a program closer to the real world.
   b) Procedural Approach (P2): Programs are modular but procedure-centric in approach. Procedures are small sections of code invoked in order to execute the programs.

B. Technical Aspects (TA)
   a) Syntax complexity (T1): It deals with the rules followed in a language while coding, i.e., a combination of statements acceptable to a compiler.
   b) Portability (T2): It refers to the degree to which an executable program can switch between several hardware platforms.
c) **Ease of Debugging (T3):** It refers to the robustness of a compiler in detecting syntactical errors in a code and style of presenting it to the programmer.

C. **Usability Aspects (UA)**

a) **Familiarity (U1):** It refers to the extent of similarity between the notations and rules followed by a language and those of other existing languages.

b) **Expressivity (U2):** It refers to how easily a programmer is able to formulate the problem in the given language notations.

c) **GUI capability (U3):** It refers to the ease of program building provided by the graphical interface of a language.

D. **Commercial Aspects (CA)**

a) **Compiler cost (C1):** It refers to the price of the compilers (some are heavily prices whereas others are freely downloadable).

b) **Industrial Demand (C2):** It deals with how much a language is desired in IT industry.

c) **Compiler availability (C3):** It deals with whether the compiler is easily available online/in the market or not.

d) **Cost of Textbooks (C4):** In an e-learning scenario, the availability of textbooks is very crucial as the learners are more or less by themselves to learn and grasp the intricacies of the subject.

The decision hierarchy formulated in this study is illustrated in Figure 5.4.

---

**Goal**

Prioritization of programming languages for introductory computing

**Criteria**

**PA** Programming Aspects

P1: Object Oriented Approach
P2: Procedural Approach

**TA** Technology Aspects

T1: Syntax Complexity
T2: Portability
T3: Ease of Debugging

**UA** Usability Aspects

U1: Familiarity
U2: Expressivity
U3: GUI Capability

**CA** Commercial Aspects

C1: Compiler Cost
C2: Industrial Demand
C3: Compiler Availability
C4: Cost of Textbooks

**Alternatives**

Java

C++

C

Visual Basic

---

**Figure 5.4** Decision hierarchy formulation
5.5.2 Elicitation of stakeholder opinion

The underlying motive of the study was to involve representation from diverse but relevant sections of the society. As the study involved prioritizing a set of programming languages for an online computing course, experts from two major fields of interest, namely academics and computing, were considered to be the prime stakeholders. Hence, faculty members, researchers, students of various colleges affiliated to University of Delhi and professionals from computing and management industry were invited through online and personal requests.

A total of 110 participants showed their willingness to participate in the survey. Out of 110 participants, 52 (47%) were academicians/researchers, 35 (32%) were students (24 on rolls and 11 alumni), 15 (14%) were from the industry and 8 (7%) were computer science experts from non-computer-oriented fields. The participants were clustered according to their expertise into academicians (Ex1), researchers (Ex2), students (Ex3), alumni (Ex4), industry (Ex5) and miscellaneous (Ex6). For the sake of simplicity and generality in naming convention, all the clusters (including students and alumni) are henceforth termed as expert clusters. An instrument was designed on the basis of AHP decision hierarchy followed in this study (Appendix A-I, Section II). The instrument consisted of two sections, first of which contained an explanatory AHP example to simplify the rating process for the participants. Second section had points for rating 6 pairs of criteria, 12 pairs of subcriteria with respect to each other and 4 alternatives with respect to each other in context to each of the 12 subcriteria. The participants were requested to rate these pairs on the basis of Saaty’s Fundamental Scale (Appendix A-I, Section I, Table A-1).

Depending upon the answers of the participants, pairwise comparison matrices for a) the criteria and b) alternatives with respect to each criterion were constructed that served as input to PARSeL. Priority weight vectors were calculated for each criterion using eigenvector method of prioritization. The consistency ratio (CR) for each matrix was calculated. In accordance to the accepted AHP methodology, if CR was found to be more than 0.1, the concerned participant was requested to re-adjust his/her judgement values. To even out the uncertainties of the human decisions, fuzzy
Adaptive Content Sequencing Incorporating Social Opinion in an e-Learning Environment

modeling was used to construct fuzzy inference systems one for each criterion. The subcriteria under each criterion served as inputs to these FIS. The results obtained from AHP calculations (Sharma, Banati and Bedi, 2011c) were used to construct the rule bases in each FIS and the final priority weights for each criterion were calculated.

