



UNIwersYTET  
IM. ADAMA MICKIEWICZA  
W POZNANIU

Wydział Matematyki i Informatyki

Marcin Szczepański

**Strategie działania inteligentnych systemów wspierających  
kształcenie operujące na danych nieprecyzyjnych**

Strategies of operation for intelligent tutoring systems operating on imprecise data

Rozprawa doktorska

**Promotor:**

prof. UAM dr hab. Jacek Marciniak

**Dyscyplina:**

Informatyka

**Dziedzina:**

Nauki ścisłe i przyrodnicze

Poznań 2024



## Streszczenie

Niniejsza praca ma na celu zaproponowanie metody adaptacji treści w kursie e-learningowym pozwalającej na wykorzystanie różnych zjawisk dydaktycznych. W rozprawie jest prezentowana architektura kursu z adaptacyjną zawartością, która posłuży do opracowania metody adaptacji opartej na wybranych nieprecyzyjnych danych dydaktycznych, wykorzystując sterowanie rozmyte. W pracy zostaje dokonany przegląd dostępnych innych rozwiązań w obszarze realizowania różnych strategii dydaktycznych. Praca ilustruje również problemy związane z pozyskiwaniem i przetwarzaniem różnych danych dydaktycznych oraz przedstawia metody modelowania danych nieprecyzyjnych. Ponadto, zostaje dokonana ewaluacja różnych sterowników rozmytych w porównaniu z wybranymi algorytmami uczenia maszynowego. Rozprawa prezentuje różne strategie działania inteligentnych systemów wspierających kształcenie, a także omówiony jest proces implementacji i pozyskiwania danych z kursu z adaptacyjną zawartością.

**Słowa kluczowe:** *kurs e-learningowy z adaptacyjną zawartością, sterownik rozmyty, klasyfikacja, uczenie maszynowe, strategia dydaktyczna*



## Abstract

This dissertation aims to propose a method for content adaptation in an e-learning course to leverage various learning phenomena. The study introduces a course architecture featuring adaptive content, which will serve as the foundation for developing an adaptation technique utilizing selected imprecise educational data through fuzzy control. The document surveys existing solutions within the domain of implementing various teaching strategies. It also delineates the challenges associated with acquiring and processing varied educational data, emphasizing the issue of modeling imprecise data. Furthermore, the effectiveness of various fuzzy controllers is compared with machine learning algorithms. Various strategies for the operation of intelligent tutoring systems are presented in the dissertation, along with discussions on the implementation process and data collection within a course with adaptive content.

**Keywords:** *course with an adaptive e-learning content, fuzzy controller, classification, machine learning, teaching strategy*



Cokolwiek robisz lub marzysz, że możesz to zrobić  
— Zacznij tylko.  
W zdecydowaniu drzemie geniusz, siła i magia.  
Zacznij teraz.

*J. W. Goethe*





# Spis treści

<b>Podziękowania</b> . . . . .	11
<b>Rozdział 1. Wprowadzenie</b> . . . . .	13
1.1. Motywacja . . . . .	13
1.2. Cele i zakres pracy . . . . .	14
1.3. Metoda badawcza . . . . .	16
1.4. Znaczenie rozprawy . . . . .	16
1.5. Struktura pracy . . . . .	18
<b>Rozdział 2. Strategie działania inteligentnych systemów wspierających kształcenie</b> . . .	19
2.1. Kursy e-learningowe w systemach LMS . . . . .	20
2.2. Inteligentne systemy wspierające kształcenie . . . . .	21
2.3. Kursy e-learningowe z adaptacyjną zawartością . . . . .	23
2.4. Dane dydaktyczne w procesie budowania systemów klasy ITS i kursów e-learningowych z adaptacyjną zawartością . . . . .	25
2.4.1. Charakterystyka i pozyskiwanie danych dydaktycznych . . . . .	25
2.4.2. Nieprecyzyjność danych dydaktycznych . . . . .	26
2.4.3. Zbiory rozmyte w modelowaniu nieprecyzyjnych danych dydaktycznych . .	27
2.5. Zachowania studentów w kursach e-learningowych . . . . .	30
2.5.1. Wybrane zachowania studentów w kursach e-learningowych . . . . .	31
2.5.2. Zjawisko disengagement w edukacji . . . . .	33
2.6. Sterowanie rozmyte w problemie klasyfikacji . . . . .	34
2.6.1. Wykorzystanie sterowania rozmytego w problemie klasyfikacji . . . . .	35
2.6.2. Sterowniki rozmyte . . . . .	36
2.6.3. Ewaluacja algorytmów klasyfikujących dane . . . . .	42
<b>Rozdział 3. Strategia adaptacji z wykorzystaniem sterowania rozmytego</b> . . . . .	45
3.1. Przygotowanie do badań . . . . .	45
3.1.1. Kurs e-learningowy „Zastosowania zbiorów rozmytych” . . . . .	45
3.1.2. Konwerter <i>fcl2js</i> . . . . .	50
3.1.3. Biblioteka <i>its-student-progress</i> . . . . .	50
3.1.4. Biblioteka <i>its-raport-generator</i> . . . . .	52
3.2. Architektura kursu e-learningowego z adaptacyjną zawartością pozwalająca na stosowanie różnych metod adaptacji . . . . .	53
3.3. Strategia dydaktyczna wykorzystująca zjawisko disengagement . . . . .	58
3.4. Modelowanie zjawiska disengagement z wykorzystaniem sterowania rozmytego . .	60
3.4.1. Zmienne wejściowe sterownika . . . . .	61
3.4.2. Baza reguł eksperckich i proces defuzyfikacji . . . . .	65

3.5. Implementacja kursu z adaptacyjną zawartością wykorzystującego zjawisko disengagement z wykorzystaniem eksperckiego sterownika rozmytego . . . . .	68
3.5.1. Wykorzystanie eksperckiego sterownika rozmytego . . . . .	69
3.5.2. Realizacja architektury kursu e-learningowego z adaptacyjną zawartością . .	71
<b>Rozdział 4. Sterowanie rozmyte a inne metody klasyfikacji studentów . . . . .</b>	<b>75</b>
4.1. Przygotowanie danych . . . . .	75
4.1.1. Wdrożenie kursu na zajęciach dydaktycznych i zbieranie danych z kursu . .	76
4.1.2. Anotowanie pozyskanych danych dydaktycznych . . . . .	80
4.2. Sterowniki rozmyte wygenerowane automatycznie . . . . .	84
4.2.1. Sterowniki Mamdaniego . . . . .	84
4.2.2. Sterowniki rozmyte Takagi-Sugeno . . . . .	88
4.3. Porównanie sterowników rozmytych z metodami uczenia maszynowego w problemie klasyfikacji studentów . . . . .	89
4.3.1. Metody klasyfikacji zawarte w porównaniu . . . . .	89
4.3.2. Porównanie metod klasyfikacji studentów . . . . .	91
4.3.3. Dyskusja . . . . .	96
<b>Rozdział 5. Rozszerzona strategia dydaktyczna wykorzystująca sterowanie rozmyte . . .</b>	<b>99</b>
5.1. Kurs e-learningowy wykorzystany w badaniach . . . . .	99
5.2. Strategia dydaktyczna zastosowana w kursie . . . . .	102
5.3. Sterowniki rozmyte w realizacji strategii dydaktycznej . . . . .	104
5.3.1. Sterownik obliczający poziom kompetencji w zakresie podstaw uczenia maszynowego . . . . .	105
5.3.2. Sterownik obliczający wartość <i>question exchange requirement</i> . . . . .	108
5.3.3. Dyskusja . . . . .	112
<b>Rozdział 6. Podsumowanie . . . . .</b>	<b>113</b>
<b>Załącznik A. Ekspercki sterownik Mamdaniego przekonwertowany do języka JavaScript przy pomocy konwertera <i>fcl2js</i> . . . . .</b>	<b>117</b>
<b>Załącznik B. Kod języka Python wykorzystujący bibliotekę <i>pyFume</i> do generowania sterownika Takagi-Sugeno . . . . .</b>	<b>125</b>
<b>Załącznik C. Kod języka Python prezentujący wygenerowany sterownik Takagi-Sugeno</b>	<b>127</b>
<b>Spis tabel . . . . .</b>	<b>129</b>
<b>Spis rysunków . . . . .</b>	<b>131</b>
<b>Spis wyciągów . . . . .</b>	<b>133</b>
<b>Bibliografia . . . . .</b>	<b>135</b>

## Podziękowania

Niniejsza praca doktorska została przygotowana pod kierunkiem Pana prof. UAM dr hab. Jacka Marciniaka, któremu składam serdeczne podziękowania za nieocenioną pomoc, cenne uwagi i inspirujące sugestie, które miały istotny wpływ na ostateczny kształt tej pracy naukowej. Dodatkowo wyrażam wdzięczność za zaangażowanie oraz życzliwość, które otrzymałem podczas wieloletniej opieki naukowej w trakcie studiów magisterskich i doktoranckich.

Chciałbym również wyrazić swoje uznanie dla wszystkich pracowników Zakładu Sztucznej Inteligencji, a w szczególności członków grupy badawczej *Intelligent Tutoring Systems*, za ich wartościowe wykłady oraz owocną współpracę, która przyczyniła się do mojego rozwoju w trakcie studiów.

Szczególne słowa uznania kieruję w stronę moich przyjaciół, Jacka i Tomasza, za ich nieustanne wsparcie, wspólne przeżycia i wiarę w sukces powstającej pracy doktorskiej. Ich obecność była niezastąpionym elementem tej trudnej, ale jednocześnie fascynującej drogi.

Dziękuję także wszystkim moim znajomym, którzy udzielali mi wsparcia zarówno w trakcie studiów, jak i w procesie pisania tej rozprawy doktorskiej. Ich przychylność i wsparcie były dla mnie niezmiernie cenne.

W szczególny sposób pragnę wyrazić głęboką wdzięczność mojej mamie Annie, jej partnerowi Zbigniewowi oraz mojej siostrze Paulinie za ich nieustające wsparcie, cierpliwość oraz wiarę we mnie, zwłaszcza w trudnych chwilach, które napotykałem w trakcie realizacji tego przedsięwzięcia. Bez ich wsparcia i motywacji ta praca nie osiągnęłaby swojego finalnego kształtu.

