



Politechnika
Śląska

POLITECHNIKA ŚLĄSKA

WYDZIAŁ AUTOMATYKI, ELEKTRONIKI I INFORMATYKI

Praca dyplomowa inżynierska

Tytuł pracy dyplomowej inżynierskiej

autor: Imię Nazwisko

kierujący pracą: dr inż. Imię Nazwisko

konsultant: dr inż. Imię Nazwisko

Gliwice, listopad 2020

Oświadczenie

Wyrażam zgodę / Nie wyrażam zgody* na udostępnienie mojej pracy dyplomowej / rozprawy doktorskiej*.

Gliwice, dnia 25 listopada 2020

.....
(podpis)

.....
(poświadczenie wiarygodności
podpisu przez Dziekanat)

* podkreślić właściwe

Spis treści

| | | |
|---|-------------------------|----|
| 1 | Wstęp | 1 |
| 2 | [Analiza tematu] | 3 |
| 3 | Wymagania i narzędzia | 5 |
| 4 | Specyfikacja zewnętrzna | 7 |
| 5 | Specyfikacja wewnętrzna | 9 |
| 6 | Weryfikacja i walidacja | 11 |
| 7 | Podsumowanie i wnioski | 13 |

Rozdział 1

Wstęp

- wprowadzenie w problem/zagadnienie
- osadzenie problemu w dziedzinie
- cel pracy
- zakres pracy
- zwięzła charakterystyka rozdziałów
- jednoznaczne określenie wkładu autora, w przypadku prac wieloosobowych
 - tabela z autorstwem poszczególnych elementów pracy

Rozdział 2

[Analiza tematu]

- analiza tematu
- wprowadzenie do dziedziny (*state of the art*) – sformułowanie problemu
- studia literaturowe [?, ?, ?, ?]
- przegląd literatury tematu (należy wskazać źródła wszystkich informacji zawartych w pracy)
- opis znanych rozwiązań (także opisanych naukowo, jeżeli problem jest poruszany w publikacjach naukowych), algorytmów, osadzenie pracy w kontekście

Rozdział 3

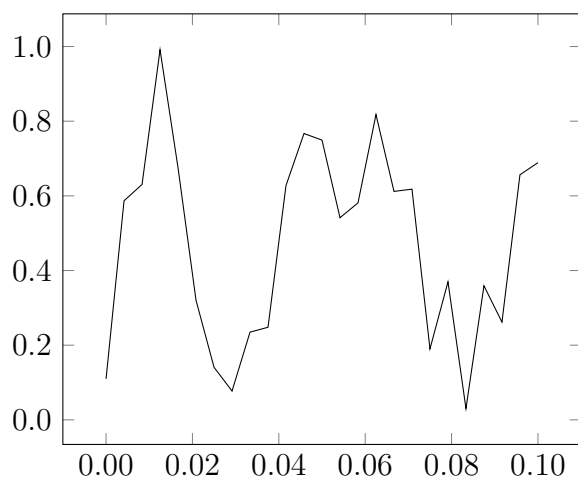
Wymagania i narzędzia

- wymagania funkcjonalne i нефункционалне
- przypadki użycia (diagramy UML)
- opis narzędzi
- metodyka pracy nad projektowaniem i implementacją

Rozdział 4

Specyfikacja zewnętrzna

- wymagania sprzętowe i programowe
- sposób instalacji
- sposób aktywacji
- kategorie użytkowników
- sposób obsługi
- administracja systemem
- kwestie bezpieczeństwa
- przykład działania
- scenariusze korzystania z systemu (ilustrowane zrzutami z ekranu lub generowanymi dokumentami)



Rysunek 4.1: Podpis rysunku po rysunkiem.

Rozdział 5

Specyfikacja wewnętrzna

- przedstawienie idei
- architektura systemu
- opis struktur danych (i organizacji baz danych)
- komponenty, moduły, biblioteki, przegląd ważniejszych klas (jeśli występują)
- przegląd ważniejszych algorytmów (jeśli występują)
- szczegóły implementacji wybranych fragmentów, zastosowane wzorce projektowe
- diagramy UML

Krótką wstawka kodu w linii tekstu jest możliwa, np. **descriptor**, a nawet **descriptor_gaussian**. Dłuższe fragmenty lepiej jest umieszczać jako rysunek, np. kod na rysunku 5.1, a naprawdę długie fragmenty – w załączniku.

```
1 class descriptor_gaussian : virtual public descriptor
2 {
3     protected:
4         /** core of the gaussian fuzzy set */
5         double _mean;
6         /** fuzzyfication of the gaussian fuzzy set */
7         double _stddev;
8
9     public:
10        /** @param mean core of the set
11            @param stddev standard deviation */
12        descriptor_gaussian (double mean, double stddev);
13        descriptor_gaussian (const descriptor_gaussian & w);
14        virtual ~descriptor_gaussian();
15        virtual descriptor * clone () const;
16
17        /** The method elaborates membership to the gaussian
18            fuzzy set. */
19        virtual double getMembership (double x) const;
20    };
```

Rysunek 5.1: Klasa **descriptor_gaussian**.

