Parallel Task Report

Artificial intelligence

2019/2020

Hanif Fadhlurrahman/1301176409/IF-41-INT02

1. Task and Problem

Given a .csv file contains 100 of influencers data, each influencer has an integer number of *Follower Count* and a real-value of *Engagement Rate*. We need to take 20 influencers based on a program that implement a *Fuzzy logic*. To create a *Fuzzy logic*, Two data must be captured at Fuzzification. Because of that, the *Follower Count* data and the *Engagement Rate* data must be separated into several types of data.

2. Membership Function and Fuzzification

The data that this program will used are as follows:

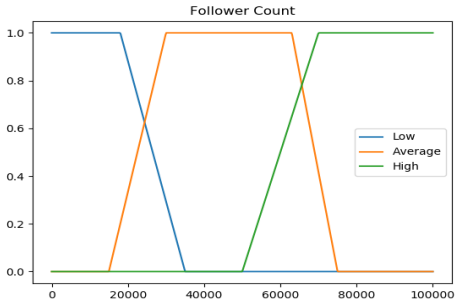
1. *Follower*:
   A. Low: has a number of *followers* from 0 – 35,000 peoples
   B. Average: Has a number of *followers* from 15,000 – 75,000 peoples
   C. High: Have a number of *followers* from 50,000 – 100,000 peoples

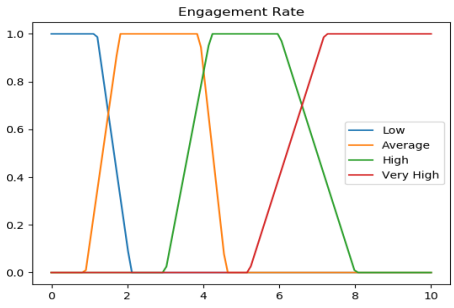
   (Source: My Knowledge)

2. *Engagement Rate*:
   A. Low: Has an *Engagement Rate* of 0.0 – 2.1
   B. Average: Has an *Engagement Rate* of 0.9 – 4.6
   C. High: Has an *Engagement Rate* of 2.7 – 8.0
   D. Very High: Has an *Engagement Rate* of 5.2 – 10.0

   (Source: https://www.scrunch.com/blog/what-is-a-good-engagement-rate-on-instagram)

The data are built into the graph then it can be seen as follows:





Then, we can do the fuzzification. For fuzzification, we just need to calculate the score that available for each *Follower Count* and *Engagement Rate*.

3. Inference Rule

To build a *Fuzzy logic*, it takes a rule of inference to set all existing rules, so the inference rules in this program are:

Engagement Rate	Follower Count	Score
Very High	High	Accepted
Very High	Average	Accepted
Very High	Low	Considered
High	High	Accepted

High	Average	Accepted
High	Low	Considered
Average	High	Considered
Average	Average	Considered
Average	Low	Reject
Low	High	Reject
Low	Average	Reject
Low	Low	Reject

4. Defuzzification Method

Once the data is changed to a fuzzy value and already has a membership degree in each input, then the value is calculated by the points to see the output that will be received, in this case the program uses a Sugeno Takagi because my data is already variant. So, I think the used of Mamdani is unnecessary. The Sugeno's formula for calculating defuzzification is:

$$x = \frac{(((50 \times low\ point) + (75 \times middle\ point) + (100 \times high\ point)}{(low\ point + middle\ point + high\ point)}$$

5. Output

After defuzzification, the program gets an x score for each influencer. The program will take 20 influencers, that get based on the biggest score. If the influencer has the same score, then the biggest follower of that score will be taken first. The id's of the chosen influencer will be inserted into a .csv file named Result.csv

6. Screenshot

```
List of people who got accepted :
No  1
Id :  11
Follower Count :  95117
Engagement Rate :  6.6
Total Score :  100.0

No  2
Id :  13
Follower Count :  90773
Engagement Rate :  6.7
Total Score :  100.0

No  3
Id :  72
Follower Count :  61456
Engagement Rate :  4.6
Total Score :  100.0

No  4
Id :  59
Follower Count :  58987
Engagement Rate :  6.7
Total Score :  100.0

No  5
Id :  60
Follower Count :  58403
Engagement Rate :  5.2
Total Score :  100.0
```