Muhhamad Affan Hasby/1301174618/IF-41-10/ARTIFICIAL INTELLIGENCE

Analisis Masalah

Fuzzy Logic adalah metode penalaran yang menyerupai penalaran manusia. Pendekatan *Fuzzy Logic* meniru cara pengambilan keputusan pada manusia yang melibatkan semua kemungkinan menengah antara nilai-nilai digital YA dan TIDAK.

Dalam menyelesaikan masalah pada Fuzzy Logic ada beberapa step, dalam kasus ini saya membaginya dengan 3 step yang mana penjelasannya sebagai berikut:

- Step 1 (Fuzzyfikasi) : Proses memetakan nilai ke dalam himpunan fuzzy dan menentukan derajat keanggotaannya di dalam himpunan fuzzy. Hal ini dilakukan karena data diproses berdasarkan teori himpunan fuzzy sehingga data yang bukan dalam bentuk fuzzy harus diubah ke dalam bentuk fuzzy.
- Step 2 (Inferensi) : Penarikan kesimpulan dari sekumpulan kaidah fuzzy. Jadi, di dalam inferensi minimal harus ada dua buah kaidah fuzzy, yaitu input dan output.
- Step 3 (Defuzzyfikasi) : Proses memetakan besaran dari himpunan fuzzy ke dalam bentuk nilai dengan menggunakan metode yang ada. Dalam menyelesaikan masalah ini saya menggunakan metode Mamdani. **Strategi Penyelesaian Masalah**

➢ Step 1 : Menentukan kategori untuk Followers dan Engagement, saya memakai 3 kategori untuk Followers yaitu nano, mikro, dan makro. Untuk Engagement saya memakai 3 kategori juga yaitu kecil, sedang, besar. Dan membuat rules influencer yang baik.

Followers	
0 - 10.000	Nano
10.000-30.000	Mikro
> 30.000	Makro

Engagement	
0 - 2.	Kecil
2 - 4.	Sedang
>4	Besar

F/E	Kecil	Sedang	Besar
Nano	Jelek	Jelek	Lumayan
Mikro	Jelek	Lumayan	Bagus
Makro	Bagus	Bagus	Bagus

➢ Step 2 : Membuat inferensi, Source code dibawah ini:

```
def inferensi(a):
  fol = fungsifollowers(followers[a])
  eng = fungsiengagement(engagement[a])
  cek = []
  L = []
  for i in range (len(fol)):
    for j in range(len(eng)):
      kelas = ruleinferensi(fol[i][1], eng[j][1])
      angka = min(fol[i][0], eng[j][0])
      L.append([kelas, angka])
      if kelas not in cek:
        cek.append(kelas)
  v = []
  for i in cek:
    y=[y for x,y in L if x==i]
    v.append([i,max(y)])
  return v
```

- Step 3 : Dengan menggunakan metode Mamdani, kategori-kategori yang telah dibuat diatas akan dicoba untuk menampilkan data yang diinginkan yaitu 20 influencer yang baik. Lalu mengoutput file ‘chosen.csv’ berupa indeks 20 influencer terbaik :

```
def defuzzi(k):
  L = [5, 15, 20, 25, 30, 35, 40, 45, 50, 55, 60, 65, 70,75, 80, 85, 90, 95]
  for i in range(3):
    x=inferensi(k)
    tj, tl, tb = 0,0,0
    for j in x:
      if j[0] == "jelek":
        tj = j[1]
      elif j[0] == "lumayan":
        tl = j[1]
      elif j[0] == "bagus":
        tb = j[1]
  up = []
  down = 0
  for i in L:
    hasil= fungsiapa(i, tj, tl, tb)
    if len(hasil) > 1 :
      maksimal = -1
      for j in range(len(hasil)):
        maksimal = max(maksimal,hasil[j][0])
      hasil= maksimal
    else :
      hasil = hasil[0][0]
    up.append(i*hasil)
    down += hasil
  return sum(up)/down
```

```
L = []
for i in range(100):
  L.append([i, defuzzi(i), followers[i], engagement[i]])
a= np.asarray([[i,j,k] for i,_,j,k in sorted(L, key=lambda x : x[1], reverse=True)[:20]])
print(sorted(L, key=lambda x : x[1], reverse=True)[:20])
np.savetxt("hasilinfluencer.csv", a.astype(int), fmt='%i', delimiter=",")
```

Hasil program:

	A	B
1	1,38237,5	
2	10,95117,6	
3	12,90773,6	
4	15,38046,0	
5	18,55217,4	
6	24,77585,2	
7	32,44833,0	
8	37,30308,0	
9	45,42111,0	
10	46,46847,0	
11	47,19925,6	
12	48,42391,0	
13	49,43475,2	
14	58,58987,6	
15	59,58403,5	
16	60,53701,4	
17	61,33443,0	
18	64,34471,0	
19	65,46833,0	
20	68,45718,0	