

Data oddania: _____

Ocena: _____

Hubert Gawłowski 224298

Kamil Kiszko-Zgierski 224328

Projekt 2. Podsumowania lingwistyczne relacyjnych baz danych

Opis projektu ma formę artykułu naukowego lub raportu z zadania badawczego/doświadczalnego/obliczeniowego (wg indywidualnych potrzeb związanych np. z pracą inżynierską/naukową/zawodową).

Wybrane sekcje (rozdziały sprawozdania) są uzupełniane wg wymagań w opisie Projektu 2. i Harmonogramie Zajęć na WIKAMP KSR jako efekty zadań w poszczególnych tygodniach.

1. Cel

Celem zadania jest zaimplementowanie lingwistycznej agregacji, tj. przedstawienie danych liczbowych za pomocą sformułowań w języku pozornie naturalnym, czyli takim, w którym sformułowania mają z góry określony schemat typu: "Q obiektów są S1[T]" lub "Q obiektów będących S2 są S1[T]" [3] (np. "Większość zawodników jest wysoka [0.63]" lub "Około 9000 zawodników mających idealny wpływ na drużynę, zdobywa bardzo dużo punktów [0.82]"; liczba w nawiasie oznacza stopień prawdziwości podanych zdań). Do tego celu posłużą nam zbiory rozmyte. Zbiory rozmyte to takie zbiory, w których funkcją charakterystyczną jest funkcja przynależności, która określa w jakim stopniu dany element przynależy do zbioru. Owa implementacja zostanie wykonana w technologii Java z graficznym interfejsem użytkownika oraz z wykorzystaniem systemu zarządzania bazą danych o nazwie MySQL, a badania zostaną przeprowadzone w oparciu o bazę danych zawierającą statystyki zawodników ligi koszykówki NBA od sezonu 1996/1997 do 2019/2020 [4].

2. Charakterystyka podsumowywanej bazy danych

Jako bazę danych w zadaniu wykorzystaliśmy bazę danych dotyczącą informacji o zawodnikach NBA (najbardziej prestiżowej lidze koszykarskiej na świecie) z lat 1996 - 2019. Cała baza danych składa się z 22 kolumn zawierających informacje o danych zawodnika oraz jego statystykach różnego rodzaju. Baza danych ma charakter pliku csv i zawiera 11144 rekordy. Spośród kolumn wybraliśmy te, które zawierają wartości, naszym zdaniem, najlepiej nadające się do rozmycia. Atrybuty do rozmycia dobraliśmy tak, aby zgadzały się z definicją zmiennej lingwistycznej [1].

Ostatecznie spośród 22 kolumn wybraliśmy następujące:

- Kolumny zawierające informacje pozwalające zidentyfikować zawodnika (nie wykorzystywane w obliczeniach):
 1. nr identyfikacyjny zawodnika (index)
 2. imię i nazwisko zawodnika (player_name)
 3. skrót nazwy drużyny (team_abbreviation)
 4. kraj urodzenia zawodnika (country)
- Kolumny wykorzystywane w obliczeniach:
 1. wiek zawodnika (age) - atrybut przyjmuje wartości liczbowe całkowite od 18 do 44. Przekładając ten atrybut na język naturalny możemy użyć wartości: "junior", "młody", "w średnim wieku", "doświadczony", "stary".
 2. wzrost zawodnika (player_height) - atrybut przyjmuje wartości liczbowe zmiennoprzecinkowe od 160,02 do 231,14. Przekładając ten atrybut na język naturalny możemy użyć wartości: "bardzo niski", "niski", "średniego wzrostu", "wysoki", "bardzo wysoki".
 3. waga zawodnika (player_weight) - atrybut przyjmuje wartości liczbowe zmiennoprzecinkowe od 60,33 do 163,29. Przekładając ten atrybut na język naturalny możemy użyć wartości: "bardzo lekki", "lekki", "o przeciętnej wadze", "ciężki", "bardzo ciężki".
 4. kolejność wyboru w drafcie (draft_number) - oznacza, który w kolejności został wybrany zawodnik w organizowanym co roku tzw. drafcie, który polega na wybieraniu przez drużyny młodych zawodników na następny sezon. Atrybut przyjmuje wartości liczbowe całkowite od 1 do 60 oraz wartość "undrafted" - nie wybrany w drafcie. Wartość "undrafted" będzie traktowana jako wartość maksymalna. Przekładając ten atrybut na język naturalny możemy użyć wartości: "błyskawicznie", "szybko", "średnio", "późno", "na koniec".
 5. rozegrane gry (gp) - liczba gier zawodnika w danym sezonie. Atrybut ten przyjmuje wartości liczbowe całkowite od 1 do 85. Przekładając ten atrybut na język naturalny możemy użyć wartości: "znikoma liczba", "mało", "średnio", "dużo", "maksymalnie".
 6. zdobyte punkty (pts) - średnia liczba punktów zdobytych przez zawodnika w meczach w danym sezonie. Atrybut ten przyjmuje wartości liczbowe zmiennoprzecinkowe od 0 do 36,1. Przekładając ten atrybut na język naturalny możemy użyć wartości: "bardzo mało", "mało", "dostatecznie", "dużo", "bardzo dużo".

7. liczba zbiórek (reb) - średnia liczba zbiórek zawodnika na mecz - zbiórka jest to złapanie piłki przez zawodnika po nieudanym rzucie do kosza. Atrybut ten przyjmuje wartości liczbowe zmiennoprzecinkowe od 0 do 16,3. Przekładając ten atrybut na język naturalny możemy użyć wartości: "bardzo mało", "mało", "dostatecznie", "dużo", "bardzo dużo".
8. liczba asyst (ast) - średnia liczba asyst zawodnika na mecz - asysta jest to ostatnie podanie między zawodnikami tej samej drużyny, po którym zdobyty zostaje punkt. Atrybut ten przyjmuje wartości liczbowe zmiennoprzecinkowe od 0 do 11,7. Podobnie jak w atrybucie wyżej, przekładając ten atrybut na język naturalny możemy użyć wartości: "bardzo mało", "mało", "dostatecznie", "dużo", "bardzo dużo".
9. wpływ na drużynę (net_rating) - jaki wpływ miał zawodnik na punkty drużyny, gdy znajdował się na parkiecie. Atrybut ten przyjmuje wartości liczbowe zmiennoprzecinkowe od -100,0 do 100,0 ze zdecydowaną większością wartości mieszczących się w przedziale $[-25, 0; 25, 0]$. Przekładając ten atrybut na język naturalny możemy użyć wartości: "fatalny", "negatywny", "neutralny", "pozytywny", "idealny"
10. skuteczność rzutów (ts_pct) - jak efektywnie zawodnik rzucał do kosza, czyli ile spośród jego rzutów kończyło się punktami. Atrybut ten przyjmuje wartości liczbowe zmiennoprzecinkowe od 0 do 1, gdzie 0 oznacza, że żaden rzut nie kończył się punktem, a 1 - każdy rzut zawodnika kończył się punktem. Najwięcej wartości znajduje się w przedziale: $[0, 30; 0, 67]$. Przekładając ten atrybut na język naturalny możemy użyć wartości: "fatalny", "nieskuteczny", "przeciętny", "skuteczny", "idealny".
11. procent asyst (ast_pct) - procent punktów przy jakich asystował zawodnik, gdy znajdował się na boisku - atrybut ten mówi dużo o wpływie zawodnika na drużynę. Przyjmuje on wartości liczbowe zmiennoprzecinkowe z przedziału $[0; 1]$ ze zdecydowaną większością wartości w przedziale $[0; 0, 40]$. Przekładając ten atrybut na język naturalny możemy użyć wartości: "fatalny", "mały", "przeciętny", "duży", "idealny".

Oczywiście wartości takie jak: wzrost, waga, czy wiek będą odnosiły się jedynie do koszykarzy, ponieważ np. słowo "stary" w kontekście gracza koszykówki będzie związane z zupełnie innym wiekiem, niż słowo "stary" używane na codzień w stosunku do opisu wieku ludzi.

W rankingu najbardziej dochodowych lig sportowych liga NBA zajmuje trzecie miejsce na świecie ¹ (wyżej w rankingu są tylko ligi NFL - hokej i MLB - baseball). W związku z dużym zainteresowaniem wokół niej zasadnym jest, aby dane liczbowe koszykarzy przedstawiać również za pomocą zmiennych lingwistycznych.

Po pierwsze, jako iż owa dyscyplina jest bardzo popularna, wielu kibiców jest również zainteresowanych statystykami koszykarzy. Dla przeciętnego wi-

¹ <https://globalsportmatters.com/business/2019/03/07/tv-is-biggest-driver-in-global-sport-league-revenue/>

dza dane liczbowe mogą być jednak mało zrozumiałe. Rozwiązaniem tego problemu wydaje się wprowadzenie zmiennych lingwistycznych w celu ułatwienia interpretacji danych, co wiąże się z lepszym przyswojeniem informacji przez odbiorcę, jak również zaoszczędzeniem czasu na próbie ich zrozumienia.

Po drugie, przedstawienie danych liczbowych w postaci zmiennych lingwistycznych zapewnia grupowanie rekordów. Dzięki temu, dużo łatwiejsze jest dokonanie filtracji zawodników, co skutkuje szybszym wyszukaniem graczy o podobnych parametrach fizycznych lub osiągniętych statystykach.

W końcu, popularność oraz statystyki przychodów klubów NBA powodują zainteresowanie wśród potencjalnych sponsorów, którzy niekoniecznie muszą być związani z koszykówką. Dane przedstawione w formie zmiennych lingwistycznych znacznie pomogłyby w analizie statystyk oraz ocenie ryzyka związanego z inwestycją w dany klub.