*Marcin Szczepański*



## ROZDZIAŁ 1

# Wprowadzenie

## 1.1. MOTYWACJA

Współczesny rozwój technologii wpływa nie tylko na formę treści dydaktycznych, ale również na sam proces nauczania i uczenia się, szczególnie w kontekście kursów e-learningowych. W rozprawie przyjęto założenie, że kurs e-learningowy to zbiór lekcji składających się z multimedialnych i interaktywnych materiałów oraz testów online. Określenie zachowań studentów podczas korzystania z takich kursów jest jednak trudnym zadaniem. Dane pozyskiwane w procesie kształcenia opartym na e-learningu stają się coraz bardziej złożone. Ponieważ studenci pracują z kursami w różny sposób, dane te mogą być różnie interpretowane. Zjawiska opisujące aktywność studenta w kursie, takie jak zaangażowanie [14], motywacja [70, 123], poziom uwagi [140] czy efektywność [160], są zatem z natury nieprecyzyjne, co utrudnia skuteczną analizę i dostosowywanie procesu nauczania do indywidualnych potrzeb. Mimo to, warto analizować te dane i zjawiska, gdyż mogą one posłużyć do doskonalenia kursów e-learningowych poprzez tworzenie strategii adaptacji treści, co pozwala na indywidualizację nauczania.

Motywacją do badań w obszarze tworzenia kursów e-learningowych jest zatem potrzeba efektywnego radzenia sobie z nieprecyzyjnymi pojęciami, które opisują zachowania studentów podczas pracy z kursami. Wspomniana koncepcja adaptacji treści jest coraz częściej wykorzystywana w ramach inteligentnych systemów wspomagających nauczanie (ang. *Intelligent Tutoring Systems; ITS* [177]). W literaturze istnieje wiele rozwiązań, takich jak systemy adaptujące treści w oparciu o style uczenia się [81, 143, 168]. Pojawia się pytanie, jaką architekturę kursu można zaproponować, aby była ona na tyle uniwersalna, by umożliwiała łatwe wdrażanie różnych strategii adaptacji treści.

Kolejnym problemem związanym z ogólną architekturą adaptacyjnego kursu e-learningowego jest opracowanie algorytmu adaptacji. Doświadczeni nauczyciele mogą wskazać zestaw zasad, które mogą sterować treściami w kursie. Dodatkowo, niektóre dane dydaktyczne oraz zjawiska w e-learningu są nieprecyzyjne, co oznacza, że nauczyciel nie musi używać precyzyjnych sformułowań, jak na przykład *jeśli  $x = 30$ , to  $y = 50$* . Może na przykład powiedzieć, że *jeśli student spędza mało czasu w kursie, to jego zaangażowanie jest niewielkie*. Tego typu reguły są zrozumiałe dla ludzi, ponieważ opierają się na nieprecyzyjnej naturze języka. Kluczowym pytaniem jest, jak takie reguły można wykorzystać w formalnym procesie budowania algorytmu adaptacji. Jedną z metod

umożliwiających zastosowanie tych nieprecyzyjnych reguł eksperckich jest **sterowanie rozmyte** (ang. *fuzzy control*) [102], bazujące na teorii zbiorów rozmytych [180]. W tym podejściu tworzone są tzw. **sterowniki rozmyte**, które definiują reguły.

Sterownik rozmyty składa się z zestawu zmiennych lingwistycznych reprezentujących nieprecyzyjne dane (w tym przypadku dane dydaktyczne) oraz eksperckiej bazy reguł, wykorzystujących nieprecyzyjny język [102]. Taki sterownik jest zrozumiały nie tylko dla informatyków czy innych osób zajmujących się sterowaniem rozmytym ale także dla osób, które nie mają doświadczenia technicznego — a takimi są zazwyczaj eksperci w dziedzinie, w ramach której modelowany jest konkretny sterownik. W dobie popularności uczenia maszynowego pojawia się pytanie, czy metoda nieoparta na zbiorach danych może być skuteczna. Jednakże, sterowniki rozmyte można także generować z danych [48, 57, 64, 75], co rodzi pytanie, czy są one lepsze od metod uczenia maszynowego. Te zagadnienia zostaną omówione w niniejszej rozprawie poprzez porównanie sterowników rozmytych z popularnymi metodami klasyfikacji stosowanymi w uczeniu maszynowym.

W problematyce adaptacji treści w kursach e-learningowych oraz ogólnie w rozwiązaniach dla e-learningu, oprócz systemów uczenia maszynowego, tworzy się także systemy regułowe oparte na precyzyjnych regułach [46, 136, 148, 154]. W kontekście dydaktyki może się wydawać, że takie systemy wystarczą do nieskomplikowanych adaptacji treści. Problem pojawia się jednak w sytuacjach granicznych, na przykład gdy system ma ocenić, czy student zaliczył kurs, składający się z określonych aktywności (ćwiczeń) i testów. Proste zsumowanie lub uśrednienie wyników oraz sztywno określony próg zaliczenia mogą spowodować, że student, któremu zabrakło ułamka punktu do zaliczenia, zostanie oceniony jako ten, który nie zaliczył kursu, podczas gdy student, który ledwo przekroczył ten próg, zostanie oceniony pozytywnie. Mimo podobnych wyników punktowych, obaj studenci zostaną ocenieni inaczej. Ponadto, studenci mogą różnie podchodzić do aktywności – jeden może pracować sumiennie we wszystkich zadaniach, ale być na granicy progu zaliczenia, podczas gdy inny może rozwiązać tylko wybrane zadania i zdobyć więcej punktów dzięki szczęśliwemu trafowi. Taki system oceniania jest niesprawiedliwy. Lepszym rozwiązaniem jest wprowadzenie nieostrych granic i zastosowanie nieprecyzyjnych reguł oceny, które pozwalają na bardziej elastyczne podejście do oceny wyników studenta. Dodatkowym problemem systemów regułowych jest często ich rozbudowana baza reguł, co sprawia, że są one trudne do modyfikacji czy poszerzania o kolejne reguły [67].

## 1.2. CELE I ZAKRES PRACY

Na cele niniejszej rozprawy składają się cztery cele główne:

1. **Zaproponowanie metody adaptacji w kursie e-learningowym pozwalającej na wykorzystanie różnych zjawisk opisujących zachowania studentów podczas pracy z kursem** — w rozprawie zostanie zaprezentowany algorytm adaptacji wykorzystujący zjawisko disengagement, czyli zjawisko opisujące spadek zaangażowania studenta podczas pracy z kursem [33]. Ze względu na fakt, że trudno jest jedno-

znacznie przetłumaczyć to pojęcie na język polski — najczęściej tłumaczy się je jako „brak zaangażowania” ale to nie jest to samo co proces obniżania się poziomu zaangażowania — dalej w rozprawie będzie wykorzystywane nazewnictwo angielskie.

2. **Zaproponowanie ogólnej architektury kursu z adaptacyjną zawartością** — w rozprawie zostaną przedstawione dwa kursy z adaptacyjną zawartością wykorzystujące omawianą w pracy architekturę.
3. **Wykorzystanie eksperckiego sterownika rozmytego Mamdaniego [102] w realizacji algorytmu adaptacji treści w kursie e-learningowym** — w rozprawie zostanie przedstawiony sterownik rozmyty oparty na regułach eksperckich wykorzystany w problemie klasyfikacji studentów pod kątem zjawiska disengagement.
4. **Porównanie działania eksperckiego sterownika rozmytego ze sterownikami wygenerowanymi automatycznie oraz z wybranymi metodami uczenia maszynowego** — sterowniki rozmyte można generować automatycznie w oparciu o zbiór danych. W rozprawie zostanie zaprezentowany kurs e-learningowy z adaptacyjną zawartością, który był wykorzystany na zajęciach dydaktycznych. Zebrane dane o postępach studentów w kursie posłużyły w procesie generowania sterowników rozmytych oraz trenowania algorytmów uczenia maszynowego a także ewaluacji metod w ramach porównania.

oraz trzy cele pomocnicze:

1. Przedstawienie metod modelowania wybranych danych dydaktycznych o charakterze nieprecyzyjnym oraz omówienie problemów związanych z pozyskiwaniem i przetwarzaniem różnych danych dydaktycznych.
2. Przygotowanie sterowników rozmytych generowanych na podstawie zebranych danych o postępach studentów w kursie e-learningowym z adaptacyjną zawartością.
3. Zaproponowanie narzędzi pozwalających na wdrożenie kursu z adaptacyjną zawartością wykorzystującego sterownik rozmyty i wykorzystanie danych o postępach studentów w tym kursie.

Zakres niniejszej rozprawy obejmuje:

- przegląd rozwiązań w obszarze strategii działania systemów klasy ITS i kursów z adaptacyjną zawartością;
- omówienie narzędzi technicznych wykorzystywanych w badaniach;
- prezentację architektury kursu z adaptacyjną zawartością pozwalającej na zastosowanie dowolnej metody klasyfikacji studentów;
- prezentację metody adaptacji w kursie e-learningowym umożliwiającej wykorzystanie wybranych zjawisk opisujących strategię studentów podczas pracy z kursem w oparciu o teorię zbiorów rozmytych i sterowanie rozmyte;
- omówienie budowy eksperckiego sterownika rozmytego wykorzystywanego w algorytmie adaptacji;
- omówienie procesu implementacji i zbierania danych w kursie e-learningowym z adaptacyjną zawartością;
- omówienie procesu generowania sterowników rozmytych w oparciu o zebrane dane;

- porównanie metod klasyfikacji w ramach realizacji prezentowanej w pracy metody adaptacji treści;
- prezentację rozszerzonej metody adaptacji w oparciu o prezentowaną w rozprawie architekturę kursu z adaptacyjną zawartością.

### 1.3. METODA BADAWCZA

W ramach prowadzonych badań został opracowany algorytm adaptacji treści opierający się na sterowaniu rozmytym. Opracowano różne sterowniki rozmyte. Pierwszym z nich był ekspercki sterownik Mamdaniego. Przygotowanie takiego sterownika wymagało zbudowania bazy reguł oraz zdefiniowania odpowiednich zmiennych lingwistycznych. W kolejnym kroku wygenerowano sterowniki rozmyte Mamdaniego i Takagi-Sugeno w oparciu o pozyskane dane dydaktyczne.

Aby ocenić działanie metody klasyfikacji wykorzystującej sterowanie rozmyte, zostało dokonane porównanie z popularnymi metodami uczenia maszynowego. Porównanie metod było możliwe dzięki zebraniu danych, które mogły później zostać wykorzystane w procesie trenowania algorytmów uczenia maszynowego. W procesie porównywania algorytmów wybrano kilka metryk ewaluacji (takich jak dokładność czy miara F1). Jest to zgodne z metodologią badań w obszarze uczenia maszynowego [88, 94].

