

**DEVELOPMENT OF A SCIENTIFIC FRAMEWORK
FOR SMARTPHONE EVALUATION WITH INTERNAL VALUED
PROBABILISTIC LINGUISTIC INFORMATION**

Submitted in partial fulfillment of the requirements for
the award of
Bachelor of Engineering degree in Computer Science and Engineering

by

V.PAVANI (Reg. No. 37110820)

T.APOORVA (Reg. No. 37110790)



**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
SCHOOL OF COMPUTING**

SATHYABAMA
INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)

**Accredited with Grade “A” by NAAC
JEPPIAAR NAGAR, RAJIV GANDHI
SALAI, CHENNAI – 600 119**

MARCH - 2021



SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY

(DEEMED TO BE UNIVERSITY)



Accredited with "A" grade by NAAC
Jeppiaar Nagar, Rajiv Gandhi Salai, Chennai – 600 119
www.sathyabama.ac.in

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

BONAFIDE CERTIFICATE

This is to certify that this project report is the bonafide work of **V.PAVANI (Reg. No. 37110820)** and **T.APOORVA (Reg. No.37110790)** who carried out the project entitled **"DEVELOPMENT OF A SCIENTIFIC DECISION FRAMEWORK FOR SMARTPHONE EVALUATION WITH INTERNAL VALUED PROBABILISTIC LINGUISTIC INFORMATION"** under my supervision from **August 2020 to March 2021**.

Internal Guide

DR.R.SUBHASHINI

Head of the Department

Submitted for Viva voce Examination held on_____

Internal Examiner

External Examiner

DECLARATION

I **V.PAVANI** hereby declare that the Project Report entitled “ **DEVELOPMENT OF A SCIENTIFIC DECISION FRAMEWORK FOR SMARTPHONE EVALUATION WITH INTERNAL VALUED PROBABILISTIC LINGUISTIC INFORMATION**”is done by me under the guidance of **DR.R.SUBHASHINI M.E.,Ph.D.**, Department of Computer Science and Engineering at Sathyabama Institute of Science and Technology is submitted in partial fulfillment of the requirements for the award of Bachelor of Engineering degree in Computer Science and Engineering.

DATE:

PLACE: CHENNAI

SIGNATURE OF THE CANDIDATE

ACKNOWLEDGEMENT

I am pleased to acknowledge my sincere thanks to **Board of Management of SATHYABAMA** for their kind encouragement in doing this project and for completing it successfully. I am grateful to them.

I convey my thanks to **Dr. T. Sasikala, M.E., Ph.D., Dean**, School of Computing, **Dr. S. Vigneswari, M.E., Ph.D., and Dr. L. Lakshmanan, M.E., Ph.D., Heads of the Department** of Computer Science and Engineering for providing me necessary support and details at the right time during the progressive reviews.

I would like to express my sincere and deep sense of gratitude to my Project Guide, **DR.R.SUBHASHINI M.E., Ph.D**, for her valuable guidance, suggestions and constant encouragement paved way for the successful completion of my project work.

I wish to express my thanks to all Teaching and Non-teaching staff members of the **Department of Computer Science and Engineering** who were helpful in many ways for the completion of the project.

Table of Contents

Title	Page No.
Bonafide Certificate	ii
Declaration	iii
Acknowledgement	iv
List of Figures	vi
Abbreviations	vii
Abstract	viii
1. Introduction	1
2. Objectives	2
3. Experiential Work	3
3.1 Data Arrangement	3
3.2 Missing value	3
3.3 Aggregation	4
3.4 Attribute Weight	5
3.5 Ranking	6
4. Results and Discussion	8
5. Conclusions and Future Work	9
6. References	10
7. Appendix	11

LIST OF FIGURES

Figure no	Figure name	Page no
Fig 4.1	Input	23
Fig 4.2	Missing values	23
Fig 4.3	Aggregation	24
Fig 4.4	Attribute weights	24
Fig 4.5	Ranking	24

ABBREVIATIONS

LT	Linguistic Term
PLTS	Probabilistic Linguistic Term Set
LTS	Linguistic Term Set
IVPLTS	Interval-Valued Probabilistic Linguistic Term Set
SDF	Scientific Decision Framework
WASPAS	Weighted Arithmetic Sum Produce Assessment
MM	Muirhead Mean
DM	Decision Maker

ABSTRACT

Smartphone is an attractive business lineup in India, with young Indian minds getting fascinated with the gadget. Smartphone companies regularly refine their strategies to build a successful gadget for customers. As the process is governed by human cognition, uncertainty becomes an inevitable part of the selection. To circumvent the issue, a Scientific Decision Framework (SDF) is proposed under Interval-Valued Probabilistic Linguistic Term Set (IVPLTS) context. At first, the relative importance of the attributes that are used for evaluation of the smartphones is calculated by using a programming model. Preference information from each DM is aggregated by extending Muirhead Mean (MM) operator to the IVPLTS context, which can easily capture the interrelationship between attributes. Moreover, smartphones are ranked systematically by extending WASPAS (Weighted Arithmetic Sum Produce Assessment) method to IVPLTS.

KEY WORDS: Interval-Valued Probabilistic Linguistic Term Set, Decision makers

CHAPTER 1

INTRODUCTION

To Understand how Interval-Valued Probabilistic Linguistic Term Sets (IVPLTS) works, we need to know what is Linguistic Term Sets (LTSS) it will obtain the values using Linguistic Terms (LTs) from the Decision Maker (DM) site. Now Probabilistic Linguistic Term Sets (PLTS) represents that we are adding the additional probability values for every (LTSS) so that it denotes that it is the support values for the obtained (LTSS) and (PLTS) is obtained by choosing the Decision Maker (DM) site how many times the (LTs) repeated in that particular (DM) site. Here, while choosing (PLTS), a problem will arise that when we watch (DM) site, the no of times the (LT) represents changes so that probability changes from time to time. So that, Interval-Valued Probabilistic Linguistic Term Sets (IVPLTSs) is chosen which has (LTs) and associated probability interval numbers that overcomes the problems of changing the probability values in (PLTS) by having more than one probability for the (LTSS) and all the attributes also follow the same order of the (IVPLTSs).

The values of the (LTSS) are in the range of 1 to 5, and the interval probability values are in the range of 0.1 to 0.9. Here the (LTSS) represent T and interval probability values represents in PL, PU, and the total (IVPLTSs) are represented in the form of T (PL, PU). Multi-criteria group decision making (MCGDM) problems are common in real-life situations. The

key of MCGDM problems is to rank feasible alternatives and search for the optimal

one based on different criteria. Due to the qualitative nature of many criteria or the dilemma of selecting appropriate numerical values, experts usually prefer to express their preferences by linguistic terms (LTs). Therefore, many linguistic approaches have been developed to model qualitative information or improve its computation. These approaches can be broadly divided into three groups: classic models suitable for the single LT, hesitant fuzzy sets based methods, probability distribution-based linguistic term sets (LTSs). For well-known linguistic models, such as the semantic model,¹ the symbolic model,² and the linguistic two-tuple model,^{3,4} they have been widely applied to different disciplines.⁵ Later on, Herrera et al.^{6,7} proposed a series of group decision making methods based on LTSs. Xu⁸ established a programming model and introduced the uncertain linguistic ordered weighted geometric (LOWG) and the induced uncertain LOWG operators⁹ for multi-attribute decision making under linguistic In order to be convenient for experts to express judgments, Rodriguez et al.¹⁰ proposed a new concept of hesitant fuzzy linguistic term sets (HFLTSSs) motivated by hesitant fuzzy

sets,¹¹ which allow decision makers (DMs) to use several possible LTSs to describe the priority over objectives. Extensive applications and reviews of such model have been developed to aggregate HFLTSS information^{12,13} and ease the computing process.^{14–16} However, all possible linguistic evaluations provided by experts or DMs have equal weights or importance in most of the current approaches about HFLTSS. Obviously, it is not appropriate in real MCGDM problems, since the DMs may prefer some LTSs to other ones so that the weights of these linguistic assessments may have different values (e.g., taking the form of probability distributions, degree of belief).