#	Field	Schema	Table	Type
1	index	nba_players	nba_players	INT
2	player_name	nba_players	nba_players	VARCHAR
3	team_abbreviation	nba_players	nba_players	VARCHAR
4	country	nba_players	nba_players	VARCHAR
5	age	nba_players	nba_players	INT
6	player_height	nba_players	nba_players	DOUBLE
7	player_weight	nba_players	nba_players	DOUBLE
8	draft_number	nba_players	nba_players	VARCHAR
9	gp	nba_players	nba_players	INT
10	pts	nba_players	nba_players	DOUBLE
11	reb	nba_players	nba_players	DOUBLE
12	ast	nba_players	nba_players	DOUBLE
13	net_rating	nba_players	nba_players	DOUBLE
14	ts_pct	nba_players	nba_players	DOUBLE
15	ast_pct	nba_players	nba_players	DOUBLE

Rysunek 1. Nazwy kolumn (atrybutów) w systemie zarządzania bazami danych MySQL

3. Atrybuty i liczności obiektów wyrażone zmiennymi lingwistycznymi

3.1. Zmienne lingwistyczne

Definicja zmiennej lingwistycznej została już wprowadzona w sekcji 1. W bieżącej sekcji skupimy się dokładniej na tej definicji podając dla każdego z 11 atrybutów z sekcji 1 nazwę zmiennej lingwistycznej, zbiór terminów lingwistycznych oraz przestrzeń rozważań. Uważamy, że podając konkretny zbiór terminów lingwistycznych, jaki wykorzystamy, reguły gramatyczna oraz semantyczna będą widoczne i nie ma potrzeby podawania ich wprost. Wszak np. dla wzrostu nie podamy jako termin lingwistyczny słowa "zielony" lub "szybki" oraz wszystkie terminy we wszystkich zmiennych będą uporządkowane w sposób semantyczny (np. słowo "ciężki" nie wystąpi przed słowem "lekki", a słowo "idealny" przed "fatalny"). Poniżej przedstawiamy wykorzystane zmienne lingwistyczne wraz ze wzorami oraz wykresami:

1. wiek zawodnika - przestrzeń rozważań: liczby naturalne z przedziału $[18; 44]$
— junior

$$\mu_{junior}(x) = \begin{cases} 1 & \text{dla } x \in [18; 20) \wedge x \in \mathbf{N} \\ -0.25x + 6 & \text{dla } x \in [20; 24] \wedge x \in \mathbf{N} \\ 0 & \text{dla } x \in (24; 44] \wedge x \in \mathbf{N} \end{cases} \quad (1)$$

— młody

$$\mu_{mlody}(x) = \begin{cases} 0 & \text{dla } x \in ([18; 20) \cup (28; 44]) \wedge x \in \mathbf{N} \\ 0.25x - 5 & \text{dla } x \in [20; 24] \wedge x \in \mathbf{N} \\ 1 & \text{dla } x \in [24; 26] \wedge x \in \mathbf{N} \\ -0.5x + 14 & \text{dla } x \in [26; 28] \wedge x \in \mathbf{N} \end{cases} \quad (2)$$

— w średnim wieku

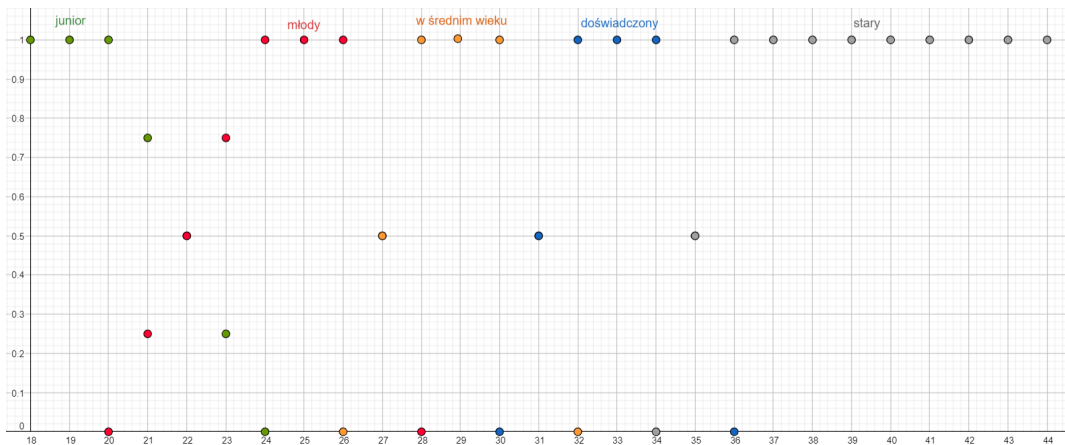
$$\mu_{wsrednimwieku}(x) = \begin{cases} 0 & \text{dla } x \in ([18; 26) \cup (32; 44]) \wedge x \in \mathbf{N} \\ 0.5x - 13 & \text{dla } x \in [26; 28] \wedge x \in \mathbf{N} \\ 1 & \text{dla } x \in [28; 30] \wedge x \in \mathbf{N} \\ -0.5x + 16 & \text{dla } x \in [30; 32] \wedge x \in \mathbf{N} \end{cases} \quad (3)$$

— doświadczony

$$\mu_{doswiadczony}(x) = \begin{cases} 0 & \text{dla } x \in ([18; 30) \cup (34; 44]) \wedge x \in \mathbf{N} \\ 0.5x - 15 & \text{dla } x \in [30; 32] \wedge x \in \mathbf{N} \\ 1 & \text{dla } x \in [32; 34] \wedge x \in \mathbf{N} \\ -0.5x + 18 & \text{dla } x \in [34; 36] \wedge x \in \mathbf{N} \end{cases} \quad (4)$$

— stary

$$\mu_{stary}(x) = \begin{cases} 0 & \text{dla } x \in [18; 34) \wedge x \in \mathbf{N} \\ 0.5x - 17 & \text{dla } x \in [34; 36] \wedge x \in \mathbf{N} \\ 1 & \text{dla } x \in [36; 44] \wedge x \in \mathbf{N} \end{cases} \quad (5)$$



Rysunek 2. Wykres funkcji przynależności dla zmiennej lingwistycznej "wiek zawodnika".

2. wzrost zawodnika - przestrzeń rozważań: liczby rzeczywiste z przedziału: $[160.03; 231.14]$

— niski

$$\mu_{niski}(x) = \begin{cases} 1 & \text{dla } x \in [160.03; 185) \\ -0.1x + 19.5 & \text{dla } x \in [185; 195] \\ 0 & \text{dla } x \in (195; 231.14] \end{cases} \quad (6)$$

— średniego wzrostu

$$\mu_{średnielowzrostu}(x) = \begin{cases} 0 & \text{dla } x \in (160.03; 185] \cup (205; 231.14] \\ 0.1x - 18.5 & \text{dla } x \in [185; 195] \\ -0.1x + 20.5 & \text{dla } x \in [195; 205] \end{cases} \quad (7)$$

— wysoki

$$\mu_{wysoki}(x) = \begin{cases} 0 & \text{dla } x \in (160.03; 200] \\ 0.1x - 20 & \text{dla } x \in [200; 210] \\ 1 & \text{dla } x \in [210; 231.14] \end{cases} \quad (8)$$

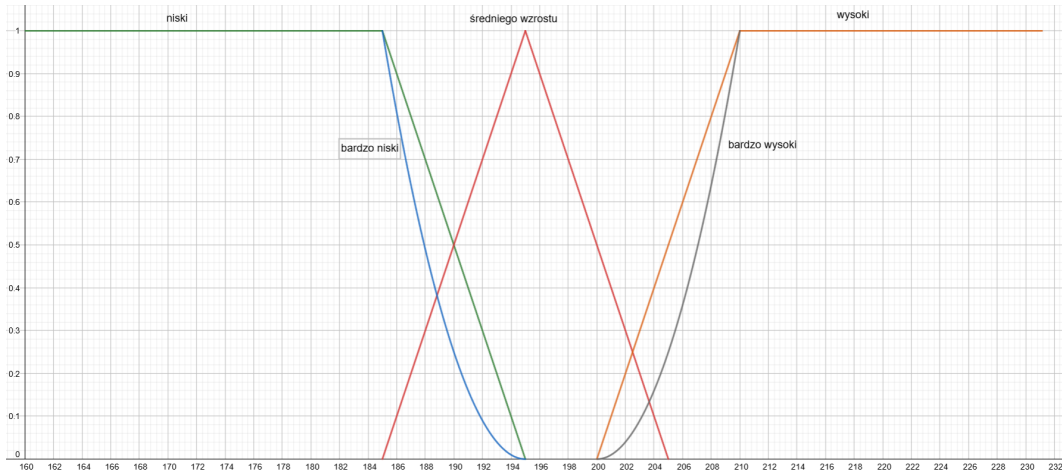
— bardzo niski

$$\mu_{bardzoniski}(x) = \mu_{niski}(x)^2 \quad (9)$$

— bardzo wysoki

$$\mu_{bardzowysoki}(x) = \mu_{wysoki}(x)^2 \quad (10)$$

Oczywiście, zgodnie z regułą semantyczną termin "bardzo niski" występuje przed terminem "niski" w zmiennej lingwistycznej. Powyżej został on wymieniony później, ponieważ definiując funkcję przynależności dla terminu "bardzo niski" korzystamy z funkcji przynależności dla terminu "niski" (konkretnie podnosimy ją do kwadratu).



Rysunek 3. Wykres funkcji przynależności dla zmiennej lingwistycznej "wzrost zawodnika".