5.5.3 Results

Priority weight vectors for the subcriteria are calculated using AHP in Tier I of PARSeL. Table 5-2 and Figure 5.5 show the cluster wise and final priority weights of the subcriteria using Geometric Mean (GM).

Table 5-2 Clusterwise and mean priority weights of subcriteria along with evaluated ranks

<table>
<thead>
<tr>
<th>Sub-criterion</th>
<th>Expert Cluster</th>
<th>GM</th>
<th>Rank</th>
</tr>
</thead>
<tbody>
<tr>
<td>Code</td>
<td>Name</td>
<td>Ex1</td>
<td>Ex2</td>
</tr>
<tr>
<td>P1</td>
<td>OO Approach</td>
<td>0.88889</td>
<td>0.87500</td>
</tr>
<tr>
<td>P2</td>
<td>Procedural Approach</td>
<td>0.11111</td>
<td>0.12500</td>
</tr>
<tr>
<td>T1</td>
<td>Syntax</td>
<td>0.54848</td>
<td>0.52468</td>
</tr>
<tr>
<td>T2</td>
<td>Portability</td>
<td>0.24091</td>
<td>0.33377</td>
</tr>
<tr>
<td>T3</td>
<td>Debugging</td>
<td>0.21061</td>
<td>0.14156</td>
</tr>
<tr>
<td>U1</td>
<td>Familiarity</td>
<td>0.16984</td>
<td>0.12632</td>
</tr>
<tr>
<td>U2</td>
<td>Expressivity</td>
<td>0.38730</td>
<td>0.41601</td>
</tr>
<tr>
<td>U3</td>
<td>GUI Capability</td>
<td>0.44286</td>
<td>0.45767</td>
</tr>
<tr>
<td>C1</td>
<td>Compiler Cost</td>
<td>0.08791</td>
<td>0.08986</td>
</tr>
<tr>
<td>C2</td>
<td>Industrial Demand</td>
<td>0.25069</td>
<td>0.18436</td>
</tr>
<tr>
<td>C3</td>
<td>Compiler Availability</td>
<td>0.48740</td>
<td>0.35452</td>
</tr>
<tr>
<td>C4</td>
<td>Cost of Textbooks</td>
<td>0.17399</td>
<td>0.37126</td>
</tr>
</tbody>
</table>
The shaded cells in the table depict the highest rated sub-criteria under each criterion for each of the expert clusters. It is evident from the figures that all of the expert groups have unanimously rated object oriented approach as the best sub-criterion in PA category to select a programming language. The figures also point out that all the expert groups except the industry cluster have rated syntax as the best sub-criterion under the TA category. The professionals working in the industry seem to prefer portability more than syntax. In the UA category, GUI capability has emerged as the best rated sub-criterion as per the aggregated result. Cluster wise, the alumni group has preferred expressivity whereas the miscellaneous group has rated familiarity as the most preferred one. In CA category, compiler availability has been given the highest priority though industrial demand is also a close runner. The alumni and the industry clusters have expressed their preferences for industrial demand. This seems a rational choice as both the fresh graduates and the professionals, being in job, know the latest trends of the computing industry and want their prospective recruits to be in sync with the industrial demand.

![Graphical representation of clusterwise and mean priority weights of sub-criteria](image)

Figure 5.5  Graphical representation of clusterwise and mean priority weights of sub-criteria
In Tier II, four fuzzy inference systems are constructed for PA, TA, UA and CA respectively. The membership curves of these variables are shown in Figure 5.6.

**Figure 5.6** Membership functions of a FIS built for the three sub-criteria of ‘Technical Aspects’. Input variables— (a) Syntax Complexity (T1), (b) Portability (T2) and (c) Ease of Debugging (T3); Output variable- (d) ‘Technicality’

The structure of the FIS for TA is shown here. The three sub-criteria, namely-syntax (T1), portability (T2) and debugging (T3) represent the inputs; and the variable “Technicality” represents the output of the FIS. The input T2 is represented using s-shaped function, whereas T1, T3 and Technicality are represented using z-shaped function respectively.
Similar membership functions are constructed for the other three criteria, namely PA, UA and CA. After a series of discussions among experts, 37 rules were selected for constructing the rule bases for the four criteria as illustrated graphically in Figure 5.7.