There have been some approaches, whose differences are summarized in Table 1, successfully proposed to consider distinguishing the preferences or weights of possible LTSs.^{17–22} First Wang and Hao²¹ put forward two proportional LTs. Then the method has been extended by a general form of numerical scale^{17,19} and probabilistic distributions.¹⁸ Moreover, Pang et al.²⁰ developed crisp-valued probabilistic linguistic term sets (PLTSSs), which allow experts to provide partially incomplete judg-

[] ments on the probabilistic distribution of possible LTSs. However, in most real-life conditions, decision

The theory of probabilistic linguistic term sets (PLTSs) is very useful in objectively dealing with the multi-criteria

group decision making (MCGDM) problems in which there is hesitancy in providing linguistic assessments; and PLTSs

allow experts to express their preferences on one linguistic term over another. In order to reflect the uncertainty

and inconsistency of decision-makers and handle incomplete linguistic information, we propose a new PLTS called

interval-valued probabilistic linguistic term set (IVPLTS).

In addition, the existing approaches associated with PLTSs are limited or highly complex in real applications. There-

fore, new operations, comparison laws, and aggregation operators are developed for IVPLTS. Furthermore, we

establish an efficient framework for MCGDM problems based on the proposed comparison method and the fuzzy

preference relation. Then we apply it to a real-life case under linguistic environment.

The extended TOPSIS meth-

ods combined with PLTSs by using different operational laws are also included for comparison. The final results

demonstrate the efficiency and practicality of the new framework.

CHAPTER 2

LITERATURE SURVEY AND OBJECTIVES

Our main objective is to find the smartphone with the highest rank among the give smartphones for this process we will use Interval-Valued Probabilistic Linguistic Term Sets (IVPLTSs). Here we divided our modules into five parts. 1st part is to form the given data into T(PL, PU) format, 2nd part is if any values from the (DM) site is missing fill those values using the 4 cases which will be discussed in next chapter, 3rd part is to aggregate the (DM) site matrices to a single matrix for calculating ranking, 4th part we need to find the attribute weights and give those attribute weights values to the MATLAB code to get final values which will be used at last module, 5th part we find the which is having the highest rank among the smartphones using aggregate matrix values and the attribute weight values to find which is the best smartphone among the given.

Here we have reviews for the smartphone but those reviews are based on person reviewing the system he might or might not be in favour to the particular company So that when a person who try's to buy a particular phone and came across this review and doesnt buy the phone though he is interested in that phone it will be loss for the company and the person

For upcoming this problem we are developing a mathematical model for the review system of the smartphone to tell which is best among them.

LITERATURE SURVEY:

The base paper investigated various methods based on Linguistic Term sets(LTSs)

For knowing appropriate numerical values using Linguistic Terms(LTs)

Here,While Choosing probability based Linguistic term set,a problem will arise to select which probability is superior over other .

So that Interval-Valued Probabilistic attributes Linguistic Terms(LTSs) and associated probability Interval numbers that overcomes the problems in probability distribution based Linguistic Term Set(LTS)

The aggregation operators are used in aggregating the evaluation information of each DM/Expert to a single one, and also helps in aggregating all attribute values into a single value DM using murihead mean operator.

CHAPTER 3

EXPERIMENTAL WORK AND METHODOLOGY

As discussed above, we are dividing the experiment into five parts we will have detail discussion now

3.1 Data arrangement:

Given data should be in the format of T(PL, PU) and the difference between PL, PU must be higher than equal to the 0.3, and the PL, PU ranges from the 0.1 to 0.9, T value ranges from 1 to 5.

3.2 Missing values:

Here the values which are taken from the above format due to there no (LTs) the values might be missing, so to solve that problem, we choose four cases.

Case 1:

One position T value is missing in only one (DM) matrix, and the values are present in the same position as the remaining (DM) matrices also PL, PU values will be missing.

For the T value we will fill the missing value position by taking the mean of the values from other (DM) matrices at the same position and the PL, PU values we will fill those missing values using the formula

$$\prod_{j=1}^{k^*} (pl)^{\lambda_j} \text{ and } \prod_{j=1}^{k^*} (pu)^{\lambda_j}, \text{ } pl, pu \in [0,1] \text{ and } pl \leq pu$$

Here, $\lambda = 1/k^* = 1/(\text{no. of marices}-1)$

Case 2:

One position T value is missing in all (DM) matrices also PL, PU values will be missing.

For T value we will fill the position of the missing values by taking the mean of the row where the value is missing and the PL, PU will also be missing to calculate that we use formula

$$\prod_{j=1}^m (pl)^{\lambda_j} \text{ and } \prod_{j=1}^m (pu)^{\lambda_j}$$

Here, $\lambda = 1/(\text{No. of instance values present in the missing row})$

Case 3:

One position T value is missing in all (DM) matrices except one also PL, PU will be missing.

For T value and PL, PU values, we will fill the missing values by replacing the only value present at that position of the other (DM) matrix.

Case 4:

Row T value is missing in all (DM) matrices, and also PL, PU will be missing.

For T value and PL, PU values, we will fill the missing values by replacing the above or below row according to which row the values are missing.

3.3 Aggregation:

From the missing values, we will fill all the missing values in all (DM) matrices we will aggregate the all matrices to form a single (DM) matrix which we will use in the Ranking module

To find that we should know about the lambda and the weight values for calculation of the aggregate matrix. Here the lambda values are in the range of several (DM) matrices, and the values represent risk attitude of the (DM) site.

Weight values are in the range from 0.1 to 0.9, and if all the weight values are added, then the output should be 1.0, and the values represent the importance of the (DM) site matrix.

To calculate the aggregation for T, PL, PU by using the following formula:

For T value:

$$\left(\prod_{i=1}^{noofmatrices} \left(\prod_{j=1}^{noofmatrices} (t_{ij})^{\lambda_j} \right)^{w_i} \right)^{1/\sum \lambda}$$

For PL, PU values:

$$\begin{aligned} & \left(1 - \left(1 - \prod_{i=1}^{noofmatrices} \left(1 - \prod_{j=1}^{noofmatrices} (1 - pl_{ij})^{\lambda_j} \right)^{w_i} \right)^{1/\sum \lambda} \right) \\ & \left(1 - \left(1 - \prod_{i=1}^{noofmatrices} \left(1 - \prod_{j=1}^{noofmatrices} (1 - pu_{ij})^{\lambda_j} \right)^{w_i} \right)^{1/\sum \lambda} \right) \end{aligned}$$

By using the above formulas, the aggregate matrix will be formed.

3.4 Attribute weight

From filling the missing values that are filled, all the (DM) matrices are formed into a vector by doing mean of each column for T, PL, PU. After finding all the Vector matrices, we make all the to form a single matrix.

Now, we need to find NIS (Negative Ideal Solution) matrix and PIS (Positive Ideal Solution) matrix by know the highest and lowest position values of the calculated matrix and this is done by using formula

$$\sum_{k=1}^{INSTANCES} \frac{t^k (pl^k + pu^k)}{2}$$

By using the above formula, we will obtain a single calculate matrix and we should know the highest and lowest position of every column and place values NIS

and PIS matrices by knowing the benefit and cost values benefit represent for what part of attributes he is getting the profit and cost represent for which part of the attributes he is paying more than usual. This value is given by taking the average of the (LT) and representing benefit and cost value.

For the PIS matrix up-to benefit, the highest position of the calculated matrix is taken, and corresponding single matrix formed by the vector matrices values is taken and that value is placed in the PIS matrix, and the opposite for the NIS matrix and, for the PIS matrix from the cost lowest position of the calculated matrix is taken. The corresponding single matrix formed by the vector matrices values is taken, and that value is placed in the PIS matrix and opposite for the NIS matrix.

By obtaining the NIS and PIS matrices, we should calculate the distance of the NIS and PIS matrices using formula:

$$d(T_{ij}, T^+) = \sqrt{\sum_{T=1}^{noofmatrices} \sum_{k=1}^{INSTANCES} (t_{ij}^k - t_j^{k+})^2 + (pl_{ij}^k - pl_j^{k+})^2 + (pu_{ij}^k - pu_j^{k+})^2}$$

$$d(T_{ij}, T^-) = \sqrt{\sum_{T=1}^{noofmatrices} \sum_{k=1}^{INSTANCES} (t_{ij}^k - t_j^{k-})^2 + (pl_{ij}^k - pl_j^{k-})^2 + (pu_{ij}^k - pu_j^{k-})^2}$$

here T+ represents PIS matrix distance and T- represents NIS matrix distance

Now, use the formula to calculate the weight of the attribute by using the formula:

$$alpha_j = *(d(T_{ij}, T^+) - d(T_{ij}, T^-))$$

Now, the obtained alpha values are unequal weight. To make them equal, we are using MATLAB code called weight optimization code in lpp.