3. waga zawodnika - przestrzeń rozważań: Liczby rzeczywiste z przedziału: $[60.33; 163.29]$

— lekki

$$\mu_{lekki}(x) = \begin{cases} 1 & \text{dla } x \in [60.33; 85) \\ -0.1x + 9.5 & \text{dla } x \in [85; 95] \\ 0 & \text{dla } x \in (95; 163.29] \end{cases} \quad (11)$$

— o przeciętnej wadze

$$\mu_{oprzecietnejwadze}(x) = \begin{cases} 0 & \text{dla } x \in [60.33; 90) \cup (110; 163.29] \\ 0.1x - 9 & \text{dla } x \in [90; 100) \\ -0.1x + 11 & \text{dla } x \in [100; 110] \end{cases} \quad (12)$$

— ciężki

$$\mu_{ciezki}(x) = \begin{cases} 0 & \text{dla } x \in [60.33; 105) \\ 0.1x - 10.5 & \text{dla } x \in [105; 115) \\ 1 & \text{dla } x \in [115; 163.29] \end{cases} \quad (13)$$

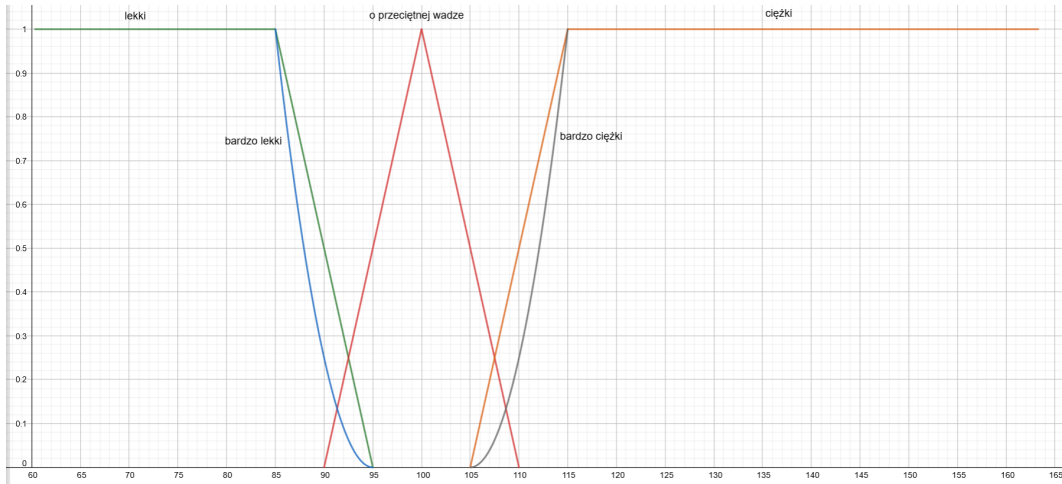
— bardzo lekki

$$\mu_{bardzolekki}(x) = \mu_{lekki}(x)^2 \quad (14)$$

— bardzo ciężki

$$\mu_{bardzociezki}(x) = \mu_{ciezki}(x)^2 \quad (15)$$

Oczywiście, zgodnie z regułą semantyczną termin ”bardzo lekki” występuje przed terminem ”lekki” w zmiennej lingwistycznej. Powyżej został on wymieniony później, ponieważ definiując funkcję przynależności dla terminu ”bardzo lekki” korzystamy z funkcji przynależności dla terminu ”lekki” (konkretnie podnosimy ją do kwadratu).



Rysunek 4. Wykres funkcji przynależności dla zmiennej lingwistycznej ”waga zawodnika”.

4. kolejność wyboru w drafcie - przestrzeń rozważań: liczby naturalne z przedziału: $[0; 60]$
 - błyskawicznie

$$\mu_{blyskawicznie}(x) = \begin{cases} 1 & \text{dla } x \in [0; 6) \wedge x \in \mathbf{N} \\ -0.125x + 1.75 & \text{dla } x \in [6; 14] \wedge x \in \mathbf{N} \\ 0 & \text{dla } x \in (14; 60] \wedge x \in \mathbf{N} \end{cases} \quad (16)$$

— szybko

$$\mu_{szybko}(x) = \begin{cases} 0 & \text{dla } x \in ([0; 10) \cup (30; 60]) \wedge x \in \mathbb{N} \\ 0.25x - 2.5 & \text{dla } x \in [10; 14) \wedge x \in \mathbb{N} \\ 1 & \text{dla } x \in [14; 22) \wedge x \in \mathbb{N} \\ -0.125x + 3.75 & \text{dla } x \in [22; 30] \wedge x \in \mathbb{N} \end{cases} \quad (17)$$

— średnio

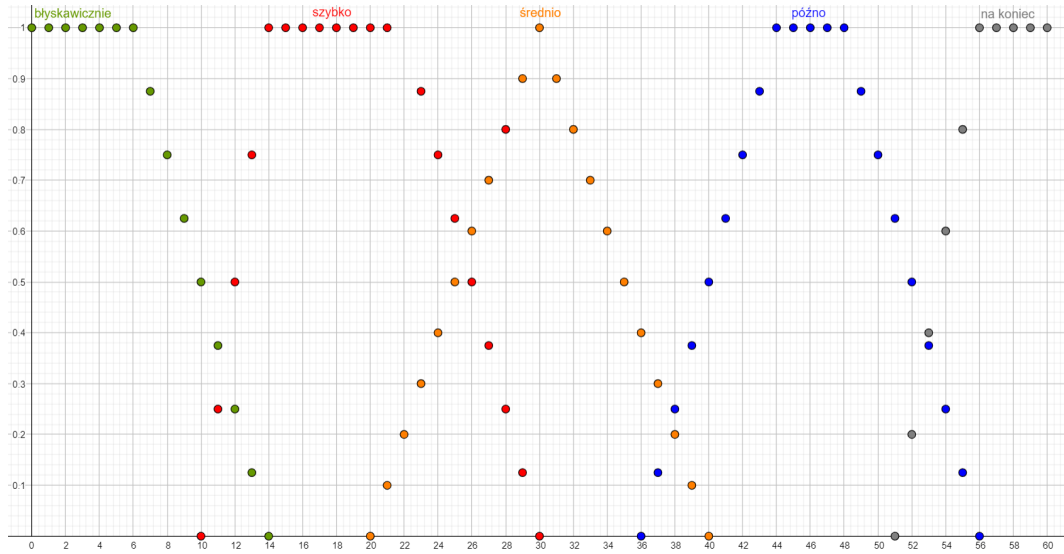
$$\mu_{średnio-draft}(x) = \begin{cases} 0 & \text{dla } x \in ([0; 20) \cup (40; 60]) \wedge x \in \mathbb{N} \\ 0.1x - 2 & \text{dla } x \in [20; 30) \wedge x \in \mathbb{N} \\ -0.1x + 4 & \text{dla } x \in [30; 40] \wedge x \in \mathbb{N} \end{cases} \quad (18)$$

— późno

$$\mu_{późno}(x) = \begin{cases} 0 & \text{dla } x \in ([0; 36) \cup (56; 60]) \wedge x \in \mathbb{N} \\ 0.125x - 4.5 & \text{dla } x \in [36; 44) \wedge x \in \mathbb{N} \\ 1 & \text{dla } x \in [44; 48) \wedge x \in \mathbb{N} \\ -0.125x + 7 & \text{dla } x \in [48; 56] \wedge x \in \mathbb{N} \end{cases} \quad (19)$$

— na koniec

$$\mu_{na koniec}(x) = \begin{cases} 0 & \text{dla } x \in [0; 51) \wedge x \in \mathbb{N} \\ 0.2x - 10.2 & \text{dla } x \in [51; 56) \wedge x \in \mathbb{N} \\ 1 & \text{dla } x \in [56; 60] \wedge x \in \mathbb{N} \end{cases} \quad (20)$$



Rysunek 5. Wykres funkcji przynależności dla zmiennej lingwistycznej "kolejność wyboru w draftie".

5. rozegrane gry w sezonie - przestrzeń rozważań: liczby naturalne z przedziału: $[1; 85]$

— znikoma liczba

$$\mu_{znikomaliczba}(x) = \begin{cases} 1 & \text{dla } x \in [1; 10) \wedge x \in \mathbb{N} \\ -0.0625x + 1.625 & \text{dla } x \in [10; 26] \wedge x \in \mathbb{N} \\ 0 & \text{dla } x \in (26; 85] \wedge x \in \mathbb{N} \end{cases} \quad (21)$$

— mało

$$\mu_{\text{mało-gier}}(x) = \begin{cases} 0 & \text{dla } x \in ([1; 20) \cup (40; 85]) \wedge x \in \mathbf{N} \\ 0.1x - 2 & \text{dla } x \in [20; 30] \wedge x \in \mathbf{N} \\ -0.1x + 4 & \text{dla } x \in [30; 40] \wedge x \in \mathbf{N} \end{cases} \quad (22)$$