Rozdział 6

Weryfikacja i walidacja

- sposób testowania w ramach pracy (np. odniesienie do modelu V)
- organizacja eksperymentów
- przypadki testowe zakres testowania (pełny/niepełny)
- wykryte i usunięte błędy
- opcjonalnie wyniki badań eksperymentalnych

Rozdział 7

Podsumowanie i wnioski

- uzyskane wyniki w świetle postawionych celów i zdefiniowanych wyżej wymagań
- kierunki ewentualnych danych prac (rozbudowa funkcjonalna ...)
- problemy napotkane w trakcie pracy

Tablica 7.1: Opis tabeli nad nią.

| ζ | metoda | | | | | | |
|---------|---------|---------|----------------|--------------|--------------|----------------------|----------------|
| | alg. 1 | alg. 2 | alg. 3 | | | alg. 4, $\gamma = 2$ | |
| | | | $\alpha = 1.5$ | $\alpha = 2$ | $\alpha = 3$ | $\beta = 0.1$ | $\beta = -0.1$ |
| 0 | 8.3250 | 1.45305 | 7.5791 | 14.8517 | 20.0028 | 1.16396 | 1.1365 |
| 5 | 0.6111 | 2.27126 | 6.9952 | 13.8560 | 18.6064 | 1.18659 | 1.1630 |
| 10 | 11.6126 | 2.69218 | 6.2520 | 12.5202 | 16.8278 | 1.23180 | 1.2045 |
| 15 | 0.5665 | 2.95046 | 5.7753 | 11.4588 | 15.4837 | 1.25131 | 1.2614 |
| 20 | 15.8728 | 3.07225 | 5.3071 | 10.3935 | 13.8738 | 1.25307 | 1.2217 |
| 25 | 0.9791 | 3.19034 | 5.4575 | 9.9533 | 13.0721 | 1.27104 | 1.2640 |
| 30 | 2.0228 | 3.27474 | 5.7461 | 9.7164 | 12.2637 | 1.33404 | 1.3209 |
| 35 | 13.4210 | 3.36086 | 6.6735 | 10.0442 | 12.0270 | 1.35385 | 1.3059 |
| 40 | 13.2226 | 3.36420 | 7.7248 | 10.4495 | 12.0379 | 1.34919 | 1.2768 |
| 45 | 12.8445 | 3.47436 | 8.5539 | 10.8552 | 12.2773 | 1.42303 | 1.4362 |
| 50 | 12.9245 | 3.58228 | 9.2702 | 11.2183 | 12.3990 | 1.40922 | 1.3724 |

Dodatki

Spis skrótów i symboli

DNA kwas deoksyrybonukleinowy (ang. *deoxyribonucleic acid*)

MVC model – widok – kontroler (ang. *model-view-controller*)

N liczebność zbioru danych

μ stopień przyleżności do zbioru

\mathbb{E} zbiór krawędzi grafu

\mathcal{L} transformata Laplace’a

Źródła

Jeżeli w pracy konieczne jest umieszczenie długich fragmentów kodu źródłowego, należy je przenieść do załącznika.

```
1 partition fcm_possibilistic :: doPartition
2                               (const dataset & ds)
3 {
4     try
5     {
6         if (_nClusters < 1)
7             throw std::string ("unknown_number_of_clusters");
8         if (_nIterations < 1 and _epsilon < 0)
9             throw std::string ("You should set a maximal
10                                number_of_iteration_or_minimal_difference_or_
11                                epsilon.");
12
13         if (_nIterations > 0 and _epsilon > 0)
14             throw std::string ("Both number_of_iterations_and_
15                                minimal_epsilon_set_or_you should set either_
16                                number_of_iterations_or_minimal_epsilon.");
17
18         auto mX = ds.getMatrix();
19         std::size_t nAttr = ds.getNumberOfAttributes();
20         std::size_t nX    = ds.getNumberOfData();
21         std::vector<std::vector<double>> mV;
22         mU = std::vector<std::vector<double>> (_nClusters);
23         for (auto & u : mU)
```

```
19         u = std::vector<double> (nX);
20     randomise(mU);
21     normaliseByColumns(mU);
22     calculateEtas(_nClusters, nX, ds);
23     if (_nIterations > 0)
24     {
25         for (int iter = 0; iter < _nIterations; iter++)
26         {
27             mV = calculateClusterCentres(mU, mX);
28             mU = modifyPartitionMatrix (mV, mX);
29         }
30     }
31     else if (_epsilon > 0)
32     {
33         double frob;
34         do
35         {
36             mV = calculateClusterCentres(mU, mX);
37             auto mUnew = modifyPartitionMatrix (mV, mX);
38
39             frob = Frobenius_norm_of_difference (mU, mUnew)
40                 ;
41             mU = mUnew;
42         } while (frob > _epsilon);
43     }
44     mV = calculateClusterCentres(mU, mX);
45     std::vector<std::vector<double>> mS =
46         calculateClusterFuzzification(mU, mV, mX);
47
48     partition part;
49     for (int c = 0; c < _nClusters; c++)
50     {
51         cluster cl;
```


Zawartość dołączonej płyty

Do pracy dołączona jest płyta CD z następującą zawartością:

- praca (źródła \LaTeX owe i końcowa wersja w pdf),
- źródła programu,
- dane testowe.

Spis rysunków

| | | |
|-----|--|----|
| 4.1 | Podpis rysunku po rysunkiem. | 8 |
| 5.1 | Klasa descriptor_gaussian | 10 |

Spis tablic

| | | |
|-----|------------------------------|----|
| 7.1 | Opis tabeli nad nią. | 14 |
|-----|------------------------------|----|