3.5 Ranking:

From the Aggregated matrix, we will find the ranking of the smartphone, which one is better from the given matrices.

To get ranking, we need to get the input of tou and weight values, and tou values range from 0.1 to 0.9, and these values represent the strategy attitude, weight values are obtained from the previous module.

By considering that, we need to find A, B, C values.

‘A’ means weighted Arithmetic, and the formula is:

$$A_i = \sum_{j=1}^n \sum_{l=1}^{INSTANCES} w_j (x_{ij}^l)$$

‘B’ means weighted geometry and the formula is:

$$B_i = \prod_{j=1}^n \prod_{l=1}^{INSTANCES} (x_{ij}^l)^{w_j}$$

Xij means t, pl, pu

By using ‘A’ and ‘B’ values we are finding ‘C’ value which means Net ranking and the formula is:

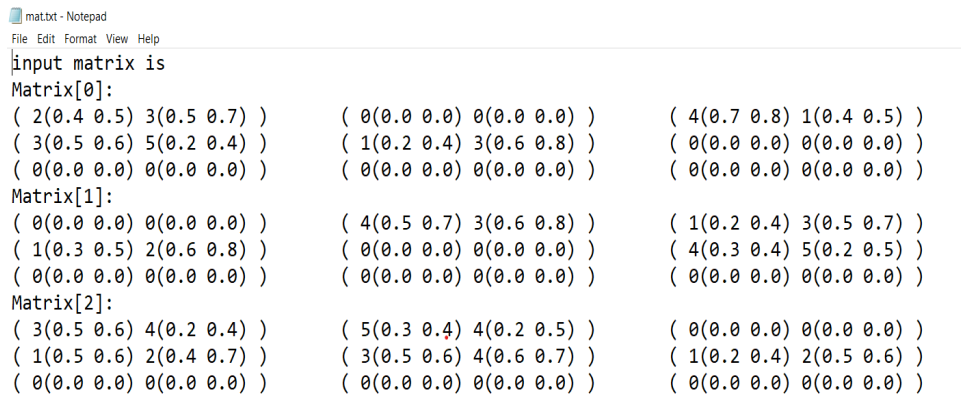
$$C_i = \tau(A_i) + (1 - \tau)B_i$$

After calculating the Ci values for the T, PL, and PU, values are added and the best smartphone among the given smartphones will be displayed.

CHAPTER 4

RESULT & DISCUSSION

INPUT DECISION MAKER MATRIX



```
mat.txt - Notepad
File Edit Format View Help
input matrix is
Matrix[0]:
( 2(0.4 0.5) 3(0.5 0.7) )      ( 0(0.0 0.0) 0(0.0 0.0) )      ( 4(0.7 0.8) 1(0.4 0.5) )
( 3(0.5 0.6) 5(0.2 0.4) )      ( 1(0.2 0.4) 3(0.6 0.8) )      ( 0(0.0 0.0) 0(0.0 0.0) )
( 0(0.0 0.0) 0(0.0 0.0) )      ( 0(0.0 0.0) 0(0.0 0.0) )      ( 0(0.0 0.0) 0(0.0 0.0) )
Matrix[1]:
( 0(0.0 0.0) 0(0.0 0.0) )      ( 4(0.5 0.7) 3(0.6 0.8) )      ( 1(0.2 0.4) 3(0.5 0.7) )
( 1(0.3 0.5) 2(0.6 0.8) )      ( 0(0.0 0.0) 0(0.0 0.0) )      ( 4(0.3 0.4) 5(0.2 0.5) )
( 0(0.0 0.0) 0(0.0 0.0) )      ( 0(0.0 0.0) 0(0.0 0.0) )      ( 0(0.0 0.0) 0(0.0 0.0) )
Matrix[2]:
( 3(0.5 0.6) 4(0.2 0.4) )      ( 5(0.3 0.4) 4(0.2 0.5) )      ( 0(0.0 0.0) 0(0.0 0.0) )
( 1(0.5 0.6) 2(0.4 0.7) )      ( 3(0.5 0.6) 4(0.6 0.7) )      ( 1(0.2 0.4) 2(0.5 0.6) )
( 0(0.0 0.0) 0(0.0 0.0) )      ( 0(0.0 0.0) 0(0.0 0.0) )      ( 0(0.0 0.0) 0(0.0 0.0) )
```

Fig 4.1 Input

From the given values, input is arranged in the form of T (PL, PU)

MISSING VALUE

```

case1 is executed for the matrix[0]
case1 is executed for the matrix[0]
case1 is executed for the matrix[0]
case1 is executed for the matrix[0]
case4 is executed for the matrix[0]
case1 is executed for the matrix[0]
case1 is executed for the matrix[1]
case1 is executed for the matrix[1]
case1 is executed for the matrix[1]
case1 is executed for the matrix[1]
case3 is executed for the matrix[1]
case3 is executed for the matrix[1]
case3 is executed for the matrix[1]
case3 is executed for the matrix[1]
case3 is executed for the matrix[1]
case1 is executed for the matrix[2]
case1 is executed for the matrix[2]
case1 is executed for the matrix[2]
case1 is executed for the matrix[2]
case1 is executed for the matrix[2]
case1 is executed for the matrix[0] of pl and pu
case1 is executed for the matrix[0] of pl and pu
case1 is executed for the matrix[0] of pl and pu
case1 is executed for the matrix[0] of pl and pu
case4 is executed for the matrix[0] of pl and pu
case1 is executed for the matrix[1] of pl and pu
case1 is executed for the matrix[1] of pl and pu
case1 is executed for the matrix[1] of pl and pu
case1 is executed for the matrix[1] of pl and pu
case3 is executed for the matrix[1] of pl and pu
case3 is executed for the matrix[1] of pl and pu
case3 is executed for the matrix[1] of pl and pu
case3 is executed for the matrix[1] of pl and pu
case1 is executed for the matrix[2] of pl and pu
case1 is executed for the matrix[2] of pl and pu
case1 is executed for the matrix[2] of pl and pu
case1 is executed for the matrix[2] of pl and pu
case1 is executed for the matrix[2] of pl and pu
case1 is executed for the matrix[2] of pl and pu
case1 is executed for the matrix[2] of pl and pu
case1 is executed for the matrix[2] of pl and pu
Missing value matrix is
Matrix[0]:
( 2(0.4 0.5) 3(0.5 0.7) )      ( 4(0.4 0.5) 3(0.4 0.6) )      ( 4(0.7 0.8) 1(0.4 0.5) )
( 3(0.5 0.6) 5(0.2 0.4) )      ( 1(0.2 0.4) 3(0.6 0.8) )      ( 2(0.2 0.4) 3(0.3 0.5) )
( 3(0.5 0.6) 5(0.2 0.4) )      ( 1(0.2 0.4) 3(0.6 0.8) )      ( 2(0.2 0.4) 3(0.3 0.5) )
Matrix[1]:
( 2(0.4 0.5) 3(0.3 0.5) )      ( 4(0.5 0.7) 3(0.6 0.8) )      ( 1(0.2 0.4) 3(0.5 0.7) )
( 1(0.3 0.5) 2(0.6 0.8) )      ( 2(0.3 0.5) 3(0.6 0.7) )      ( 4(0.3 0.4) 5(0.2 0.5) )
( 3(0.5 0.6) 5(0.2 0.4) )      ( 1(0.2 0.4) 3(0.6 0.8) )      ( 2(0.2 0.4) 3(0.3 0.5) )
Matrix[2]:
( 3(0.5 0.6) 4(0.2 0.4) )      ( 5(0.3 0.4) 4(0.2 0.5) )      ( 2(0.4 0.6) 2(0.4 0.6) )
( 1(0.5 0.6) 2(0.4 0.7) )      ( 3(0.5 0.6) 4(0.6 0.7) )      ( 1(0.2 0.4) 2(0.5 0.6) )
( 3(0.5 0.6) 5(0.2 0.4) )      ( 1(0.2 0.4) 3(0.6 0.8) )      ( 2(0.2 0.4) 3(0.3 0.5) )

```

Fig 4.2 Missing values

Here from the input of the matrix, we can see there are missing values. So, we used 4 cases to solve this problem.