— średnio

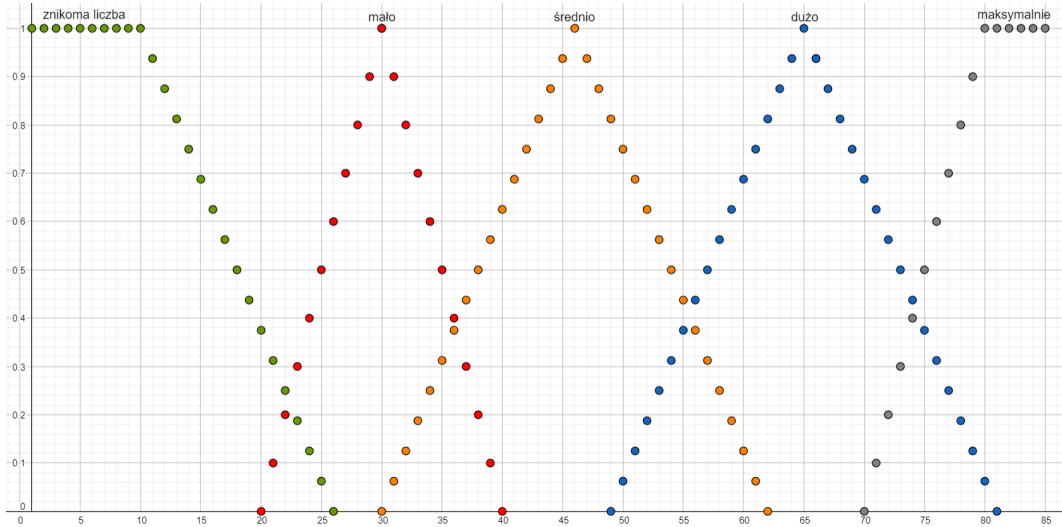
$$\mu_{\text{średnio-gier}}(x) = \begin{cases} 0 & \text{dla } x \in ([1; 30) \cup (62; 85]) \wedge x \in \mathbf{N} \\ 0.0625x - 1.875 & \text{dla } x \in [30; 46] \wedge x \in \mathbf{N} \\ -0.0625x + 3.875 & \text{dla } x \in [46; 62] \wedge x \in \mathbf{N} \end{cases} \quad (23)$$

— dużo

$$\mu_{\text{dużo-gier}}(x) = \begin{cases} 0 & \text{dla } x \in ([1; 49) \cup (81; 85]) \wedge x \in \mathbf{N} \\ 0.0625x - 3.0625 & \text{dla } x \in [49; 65] \wedge x \in \mathbf{N} \\ -0.0625x + 5.0625 & \text{dla } x \in [65; 81] \wedge x \in \mathbf{N} \end{cases} \quad (24)$$

— maksymalnie

$$\mu_{\text{maksymalnie}}(x) = \begin{cases} 0 & \text{dla } x \in [1; 70) \wedge x \in \mathbf{N} \\ 0.1x - 7 & \text{dla } x \in [70; 80] \wedge x \in \mathbf{N} \\ 1 & \text{dla } x \in [80; 85] \wedge x \in \mathbf{N} \end{cases} \quad (25)$$



Rysunek 6. Wykres funkcji przynależności dla zmiennej lingwistycznej "rozegrane gry".

6. średnia liczba zdobytych punktów na mecz - przestrzeń rozważań: liczby rzeczywiste z przedziału: $[0; 36.1]$

— mało

$$\mu_{\text{mało-punktów}}(x) = \begin{cases} 1 & \text{dla } x \in [0; 1) \\ -0.1x + 1.1 & \text{dla } x \in [1; 11] \\ 0 & \text{dla } x \in (11; 36.1] \end{cases} \quad (26)$$

— dostatecznie

$$\mu_{dostatecznie_punktow}(x) = \begin{cases} 0 & \text{dla } x \in [0; 1) \cup (16; 36.1] \\ 0.2x - 0.2 & \text{dla } x \in [1; 6) \\ 1 & \text{dla } x \in [6; 10) \\ -0.125x + 2.25 & \text{dla } x \in [10; 16] \end{cases} \quad (27)$$

— dużo

$$\mu_{duzo_punktow}(x) = \begin{cases} 0 & \text{dla } x \in [0; 11) \\ 0.125x - 1.375 & \text{dla } x \in [11; 19) \\ 1 & \text{dla } x \in [19; 36.1] \end{cases} \quad (28)$$

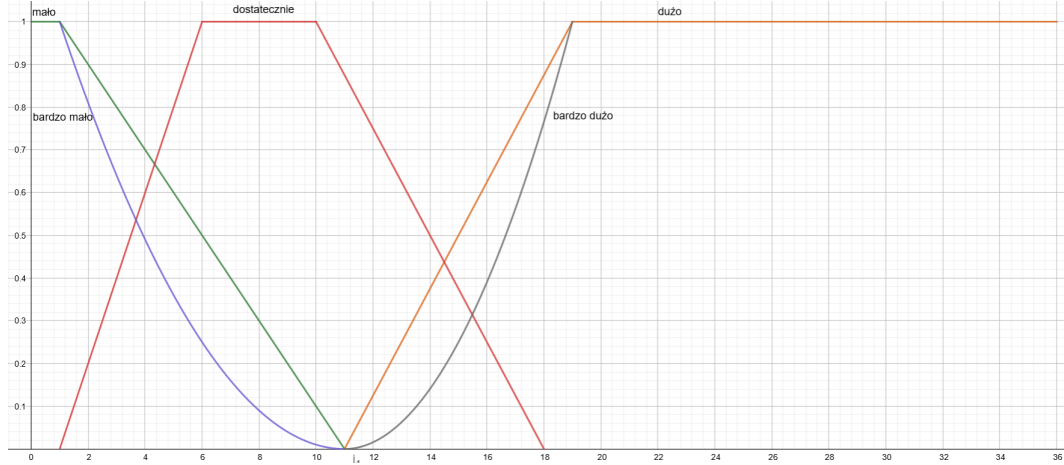
— bardzo mało

$$\mu_{bardzomalo_punktow}(x) = \mu_{malo}(x)^2 \quad (29)$$

— bardzo dużo

$$\mu_{bardzoduzo_punktow}(x) = \mu_{duzo}(x)^2 \quad (30)$$

Oczywiście, zgodnie z regułą semantyczną termin "bardzo mało" występuje przed terminem "mało" w zmiennej lingwistycznej. Powyżej został on wymieniony później, ponieważ definiując funkcję przynależności dla terminu "bardzo mało" korzystamy z funkcji przynależności dla terminu "mało" (konkretnie podnosimy ją do kwadratu).



Rysunek 7. Wykres funkcji przynależności dla zmiennej lingwistycznej "zdobyte punkty".

7. średnia liczba zbiorów na mecz - przestrzeń rozważań: liczby rzeczywiste z przedziału: $[0; 16.3]$

— mało

$$\mu_{malo_zbiorow}(x) = \begin{cases} 1 & \text{dla } x \in [0; 1) \\ -0.25x + 1.25 & \text{dla } x \in [1; 5] \\ 0 & \text{dla } x \in (5; 16.3] \end{cases} \quad (31)$$

— dostatecznie

$$\mu_{dostatecznie_zbiorek}(x) = \begin{cases} 0 & \text{dla } x \in [0; 2) \cup (6; 16.3] \\ 0.5x - 1 & \text{dla } x \in [2; 4) \\ 1 & \text{dla } x \in [4; 5) \\ -x + 6 & \text{dla } x \in [5; 6] \end{cases} \quad (32)$$

— dużo

$$\mu_{duzo_zbiorek}(x) = \begin{cases} 0 & \text{dla } x \in [0; 4) \\ 0.4x - 1.6 & \text{dla } x \in [4; 6.5) \\ 1 & \text{dla } x \in [6.5; 16.3] \end{cases} \quad (33)$$

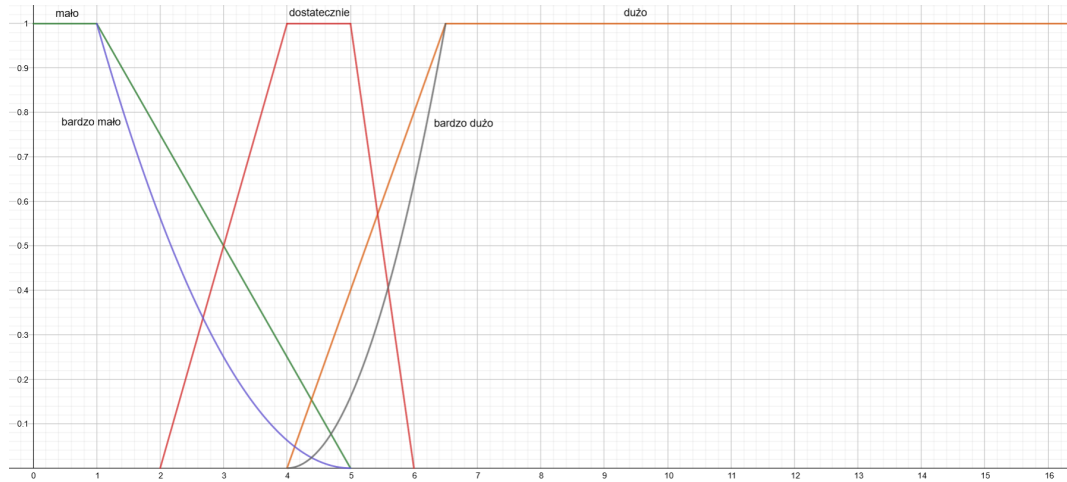
— bardzo mało

$$\mu_{bardzomalo_zbiorek}(x) = \mu_{malo}(x)^2 \quad (34)$$

— bardzo dużo

$$\mu_{bardzoduzo_zbiorek}(x) = \mu_{duzo}(x)^2 \quad (35)$$

Oczywiście, zgodnie z regułą semantyczną termin "bardzo mało" występuje przed terminem "mało" w zmiennej lingwistycznej. Powyżej został on wymieniony później, ponieważ definiując funkcję przynależności dla terminu "bardzo mało" korzystamy z funkcji przynależności dla terminu "mało" (konkretnie podnosimy ją do kwadratu).



Rysunek 8. Wykres funkcji przynależności dla zmiennej lingwistycznej "liczba zbiórek".