Metodologia badań w kontekście uczenia maszynowego charakteryzuje się pozyskaniem, analizą i przetwarzaniem danych w celu stworzenia nowych metod, a następnie oceną ich efektywności na wydzielonym zbiorze danych testowych. Metodologia w badaniach nad uczeniem maszynowym wymaga starannego doboru danych oraz metod ewaluacji. Dobór danych odgrywa kluczową rolę w procesie badawczym, ponieważ jakość zbioru danych bezpośrednio wpływa na działanie i ogólność stworzonych modeli. W ramach prowadzonych badań wybrano dane istotne dydaktycznie, które zostały przetworzone do postaci znormalizowanej a następnie przypisano im etykiety, dzięki czemu możliwe było wykorzystanie w procesie generowania sterowników rozmytych oraz trenowania algorytmów uczenia maszynowego. Równie ważne jest zastosowanie adekwatnych metod ewaluacji, które pozwolą ocenić rzeczywistą efektywność opracowanych rozwiązań. W rozprawie wybrano podstawowe metryki wykorzystywane w problemie ewaluacji algorytmów klasyfikujących, tzn. dokładność, precyzję, czułość oraz miarę F1.

Wspomniany wcześniej algorytm adaptacji został wdrożony w autorskim kursie e-learningowym w oparciu o ogólną architekturę, która pozwala na stosowanie różnych metod adaptacji [163]. Wybór tej architektury jest uzasadniony poprzez pokazanie rozszerzonej strategii dydaktycznej wdrożonej z wykorzystaniem tej architektury.

### 1.4. ZNACZENIE ROZPRAWY

Prace nad inteligentnymi systemami wspierającymi kształcenie były prowadzone już w latach osiemdziesiątych XX wieku [158, 177]. Trwają one do dzisiaj i polegają na badaniu



architektury, algorytmów w oparciu o które działają oraz zastosowań tego typu systemów [15, 26, 66, 74, 97, 101, 150, 157]. Zadaniem systemów klasy ITS jest dostarczanie osobom uczącym się treści zgodnych z ich indywidualnymi potrzebami, często bez interwencji nauczyciela. Treści podawane są w oparciu o zaimplementowane w systemie i/lub kursie e-learningowym strategie dydaktyczne, czyli schematy postępowania systemu w trakcie interakcji ze studentem [106]. Zestawy strategii, które powinny być wykorzystywane są dobierane przez autora kursu bądź przez system na przykład na podstawie określonego poprzez kwestionariusz stylu uczenia się. Dzięki zróżnicowaniu strategii student może otrzymywać treści w najbardziej odpowiedniej dla niego formie, np. jako e-podręcznik, film, instruktaż czy ćwiczenie.

Praca studentów w systemach e-learningowych niesie za sobą wiele zalet ale także i wad. Nauczanie zdalne niesie ze sobą zwiększone ryzyko pracy niesamodzielnej studentów, co powoduje, że wiedza i umiejętności nie mogą być właściwie sprawdzone i ocenione. Systemy klasy ITS pozwalają na indywidualizację kształcenia [131] a co za tym idzie możliwe jest znalezienie wzorców definiujących pracę niesamodzielną czy przystępowanie do testów bez wcześniejszego przygotowania się w strategiach dydaktycznych.

Coraz częściej w ramach systemów ITS budowane są kursy z adaptacyjną zawartością [7, 81, 108, 159]. Adaptacja treści ma na celu usprawnienie procesu dydaktycznego poprzez dostosowanie kursu do zachowania i potrzeb studenta [43, 47, 130, 132]. W kontekście badań nad tą problematyką, w rozprawie proponowana jest metoda adaptacji treści, która umożliwia wykorzystanie różnych metod klasyfikacji studentów w ramach badanego zjawiska dydaktycznego (takim zachowaniem może być na przykład zaangażowanie, motywacja czy poziom skupienia studenta).

Algorytmy adaptacji mogą bazować na danych zebranych w ankietach (na przykład, kiedy adaptacja bazuje na profilu psychologicznym czy preferowanym stylu uczenia się) [12, 155, 170] lub na danych zbieranych w trakcie pracy studenta z kursem [52, 121]. Takimi danymi mogą być informacje na temat czasu spędzonego w różnych częściach kursu, informacje o wykonanych ćwiczeniach czy punkty zdobywane w testach [134, 149]. W zależności od kursu takie dane mogą być rozumiane przez dydaktyków niejednakowo, bo chociażby ten sam czas spędzony w jednym kursie może być uznany za niski a w innym, mniejszym kursie za wysoki. Zatem trudno jest zdefiniować nieskomplikowany zestaw precyzyjnych reguł adaptacji, aby mógł on być wykorzystany w różnych kursach. Jeżeli jednak reguły bazowałyby na nieprecyzyjnych pojęciach, to możliwe by było wykorzystanie ich w wielu kursach. Plusem takiego rozwiązania jest to, że nie jest potrzebny żaden zbiór do uczenia algorytmów (co jest istotne, bo rzadko kiedy dydaktycy posiadają takie zbiory), ponieważ reguły mogą być pozyskane od ekspertów, którzy nie muszą posiadać wiedzy informatycznej czy matematycznej — baza reguł opiera się na języku naturalnym [102]. W rozprawie zaproponowany zostanie algorytm adaptacji opierający się właśnie na bazie nieprecyzyjnych reguł a następnie jego działanie zostanie porównane z popularnymi metodami uczenia maszynowego.

Kolejne badania w obszarze inteligentnych systemów wspierających kształcenie, w tym kursów z adaptacyjną zawartością, są zasadne, ponieważ pandemia COVID-19 pokazała, że nauczanie zdalne będzie odgrywać coraz większą rolę w edukacji [28, 42, 69]. Zatem potrzebne są systemy, które pozwolą na efektywną i zindywidualizowaną naukę. W rozprawie zostanie zaprezentowane rozwiązanie, w którym adaptacja treści

w kursie e-learningowym będzie się opierać na sterowaniu rozmytym wykorzystującym m.in. reguły eksperckie a dane wykorzystywane w sterowniku zostaną zamodelowane z wykorzystaniem teorii zbiorów rozmytych.

## 1.5. STRUKTURA PRACY

Pozostała część rozprawy jest podzielona na następujące rozdziały:

**Rozdział 2** stanowi przegląd dostępnych rozwiązań w zakresie realizowania różnych strategii dydaktycznych w kursach e-learningowych, inteligentnych systemach wspierających kształcenie — w tym kursach z adaptacyjną zawartością. W rozdziale zostaje poruszony problem zbierania danych dydaktycznych w celu ich dalszego wykorzystania do budowania strategii dydaktycznych. Dokonany zostaje także przegląd zjawisk występujących w e-learningu oraz rozwiązań modelujących zachowania studentów w systemach klasy ITS i kursach z adaptacyjną zawartością. Ostatnia część rozdziału dotyczy natomiast koncepcji sterowania rozmytego w problemie klasyfikacji oraz metod ewaluacji algorytmów stosowanych w tym problemie.

**Rozdział 3** pokazuje narzędzia wykorzystane w badaniach a także przedstawia architekturę kursu e-learningowego z adaptacyjną zawartością oraz strategię dydaktyczną opierającą się na zjawisku disengagement, która wykorzystuje prezentowaną architekturę. Proponowana strategia dydaktyczna jest budowana przy użyciu eksperckiego sterownika rozmytego. Ostatnia część rozdziału stanowi opis implementacji kursu e-learningowego z adaptacyjną zawartością w oparciu o proponowaną strategię dydaktyczną.

**Rozdział 4** opisuje proces automatycznego generowania sterowników rozmytych modelujących zjawisko disengagement w oparciu o dane zbierane podczas pracy studentów z kursem e-learningowym z adaptacyjną zawartością. Rozdział przedstawia proces wdrożenia kursu i zbierania danych a także ich późniejszego anotowania. Następnie w rozdziale zostaje przedstawione porównanie prezentowanych sterowników rozmytych z innymi metodami klasyfikacji wykorzystującymi algorytmy uczenia maszynowego.

**Rozdział 5** przedstawia przykład rozszerzonej strategii dydaktycznej wykorzystującej zjawisko disengagement realizowanej w oparciu o prezentowaną w rozprawie architekturę kursu e-learningowego z adaptacyjną zawartością. Rozszerzona strategia potwierdza zasadność modelowania bardziej złożonych zjawisk i strategii dydaktycznych.

**Rozdział 6** zawiera podsumowanie rozprawy, przedstawia płynące z niej wnioski i zarysowuje tematykę potencjalnych badań w przyszłości.

## ROZDZIAŁ 2

# Strategie działania inteligentnych systemów wspierających kształcenie

Proste strategie dydaktyczne, umożliwiające niezbyt złożoną indywidualizację procesu nauczania, mogą być realizowane za pomocą platform e-learningowych, takich jak np. Moodle [115]. Jednakże takie systemy nie umożliwiają wdrażania strategii opartych na analizie zachowań studentów podczas korzystania z kursów online. Próby modelowania cech i zachowań uczniów są podejmowane w inteligentnych systemach wspierających kształcenie [177]. Niestety, znane systemy tej klasy nie posiadają wystarczająco elastycznej architektury, aby w ramach jednego kursu można było wdrażać różne strategie dydaktyczne, zależnie od potrzeb nauczyciela. W ramach takich systemów powstawały różne kursy e-learningowe z adaptacyjną treścią, bazujące na danych o postępach studentów, takich jak wyniki testów czy profil psychologiczny studenta [62]. Pośród znanych implementacji brak jest jednak prób budowy strategii adaptacyjnych opartych na różnorodnych zachowaniach studentów w trakcie pracy z kursami. Aby to umożliwić, konieczne jest przeprowadzenie analizy danych dydaktycznych, jakie można gromadzić podczas kursów online, a następnie wybór metod klasyfikacji studentów na podstawie tych danych w ramach badanego zjawiska edukacyjnego.

W niniejszym rozdziale zostaną przedstawione istniejące metody tworzenia prostych strategii dydaktycznych w systemach LMS, metody wykorzystujące systemy ITS oraz kursy z adaptacyjną treścią. Przeprowadzona zostanie także analiza dostępnych danych dydaktycznych oraz metod modelowania tych danych, zwłaszcza w przypadkach, gdy mają one nieprecyzyjny charakter. W tym celu zostaną wprowadzone elementy teorii zbiorów rozmytych. Omówione zostaną także różnorodne zachowania studentów w kursach e-learningowych, ze szczególnym uwzględnieniem zjawiska spadku zaangażowania w kursie (tzw. disengagement). Ze względu na nieprecyzyjny charakter różnych zachowań studentów, mogą być one modelowane metodami bazującymi na teorii zbiorów rozmytych. Zostaną zatem wprowadzone pojęcia sterowania rozmytego i sterowników rozmytych. Zachowania te mogą być jednak modelowane również metodami uczenia maszynowego i w niniejszej rozprawie zostanie dokonane porównanie tych metod ze sterownikami rozmytymi. W tym celu na końcu rozdziału wprowadzono definicje metryk ewaluacji algorytmów wykorzystywanych w problemie klasyfikacji.