AGGREGRATION

Aggregation:

$$\begin{pmatrix} 3(0.4 \ 0.6) & 4(0.3 \ 0.5) \\ 2(0.4 \ 0.5) & 4(0.4 \ 0.5) \\ 1(0.3 \ 0.5) & 4(0.4 \ 0.6) \end{pmatrix} \quad \begin{pmatrix} 2(0.4 \ 0.6) & 3(0.5 \ 0.7) \\ 2(0.4 \ 0.5) & 3(0.6 \ 0.8) \\ 2(0.4 \ 0.5) & 4(0.5 \ 0.7) \end{pmatrix} \quad \begin{pmatrix} 2(0.4 \ 0.6) & 2(0.5 \ 0.7) \\ 2(0.4 \ 0.5) & 3(0.5 \ 0.7) \\ 2(0.4 \ 0.5) & 4(0.5 \ 0.7) \end{pmatrix}$$

Fig 4.3 Aggregation

After filling all those values, we need to for a single matrix from the given three matrices using aggregation, and this final matrix is used in the part of finding the best smartphone.

ATTRIBUTE WEIGHT

attribute weights are:
0.3 0.3 0.2

Fig 4.4 Attribute weights

After filling the missing values, we will calculate using Muirhead mean to get these values, and these values are taken as input for the MATLAB code for the consistent weight, which will be given as input for finding the best smart-phone.

RANKING

Ranking:

oneplus having highest strategy attitude at 0.100000 and that value is 1.950065
oneplus having highest strategy attitude at 0.200000 and that value is 3.900058
oneplus having highest strategy attitude at 0.300000 and that value is 5.850052
oneplus having highest strategy attitude at 0.400000 and that value is 7.800045
oneplus having highest strategy attitude at 0.500000 and that value is 9.750038
oneplus having highest strategy attitude at 0.600000 and that value is 11.700031
oneplus having highest strategy attitude at 0.700000 and that value is 13.650024
oneplus having highest strategy attitude at 0.800000 and that value is 15.600018
oneplus having highest strategy attitude at 0.900000 and that value is 17.550011

Fig 4.5 Ranking

From the obtained result, we can conclude that from among all the smartphones, OnePlus has the highest rank for every strategy attitude. So, we are considering OnePlus as the best among the given smart-phones.

CHAPTER 5

CONCLUSION AND FUTURE PLANS

5.1 CONCLUSION

By using the Interval-Valued Probabilistic Linguistic Term Set (IVPLTS) method, we find the best smartphone among the given by using all the available modules and resources. The main advantages of the new method are not only that the proposed operations consider the interval-valued probability but also they accommodate an incomplete linguistic information environment.

5.2 FUTURE PLANS

We are proposing a mathematical model from the given data so that we are concluding which is better for the given phones, and we are planning to publish the project by taking the help of our guide within two months.

CHAPTER 6

REFERENCES

1. Chengzu Bai, Ren Zhang, Shuang Shen, Chaofan Huang, Xin Fan, Interval-valued probabilistic linguistic term sets in multi-criteria group decision making. *Int J Intell Syst.* 2018;14
2. Delgado M, Verdegay JL, Vila MA. On aggregation operations of linguistic labels. *Int J Intell Syst.* 1993;8: 351–370.
3. Herrera F, Martinez L. A 2-tuple fuzzy linguistic representation model for computing with words. *IEEE Trans Fuzzy Syst.* 2000;8:746–752.
4. Chen N, Xu Z, Xia M. Interval-valued hesitant preference relations and their applications to group decision making. *Knowl Based Syst.* 2013.
5. Gou X, Xu ZS, Liao H. Multiple criteria decision making based on Bonferroni means with hesitant fuzzy linguistic information. *Soft Comput.* 2016.

CHAPTER 7

APPENDIX

APPENDIX-A: INPUT DATA

```
for(t=0;t<q;t++)
{
    for(i=0;i<m;i++)
    {
        for(j=0;j<n;j++)
        {
            for(e=0;e<l;e++)
            {
                printf("Enter data in [%d][%d][%d][%d]:", t,i,j,e);
                scanf("%d",&matrix[t][i][j][e]);
                if(matrix[t][i][j][e]>5 || matrix[t][i][j][e]<0)
                {
                    printf("Enter correct value \n");
                    e--;
                    continue;
                }
                printf("Enter the pl value of [%d] matrix of[%d][%d][%d] :",t,i,j,e);
                scanf("%f",&pl[t][i][j][e]);
                if(pl[t][i][j][e]<0)
                {
                    printf("Enter correct value \n");
                    e--;
                    continue;
                }
                printf("\n Enter the pu value of [%d] matrix of[%d][%d][%d] : ",t,i,j,e);
                scanf("%f",&pu[t][i][j][e]);
                printf("\n");
                if(pu[t][i][j][e]<0)
                {
                    printf("Enter correct value \n");
                    e--;
                    continue;
                }
                if(pl[t][i][j][e]>pu[t][i][j][e])
                {
                    printf("Enter correct value \n");
                    e--;
                    continue;
                }
                if(pu[t][i][j][e]-pl[t][i][j][e]<0.0 || pu[t][i][j][e]-pl[t][i][j][e]>0.4)
                {
                    printf("Enter correct value \n");
                    e--;
                    continue;
                }
            }
        }
    }
}
```

APPENDIX-B: MISSING VALUES

```
for(t=0;t<q;t++)
{
    for(i=0;i<m;i++)
    {
        for(j=0;j<n;j++)
        {
            for(e=0;e<l;e++)
            {
                if(matrix[t][i][j][e]==0)
                {
                    A=i;
                    B=j;
                    M=t;
                    Z=e;
                    count=0;
                    for(tt=0;tt<q;tt++)
                    {
                        if(matrix[tt][A][B][Z]==0)
                        {
                            count++;
                        }
                    }
                    r=0;
                    //printf("count value %d \n",count);
                    if(count==1)
                    {
                        for(tt=0;tt<q;tt++)
                        {
                            r+=matrix[tt][A][B][Z];
                        }
                        matrix[M][A][B][Z]+=(r/(q-1));
                        fprintf(fptr,"case1 is executed for the matrix[%d]\n",M);
                    }
                    r=0;
                    if(count>1 && count<q)
                    {
                        for(tt=0;tt<q;tt++)
                        {
                            r+=matrix[tt][A][B][Z];
                        }
                        matrix[M][A][B][Z]+=r;
                        fprintf(fptr,"case3 is executed for the matrix[%d]\n",M);
                    }
                }
            }
        }
    }
}
```



```

k=1/(q-1);
v=1/(n-1);
for(t=0;t<q;t++)
{
    for( i=0;i<m;i++)
    {
        for( j=0;j<n;j++)
        {
            for( e=0;e<l;e++)
            {
                if(pl[t][i][j][e]==0 && pu[t][i][j][e]==0 )
                {
                    A=i;B=j;M=t;Z=e;
                    rount=0;
                    for( tt=0;tt<q;tt++)
                    {
                        if(pl[tt][A][B][Z]==0 && pu[tt][A][B][Z]==0 )
                        {
                            rount++;
                        }
                    }
                    // printf("rount %d \n",rount);
                    C=1;D=1;
                    if(rount==1)
                    {
                        //printf("enters \n");
                        for( tt=0;tt<q;tt++)
                        {
                            if(tt!=M)
                            {
                                C*=pow(pl[tt][A][B][Z],k);
                                D*=pow(pu[tt][A][B][Z],k);
                            }
                        }
                        pl[M][A][B][Z]+=C;
                        pu[M][A][B][Z]+=D;
                        fprintf(fptr,"case1 is executed for the matrix[%d] of pl and pu \n",M);
                        // printf("leaves");
                    }
                    //printf("%f %f",C,D);
                    E=0;
                    F=0;
                    if(rount>1 && rount<q)
                    {
                        for(tt=0;tt<q;tt++)
                        {
                            E+=pl[tt][A][B][Z];
                            F+=pu[tt][A][B][Z];
                        }
                        pl[M][A][B][Z]+=E;
                        pu[M][A][B][Z]+=F;
                        fprintf(fptr,"case3 is executed for the matrix[%d] of pl and pu\n",M);
                    }
                }
            }
        }
    }
}