8. średnia liczba asyst na mecz - przestrzeń rozważań: liczby rzeczywiste z przedziału: $[0; 11.7]$

— mało

$$\mu_{malo_asyst}(x) = \begin{cases} 1 & \text{dla } x \in [0; 0.5) \\ -x + 1.5 & \text{dla } x \in [0.5; 1.5] \\ 0 & \text{dla } x \in (1.5; 11.7] \end{cases} \quad (36)$$

— dostatecznie

$$\mu_{dostatecznie_asyst}(x) = \begin{cases} 0 & \text{dla } x \in [0; 0.5) \cup (3; 11.7] \\ x - 0.5 & \text{dla } x \in [0.5; 1.5) \\ 1 & \text{dla } x \in [1.5; 2) \\ -x + 3 & \text{dla } x \in [2; 3] \end{cases} \quad (37)$$

— dużo

$$\mu_{duzo_asyst}(x) = \begin{cases} 0 & dla\ x \in [0; 2) \\ 0.625x - 1.25 & dla\ x \in [2; 3.6) \\ 1 & dla\ x \in [3.6; 11.7] \end{cases} \quad (38)$$

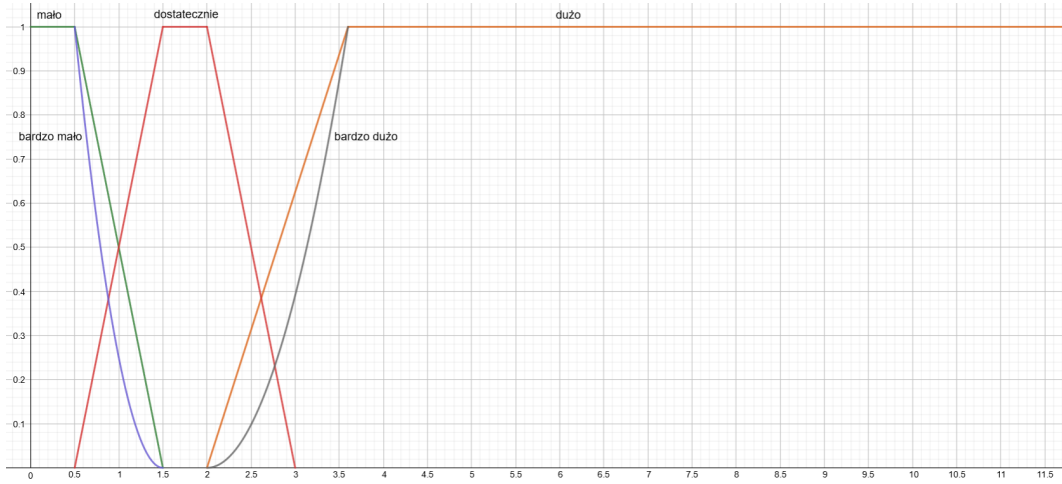
— bardzo mało

$$\mu_{bardzomalo_asyst}(x) = \mu_{malo_asyst}(x)^2 \quad (39)$$

— bardzo dużo

$$\mu_{bardzoduzo_asyst}(x) = \mu_{duzo_asyst}(x)^2 \quad (40)$$

Oczywiście, zgodnie z regułą semantyczną termin ”bardzo mało” występuje przed terminem ”mało” w zmiennej lingwistycznej. Powyżej został on wymieniony później, ponieważ definiując funkcję przynależności dla terminu ”bardzo mało” korzystamy z funkcji przynależności dla terminu ”mało” (konkretnie podnosimy ją do kwadratu).



Rysunek 9. Wykres funkcji przynależności dla zmiennej lingwistycznej ”liczba asyst”.

9. wpływ na drużynę - przestrzeń rozważań: liczby rzeczywiste z przedziału: $[-100; 100]$

— fatalny

$$\mu_{fatalny_wpływ}(x) = \begin{cases} 1 & dla\ x \in [-100; -15) \\ -0.1x - 0.5 & dla\ x \in [-15; -5] \\ 0 & dla\ x \in (-5; 100] \end{cases} \quad (41)$$

— negatywny

$$\mu_{negatywny_wpływ}(x) = \begin{cases} 0 & dla\ x \in [-100; -15) \cup (0; 100] \\ 0.125x + 1.875 & dla\ x \in [-15; -7) \\ 1 & dla\ x \in [-7; -2.5) \\ -0.4x & dla\ x \in [-2.5; 0] \end{cases} \quad (42)$$

— neutralny

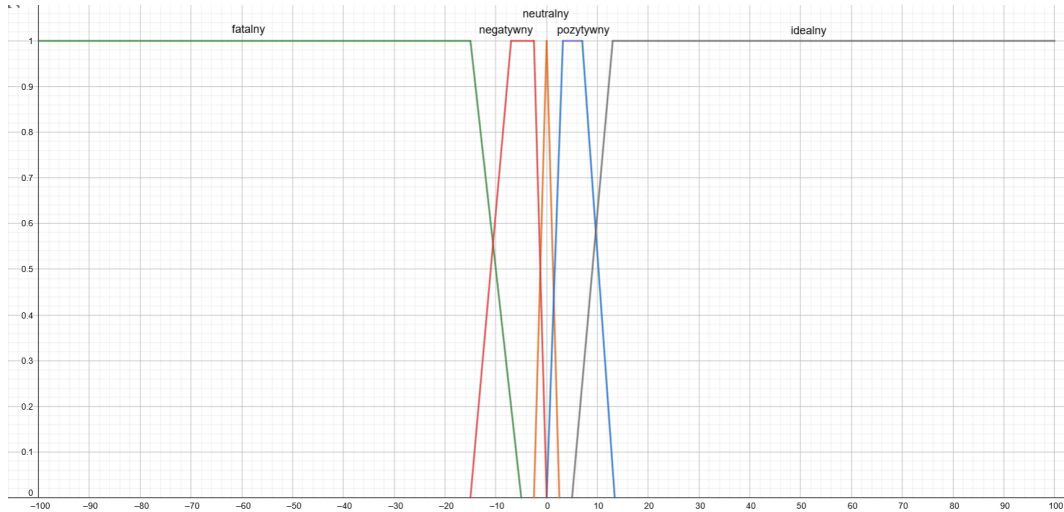
$$\mu_{neutralny_wpływ}(x) = \begin{cases} 0 & \text{dla } x \in [-100; -2.5) \cup (2.5; 100] \\ 0.4x + 1 & \text{dla } x \in [-2.5; 0) \\ -0.4x + 1 & \text{dla } x \in [0; 2.5] \end{cases} \quad (43)$$

— pozytywny

$$\mu_{pozytywny_wpływ}(x) = \begin{cases} 0 & \text{dla } x \in [-100; 0) \cup (13.4; 100] \\ 0.3125x & \text{dla } x \in [0; 3.2) \\ 1 & \text{dla } x \in [3.2; 7) \\ -0.15625x + 2.09375 & \text{dla } x \in [7; 13.4] \end{cases} \quad (44)$$

— idealny

$$\mu_{idealny_wpływ}(x) = \begin{cases} 0 & \text{dla } x \in [-100; 5) \\ 0.125x - 0.625 & \text{dla } x \in [5; 13) \\ 1 & \text{dla } x \in [13; 100] \end{cases} \quad (45)$$



Rysunek 10. Wykres funkcji przynależności dla zmiennej lingwistycznej "wpływ na drużynę".

10. skuteczność rzutów - przestrzeń rozważań: liczby rzeczywiste z przedziału: $[0; 1]$

— fatalny

$$\mu_{fatalny_skuteczność}(x) = \begin{cases} 1 & \text{dla } x \in [0; .35) \\ -10x + 4.5 & \text{dla } x \in [0.35; 0.45] \\ 0 & \text{dla } x \in (0.45; 1] \end{cases} \quad (46)$$

— nieskuteczny

$$\mu_{nieskuteczny}(x) = e^{-\frac{(x-0.4)^2}{0.003}} \text{ dla } x \in [0; 1] \quad (47)$$

— przeciętny

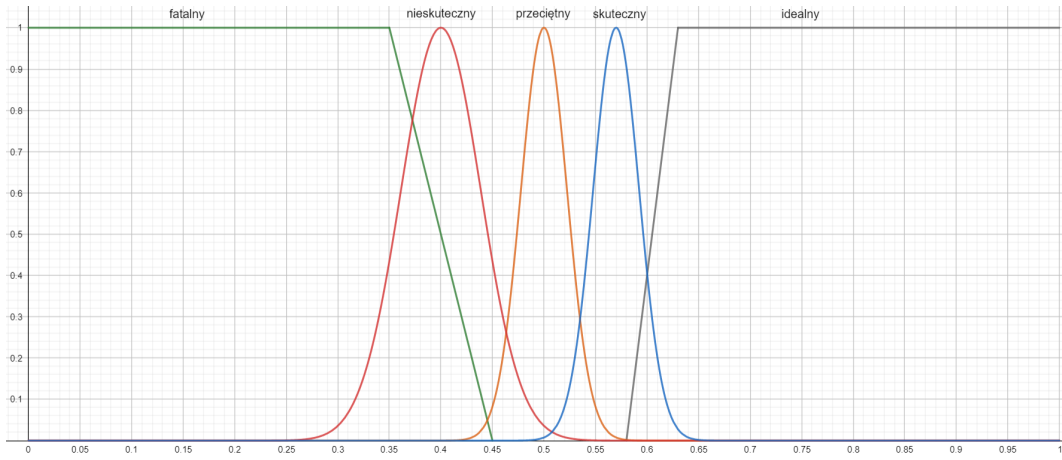
$$\mu_{przecietny_skuteczosc}(x) = e^{-\frac{(x-0.5)^2}{0.001}} \text{ dla } x \in [0; 1] \quad (48)$$

— skuteczny

$$\mu_{skuteczny}(x) = e^{-\frac{(x-0.57)^2}{0.001}} \text{ dla } x \in [0; 1] \quad (49)$$

— idealny

$$\mu_{idealny_skuteczosc}(x) = \begin{cases} 0 & \text{dla } x \in [0; 0.58) \\ 20x - 11.6 & \text{dla } x \in [0.58; 0.63) \\ 1 & \text{dla } x \in [0.63; 1] \end{cases} \quad (50)$$



Rysunek 11. Wykres funkcji przynależności dla zmiennej lingwistycznej "skuteczność rzutów".