## 2.1. KURSY E-LEARNINGOWE W SYSTEMACH LMS

Kursy e-learningowe stanowią fundamentalny element współczesnego kształcenia na odległość, umożliwiając uczestnikom zdobywanie wiedzy bez konieczności fizycznej obecności w sali lekcyjnej. Ta forma edukacji jest szczególnie istotna w erze cyfrowej, gdzie dostępność informacji i elastyczność czasowa są kluczowe dla skutecznego uczenia się.

Kursy e-learningowe mogą być udostępniane jako swobodnie dostępne strony internetowe, jednak pojawia się tutaj istotny problem w kwestii śledzenia postępów uczestników. Przy takim podejściu, brak systemu umożliwiającego zapisywanie i monitorowanie osiągnięć studentów może ograniczać efektywność procesu nauczania. Z tego powodu stosowane są systemy klasy LMS (ang. *Learning Management System*), takie jak chociażby platforma Moodle [115], OpenOlat [124] czy Blackboard [21].

Systemy klasy LMS pozwalają przechowywać stan kursu danego studenta na serwerze, co umożliwia nie tylko kontynuację nauki po przerwie, ale także umożliwia nauczycielowi monitorowanie postępów uczestników [76]. Dzięki temu, nawet po opuszczeniu kursu, studenci nie tracą zgromadzonych informacji ani wyników osiągnięć w poszczególnych zadaniach.

Warto zauważyć, że poza tym, że systemy LMS umożliwiają monitorowanie postępów studentów, także pozwalają nauczycielom na budowanie prostych strategii dydaktycznych. Przez strategię dydaktyczną w kontekście e-learningu rozumie się schemat postępowania systemu w trakcie interakcji ze studentem [106].

Na budowanie prostych strategii dydaktycznych pozwala np. platforma Moodle [115]. W 2017 roku Marian Mudrák dokonał analizy tej platformy pod względem możliwości wdrażania różnych strategii dydaktycznych [119]. System Moodle umożliwia zarządzanie widocznością zasobów i aktywności kursu poprzez reguły *jeśli...to...* — mogą być one oparte na informacjach ogólnych takich jak data i czas, używany język, dane z profilu studenta (np. imię i nazwisko), przynależność do grupy a także informacje o wynikach i statusie ukończenia wcześniejszych aktywności. Takie reguły pozwalają na pewną indywidualizację ścieżek uczenia się natomiast nie jest możliwe uwzględnienie cech osobowości studenta czy jego stylu uczenia się.

W literaturze można znaleźć rozwiązania pozwalające na wdrażanie w systemach LMS bardziej zaawansowanych strategii polegających np. na wykorzystaniu stylu uczenia się definiowanego w modelu Feldera-Silverman [51]. Branka Arsovic i Nenad Stefanovic w 2020 roku zaprezentowali badanie pokazujące istotne statystycznie różnice w nauce studentów, którzy pracowali z kursem e-learningowym wdrożonym na platformie Moodle z wykorzystaniem wcześniej przygotowanej i zainstalowanej wtyczki pozwalających na strategię polegającą na adaptacji treści w oparciu o styl uczenia się — studenci ci uzyskiwali lepsze wyniki niż osoby uczące się w modelu tradycyjnym z nauczycielem [10]. Arsovic i Stefanovic pokazali, że w systemie Moodle można zbudować bardziej złożoną adaptację ale trzeba wcześniej napisać odpowiednią wtyczkę i ją zainstalować. Nie zawsze jest to możliwe do zrealizowania, szczególnie na uczelniach, gdzie wdrożenie nowej wtyczki w systemie jest czasochłonne, wymaga zaangażowania wielu osób, a ostateczna decyzja o jej dopuszczeniu zależy od głównego administratora, który może nie wyrazić na to zgody. Dobrym rozwiązaniem może być zatem wykorzystanie architektury pozwalającej na wdrażanie strategii dydaktycznych bezpośrednio w kursie (np. dystrybuowanym

w standardzie SCORM [153] i umieszczanym jako zasób na platformie Moodle czy innym systemie LMS). Właśnie takie rozwiązanie zostanie zaprezentowane w rozprawie.

Inną popularną platformą LMS jest OpenOlat [124]. Zgodnie z dokumentacją możliwości budowania różnych strategii dydaktycznych w tej platformie są mocno ograniczone podobnie jak w systemie Moodle. Autorzy systemu proponują budowanie reguł widoczności elementów kursu w oparciu o dane takie jak data i czas, przynależność do grupy, atrybuty przypisane do studenta w systemie czy wyniki w testach i zadaniach [125]. Analogiczny sposób budowania strategii dydaktycznych wdrożony jest w systemie Blackboard [21]. System ten pozwala na określanie reguł widoczności zasobów kursu w ramach modułu o nazwie „*Adaptive Release*” — reguły budowane są w oparciu o datę, przynależność do grupy czy osiągnięte wyniki [22]. Niestety także i ten system nie pozwala na analizę pracy studentów w celu budowania bardziej złożonych strategii.

Systemy LMS takie jak np. Moodle pozwalają jedynie na wdrażanie bardzo prostych strategii dydaktycznych w kursach budowanych w LMS (bez implementacji i wykorzystania dodatkowych wtyczek), które nie uwzględniają zachowań studentów podczas pracy z kursem e-learningowym. W rozprawie zostanie zaprezentowane rozwiązanie wykorzystujące informacje na temat zachowań studentów podczas pracy z kursem.

## 2.2. INTELIGENTNE SYSTEMY WSPIERAJĄCE KSZTAŁCENIE

Platformy LMS umożliwiają budowanie bardzo prostych strategii dydaktycznych. Istnieje jednak klasa systemów, która pozwala na realizację strategii opartych na analizie pracy studenta z kursem a także informacjach pozyskanych w ankietach. Są to systemy ITS (ang. Intelligent Tutoring Systems), czyli inteligentne systemy wspierające kształcenie. Zgodnie z definicją, **inteligentny system wspierający kształcenie** to system informatyczny, którego celem jest dostarczanie studentowi **zindywidualizowanych** treści oraz informacji zwrotnych **bez interwencji nauczyciela** w oparciu o bezpośrednią interakcję z tym systemem [131].

Badania nad inteligentnymi systemami wspierającymi kształcenie sięgają lat osiemdziesiątych XX wieku [158, 177]. Celem tych systemów jest dostarczanie spersonalizowanych treści edukacyjnych, dostosowanych do indywidualnych potrzeb uczących się osób, z minimalną a najlepiej zerową ingerencją nauczyciela. Wśród tych systemów znajdują się projekty, takie jak *SOPHIE* [24] oraz *GUIDON* [32], które zainicjowały rozwój inteligentnych systemów wspierających kształcenie. Były to aplikacje, które posiadały zdolność do dostosowywania się do potrzeb uczącego się za pomocą algorytmów oraz treści zaimplementowanych na stałe w aplikacji. W późniejszej fazie badań nad inteligentnymi systemami wspierającymi kształcenie, systemy te charakteryzowały się podobnym podejściem [95, 138, 152]. Architektura, w której treści i algorytmy są wbudowane na stałe w aplikację bez możliwości ich wymiany, odzwierciedla kontekst, w którym dane rozwiązanie jest stosowane. Brak potrzeby wymiany treści i algorytmów wynika z faktu, że mocną stroną danego systemu jest właśnie ich starannie opracowana forma i zawartość, a także zaprojektowany sposób dostarczania ich użytkownikowi. Przykładami takich systemów są *Logicando* [95], *Fractions Tutor* [138], czy *ALS-KL* [152].

*Logicando* skupia się na nauczaniu podstaw logiki w szkole podstawowej [95], *Fractions Tutor* na nauce ułamków [138], a *ALS-KL* na nauczaniu języka angielskiego na określonym poziomie ustalonym przez wynik testu wstępnego [152]. Ze względu na brak możliwości wymiany algorytmów regulujących zachowanie systemu oraz dostarczanych treści dla uczących się, co w wielu przypadkach stanowiło istotne ograniczenie, zaczęły pojawiać się systemy, które umożliwiały dokonywanie takich modyfikacji. Przykładami takich systemów są *ADAPT System* [39], *AeLF* [133] czy *KPP System* [182].

Systemy klasy ITS zdolne są do wykrywania stylów uczenia się i dostarczania treści zgodnie z nimi. W ramach rozwiązań w obszarze systemów ITS, style uczenia się są najczęściej określane w oparciu o model Feldera-Silverman [51] lub model VARK [54]. Przykładem systemu ITS, który opiera się na stylach uczenia się jest aplikacja przygotowująca do egzaminu *ICDL* [5], gdzie treści są podzielone na jednostki *learning objects* i zapisane w standardzie SCORM [153]. W systemie wykorzystano narzędzie WEKA [176], które analizując dane z kwestionariusza o uczącym się, identyfikuje jego styl pracy zgodnie z modelem Feldera-Silverman [51]. Uczący się otrzymuje dostęp do kursu w formie najbardziej odpowiadającej jego stylowi, np. e-podręczniku, filmu, instruktażu czy ćwiczenia.

Inteligentne systemy wspierające kształcenie pozwalają również na dostarczanie treści na podstawie bieżącej aktywności uczącego się. Przykładami takich systemów są chociażby *ProTuS* [89], który rekomenduje uczącemu się kolejne treści do nauki programowania, bazując na systemie tagowania treści w aplikacji, oraz *RiPPLE* [87], ogólny system *Learning Content Management System* (LCMS), w którym materiały do nauki są sortowane na podstawie rekomendacji generowanych na podstawie wcześniejszych postępów ucznia. W systemie *ProTuS* w oparciu o historię aktywności studenta z systemem określano styl uczenia się i na podstawie tej informacji zgodnie z zainteresowaniami studenta były wybierane materiały najbardziej odpowiadające danemu stylowi uczenia się [89]. Natomiast w systemie *RiPPLE* algorytm rekomendacji treści opierał się na odpowiedziach studentów w testach a także ich zainteresowaniach oraz indywidualnej oceny trudności pytań w testach [86, 87]. Innym przykładem systemu ITS bazującego na postępach osoby uczącej się jest *SeisTutor* — system, który kieruje studenta na daną ścieżkę uczenia się w oparciu o wynik testu wstępnego [157].