```

```

    mount=0;mountt=0;C=1;D=1;
    if(rount==q)
    {
        for( jj=0;jj<n;jj++)
        {
            if(B!=jj )
            {
                mount+=p1[M][A][jj][Z];
                mountt+=pu[M][A][jj][Z];
            }
        }
        if(mount==0 && mountt==0)
        {
            for( ii=0;ii<m;ii++)
            {
                for( jj=0;jj<n;jj++)
                {
                    for(ZZ=0;ZZ<n;ZZ++)
                    {
                        if(ii==m-1)
                        {
                            p1[M][A][jj][ZZ]+=p1[M][A-1][jj][ZZ];
                            pu[M][A][jj][ZZ]+=pu[M][A-1][jj][ZZ];
                        }
                        else
                        {
                            p1[M][A][jj][ZZ]+=p1[M][A+1][jj][ZZ];
                            pu[M][A][jj][ZZ]+=pu[M][A+1][jj][ZZ];
                        }
                    }
                }
            }
            fprintf(fptr,"case4 is executed for the matrix[%d] of p1 and pu\n",M);
        }
        else
        {
            for( jj=0;jj<n;jj++)
            {
                if(B!=jj )
                {
                    C*=pow(p1[M][A][jj][Z],v);
                    D*=pow(pu[M][A][jj][Z],v);
                }
            }
            p1[M][A][B][Z]+=C;
            pu[M][A][B][Z]+=D;
            fprintf(fptr,"case2 is executed for the matrix[%d] of p1 and pu\n",M);
        }
    }
}
}
}

```

APPENDIX-C: ATTRIBUTE WEIGHT

```

for( t=0;t<q;t++)
{
    for( j=0;j<n;j++)
    {
        for( e=0;e<l;e++)
        {
            for( i=0;i<m;i++)
            {
                sumofcol[t][j][e]+=matrix[t][i][j][e];
                sumcolofpl[t][j][e]+=pl[t][i][j][e];
                sumcolofpu[t][j][e]+=pu[t][i][j][e];
            }
        }
    }
}
for( t=0;t<q;t++)
{
    for( j=0;j<n;j++)
    {
        for( e=0;e<l;e++)
        {
            sumofcol[t][j][e]=round(10*(sumofcol[t][j][e]/m))/10;
            sumcolofpl[t][j][e]=round(10*(sumcolofpl[t][j][e]/m))/10;
            sumcolofpu[t][j][e]=round(10*(sumcolofpu[t][j][e]/m))/10;
        }
    }
}
for( i=0;i<q;i++)
{
    for(j=0;j<n;j++)
    {
        for(e=0;e<l;e++)
        {
            matsta[i][j][e]=sumofcol[TT][j][e];
            plsta[i][j][e]=sumcolofpl[TT][j][e];
            pusta[i][j][e]=sumcolofpu[TT][j][e];
        }
    }
    TT++;
}
for( i=0;i<q;i++)
{
    for( j=0;j<n;j++)
    {
        for( e=0;e<l;e++)
        {
            QQ1=matsta[i][j][e];
            QQ=((plsta[i][j][e]+pusta[i][j][e])/2);
            QQT+=QQ1*QQ;
            QQ=0,QQ1=0;
        }
        mat1[i][j]=QQT;
        QQT=0;
    }
}

```

```

float maxElement=0;
for( j=0;j<n;j++)
{
    maxElement=mat1[0][j];
    maxx[j]=0;
    for( i=1;i<q;i++)
    {
        if(mat1[i][j]>maxElement)
        {
            maxElement=mat1[i][j];
            maxx[j]=i;
        }
    }
}

printf("Enter benefit value:");
scanf("%d",&ben);
printf("Enter cost value:");
scanf("%d",&cos);

for( j=0;j<n;j++)
{
    if(j<ben)
    {
        aa1=0,bb1=0;
        aa1=maxx[j];
        bb1=minn[j];
        for( e=0;e<l;e++)
        {
            pisoft[0][j][e]=matsta[aa1][j][e];
            nisoft[1][j][e]=matsta[bb1][j][e];
            pisoftp1[0][j][e]=plsta[aa1][j][e];
            nisoftp1[1][j][e]=plsta[bb1][j][e];
            pisoftp0[0][j][e]=pusta[aa1][j][e];
            nisoftp0[1][j][e]=pusta[bb1][j][e];
        }
    }
    else
    {
        aa1=0,bb1=0;
        aa1=minn[j];
        bb1=maxx[j];
        for( e=0;e<l;e++)
        {
            pisoft[0][j][e]=matsta[aa1][j][e];
            nisoft[1][j][e]=matsta[bb1][j][e];
            pisoftp1[0][j][e]=plsta[aa1][j][e];
            nisoftp1[1][j][e]=plsta[bb1][j][e];
            pisoftp0[0][j][e]=pusta[aa1][j][e];
            nisoftp0[1][j][e]=pusta[bb1][j][e];
        }
    }
}
}

float minElement=0;
for( j=0;j<n;j++)
{
    minElement=mat1[0][j];
    minn[j]=0;
    for( i=1;i<q;i++)
    {
        if(mat1[i][j]<minElement)
        {
            minElement=mat1[i][j];
            minn[j]=i;
        }
    }
}
}

```

```

float dd1,dd2,ddn1,ddn2;
for( j=0;j<n;j++)
{
    for( i=0;i<q;i++)
    {
        for( e=0;e<l;e++)
        {
            dd1=(round(10*(pow((matsta[i][j][e]-pisoft[0][j][e]),2)))/10);
            dd2=(round(10*(pow((plsta[i][j][e]-pisoftp1[0][j][e]),2)))/10)+(round(10*(pow((pusta[i][j][e]-pisoftp0[0][j][e]),2)))/10);
            disofpis[j]=dd1+dd2;
            ddn1=(round(10*(pow((matsta[i][j][e]-nisoft[1][j][e]),2)))/10);
            ddn2=(round(10*(pow((plsta[i][j][e]-nisoftp1[1][j][e]),2)))/10)+(round(10*(pow((pusta[i][j][e]-nisoftp1[1][j][e]),2)))/10);
            disofnis[j]=ddn1+ddn2;
            dd1=0,dd2=0,ddn1=0,ddn2=0;
        }
    }
    rando=disofpis[j];
    rando1=disofnis[j];
    rando=round(10*(sqrt((rando)/(q-1)))/10);
    rando1=round(10*(sqrt((rando1)/(q-1)))/10);
    disofpis[j]=rando;
    disofnis[j]=rando1;
    rando=0;
    rando1=0;
}
float ald=0;
for(j=0;j<n;j++)
{
    ald=fabs(disofpis[j]-disofnis[j]);
    alphasofdistances[j]=round(10*ald)/10;
}
for(j=0;j<n;j++)
{
    printf("%f \n",alphasofdistances[j]);
}
fprintf(fp, "attribute weights: \n");
for(j=0;j<n;j++)
{
    fprintf(fp, " %0.1f ",alphasofdistances[j]);
}
fprintf(fp, "\n");

```

Matlab code for optimized attribute weights

```

lpp(alpha values),attribute value1<30%,20%<attribute value2<40%,attribute value 3,4,5,6,<25%

```

APPENDIX-D: AGGREGATION

```
for(i=0;i<3;i++)
{
    printf("enter the lambda values [%d]",i);
    scanf("%d",&bb[i]);
    printf("enter the weight of the desicion makers values [%d]",i);
    scanf("%f",&b1[i]);
    pp+=bb[i];
}
for(i=0;i<m;i++)
{
    for(j=0;j<n;j++)
    {
        for(e=0;e<l;e++)
        {
            rrr=1;
            xx=1;
            for(t=0;t<q;t++)
            {
                rrr=1;
                ttt=matrix[t][i][j][e];
                for(zz=0;zz<q;zz++)
                {
                    rrr*= pow(ttt,bb[zz]);
                }
                xx*= pow(rrr,b1[t]);
            }
            rrr=pow(xx,(1/pp));
            rrr=ceil(rrr);
            ee[i][j][e]=rrr;
        }
    }
}
for(i=0;i<m;i++)
{
    for(j=0;j<n;j++)
    {
        for(e=0;e<l;e++)
        {
            rrr=1;
            xx=1;
            for(t=0;t<q;t++)
            {
                rrr=1;
                ttt=pl[t][i][j][e];
                for(zz=0;zz<q;zz++)
                {
                    rrr*= pow((1-ttt),bb[zz]);
                }
                xx*= pow((1-rrr),b1[t]);
            }

            rrr= pow((1-xx),(1/pp));
            cc[i][j][e]=ceilf((1-rrr)*10)/10;
        }
    }
}
```

```

for(i=0;i<m;i++)
{
    for(j=0;j<n;j++)
    {
        for(e=0;e<l;e++)
        {
            rrr=1;
            xx=1;
            for(t=0;t<q;t++)
            {
                rrr=1;
                ttt=pu[t][i][j][e];
                for(zz=0;zz<q;zz++)
                {
                    rrr*= pow((1-ttt),bb[zz]);
                }
                xx*= pow((1-rrr),b1[t]);
            }
            rrr= pow((1-xx),(1/pp));
            dd[i][j][e]=ceilf((1-rrr)*10)/10;
        }
    }
}
fprintf(fptr,"Aggregration:  \n");
for( i=0;i<m;i++)
{
    for( j=0;j<n;j++)
    {
        fprintf(fptr,"( ");
        for( e=0;e<l;e++)
        {
            fprintf(fptr,"%d(%0.1f %0.1f) ",ee[i][j][e],cc[i][j][e],dd[i][j][e]);
        }
        fprintf(fptr,") \t");
    }
    fprintf(fptr,"\n");
}