11. procent asyst - przestrzeń rozważań: liczby rzeczywiste z przedziału: $[0; 1]$

— fatalny

$$\mu_{fatalny_asyst}(x) = \begin{cases} 1 & \text{dla } x \in [0; 0.01) \\ -25x + 1.25 & \text{dla } x \in [0.01; 0.05] \\ 0 & \text{dla } x \in (0.05; 1] \end{cases} \quad (51)$$

— mały

$$\mu_{maly_asyst}(x) = \begin{cases} 0 & \text{dla } x \in [0; 0.01) \cup (0.06; 1] \\ 25x - 0.25 & \text{dla } x \in [0.01; 0.05] \\ 1 & \text{dla } x \in [0.05; 0.06] \\ -25x + 2.5 & \text{dla } x \in [0.06; 0.10] \end{cases} \quad (52)$$

— przeciętny

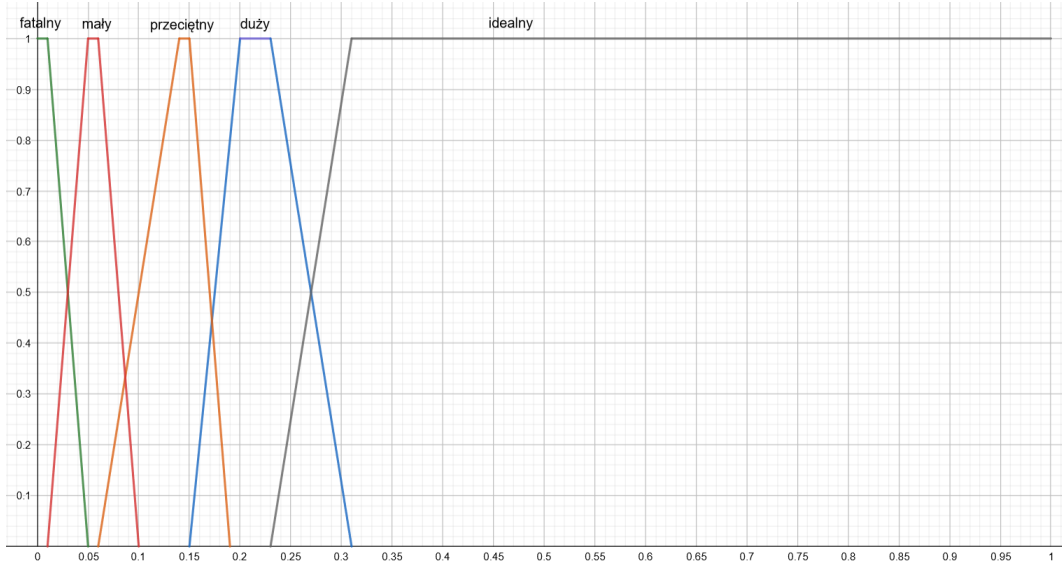
$$\mu_{przecietny_asyst}(x) = \begin{cases} 0 & \text{dla } x \in [0; 0.06) \cup (0.19; 1] \\ 12.5x - 0.75 & \text{dla } x \in [0.06; 0.14] \\ 1 & \text{dla } x \in [0.14; 0.15] \\ -25x + 4.75 & \text{dla } x \in [0.15; 0.19] \end{cases} \quad (53)$$

— duży

$$\mu_{duzy_asyst}(x) = \begin{cases} 0 & \text{dla } x \in [0; 0.15) \cup (0.31; 1] \\ 20x - 3 & \text{dla } x \in [0.15; 0.20) \\ 1 & \text{dla } x \in [0.20; 0.23) \\ -12.5x + 3.875 & \text{dla } x \in [0.23; 0.31] \end{cases} \quad (54)$$

— idealny

$$\mu_{idealny_asyst}(x) = \begin{cases} 0 & \text{dla } x \in [0; 0.23) \\ 12.5x - 2.875 & \text{dla } x \in [0.23; 0.31) \\ 1 & \text{dla } x \in [0.31; 1] \end{cases} \quad (55)$$



Rysunek 12. Wykres funkcji przynależności dla zmiennej lingwistycznej ”procent asyst”.

3.2. Kwantyfikatory lingwistyczne

W programie zostały wykorzystane 2 typy kwantyfikatorów: absolutne oraz względne.

Kwantyfikatory absolutne wraz ze wzorami oraz wykresem prezentują się następująco:

— Więcej niż 0

$$\mu_{wiecej_niz_0}(x) = \begin{cases} -0.000667x + 1 & \text{dla } x \in [0; 1500] \\ 0 & \text{dla } x \in (1500; 11144] \end{cases} \quad (56)$$

— Około 2000

$$\mu_{okolo_2000}(x) = \begin{cases} 0 & \text{dla } x \in [0; 500) \\ 0.000667x - 0.333 & \text{dla } x \in [500; 2000) \\ -0.000667x + 2.333 & \text{dla } x \in [2000; 3500) \\ 0 & \text{dla } x \in [3500; 11144] \end{cases} \quad (57)$$

— Około 4000

$$\mu_{okolo_4000}(x) = \begin{cases} 0 & dla\ x \in [0; 2500) \\ 0.000667x - 1.667 & dla\ x \in [2500; 4000) \\ -0.000667x + 3.667 & dla\ x \in [4000; 5500) \\ 0 & dla\ x \in [5500; 11144] \end{cases} \quad (58)$$

— Około 6000

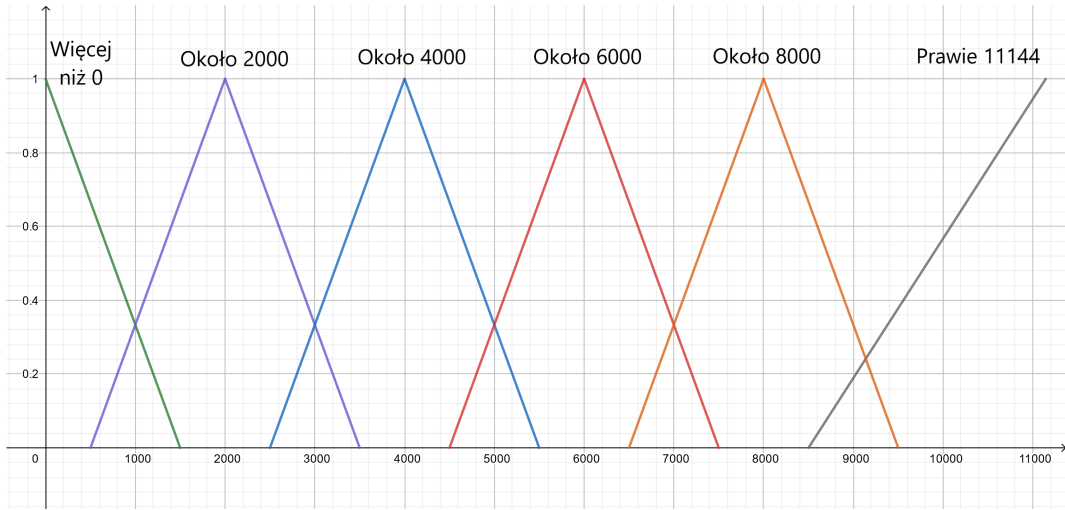
$$\mu_{okolo_6000}(x) = \begin{cases} 0 & dla\ x \in [0; 4500) \\ 0.000667x - 3 & dla\ x \in [4500; 6000) \\ 0.000667x + 5 & dla\ x \in [6000; 7500) \\ 0 & dla\ x \in [7500; 11144] \end{cases} \quad (59)$$

— Około 8000

$$\mu_{okolo_8000}(x) = \begin{cases} 0 & dla\ x \in [0; 6500) \\ 0.000667x - 4.333 & dla\ x \in [6500; 8000) \\ -0.000667x + 6.333 & dla\ x \in [8000; 9500) \\ 0 & dla\ x \in [9000; 11144] \end{cases} \quad (60)$$

— Prawie 11144

$$\mu_{prawie_11144}(x) = \begin{cases} 0 & dla\ x \in [0; 8500) \\ 0.000378x - 3.215 & dla\ x \in [8500; 11144] \end{cases} \quad (61)$$



Rysunek 13. Wykres funkcji przynależności dla kwantyfikatora absolutnego.