Na przestrzeni lat powstawały kolejne systemy klasy ITS, w których implementowano strategię dydaktyczne różnymi metodami. Czasami opierano się na logice pierwszego rzędu [129], na sieciach neuronowych [52] a także na zbiorach rozmytych [36]. Strategie mogą być budowane w oparciu o potrzeby studenta, które mogą być identyfikowane zarówno w trakcie pracy studenta w kursie jak i przed jej rozpoczęciem [12, 52, 62, 121, 146, 155, 170]. W pierwszym przypadku (tzn. w trakcie pracy studenta) o tym, jakie kolejne treści i w jakiej postaci zostaną dostarczone uczącemu się decydować mogą reguły bazujące na zebranych informacjach o studencie [52, 121, 146]. Informacje te to m.in. wyniki osiągnięte w testach cząstkowych a także dane o liczbie rozwiązanych ćwiczeń czy informacji, za którym podejściem dane zadanie zostało rozwiązane prawidłowo oraz w jakim czasie. Natomiast reguły składają się na algorytm decydujący, którą strategię dydaktyczną, tj. schemat postępowania systemu w trakcie interakcji ze studentem [106], wybrać w momencie, kiedy system ma podjąć decyzję jaką treść podać studentowi. Reguły te mogą mieć postać prostych implikacji w logice pierw-

szego rzędu. W drugim przypadku [12, 62, 155, 170], tzn. przed rozpoczęciem pracy z kursem można zdiagnozować chociażby style uczenia się studenta [51, 54], określić profil psychologiczny na podstawie odpowiednio przygotowanych ankiet (w literaturze pojawiają się rozwiązania bazujące na kwestionariuszu MBTI [12, 122, 144], który określa profil psychologiczny człowieka zgodnie z teorią Junga [79]) czy zebrać dane z historii pracy z innymi kursami. Takie informacje mogą pomóc w określeniu początkowej ścieżki kształcenia studenta tak, aby jego możliwe braki i potrzeby zostały zidentyfikowane.

W ramach badań w obszarze inteligentnych systemów wspierających kształcenie, na Uniwersytecie im. Adama Mickiewicza w Poznaniu wykorzystywano system *Edumatic ITS* [45], który pozwala m.in. na indywidualizację nauczania poprzez system rekomendowania treści z repozytorium [105]. Dzięki tej platformie dokonano także badań w obszarze indywidualizacji postaci materiałów [110] oraz w kontekście zarządzania harmonogramem pracy z danym kursem e-learningowym [107, 109].

Prezentowane przykłady inteligentnych systemów wspierających kształcenie nie posiadają na tyle ogólnej architektury, aby możliwe było wdrażanie różnych strategii dydaktycznych w obrębie tego samego kursu e-learningowego. W rozprawie zostanie przedstawiona ogólna architektura kursu pozwalająca na realizację różnych strategii dydaktycznych.

## 2.3. KURSY E-LEARNINGOWE Z ADAPTACYJNĄ ZAWARTOŚCIĄ

W kontekście inteligentnych systemów wspierających kształcenie coraz częściej budowane są kursy e-learningowe z adaptacyjną zawartością [7, 81, 108, 159]. Takie kursy charakteryzują się tym, że ich zawartość jest dostosowywana do osoby uczącej się na podstawie np. osobistych preferencji studenta czy jego postępów w kursie [43, 47, 130, 132]. Tym, czym różnią się kursy z adaptacyjną zawartością od systemów ITS jest fakt, że adaptacja jest realizowana bezpośrednio w kursie — system, w którym osadzony jest kurs nie ingeruje w jego zawartość. Sama adaptacja natomiast może bazować na danych zebranych w ankietach (np. w momencie, gdy określany jest profil psychologiczny studenta czy jego styl uczenia się) [12, 155, 170] lub na danych zebranych w trakcie pracy studenta z kursem [52, 121].

W literaturze pojawia się wiele rozwiązań wykorzystujących style uczenia się w kontekście adaptacji treści w kursach e-learningowych z adaptacyjną zawartością [10, 43, 81, 136, 143, 155, 168]. Na przykład w 2017 roku Fabiano A. Dorça wraz z innymi naukowcami zaproponował rozwiązanie, w którym były rekomendowane kolejne treści w oparciu o wcześniej zbudowaną ontologię łączącą relacjami learning objects ze stylami uczenia się definiowanymi w modelu Feldera-Silverman [43]. Natomiast w 2019 roku Nisha S. Raj oraz Renumol V. G. zaprezentowali adaptację treści także w oparciu o model Feldera-Silverman, jednak same treści w kursie były dostarczane w oparciu o zbiór reguł [136]. Zaś w 2021 roku Hassan A. El-Sabagh zaproponował podejście, w którym styl uczenia się był określany w modelu VARK [54, 143] — co więcej zaprezentował on re-

zultaty badań, z których wynika, że adaptacja treści w oparciu o style uczenia się miała istotny statystycznie pozytywny wpływ na zaangażowanie studentów mierzone skalą Marcii Dixson, czyli kwestionariuszem zawierającym 48 zdań opisujących zaangażowanie w obszarze umiejętności, interakcji, wydajności i sfery emocjonalnej [40].

Algorytmy adaptacji mogą opierać się nie tylko na stylach uczenia się. Przykładem takiej adaptacji jest rozwiązanie zaproponowane w 2017 roku przez Giuseppe Fenze, Francesco Orciuoli'ego i Demetriosą Sampsona [52]. Autorzy proponują wykorzystanie modelu sieci neuronowej wyuczonej w oparciu o dane pozyskane m.in. od dydaktyków, która pozwala na wygenerowanie reguł określających postać kolejnego zadania, jakie powinien rozwiązać student. Same reguły opierają się na aktywnościach studenta w poprzednich zadaniach.

W ramach badań nad kursami z adaptacyjną zawartością, na Uniwersytecie im. Adama Mickiewicza we współpracy z Uniwersytetem SWPS został opracowany i wdrożony kurs e-learningowy na temat efektywnej współpracy z chińskim partnerem biznesowym [108]. W kursie zaimplementowano strategię adaptacji polegającą na kierowaniu osoby uczącej się na odpowiednią ścieżkę nauki w zależności od odpowiedzi udzielonych w pytaniach typu *case study* w początkowych modułach kursu. Na podstawie tych informacji obliczana była tzw. kompetencja kulturowa, do wyznaczenia której wykorzystano ekspercki sterownik rozmyty. Wdrożono także dodatkową strategię polegającą na kierowaniu osoby uczącej się na ścieżkę swobodnego dostępu do wszystkich materiałów w przypadku wysokiej inteligencji kulturowej obliczonej na podstawie specjalnej ankiety wstępnej. Wdrożenie strategii adaptacji treści w kursie możliwe było dzięki mechanizmom realizowanym w narzędziu Eduexe [44].

Eduexe to program typu authoring tool, który pozwala na tworzenie kursów e-learningowych z gotowych komponentów a następnie ich eksport m.in. do standardu SCORM. System ten pozwala na definiowanie różnych reguł widoczności elementów kursu (podobnie jak np. w systemie Moodle) aczkolwiek reguły te mogą bazować na pojęciach zdefiniowanych przez autora kursu. Autor ma również możliwość wykorzystania dowolnego skryptu języka JavaScript, który obliczy wartości dla zdefiniowanych pojęć w oparciu o różne informacje o aktywnościach studenta w kursie.

Innym narzędziem typu authoring tool pozwalającym na adaptację treści w kursie jest EDUCA [25]. Narzędzie umożliwia budowanie kursu ze wcześniej przygotowanych zasobów i aktywności (np. quizów) w postaci drzewa, w którym autor definiuje różne ścieżki uczenia się. Natomiast, aby budować strategię dydaktyczne, w których treści dostosowywane są do profilu studenta, jego wiedzy, umiejętności czy zainteresowań potrzebne jest otagowanie zasobów dostępnych w systemie, zastosowanie aplikacji serwerowej z silnikiem rekomendacyjnym bazującym na repozytorium treści w osobnej przestrzeni na serwerze. Zatem jest to bardziej skomplikowana architektura niż ta, którą proponuje Eduexe a niekoniecznie przynosi dodatkowe korzyści. W dodatku silnik rekomendacyjny jest zbudowany w oparciu o sieci neuronowe, więc nie zawsze autor kursu ma pełną kontrolę nad tym, jakie treści są dostarczane do studenta.



Prezentowane rozwiązania pokazują, że zawartość kursu e-learningowego może się zmieniać w oparciu o dane o pracy studenta takie jak wyniki w testach czy profil psychologiczny osoby uczącej się lub styl uczenia się. W rozprawie zostanie zaprezentowana strategia oparta na modelowaniu różnych zachowań studentów podczas pracy z kursem e-learningowym oparta na architekturze zbudowanej z wykorzystaniem możliwości technicznych oferowanych przez narzędzie Eduexe.

## 2.4. DANE DYDAKTYCZNE W PROCESIE BUDOWANIA SYSTEMÓW KLASY ITS I KURSÓW E-LEARNINGOWYCH Z ADAPTACYJNĄ ZAWARTOŚCIĄ

Aby zbudować dobrze działający system ITS czy kurs ze skuteczną strategią adaptacji, należy dobrze rozumieć jakie strategie przyjmują studenci podczas pracy w różnych kursach. Przeprowadzenie takiej analizy jest możliwe w przypadku zebrania danych o aktywnościach studentów (na przykład w postaci logów czy raportów). W dodatku w zależności od kontekstu te same dane w różnych kursach dydaktycy mogą interpretować na różne sposoby, ponieważ chociażby czas spędzony w jednym kursie może być dla nauczyciela duży a w innym większym kursie ta sama wartość czasowa będzie uznana za niską. Konieczne jest zatem wybranie odpowiedniej metody modelowania danych, tzn. przygotowania do ich dalszego przetwarzania.

### 2.4.1. Charakterystyka i pozyskiwanie danych dydaktycznych

Nauczyciel wykorzystujący w procesie nauczania kursy e-learningowe na danej platformie LMS ma dostęp do różnych informacji takich jak na przykład czas spędzony w kursie, liczba wejść czy punkty zdobyte przez studentów. Dzięki systematycznemu monitorowaniu postępów uczestników, ich wyników z testów oraz zadań możliwa jest bieżąca ocena skuteczności przekazywanej wiedzy w formie kursu e-learningowego i identyfikację obszarów wymagających dodatkowej uwagi.