```

APPENDIX-D: RANKING

```
float tou[9]={0.1,0.2,0.3,0.4,0.5,0.6,0.7,0.8,0.9};
for(i=0;i<n;i++)
{
    printf("enter the w2[%d]",i);
    scanf("%f",&w2[i]);
}

for(i=0;i<m;i++)
{
    for(j=0;j<n;j++)
    {
        for(e=0;e<1;e++)
        {
            aaa1+=ee[i][j][e]*w2[j];
        }
        Aa1[i]=aaa1;
        aaa1=0;
    }
    for(i=0;i<m;i++)
    {
        for(j=0;j<n;j++)
        {
            for(e=0;e<1;e++)
            {
                bbb1*=ee[i][j][e]*w2[j];
            }
            Bb1[i]=bbb1;
            bbb1=1;
        }
        for(i=0;i<m;i++)
        {
            for(j=0;j<9;j++)
            {
                ccc1=(Aa1[i]*tou[j]);
                ccc2=(Bb1[i]*(1-tou[j]));
                Cc[i][j]=ccc1+ccc2;
                ccc1=0;
                ccc2=0;
            }
        }
    }
}
```

```

for(i=0;i<m;i++)
{
    for(j=0;j<n;j++)
    {
        for(e=0;e<l;e++)
        {
            aaa1+=cc[i][j][e]*w2[j];
        }
    }
    Aa2[i]=aaa1;
    aaa1=0;
}
for(i=0;i<m;i++)
{
    for(j=0;j<9;j++)
    {
        ccc1=(Aa1[i]*tou[j]);
        ccc2=(Bb1[i]*(1-tou[j]));
        Cc2[i][j]=ccc1+ccc2;
        ccc1=0;
        ccc2=0;
    }
}
for(i=0;i<m;i++)
{
    for(j=0;j<n;j++)
    {
        for(e=0;e<l;e++)
        {
            aaa1+=dd[i][j][e]*w2[j];
        }
    }
    Aa3[i]=aaa1;
    aaa1=0;
}
for(i=0;i<m;i++)
{
    for(j=0;j<n;j++)
    {
        for(e=0;e<l;e++)
        {
            bbb1+=dd[i][j][e]*w2[j];
        }
    }
    Bb3[i]=bbb1;
    bbb1=1;
}

```

```

for(i=0;i<m;i++)
{
    for(j=0;j<n;j++)
    {
        for(e=0;e<l;e++)
        {
            bbb1*=cc[i][j][e]*w2[j];
        }
    }
    Bb1[i]=bbb1;
    bbb1=1;
}

```

```

for(i=0;i<m;i++)
{
    for(j=0;j<9;j++)
    {
        ccc1=(Aa1[i]*tou[j]);
        ccc2=(Bb1[i]*(1-tou[j]));
        Cc3[i][j]=ccc1+ccc2;
        ccc1=0;
        ccc2=0;
    }
}
maxElement=0;
float maxxx[9];
int minnn[9];
for( j=0;j<9;j++)
{
    maxElement=Cctotal[0][j];
    maxxx[j]=maxElement;
    for( i=1;i<m;i++)
    {
        if(Cctotal[i][j]>maxElement)
        {
            maxElement=Cctotal[i][j];
            maxxx[j]=maxElement;
            minnn[j]=i;
        }
    }
}
fprintf(fptr,"Ranking: \n");
for( j=0;j<9;j++)
{
    if(minnn[j]==0)
    {
        fprintf(fptr,"samsung having highest strategy attitude at %f",tou[j]);
    }
    else if(minnn[j]==1)
    {
        fprintf(fptr,"i-phone having highest strategy attitude at %f",tou[j]);
    }
    else if(minnn[j]==2)
    {
        fprintf(fptr,"oneplus having highest strategy attitude at %f",tou[j]);
    }
    else if(minnn[j]==3)
    {
        fprintf(fptr,"realme having highest strategy attitude at %f",tou[j]);
    }
    else if(minnn[j]==4)
    {
        fprintf(fptr,"Nokia having highest strategy attitude at %f",tou[j]);
    }
    fprintf(fptr," and that value is %f \n",maxxx[j]);
}

```


DEVELOPMENT OF A SCIENTIFIC DECISION FRAMEWORK FOR SMARTPHONE EVALUATION WITH INTERVAL VALUED PROBABILISTIC LINGUISTIC INFORMATION

VELPURI PAVANI [1], T APOORVA [2], R SUBHASHINI [3]

[1][2] UG Student, Dept. of CSE, Sathyabama Institute of Science and Technology, Chennai, India

[3] Associate professor, Dept. of CSE, Sathyabama Institute of Science and Technology, Chennai, India

ABSTRACT: - Smartphone is associate degree appealing business arrangement in India, with young Indian personalities obtaining entranced with the device. Smartphone organizations systematically refine their systems to assemble an effective device for shoppers. Because the interaction is pictured by human insight, vulnerability turns into associate degree ineluctable piece of selection. To evade the problem, a Scientific Choice Framework (SDF) is planned beneath Interval-Valued Probabilistic Linguistic Term Set (IVPLTS) setting. From the beginning, the overall significance of the ascribes that area unit used for assessment of the smartphones is set by utilizing a programming model. Inclination data from each DM is destroyed by broadening Muirhead Mean (MM) administrator to the IVPLTS setting, which might while not abundant of a stretch catch the interrelatedness between credits. Also, smartphones area unit positioned deliberately by broadening ASPAS (Weighted Math add manufacture Assessment) technique to IVPLTS. Catchphrases: Interval-Valued Probabilistic Linguistic Term Set, call producers.

Keywords- Interval-valued probabilistic linguistic term sets, Multi-criteria group decision making, possibility degree formula

I INTRODUCTION

To understand however Interval-Valued Probabilistic Linguistic Term Sets (IVPLTS) works, we need to understand what's Linguistic Term Sets (LTSs) it'll acquire the qualities utilizing Linguistic Terms (LTs) from the choice Maker (DM) web site. Presently Probabilistic Linguistic Term Sets (PLTS) addresses that we have a tendency to area unit adding the additional probability esteems for every (LTSs) thus that it indicates that it's the assistance esteems for the got (LTSs) and (PLTS) is gotten by picking the choice Maker (DM) web site however oft the (LTs) rehashed therein specific (DM) web site. Here, whereas choosing (PLTS), a problem can emerge that after we observe (DM) site, the no of times the (LT) addresses changes with the goal that probability changes from time to time. So that, Interval-Valued Probabilistic Linguistic Term Sets (IVPLTSs) is picked which has (LTs) and connected probability stretch numbers that

defeats the problems of changing the probability esteems in (PLTS) by having over one probability for the (LTSs) and each one among the credits to boot follow an analogous request of the (IVPLTSs). The estimations of the (LTSs) area unit within the scope of one to five, and therefore the stretch probability esteems area unit in the scope of zero.1 to 0.9. Here the (LTSs) address T and stretch probability esteems addresses in PL, PU, and therefore the mixture (IVPLTSs) area unit addressed as T (PL, PU).