Kwentyfikatory względne wraz z wzorami oraz wykresem wyglądają następująco:

— Prawie żaden

$$\mu_{prawie_zaden}(x) = e^{-\frac{x^2}{0.01}} \quad dla\ x \in [0; 1]. \quad (62)$$

— Około 1/4

$$\mu_{okolo_1/4}(x) = e^{-\frac{(x-0.25)^2}{0.01}} \quad dla\ x \in [0; 1]. \quad (63)$$

— Około 1/2

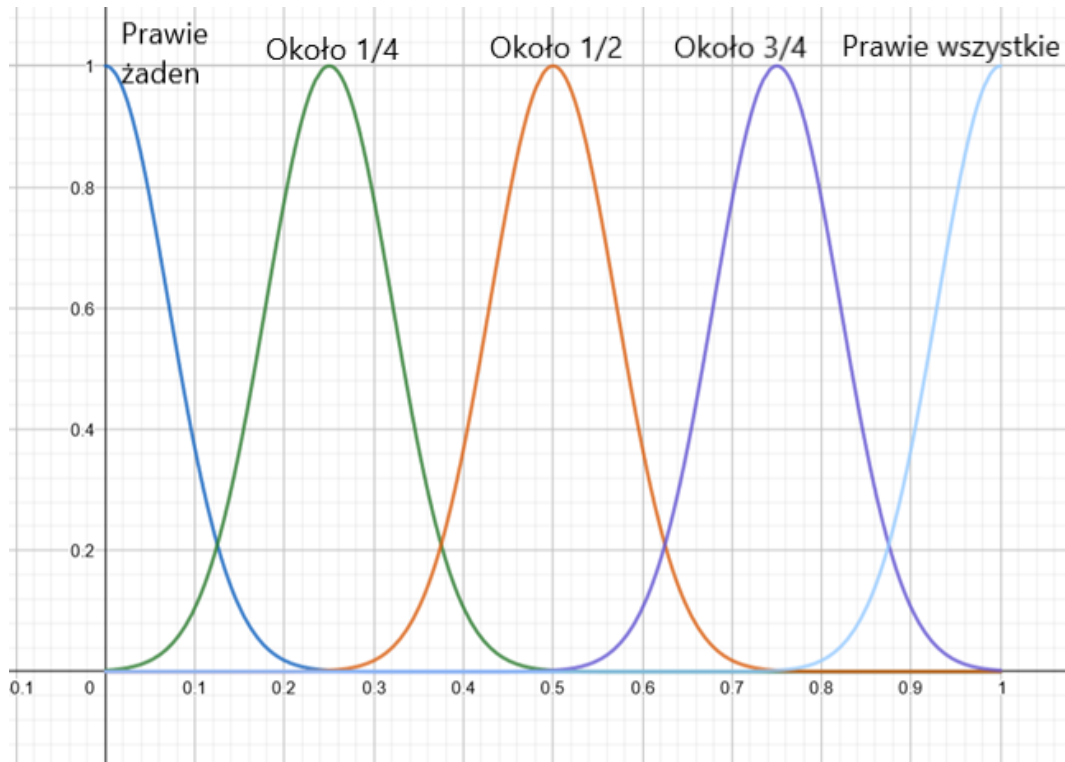
$$\mu_{około_{1/2}}(x) = e^{-\frac{(x-0.5)^2}{0.01}} \quad \text{dla } x \in [0; 1]. \quad (64)$$

— Około 3/4

$$\mu_{około_{3/4}}(x) = e^{-\frac{(x-0.75)^2}{0.01}} \quad \text{dla } x \in [0; 1]. \quad (65)$$

— Prawie wszystkie

$$\mu_{prawie_wszystkie}(x) = e^{-\frac{(x-1)^2}{0.01}} \quad \text{dla } x \in [0; 1]. \quad (66)$$



Rysunek 14. Wykres funkcji przynależności dla kwantyfikatora relatywnego.

4. Narzędzia obliczeniowe: projekt (wybór, implementacja) i diagram UML pakietu obliczeń rozmytych. Diagram UML generatora podsumowań

4.1. Diagram pakietu obliczeń rozmytych

Jako iż podsumowania lingwistyczne opierają się na teorii zbiorów rozmytych, na potrzeby tego zadania postanowiliśmy zaimplementować własny pakiet przeznaczony do obliczeń rozmytych. Pakiet nazywa się *FuzzySetCalculations* i składa się z następujących elementów:

FuzzySet - klasa, która reprezentuje zbiór rozmyty. Metody tej klasy zapewniają najbardziej istotne funkcjonalności dla podsumowań własności. Pierwszą z nich jest metoda o nazwie `getHeight()`, która zwraca wysokość

zbioru rozmytego, tj. maksymalną wartość jaką przyjmuje funkcja przynależności w całej przestrzeni rozważań. Kolejną metodą to `getSupport()`, której wynikiem wykonania są wszystkie te elementy zbioru rozmytego, które przynależą do zbioru rozmytego w stopniu większym niż 0. Bardzo podobną metodą jest `getAlphaCount(double y)`, która jako argument przyjmuje wartość zbioru rozmytego. Argument ten stanowi próg dla selekcji elementów, dla których wartość funkcji przynależności jest powyżej niego. Kolejną w kolejności jest metoda `getComplement(double x)`, której zadaniem jest zwrócenie dopełnienia zbioru rozmytego dla argumentu x , czyli liczby, którą trzeba dodać w celu uzyskania wartości 1. Następne dwie metody, czyli `getAndResult(List<FuzzySet> fuzzySet)` i `getOrResult(List<FuzzySet> fuzzySet)` odpowiadają za operacje sumy (OR) i iloczynu (AND) zbiorów rozmytych (wynikiem tych operacji również jest zbiór rozmyty). Suma zbiorów rozmytych jest obliczana jako maksimum (wykorzystanie T-normy) ze wszystkich wartości zbiorów rozmytych przekazanych w argumencie dla elementu x . Iloczyn zbiorów rozmytych jest obliczany jako minimum (wykorzystanie S-normy) ze wszystkich wartości zbiorów rozmytych przekazanych w argumencie dla wszystkich elementów w przestrzeni. Następną metodą jest metoda `getDegreeOfMembership(double x)`, która dla elementu x zwraca stopień przynależności do zbioru rozmytego (w metodzie tej następuje de facto wywołanie metody `countMembership(double x)` z klasy `MembershipFunction`. Kolejną metody to `isConvex()` oraz `isConcave()`, które zwracają wartość logiczną `true` w momencie, gdy zbiór rozmyty jest wypukły oraz wklęsły. Ostatnią z metod, którą planujemy zaimplementować jest metoda `getCardinality()`, dzięki której otrzymujemy moc zbioru rozmytego tj. sumę wartości funkcji charakterystycznej z wszystkich elementów zbioru rozmytego. [3]

MembershipFunction - klasa abstrakcyjna, która reprezentuje funkcje przynależności do zbioru rozmytego. Klasa ta zawiera jedną metodę abstrakcyjną o nazwie `countMembership(double x)`. Dzięki temu, w klasach dziedziczących można zaimplementować obliczanie stopnia przynależności w zależności od tego, jakiego rodzaju funkcji przynależności chcemy użyć. [5] Z klasy `MembershipFunction` dziedziczy pięć klas:

- **TrapezoidalMembershipFunction** - reprezentuje trapezoidalną funkcję przynależności. W przypadku klasy `LeftTrapezoidalMembershipFunction` (funkcja malejąca) wykorzystane zostają 3 argumenty:

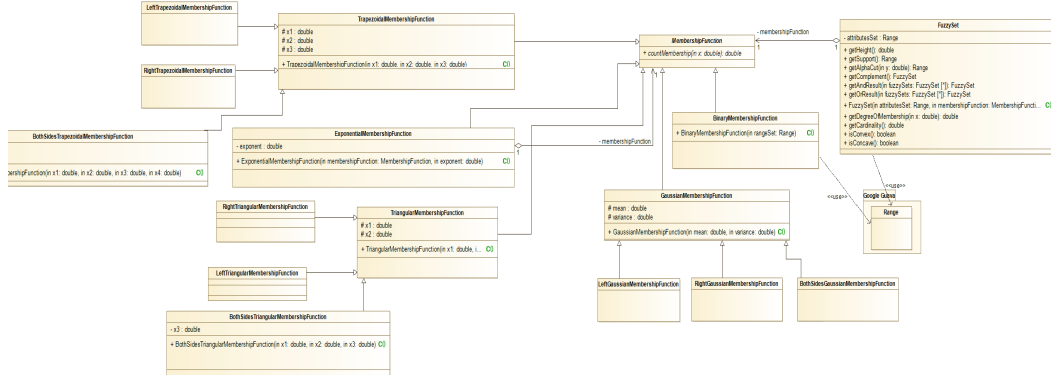
- x1 - początek przedziału od którego funkcja przyjmuje wartość 1
- x2 - koniec przedziału dla którego funkcja przyjmuje wartość 1 i początek przedziału dla funkcji malejącej
- x3 - koniec przedziału dla funkcji malejącej i początek przedziału dla funkcji malejącej

W przypadku klasy `RightTrapezoidalMembershipFunction` również występują 3 argumenty:

- x1 - początek przedziału od którego funkcja jest rosnąca
- x2 - koniec przedziału dla którego funkcja jest rosnąca i początek przedziału dla funkcji stałej o wartości 1
- x3 - koniec przedziału dla funkcji stałej o wartości równej 1

Do funkcjonowania klasy `BothSidesTrapezoidalMembershipFunction` niezbędne są natomiast 4 argumenty:

- x1 - początek przedziału od którego funkcja zaczyna rosnąć
- x2 - koniec przedziału dla funkcji rosnącej i początek przedziału dla funkcji stałej o wartości 1
- x3 - koniec przedziału dla funkcji stałej o wartości 1 i początek przedziału dla funkcji malejącej
- x4 - koniec przedziału dla funkcji malejącej
- **TriangularMembershipFunction** - reprezentuje trójkątną funkcję przynależności. Jako, że funkcja ta może być malejąca, rosnąca lub też składać się z obu tego rodzaju funkcji, w tym celu z klasy GaussianMembershipFunction dziedziczą trzy funkcje, które reprezentują wynienione wyżej rodzaje monotoniczności. W przypadku klasy LeftTriangularMembershipFunction (funkcja malejąca) wykorzystane zostają 3 argumenty:
 - x1 - początek przedziału od którego funkcja zaczyna maleć
 - x2 - koniec przedziału dla funkcji malejącej i początek przedziału dla funkcji stałej o wartości 1
 W przypadku klasy RightTriangularMembershipFunction również występują 2 argumenty:
 - x1 - początek przedziału od którego funkcja zaczyna rosnąć
 - x2 - koniec przedziału dla funkcji rosnącej.
 Do funkcjonowania klasy BothSidesTriangularMembershipFunction niezbędne są natomiast 3 argumenty:
 - x1 - początek przedziału od którego funkcja zaczyna rosnąć
 - x2 - koniec przedziału dla funkcji rosnącej i początek przedziału dla funkcji stałej o wartości 1
 - x3 - koniec przedziału dla funkcji malejącej.
- **ExponentialMembershipFunction** - reprezentuje wykładniczą funkcję przynależności. Jej zadaniem jest wzmocnienie/osłabienie innej funkcji przynależności (np. poprzez dodanie słowa bardzo). Argumenty, które są przez nią przyjmowane to:
 - exponent - wykładnik funkcji
 - membershipFunction - wybrana funkcja przynależności
- **GaussianMembershipFunction** - reprezentuje funkcję Gaussa, jako funkcję przynależności. Parametry przez nią przyjmowane to średnia oraz wariancja, które mają bezpośredni wpływ na element, gdzie znajduje się maksimum oraz na rozłożenie funkcji w dziedzinie. Jako, że funkcja ta może być malejąca, rosnąca lub też składać się z obu tego rodzaju funkcji, w tym celu z klasy GaussianMembershipFunction dziedziczą trzy funkcje, które reprezentują wynienione wyżej rodzaje monotoniczności.
- **BinaryMembershipFunction** - stanowi funkcję przynależności dla zbiorów klasycznych, tj. takich w których stopień przynależności do zbioru wynosi 0 lub 1. Jedynym argumentem jest rangeSet, który to zakres określa, dla jakich elementów funkcja zwraca 1.