W zależności od platformy LMS, dostępne są informacje takie jak czas spędzony w kursie, liczba odwiedzin czy informacje o rozwiązanych ćwiczeniach w kursie. Te dane pomagają zrozumieć zaangażowanie studenta. Na podstawie historii odwiedzin elementów kursu można dokonać analizy ścieżek nauki, co pomaga w zrozumieniu preferencji studenta oraz dostosowaniu treści do indywidualnych potrzeb i stylu nauki.

Przykładowo, zgodnie z dokumentacją [116], na platformie Moodle zbierane są dane o wynikach we wszelkich aktywnościach dostępnych w tym systemie. Wyniki te mogą być zapisywane w formie punktów i ocen. Poza wynikami nauczyciel może oglądać raporty zawierające informacje o aktywnościach w formie podsumowań, zestawień i logów z danymi o dacie wejścia do danego elementu kursu czy liczbie wejść.

Choć można zbierać wiele danych o pracy studenta z kursem e-learningowym, istotne jest wybieranie tych kluczowych oraz dobór odpowiedniego formatu zapisu. Ważny jest także wybór standardu, na przykład SCORM lub xAPI, który umożliwia współpracę kursu z różnymi platformami LMS [153, 179].

Standard **SCORM** został opracowany w 2001 roku przez naukowców grupy **ADL** (*Advanced Distribution Learning*) sponsorowanych przez Departament Obrony Stanów Zjednoczonych – najpierw jako SCORM 1.1, który szybko przeszedł do wersji 1.2 a następnie 2004. Najnowszą wersją jest **SCORM 2004 4th Edition**. Nazwa standardu jest akronimem od *Sharable Content Object Reference Model*. Nazwa nie jest przypadkowa bowiem przez *Sharable Content* należy rozumieć treść, którą można współdzielić w różnych systemach LMS, *Object* oznacza, że treści kursu są dzielone na pewne obiekty, części — lekcje, *Reference Model* natomiast oznacza, że standard powstał na bazie wcześniej dostępnych innych standardów jak **AICC** [3], które zostały zebrane w jedną techniczną specyfikację.

Innym wykorzystywanym w e-learningu standardem jest **xAPI** (Experience API), który umożliwia gromadzenie i analizę danych związanych z aktywnościami uczestników w procesie uczenia się online [179]. Ten standard umożliwia zbieranie informacji o aktywnościach uczestników, takich jak interakcje studenta w kursie, ukończone zadania czy wyniki testów — czyli podobnie jak standard SCORM. Jednak w xAPI można śledzić więcej typów aktywności oraz zapisywać więcej szczegółów o wszelkich działaniach studenta.

#### 2.4.2. Nieprecyzyjność danych dydaktycznych

W zależności od nauczyciela oraz samego kursu e-learningowego, jego formatu, zawartości i poziomu trudności, te same informacje dotyczące aktywności studenta w kursie mogą być interpretowane na różne sposoby. Na przykład, czas spędzony na kursie może być oceniany inaczej w zależności od długości i złożoności tego kursu. W krótkim kursie spędzony czas wynoszący 20 minut może być uznany za wysoki, podczas gdy w kursie o rozbudowanej strukturze może być uznany za niski. Innym przykładem może być liczba wykonanych ćwiczeń — w kursie z trzema ćwiczeniami, ukończenie dwóch z nich może być uznane za znaczące, natomiast w kursie z 30 ćwiczeniami, wykonanie tylko dwóch oznacza niski poziom liczby rozwiązanych ćwiczeń. Zbiorcza analiza różnych kursów byłaby dla nauczyciela prostsza gdyby nie używać liczb lecz wartości lingwistycznych, np. *niewielki czas spędzony w kursie, wysoki poziom rozwiązanych ćwiczeń*, itp. Wartości te są nieprecyzyjne ale zrozumiałe dla człowieka. Natomiast, aby możliwe było ich komputerowe przetwarzanie, należy wybrać odpowiednie narzędzie formalne.

W problemie matematycznego modelowania informacji nieprecyzyjnej wyróżnia się dwa główne podejścia — wykorzystanie zbiorów rozmytych [180] lub zbiorów przybliżonych [127]. Teoria zbiorów rozmytych zajmuje się stopniową przynależnością elementów do zbioru, gdzie każdy element może mieć różny stopień przynależności. Natomiast teoria zbiorów przybliżonych koncentruje się na przybliżonym opisie zbiorów na podstawie informacji niepełnych lub niedokładnych, dzieląc zbiór na elementy pewne i niemożliwe do włączenia w daną definicję [84]. W kontekście modelowania nieprecyzyjnych danych dydaktycznych dane nie są niedokładne czy niepewne zatem lepszym wyborem w tym wypadku będą zbiory rozmyte.

### 2.4.3. Zbiory rozmyte w modelowaniu nieprecyzyjnych danych dydaktycznych

W celu modelowania nieprecyzyjnych danych dydaktycznych, w niniejszej rozprawie zostaną wykorzystane elementy teorii zbiorów rozmytych.

**Definicja 2.1.** Zbiorem rozmytym  $A$  w pewnej (niepustej) przestrzeni  $X$ , nazywa się zbiór par:

$$A = \{(x, \mu_A(x)); x \in X\}$$

gdzie  $\mu_A : X \rightarrow [0; 1]$  jest funkcją przynależności elementu  $x$  do zbioru rozmytego  $A$ .

**Definicja 2.2.** Funkcją przynależności  $\mu$  elementu  $x$  do zbioru rozmytego  $A$ , nazywa się funkcję  $\mu_A : X \rightarrow [0; 1]$ , która każdemu elementowi  $x \in X$  przyporządkowuje jego stopień przynależności do zbioru rozmytego  $A$ , przy czym:

- $\mu_A(x) = 1$  oznacza pełną przynależność do zbioru rozmytego  $A$ ,
- $\mu_A(x) = 0$  oznacza brak przynależności do zbioru rozmytego  $A$ ,
- $\mu_A(x) \in (0; 1)$  oznacza częściową przynależność do zbioru rozmytego  $A$ , przy czym im wyższa wartość, tym większa przynależność.

Funkcje przynależności mogą być dowolne, natomiast powinny odzwierciedlać modelowane zagadnienie. Najczęściej jako funkcje przynależności przyjmuje się funkcje liniowe, trójkątne, trapezoidalne, czy funkcje Gaussa. Natomiast, jeżeli uniwersum zbioru rozmytego jest skończone funkcję można opisać przez zbiór par, w którym zostanie zdefiniowana wartość funkcji dla każdego elementu z uniwersum. Przykładowy zbiór rozmyty został przedstawiony poniżej:

**Przykład 2.1.** Zbiór rozmyty opisujący liczbę wykonanych ćwiczeń w kursie (w procentach) zamodelowany przykładową funkcją przynależności:

$$\mu_A(x) = \begin{cases} 0 & \text{dla } x < 20 \\ \frac{x-20}{60} & \text{dla } x \in [20, 80] \\ 1 & \text{dla } x > 80 \end{cases}$$

W zależności od kontekstu i zastosowania funkcje przynależności mogą być inne.

**Definicja 2.3.** Sumą zbiorów rozmytych  $A$  i  $B$  określa się zbiór  $C$ :

$$C = A \cup B \Leftrightarrow \forall x \in X : C(x) = \mu_A(x) \vee \mu_B(x)$$

przy czym operację  $\vee$  określa się jako maksimum dwóch liczb.

**Definicja 2.4. Iloczynem (przekrojem) zbiorów rozmytych**  $A$  i  $B$  określa się zbiór  $C$ :

$$C = A \cap B \Leftrightarrow \forall x \in X : C(x) = \mu_A(x) \wedge \mu_B(x)$$

przy czym operację  $\wedge$  określa się jako minimum dwóch liczb.

**Definicja 2.5. Notacja singletonowa (Zadeha)**

Zbiór rozmyty  $A$  w przestrzeni  $X$  o skończonej liczbie elementów  $X = \{x_1, x_2, \dots, x_n\}$  można zapisać następująco:

$$A = \frac{\mu_A(x_1)}{x_1} + \frac{\mu_A(x_2)}{x_2} + \dots + \frac{\mu_A(x_k)}{x_k} = \sum_{i=1}^k \frac{\mu_A(x_i)}{x_i}$$

Poniższy przykład przedstawia zastosowanie operacji na zbiorach rozmytych i notacji singletonowej:

**Przykład 2.2.** Niech  $X$  oznacza zbiór studentów pracujących w kursie, dla których informacja  $a$  oznacza procent ćwiczeń wykonanych w kursie, zaś  $b$  oznacza procent odwiedzonych elementów kursu. Natomiast  $A$  oznacza zbiór rozmyty opisujący studentów ze zbioru  $X$ , którzy wykonali niewielką liczbę ćwiczeń w kursie, zaś  $B$  — studentów ze zbioru  $X$ , którzy odwiedzili mało elementów kursu. Do opisu zbiorów  $A$ ,  $B$  oraz  $C$  zostanie wykorzystana notacja singletonowa Zadeha, ponieważ zbiór  $X$  jest dyskretny. Jako funkcje przynależności przyjęto zbiory par, w których dla każdego elementu z uniwersum przypisano konkretną wartość funkcji.

$$X = \{x_1 = (a = 10, b = 40), x_2 = (a = 30, b = 70), x_3 = (a = 70, b = 90), \\ x_4 = (a = 20, b = 30), x_5 = (a = 10, b = 10), x_6 = (a = 15, b = 25)\}$$

$$A = \frac{0,9}{x_1} + \frac{0,5}{x_2} + \frac{0}{x_3} + \frac{0,7}{x_4} + \frac{0,9}{x_5} + \frac{0,8}{x_6}$$

$$B = \frac{0,3}{x_1} + \frac{0}{x_2} + \frac{0}{x_3} + \frac{0,5}{x_4} + \frac{0,9}{x_5} + \frac{0,6}{x_6}$$

Zbiór rozmyty  $C$  oznacza studentów, którzy wykonali niewielką liczbę ćwiczeń w kursie i odwiedzili mało elementów kursu:

$$C = A \cap B = \frac{0,9 \wedge 0,3}{x_1} + \frac{0,5 \wedge 0}{x_2} + \frac{0 \wedge 0}{x_3} + \frac{0,7 \wedge 0,5}{x_4} + \frac{0,9 \wedge 0,9}{x_5} + \frac{0,8 \wedge 0,6}{x_6} =$$

$$= \frac{0,3}{x_1} + \frac{0}{x_2} + \frac{0}{x_3} + \frac{0,5}{x_4} + \frac{0,9}{x_5} + \frac{0,6}{x_6}$$

Zatem studenci, którzy wykonali niewielką liczbę ćwiczeń w kursie i odwiedzili mało elementów kursu, to w kolejności od najwyższego stopnia przynależności osoby:  $x_5, x_6, x_4$  oraz  $x_1$ . Dla osób  $x_2$  oraz  $x_3$  stopień przynależności wynosi zero, zatem tych osób nie uwzględnia się w powyższym zbiorze studentów.