II RELATED WORK

In this phase, the essential ideas and activities of stretch numbers and PLTSs area unit momentarily investigated furthermore, talked regarding. We tend to at the start settle for every LTS documented during this paper is proscribed and requested utterly. Definition one.24 Let a phonetic term set be $S = \{t = -\tau, \dots, -1, 0, 1, \dots, \tau\}$, with odd cardinality wherever the midterm tends to the investigation of

"generally zero.5" or "lack of concern", and moreover the remainder phonetic names zone unit place equally around it **Interval numbers** Definition a pair of.25 Let $\tilde{a}$ be a span range or an internal worth, and later $= [aL, aU] = aL \leq x \leq aU$. On the off likelihood that $aL \geq x \geq aU$, $\tilde{a}$ is understood as a positive span range. For any two positive interval numbers

- $\tilde{a} = [aL, aU]$ and $\tilde{b} = [bL, bU]$, $\lambda \in [0, 1]$, then
 1. $\tilde{a} + \tilde{b} = [aL + bL, aU + bU]$;
 2. $\tilde{a} \lambda = [\lambda aL, \lambda aU]$;
 3. $\tilde{a} \cdot \tilde{b} = [\lambda aL, \lambda aU]$;
 4. $\tilde{a} \cdot \tilde{b} = [\lambda aL \cdot bL, \lambda aU \cdot bU]$.

2.3 The comparison between PLTSs

Peng et al.^[20] defined the score and the deviation degree of a PLTS to compare PLTSs with each other.

Definition 8.^[20] Let $E(g) = \{E^{(k)}(g^{(k)} | k = 1, 2, \dots, K)\}$ be a PLTS, and $E^{(k)}$ be the subscript of $E^{(k)}$. Then the score of $E(g)$ is

$$S(E(g)) = s_g = \frac{\sum_{k=1}^K E^{(k)}(g^{(k)})}{\sum_{k=1}^K E^{(k)}},$$

and the deviation degree of $E(g)$ is

$$\sigma(E(g)) = \left(\frac{\sum_{k=1}^K (E^{(k)}(g^{(k)}) - S(E(g)))^2}{\sum_{k=1}^K E^{(k)}} \right)^{1/2}.$$

Using the score and the deviation degree of a PLTS, Peng et al.^[20] constructed a procedure for PLTSs:

$$\begin{cases} \text{If } S(E_1(g)) > S(E_2(g)), \text{ then } E_1(g) > E_2(g) \\ \text{If } S(E_1(g)) = S(E_2(g)), \text{ then } E_1(g) = E_2(g) \\ \text{If } S(E_1(g)) < S(E_2(g)), \text{ then } E_1(g) < E_2(g) \end{cases} \quad (1)$$

FIG 1 COMPARISION BETWEEN PLTs

The hypothesis for viewing PLTSs and IVPLTSs will facilitate one position targets or choos¹ the most effective different choice. Ache et al.^[20] utilised the score capability and deviation degree to rank PLTSs. Chen et al.^[31] too acquainted the score work with look into span reputable reluctant down like elements. All the same¹ examined within the Section two, it alright may be ludicrous to mention one PLTS or IVPLTS is completely predominant to succeeding if these 2 PLTSs or IVPLTSs have a number of traditional elements. During this¹ method, we'll place forward an extra chance degree formula to investigate PLTSs and IVPLTSs spurred by dynasty et al.^[13] From the commencement, a model is gave the impression to define the first thought of our technique for

peruses straightforward agreement.

3 INTERVAL-VALUED PROBABILISTIC LINGUISTIC TERM SETS

PLTSs not only allow DMs to express their preferences with several possible LTs, but also contain probabilistic information over an alternative or object. However, in many real MCGDM problems, it is difficult for experts to exactly quantify their opinions using a crisp number, but they can be represented by an interval number. Therefore, it is very necessary to introduce the concept of interval-valued probabilistic linguistic term sets, which present the membership degrees of several elements to a given set with their own interval values.

3.1 The concept of IVPLTSs

Definition 5. Let $S = \{s_1, s_2, \dots, s_{-1}, s_0, s_1, s_2, \dots, s_l\}$ be an LTS. An IVPLTS is defined as

$$\tilde{L} = \{L^{(k)}(g^{(k)}) | L^{(k)} \in X, g^{(k)} \in S, k = 1, 2, \dots, K\}$$

where $L^{(k)}(g^{(k)})$ is the IVLT associated with the interval-valued probability $g^{(k)}$, $L^{(k)}$ is the subscript of $L^{(k)}$ and K is the number of all LTs in $\tilde{L}$, $g^{(k)} = [g_1^{(k)}, g_2^{(k)}]$ is an interval value, where $g_1^{(k)} = \inf g^{(k)}$ and $g_2^{(k)} = \sup g^{(k)}$ represent the lower and upper limits of $g^{(k)}$, respectively, and $\sum_{k=1}^K g_1^{(k)} \leq 1$.

Remark 1.

- If $g_1^{(k)} = g_2^{(k)}$ ($k = 1, 2, \dots, K$), then an IVPLTS reduces to the PLTS.
- It is crucial for IVPLTSs to handle the ignorance of $\tilde{L}$, so $\sum_{k=1}^K g_1^{(k)} \leq 1$ and $\sum_{k=1}^K g_2^{(k)} = 1$, which means partial ignorance exists.
- $L^{(k)}(g^{(k)})$ ($k = 1, 2, \dots, K$) are arranged according to the values of $g^{(k)}$ in ascending order.

FIG 2 INTERVAL- VALUED PROBABILISTIC LINGUISTIC TERM SETS

III EXISTING SYSTEM

The hypothesis of probabilistic linguistics term sets (PLTSs) is helpful in impartially managing¹ multi-models cooperative selection creating (MCGDM) problems during which there is aversion in giving account evaluations; and PLTSs permit specialists to speak their inclinations on one account term over another. To mirror the vulnerability furthermore, irregularity of chiefs and handle inadequate account knowledge, we tend to propose another PLTS referred to as span honoured probabilistic account term set (IVPLTS). Furthermore, the present methodologies connected with PLTSs Are restricted or exceptionally complicated in real applications. Consequently, new tasks, correlation laws, and accumulation administrator's square measure created for IVPLTS. Moreover, we set up a productive system for MCGDM problems in view of the projected correlation technique and also the downy¹ clination association. At that time we tend to apply it to a real case under linguistics climate. The all-encompassing TOPSIS ways joined with PLTSs by utilizing numerous operational laws square measure in addition enclosed for correlation. The ultimate outcomes Exhibit the productivity and reasonableness of the new system.

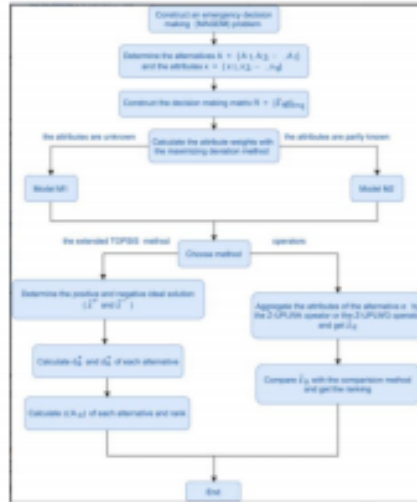


FIG 3 OVERVIEW OF THE EXISTING SYSTEM

IV PROPOSED SYSTEM

Our primary goal is to get the smartphone with the foremost elevated position among the provide smartphones for this cycle we'll utilize Interval-Valued Probabilistic Linguistic Term Sets (IVPLTSs). Here we have a tendency to isolate our modules into 5 sections. 1st half is to border the given info into T(PL, PU) design, 2nd half is that if any qualities from the (DM) website is missing fill those qualities utilizing the four cases which is able to be examined in next section, 3rd half is to total the (DM) site frameworks to a solitary grid for problem solving positioning, 4th half we'd like to get the standard loads and provides those property hundreds esteems to the MATLAB code to urge last qualities which will be utilised finally module, 5th half we have a tendency to discover the that has the foremost elevated position among the smartphones utilizing total framework esteems and also the characteristic weight esteems to get which is that the best smartphone among the given.

This framework can be utilized by the different

clients to get the directing meetings on the web. The information of the chatbot put away in the data set as example layout. Bot will give analgesics and food proposals that implies which food you need to take dependent on the illness. The real government assistance of the chatbot is the encourage individuals by giving legitimate direction with respect to the great and sound living. For the explanation that a considerable lot of individuals don't have basic consciousness of state of being. A few people live for quite a long time with weakening however they don't focus on indications essentially in light of the fact that they figure they don't need a specialist.

4.1 ADVANTAGES OF THE PROPOSED SYSTEM

- Can discover the smartphone with the foremost elevated position among the provide smartphones.

V MODULES DESCRIPTION

This project consists of mainly five modules:

5.1 DATA ARRANGEMENT

Given data need to be within the organization of T(PL, PU) and therefore the distinction between PL, PU should be over similar to the zero.3, and the PL, element goes from the zero.1 to 0.9, T esteem goes from one to five.