![Graphical representation of the chosen rule bases for the four FIS built for the four criteria: (a) PA, (b) TA, (c) UA and (d) CA respectively](image)

**Figure 5.7** Graphical representation of the chosen rule bases for the four FIS built for the four criteria: (a) PA, (b) TA, (c) UA and (d) CA respectively

The resultant weight vector $W_{RC} = (5.87, 4.15, 6.92, 5.58)^T$ consisting of weights of the four criteria is obtained by incorporating priority weights of the subcriteria into FIS is also shown in the figure. The priority weight vector for the alternatives is
calculated using AHP. The priorities of the alternatives given by individual participants in context to each of the four criteria have been aggregated cluster wise (using GM) to compute the criterion wise ranks of the alternatives as tabulated in Table 5-3.

Table 5-3  Clusterwise and mean priority weights of subcriteria along with evaluated ranks

<table>
<thead>
<tr>
<th>Alternative</th>
<th>Expert Cluster</th>
<th>GM</th>
<th>Criterion wise Rank</th>
</tr>
</thead>
<tbody>
<tr>
<td>Code</td>
<td>Name</td>
<td>Ex1</td>
<td>Ex2</td>
</tr>
<tr>
<td>A1</td>
<td>C++</td>
<td>0.55598</td>
<td>0.50714</td>
</tr>
<tr>
<td>A2</td>
<td>Java</td>
<td>0.30413</td>
<td>0.24643</td>
</tr>
<tr>
<td>A3</td>
<td>C</td>
<td>0.07271</td>
<td>0.11280</td>
</tr>
<tr>
<td>A4</td>
<td>VB</td>
<td>0.06718</td>
<td>0.13363</td>
</tr>
</tbody>
</table>

The final priority weight vector of the alternatives is calculated as the sum-product of the weight vector $W_{Ri}$ and the cluster wise aggregated weights of the alternatives. The final priority weights and the ranks of the alternatives are tabulated in Table 5-4.
Table 5-4  Final priority weights of alternatives with respect to each criterion

<table>
<thead>
<tr>
<th>Alternative Code</th>
<th>Name</th>
<th>PA</th>
<th>TA</th>
<th>UA</th>
<th>CA</th>
<th>Final Priority Weight</th>
</tr>
</thead>
<tbody>
<tr>
<td>A1</td>
<td>C++</td>
<td>0.35481</td>
<td>0.34318</td>
<td>0.16229</td>
<td>0.21032</td>
<td>0.58035</td>
</tr>
<tr>
<td>A2</td>
<td>Java</td>
<td>0.34976</td>
<td>0.35406</td>
<td>0.29825</td>
<td>0.47916</td>
<td>0.82600</td>
</tr>
<tr>
<td>A3</td>
<td>C</td>
<td>0.12044</td>
<td>0.14522</td>
<td>0.08869</td>
<td>0.15625</td>
<td>0.27952</td>
</tr>
<tr>
<td>A4</td>
<td>VB</td>
<td>0.13110</td>
<td>0.14283</td>
<td>0.44404</td>
<td>0.14556</td>
<td>0.52473</td>
</tr>
</tbody>
</table>

Clusterwise and final priority weights of alternatives with respect to the four criteria are illustrated graphically in Figure 5.8. In context to programming aspects, C++ has been rated as the most preferred language. This may be due to the fact that...

![Graphical representation of clusterwise and final priority weights of alternatives with respect to the four criteria: (moving clockwise from the top left) PA, TA, UA and CA respectively](image-url)
C++ provides strong object oriented framework to the learners, and unlike Java, exploring advanced concepts like pointer handling is also feasible in C++. Three groups, namely— student, industry and the miscellaneous have preferred Java over C++. The alumni group has equal preference for both Java and C++. Under technical aspects, academicians, alumni and industry clusters have preferred C++ over Java; still Java has come up as the most preferred language. Programming in Java leads to highly portable, robust and secure programs that are lightweight and web friendly. When usability aspects are in question, VB seems the best choice because of its robust graphical user interface, ease of debugging and syntax. The same has been the outcome of this study as all the clusters have unanimously given the highest priority to VB. Under commercial considerations, Java is the best choice as it is an open source language with rich API libraries that provide a wide spectrum of reusable code to the computing community.

As already discussed, C++ is the highest rated language in the programming aspects category. Java has been rated the highest in technical and commercial aspects categories. VB is the winner in the usability category. Overall, Java has emerged as the most preferred language after the individual preferences under different criteria are aggregated.

The Externalization phase of KMeLS framework provides prioritized lists of content modules essential for designing e-learning courses from the perspectives of target learner groups as well as other stakeholders belonging to various strata of the society. The results obtained in this phase go as input to the next phase of KMeLS, that is, the Combination phase. The Combination phase consumes this input and builds customized learning paths for the learners with different needs. The forthcoming chapter presents and discusses the Combination phase in detail.