Zaprezentowany przykład prezentuje prosty sposób łączenia ze sobą różnych nieprecyzyjnych pojęć. Daje to możliwość budowania bazy reguł (np. opisujących pewne strategie dydaktyczne oparte na aktywności studenta w kursie) w postaci implikacji, w których poprzednik składa się ze skończonej liczby koniunkcji (lub alternatyw) różnych nieprecyzyjnych danych dydaktycznych.

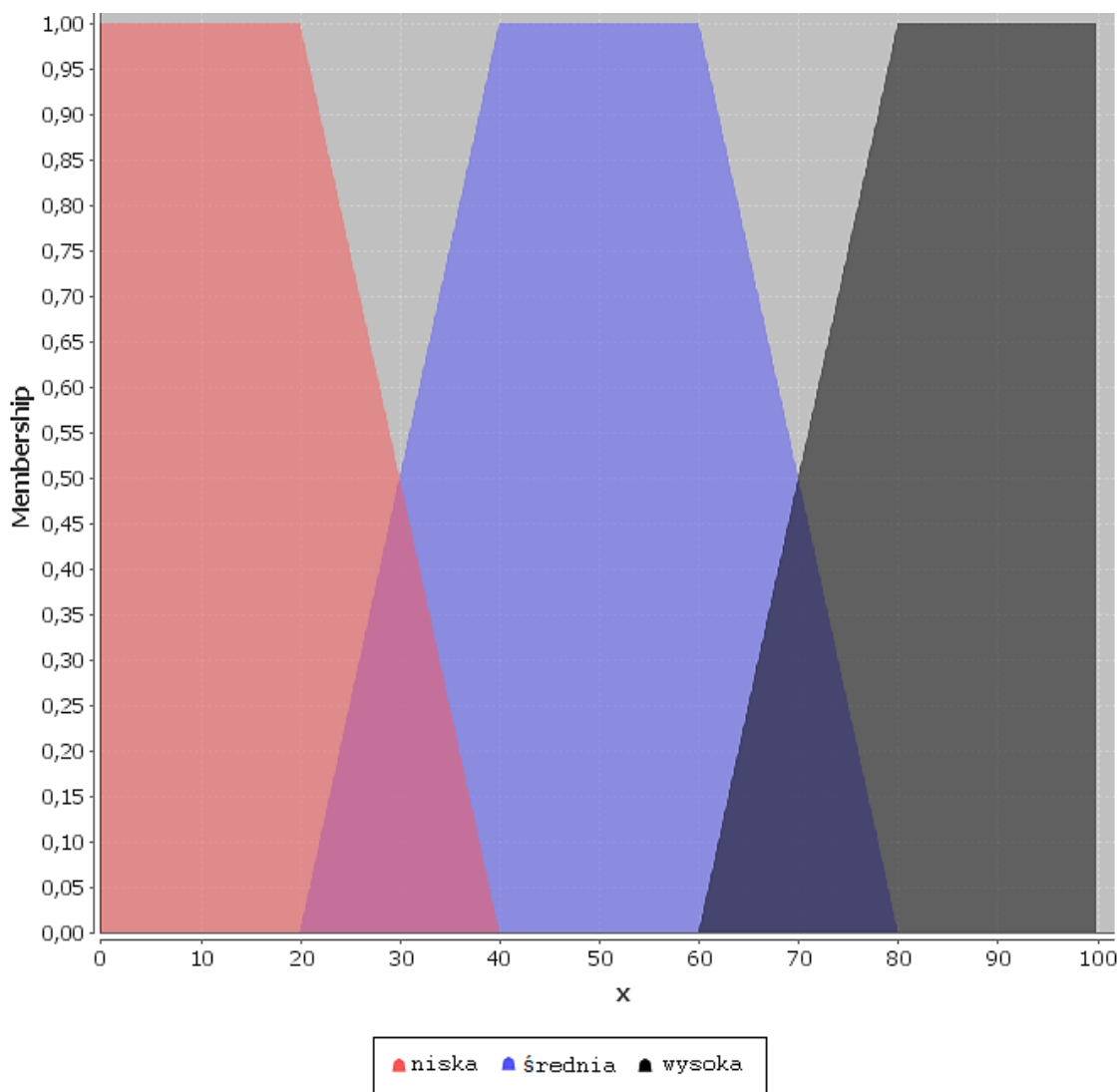
W oparciu o zaprezentowane wyżej definicje możliwe jest modelowanie konkretnego zbioru rozmytego. Na przykład: zbiór ludzi, którzy odwiedzili niewielką liczbę elementów w kursie. Analogicznie można określić zbiór ludzi, którzy odwiedzili dużo elementów, bardzo dużo, itd. Można to uogólnić i zdefiniować pojęcie *liczba odwiedzonych elementów w kursie* z lingwistycznymi wartościami typu *niska, średnia, wysoka*, itp., wykorzystując pojęcie zmiennej lingwistycznej.

**Definicja 2.6.** Zmienna lingwistyczna to zmienna przyjmująca wartości lingwistyczne. Opisana jest za pomocą czwórki:

- **nazwa** — nazwa zmiennej lingwistycznej, np. *czas spędzony w kursie*;
- **uniwersum** — zbiór dopuszczalnych wartości zmiennej lingwistycznej, np. *od 0 do 30 minut*;
- **termy (zbiór wartości)** — wartości lingwistyczne przyjmowane przez zmienną, np. *niski, średni, wysoki*;
- **interpretacje** — zbiory rozmyte definiujące termy.

Poniższy przykład prezentuje zastosowanie powyższej definicji w modelowaniu zmiennych lingwistycznych:

**Przykład 2.3.** Liczbę odwiedzonych elementów kursie (podaną w procentach) można zamodelować zmienną lingwistyczną w uniwersum:  $[0; 100]$ . Na potrzeby przykładu zostaną zaproponowane trzy termy: *niska, średnia, wysoka*. Interpretacje tych zmiennych mogą być definiowane różnymi zbiorami rozmytymi w zależności od kontekstu i potrzeby. Przykładowe interpretacje stanowią zbiory rozmyte o funkcjach przynależności zaprezentowanych na rysunku 2.1.



**Rysunek 2.1.** Interpretacje termów zmiennej: *liczba odwiedzonych elementów w kursie*

Każde nieprecyzyjne pojęcie opisujące jakąś daną dydaktyczną można precyzyjnie zdefiniować wykorzystując zmienne lingwistyczne. Tak określony zbiór pojęć reprezentujących zebrane dane o aktywnościach studentów w kursie oraz baza rozmytych reguł oparta na łączeniu ze sobą różnych nieprecyzyjnych pojęć będzie stanowić narzędzie do budowania strategii dydaktycznych w oparciu o wiedzę ekspercką czyli wiedzę pozyskaną od dydaktyków.

Przetwarzanie i późniejsze wykorzystanie danych dydaktycznych może być trudne, ponieważ część danych jest nieprecyzyjna. Nieprecyzyjność może być zaletą, jeżeli zostaną wykorzystane narzędzia formalne, które bazują na takich danych. Jednym z takich narzędzi są zbiory rozmyte. W rozprawie zostanie zaprezentowany sposób modelowania a następnie wykorzystania nieprecyzyjnych danych dydaktycznych w oparciu o zbiory rozmyte.

## 2.5. ZACHOWANIA STUDENTÓW W KURSACH E-LEARNINGOWYCH

Strategie dydaktyczne w kursach i systemach e-learningowych są budowane na przykład w oparciu o style uczenia się. Natomiast w e-learningu można mówić także o różnych zjawiskach opisujących aktywności studenta w kursie, np. zaangażowanie [14], motywacja [70, 123], poziom uwagi [140] czy efektywność nauki [160]. Dzięki zamodelowaniu takich zjawisk możliwe by było opracowanie strategii dydaktycznej pozwalającej na większą indywidualizację procesu kształcenia. W niniejszej rozprawie zostanie zaprezentowana strategia, w której wykorzystane zostanie zjawisko disengagement, czyli sytuacja, kiedy poziom zaangażowania studenta w kursie spada.

### 2.5.1. Wybrane zachowania studentów w kursach e-learningowych

Od wielu lat prowadzone są badania w obszarze analizy różnych zjawisk występujących w trakcie nauki z wykorzystaniem rozwiązań e-learningowych. Takimi zjawiskami są chociażby satysfakcja z nauki w kursie [145], zaangażowanie [14], motywacja [70, 123], poziom uwagi [140] czy stopień porzucenia kursu [166].

#### Satysfakcja z nauki w kursie

Satysfakcja/zadowolenie studentów jest istotnym wskaźnikiem ich ogólnych doświadczeń i osiągnięć [172]. Istnieje wiele narzędzi do pomiaru zadowolenia studentów w kursach online. Powszechną praktyką w tym zakresie jest używanie kwestionariuszy. Przez lata opracowano i stosowano różne narzędzia do pomiaru satysfakcji studentów, takie jak kwestionariusz doświadczeń w kursie [137] czy krajowe badania studentów [11]. W 2017 roku Antoinette Davis stworzyła narzędzie do pomiaru satysfakcji z nauki online (SOL), które składa się z elementów, takich jak efektywność informacji zwrotnej, wykorzystanie forów dyskusyjnych, dialog instruktor-uczeń, cechy instruktora, poczucie wspólnoty czy komunikacja za pośrednictwem komputera [37]. W literaturze pojawiają się rozwiązania polegające na modelowaniu satysfakcji z nauki w kursie metodami uczenia maszynowego. Wykorzystuje się m.in. drzewa decyzyjne [29, 72] czy sieci neuronowe [6].

#### Motywacja studentów

W ciągu lat motywacja staje się coraz bardziej uznawana za istotny czynnik wpływający na proces uczenia się, zarówno w tradycyjnych, jak i online systemach edukacyjnych. W rezultacie powstały różne skale oceny służące do pomiaru motywacji uczniów i studentów w procesie uczenia się [68, 83]. Te skale motywacji wykorzystują różne podejścia, aby uwzględnić zmienne związane z motywacją i uczeniem się w kontekście różnych środowisk edukacyjnych. Kilka badań wykorzystało te skale do różnych celów, aby ocenić ich użyteczność i wiarygodność w pomiarze motywacji uczniów. Na przykład, badanie opierające się na kwestionariuszu motywacji uczniów do nauki przedmiotów ścisłych i przyrodniczych (SMTSL [169]) wykazało, że poziom motywacji uczniów ma wpływ na ich osiągnięcia i postawę w nauce przedmiotów ścisłych [27].