5.2 MISSING VALUES

One position T price is missing in mere one (DM) matrix, and therefore the values area unit Present within the same position because the remaining (DM) matrices additionally PL, Pu values can be missing. For the T price we'll fill the missing price position by taking the mean of The values from different (DM) matrices at constant position and therefore the PL, Pu values we tend to will fill those missing values victimization the formula.

One position T esteem is absent altogether (DM) frameworks likewise PL, element esteems can be missing. 4 For T esteem we are going to fill true of the missing qualities by

"generally zero.5" or "lack of concern", and moreover the remainder phonetic names zone unit place equally around it. **Interval numbers** Definition a pair of.25 Let $\tilde{a}$ be a span range or an interval worth, and later $= [aL, aU] = aL \leq x \leq aU$. On the off likelihood that $aL \geq x \geq aU$, $\tilde{a}$ is understood as a positive span range. For any two positive interval numbers

- $\tilde{a} = [aL, aU]$ and $\tilde{b} = [bL, bU]$, $\lambda \in [0, 1]$, then
1. $\tilde{a} + \tilde{b} = [aL + bL, aU + bU]$;
 2. $\tilde{a} \cdot \lambda = [(aL) \cdot \lambda, (aU) \cdot \lambda]$; 4 BAI ET AL.
 3. $\tilde{a} \cdot a = [\lambda aL, \lambda aU]$;
 4. $\tilde{a} \cdot \tilde{b} = [aL \cdot bL, aU \cdot bU]$.

2.3 The comparison between PLTSs

Pang et al.²⁰ Defined the score and the deviation degree of a PLTS to compare PLTSs with each other.

Definition 8.²⁰ Let $L(p) = \{L^{(i)}(p) | i = 1, 2, \dots, \#L(p)\}$ be a PLTS, and $L^{(i)}$ be the subscript of LT $L^{(i)}$. Then the score of $L(p)$ is

$$S(L(p)) = s_p = \frac{\sum_{i=1}^{\#L(p)} L^{(i)}(p)}{\sum_{i=1}^{\#L(p)} 1},$$

and the deviation degree of $L(p)$ is

$$\sigma(L(p)) = \left(\frac{\sum_{i=1}^{\#L(p)} (L^{(i)}(p) - s_p)^2}{\sum_{i=1}^{\#L(p)} 1} \right)^{1/2}.$$

Using the score and the deviation degree of a PLTS, Pang et al.²⁰ constructed a preorder for PLTSs:

$$\begin{cases} \text{if } S(L_1(p)) > S(L_2(p)), \text{ then } L_1(p) > L_2(p) \\ \text{if } S(L_1(p)) = S(L_2(p)), \text{ then } L_1(p) = L_2(p) \\ \text{if } S(L_1(p)) = S(L_2(p)), \text{ then } \begin{cases} \sigma(L_1(p)) < \sigma(L_2(p)), & L_1(p) > L_2(p) \\ \sigma(L_1(p)) > \sigma(L_2(p)), & L_1(p) < L_2(p) \\ \sigma(L_1(p)) = \sigma(L_2(p)), & L_1(p) = L_2(p) \end{cases} \end{cases} \quad (1)$$

FIG 1 COMPARISON BETWEEN PLTSs

The hypothesis for viewing PLTSs and IVPLTSs will facilitate one position targets or choos the most effective different choice. Ache et al.²⁰ utilised the score capability and deviation degree to rank PLTSs. Chen et al.³¹ too acquainted the score work with look into span reputable reluctant down like elements. All the same, examined within the Section two, it alright may be ludicrous to mention one PLTS or IVPLTS is completely predominant to succeeding if these 2 PLTSs or IVPLTSs have a number of traditional elements. During this method, we'll place forward an extra chance degree formula to investigate PLTSs and IVPLTSs spurred by dynasty et al.¹³ From the commencement, a model is gave the impression to define the first thought of our technique for

peruses straightforward agreement.

3 INTERVAL-VALUED PROBABILISTIC LINGUISTIC TERM SETS

PLTSs not only allow DMs to express their preferences with several possible LTs, but also contain probabilistic information over an alternative or objects. However, in many real MCGDM problems, it is difficult for experts to exactly quantify their opinions using a crisp number, but they can be represented by an interval number. Therefore, it is very necessary to introduce the concept of interval-valued probabilistic linguistic term sets, which permit the membership degrees of several elements to a given set with their own interval values.

3.1 The concept of IVPLTSs

Definition 9. Let $S = \{s_1, s_2, \dots, s_{\#S}\}$ be an LTS. An IVPLTS is defined as

$$\tilde{L} = \{L^{(i)}(p^{(i)}) | L^{(i)} \in S, p^{(i)} \in [0, 1], i = 1, 2, \dots, \#\tilde{L}\}$$

where $L^{(i)}(p^{(i)})$ is the LT $L^{(i)}$ associated with the interval-valued probability $p^{(i)}$, $p^{(i)}$ is the subscript of $L^{(i)}$ and $\#\tilde{L}$ is the number of all LTs in $\tilde{L}$, $p^{(i)} = [p_1^{(i)}, p_2^{(i)}]$ is an interval value, where $p_1^{(i)} = \inf p^{(i)}$ and $p_2^{(i)} = \sup p^{(i)}$ represent the lower and upper limits of $p^{(i)}$, respectively, and $\sum_{i=1}^{\#\tilde{L}} p_1^{(i)} \leq 1$.

Remark 1.

1. If $p_1^{(i)} = p_2^{(i)}$ ($i = 1, 2, \dots, \#\tilde{L}$), then an IVPLTS reduces to the PLTS.
2. It is crucial for IVPLTSs to handle the ignorance of $\tilde{L}$, so $\sum_{i=1}^{\#\tilde{L}} p_1^{(i)} \leq 1$ not $\sum_{i=1}^{\#\tilde{L}} p_2^{(i)} = 1$, which means partial ignorance exists.
3. $L^{(i)}(p^{(i)})$ ($i = 1, 2, \dots, \#\tilde{L}$) are arranged according to the values of $p^{(i)}$ in ascending order.

FIG 2 INTERVAL-VALUED PROBABILISTIC LINGUISTIC TERM SETS

III EXISTING SYSTEM

The hypothesis of probabilistic linguistics term sets (PLTSs) is helpful in impartially managing multi-models cooperative selection creating (MCGDM) problems during which there is aversion in giving account evaluations; and PLTSs permit specialists to speak their inclinations on one account term over another. To mirror the vulnerability furthermore, irregularity of chiefs and handle inadequate account knowledge, we tend to propose another PLTS referred to as span honoured probabilistic account term set (IVPLTS). Furthermore, the present methodologies connected with PLTSs Are restricted or exceptionally complicated in real applications. Consequently, new tasks, correlation laws, and accumulation administrator's square measure created for IVPLTS. Moreover, we set up a productive system for MCGDM problems in view of the projected correlation technique and also the downy relation association. At that time we tend to apply it to a real case under linguistics climate. The all-encompassing TOPSIS ways joined with PLTSs by utilizing numerous operational laws square measure in addition enclosed for correlation. The ultimate outcomes Exhibit the productivity and reasonableness of the new system.

Paper 15.pdf

ORIGINALITY REPORT

11%

SIMILARITY INDEX

2%

INTERNET SOURCES

11%

PUBLICATIONS

2%

STUDENT PAPERS

PRIMARY SOURCES

1

Chengzu Bai, Ren Zhang, Shuang Shen, Chaofan Huang, Xin Fan. "Interval-valued probabilistic linguistic term sets in multi-criteria group decision making", International Journal of Intelligent Systems, 2018

Publication

9%

2

Submitted to Humble Independent School District

Student Paper

2%

3

Chengzu Bai, Ren Zhang, Longxia Qian, Yaning Wu. "Comparisons of probabilistic linguistic term sets for multi-criteria decision making", Knowledge-Based Systems, 2017

Publication

<1%

4

Peng Li, Huanhuan Peng. "A Novel IVPLTS Decision Method Based on Regret Theory and Cobweb Area Model", Mathematical Problems in Engineering, 2020

Publication

<1%

5

Xiangqian Feng, Xiaodong Pang, Lan Zhang.

<1%

"On consistency and priority weights for interval probabilistic linguistic preference relations", Fuzzy Optimization and Decision Making, 2020

Publication

Exclude quotes Off

Exclude matches Off

Exclude bibliography Off