Rysunek 15. Diagram UML pakietu FuzzySetCalculations.

4.2. Diagram UML generatora podsumowań. Krótka instrukcja użytkownika

Drugim pakietem użytym przez nas w aplikacji jest pakiet generatora podsumowań. Składa się on z następujących elementów: **Label** - klasa reprezentująca pojedynczą etykietę. Etykieta zawiera swoją nazwę oraz pole odpowiadające za klasę FuzzySet. Dzięki temu w etykiecie mamy dostęp do metod klasy FuzzySet, w tym `getDegreeOfMembership(double x)`, czyli stopień przynależności dla zadanego argumentu.

LinguisticVariable - klasa odpowiadająca za zmienną lingwistyczną. Posiada ona swoją nazwę oraz pole `rangeSet`, które odpowiada za przestrzeń rozważań dla podanej zmiennej lingwistycznej. Poza tym klasa ta posiada pole typu `Label`, które odpowiada za wyżej wymienioną etykietę.

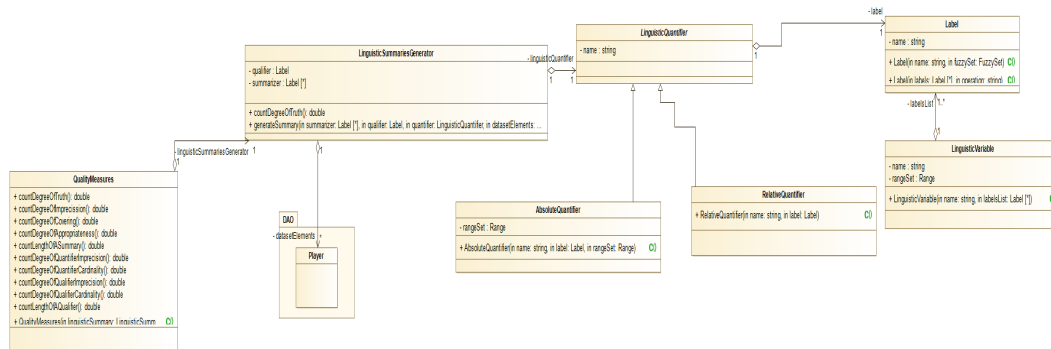
LinguisticQuantifier - klasa abstrakcyjna reprezentująca kwantyfikator lingwistyczny. Kwantyfikator lingwistyczny posiada etykietę. Z klasy abstrakcyjnej `LinguisticQuantifier` dziedziczą dwie następujące klasy:

- **AbsoluteQuantifier** - kwantyfikator absolutny posiadający dodatkowo poza polami, które dziedziczy przedział rozważań.
- **RelativeQuantifier** - kwantyfikator względny.

Player - klasa z modułu DAO odpowiadającego za wczytywanie danych. Klasa ta odpowiada za pojedynczego zawodnika wczytanego z bazy danych.

LinguisticSummariesGenerator - klasa odpowiadająca za generowanie podsumowań, a w zasadzie pojedynczego podsumowania. Zawiera ona kwantyfikator lingwistyczny, pole typu `Player`, co odpowiada podmiotowi podsumowania. Klasa zawiera także pola: `qualifier` (kwalifikator) oraz `summarizer` (sumaryzator). Pierwsze z nich jest typu `Label`, a drugie jest Listą typu `Label` (jako, że sumaryzatorów może być więcej niż 1). Klasa ta pozwala także oczywiście na wygenerowanie podsumowania.

QualityMeasures - klasa zawierająca jedno pole - `LinguisticSummariesGenerator`. Klasa ta pozwala na obliczenie miar jakości od T_1 do T_{11} [1] dla wygenerowanego podsumowania.



Rysunek 16. Diagram UML generatora podsumowań.

Diagram UML generatora podsumowań (warstwy obliczeniowej oraz interfejsu użytkownika). Krótki ilustrowany opis jak użytkownik może korzystać z aplikacji, w szczególności wprowadzać parametry podsumowań, odczytywać wyniki oraz definiować własne etykiety i kwantyfikatory. Wersja JRE i inne wymagania niezbędne do uruchomienia aplikacji przez użytkownika na własnym komputerze.

Sekcja uzupełniona jako efekt zadania Tydzień 11 wg Harmonogramu Zajęć na WIKAMP KSR.

5. Jednopoimotowe podsumowania lingwistyczne.

Miary jakości, podsumowanie optymalne

Wyniki kolejnych eksperymentów wg punktów 2.-4. opisu projektu 2. Listy podsumowań jednopoimotowych i tabele/rankingi podsumowań dla danych atrybutów obowiązkowe i dokładnie opisane w „captions” (tytułach), konieczny opis kolumn i wierszy tabel. Dla każdego podsumowania podane miary jakości oraz miara jakości podsumowania optymalnego.

Sekcja uzupełniona jako efekt zadania Tydzień 11 wg Harmonogramu Zajęć na WIKAMP KSR.

6. Wielopoimotowe podsumowania lingwistyczne i ich miary jakości

Wyniki kolejnych eksperymentów wg punktów 2.-4. opisu projektu 2. Uzasadnienie i metoda podziału zbioru danych na rozłączne podmioty. Listy podsumowań wielopoimotowych i tabele/rankingi podsumowań dla danych atrybutów obowiązkowe i dokładnie opisane w „captions” (tytułach), konieczny opis kolumn i wierszy tabel. Konieczne uwzględnienie wszystkich 4-ch form podsumowań wielopoimotowych.

****** Możliwe sformułowanie zagadnienia wielopoimotowego podsumowania optymalnego ******.

****Ewentualne wyniki realizacji punktu „na ocenę 5.0” wg opisu Projektu 2. i ich porównanie do wyników z części obowiązkowej**.**

Sekcja uzupełniona jako efekt zadania Tydzień 12 wg Harmonogramu Zajęć na WIKAMP KSR.

7. Dyskusja, wnioski

Dokładne interpretacje uzyskanych wyników w zależności od parametrów klasyfikacji opisanych w punktach 3.-4 opisu Projektu 2. Szczególnie istotne są wnioski o charakterze uniwersalnym, istotne dla podobnych zadań. Omówić i wyjaśnić napotkane problemy (jeśli były). Każdy wniosek/problem powinien mieć poparcie w przeprowadzonych eksperymentach (odwołania do konkretnych wyników: tabel i miar jakości). Ocena które wybrane kwantyfikatory, sumaryzatory, kwalifikatory i/lub ich miary jakości mają małe albo duże znaczenie dla wiarygodności i jakości otrzymanych agregacji/podsumowań. Dla końcowej oceny jest to najważniejsza sekcja sprawozdania, gdyż prezentuje poziom zrozumienia rozwiązywanego problemu.

****** Możliwości kontynuacji prac w obszarze logiki rozmytej i wnioskowania rozmytego, zwłaszcza w kontekście pracy inżynierskiej, magisterskiej, naukowej, itp. ******

Sekcja uzupełniona jako efekt zadań Tydzień 11 i Tydzień 12 wg Harmonogramu Zajęć na WIKAMP KSR.

8. Braki w realizacji projektu 2.

Wymienić wg opisu Projektu 2. wszystkie niezrealizowane obowiązkowe elementy projektu, ewentualnie podać merytoryczne (ale nie czasowe) przyczyny tych braków.

Literatura

- [1] A. Niewiadomski, Zbiory rozmyte typu 2. Zastosowania w reprezentowaniu informacji. Seria „Problemy współczesnej informatyki” pod redakcją L. Rutkowskiego. Akademicka Oficyna Wydawnicza EXIT, Warszawa, 2019.
- [2] S. Zadrozny, Zapytania nieprecyzyjne i lingwistyczne podsumowania baz danych, EXIT, 2006, Warszawa
- [3] A. Niewiadomski, Methods for the Linguistic Summarization of Data: Applications of Fuzzy Sets and Their Extensions, Akademicka Oficyna Wydawnicza EXIT, Warszawa, 2008.
- [4] Baza danych zawierająca statystyki koszykarzy z ligi NBA z lat 1996-2020 [przeglądany 04.05.2021] Dostępna w: <https://www.kaggle.com/justinas/nba-players-data>
- [5] Dokumentacja diagramów UML [przeglądana 24.05.2021] Dostępna w: <https://www.uml-diagrams.org/>

Literatura zawiera wyłącznie źródła recenzowane i/lub o potwierdzonej wiarygodności, możliwe do weryfikacji i cytowane w sprawozdaniu.