W badaniach dotyczących motywacji studentów w e-learningu wykorzystywano różnorodne techniki, obejmujące zarówno pomiary jawne, jak i ukryte. Na przykład, Angel De Vicente i Helen Pain zastosowali technikę MOODS (Motivational Diagnosis Study), która wykorzystuje tzw. suwak motywacyjny i kwestionariusz samooceny motywacji studentów [38]. Poprzez monitorowanie stanów motywacji uczniów w różnych fazach interakcji z MOODS, badacze generowali krótkie raporty zwrotne na temat motywacji. McQuiggan i inni modelowali poczucie własnej skuteczności w systemie ITS za pomocą drzew decyzyjnych, tworząc dynamiczny model, który automatycznie aktualizował się na podstawie danych z testów, danych fizjologicznych rejestrowanych za pomocą tzw. biofeedbacku oraz interakcji uczniów w ich środowisku nauki [114]. Wyniki badań sugerują, że ten dynamiczny model lepiej przewidywał poczucie własnej skuteczności uczniów niż statyczny model oparty na zwalidowanych danych. Lei Qu i współpracownicy badali zastosowanie modelu Bayesa, który integrował dane dotyczące skupienia uwagi i interakcji uczniów w systemie ITS w celu wykrywania pewności siebie i dezorientacji [135]. Eksperymenty wykazały, że modele te osiągały dokładność rozpoznawania motywacji przekraczającą 70%. Santos i inni z kolei zastosowali splotowe sieci neuronowe do wykrywania wewnętrznej motywacji na podstawie analizy mimiki uczniów [147]. Badania wykazały, że poziom wewnętrznej motywacji można efektywnie wykryć tylko na podstawie bodźców wizualnych.

### **Zjawisko porzucania kursów przez studentów**

O zjawisku porzucania kursów przez studentów najczęściej mówi się w kontekście kursów typu MOOC [63]. W tradycyjnym nauczaniu online nieukończenie kursu może obniżyć pewność siebie lub samoocenę studenta i zniechęcić go do zapisania się na kolejne kursy [99]. Dla instytucji wysokie wskaźniki rezygnacji oznaczają nieskuteczne programy online i w konsekwencji utratę zysków. Te problemy wskazują na fakt, że administratorzy kursów online borykają się z trudnościami w poprawie wskaźników ukończenia kursów i starają się zidentyfikować przyczyny rezygnacji studentów, aby zminimalizować zjawisko utraty uczestników [99]. Badania pokazują, że ukończenie kursu nie jest celem wielu studentów zapisujących się na kursy typu MOOC. Istnieją różne intencje i motywacje poza ukończeniem kursu czy zdobyciem certyfikatu [23, 71, 173]. Na przykład Khe Foon Hew i Wing Sum Cheung w 2014 zidentyfikowali liczne powody, dla których studenci zapisują się na MOOC, m.in. chęć nauki nowych tematów, poszerzenia wiedzy, prosta ciekawość, podjęcie osobistego wyzwania czy zdobycie wielu certyfikatów ukończenia [71]. Ponieważ porzucanie kursów przez studentów jest częstym zjawiskiem, to naukowcy próbowali różnych metod modelowania tego fenomenu — na przykład w oparciu o sieci neuronowe na podstawie cech indywidualnych studentów (np. wiek, płeć czy narodowość) oraz informacji o aktywnościach w kursach [166].

### **Zaangażowanie studentów**

Zaangażowanie studenta w e-learningu rozumie się w kontekście jego psychologicznego zaangażowania w proces uczenia się, zdobywania wiedzy i budowania krytycznego myślenia [40]. Chociaż istnieją różne konceptualizacje zaangażowania studentów [181], zwoleńnicy analityki edukacyjnej kładą nacisk na analizę logów z platform e-learningowych,



w tym np. kliknięć w zasoby edukacyjne, w kontekście zaangażowania w nauczaniu online [141].

Istnieją różne modele pomiaru zaangażowania uczniów w kontekście uczenia się. W 2012 roku Eitel J. M. Lauría wraz z innymi potwierdzili fakt, że liczba przesłanych zadań, postów na forach i ukończonych quizów online może ilościowo określić zaangażowanie uczącego się w kursach typu MOOC [96]. Badania wykorzystujące statystyki opisowe wykazały, że konsekwencja i wytrwałość w działaniach edukacyjnych są związane z zaangażowaniem uczącego się i jego wynikami [65]. Zaangażowanie uczącego się polega również na badaniu tych działań, które wymagają obecności online na platformie e-learningowej [8]. Te działania online mogą mieć formę uczestnictwa w forach dyskusyjnych, wiki, blogach, wspólnych zadaniach, quizach online, które wymagają pewnego poziomu zaangażowania ze strony ucznia. W 2019 roku Jeongju Lee, Hae-Deok Song i Ah Jeong Hong stwierdzili, że wskaźniki zaangażowania uczniów, takie jak motywacja psychologiczna, współpraca rówieśnicza, poznawcze rozwiązywanie problemów, interakcja z nauczycielami i rówieśnikami, mogą pomóc zwiększyć zaangażowanie uczniów i ostatecznie pomóc nauczycielom w skutecznym projektowaniu programów nauczania [98]. W literaturze pojawiają się rozwiązania polegające na modelowaniu zaangażowania studentów wykorzystujące chociażby drzewa decyzyjne [117], algorytm k-średnich [1] czy nawet konwolucyjne sieci neuronowe [120].

### 2.5.2. Zjawisko disengagement w edukacji

Jednym z coraz częściej opisywanych w literaturze zjawisk jest nie samo zaangażowanie studenta w kursie a proces obniżania się poziomu zaangażowania. Pomysł na badanie tego zjawiska nie jest nowy, co pokazuje wiele badań — nawet jeszcze z pierwszej dekady XXI wieku. Naukowcy od lat zajmują się badaniem zaangażowania i jego braku podczas nauki na odległość [9, 82]. Powstawały systemy, które pozwalały na analizę logów z kursów i w oparciu o takie dane próbowano modelować tzw. **disengagement**, czyli właśnie zjawisko opisujące spadek zaangażowania studenta podczas pracy z kursem [33]. Ze względu na fakt, że trudno jest jednoznacznie przetłumaczyć to pojęcie na język polski — najczęściej tłumaczy się je jako „brak zaangażowania” ale to nie jest to samo co proces obniżania się poziomu zaangażowania — dalej w rozprawie będzie wykorzystywane nazewnictwo angielskie.

W literaturze w kontekście edukacji, pojęcie *disengagement* odnosi się do zjawiska, w którym uczniowie/studenci tracą zainteresowanie, zaangażowanie lub motywację do nauki [167]. Jest to proces, w którym uczniowie stają się bardziej obojętni lub dystansują się od aktywnego uczestnictwa w procesie edukacyjnym [91]. Disengagement może mieć różne przyczyny, takie jak niewłaściwe metody nauczania, brak dostosowania programu nauczania do indywidualnych potrzeb uczniów, nadmierna ilość prac domowych, czy też niewłaściwe relacje społeczne w środowisku szkolnym/akademickim [126]. Skutki tego zjawiska mogą obejmować spadek osiągnięć w nauce, utratę zainteresowania nauką oraz negatywny wpływ na rozwój społeczny i emocjonalny uczniów [30].

W literaturze można znaleźć rozwiązania wykorzystujące różne metody uczenia maszynowego pozwalające na modelowanie zjawiska disengagement — zaczynając od regresji logistycznej a kończąc na sieciach neuronowych [34, 50, 100, 162, 164, 171].

Modelowaniem zjawiska disengagement zajmowali się na przykład Mihaela Cocea i Stephan Weibelzahl. W 2009 roku zaprezentowali rozwiązanie, w którym na podstawie analizy logów z platformy e-learningowej wykrywano studentów, którzy cechowali się występowaniem w ich pracy zjawiska disengagement [34]. Detekcja zjawiska opierała się na danych takich jak liczba przeczytanych stron kursu, czas spędzony w kursie czy liczba rozwiązanych zadań. Dalej na podstawie analizy logów, w 2015 roku T. Gopalakrishnan i P. Sengottuvelan wyuczyli naiwny klasyfikator Bayesa w celu wykrywania zjawiska disengagement [164]. W 2018 roku natomiast Jacqueline Feild wraz z innymi zaproponowała klasyfikator, który w oparciu o informacje o rozwiązanych testach w kursie, przypisywał stopień występowania zjawiska disengagement [50]. Informacjami o testach były przede wszystkim zdobyte punkty, datę rozpoczęcia rozwiązywania, spędzony czas czy liczba podejść. Proponowany klasyfikator bazował na regresji logistycznej i został wyuczony na zbiorze danych, które były zbierane przez cztery lata w różnych kursach e-learningowych typu MOOC. W 2022 roku znowu Manisha Verma wraz z innymi naukowcami zaproponowali wykorzystanie sieci neuronowych w celu wykrywania poziomu zjawiska disengagement [171].

W oparciu o disengagement możliwe jest kategoryzowanie studentów w różnych kontekstach. Na przykład, Nina Bergdahl, Jalal Nouri i Uno Fors w 2019 roku zaprezentowali rezultaty badań, z których wynika, że występują istotne statystycznie różnice w obszarze umiejętności cyfrowych między studentami zaangażowanymi i niezaangażowanymi w naukę [17]. Studenci z wyższym poziomem umiejętności cyfrowych są bardziej zaangażowani w naukę. Co ciekawe w oparciu o informację o zaangażowaniu studenta wg wyników tych badań, można przewidywać końcową ocenę studenta. W 2022 roku Nina Bergdahl poczyniła kolejne badania w obszarze zaangażowania i jego braku w nauczaniu zdalnym [16]. Z analizy ankiet wypełnianych przez różnych dydaktyków wynika, że zaangażowanie i disengagement są zależne od kontekstu. Zjawiska te są odrębnymi konstruktami, co oznacza, że disengagement to nie jest zupełne przeciwieństwo dla zaangażowania — tzn. to nie jest brak zaangażowania. Jest to ważne chociażby w kontekście modelowania tego zjawiska i wykorzystania go dalej np. w problemie adaptacji treści w kursach e-learningowych.

Studenci mogą przyjmować różne strategie pracy z kursem e-learningowym w zależności od tematyki i formy tego kursu. Zjawisko disengagement nie było do tej pory wykorzystywane w kursach z adaptacyjną zawartością, choć były różne próby modelowania tego zjawiska w ramach systemów LMS. W rozprawie zostanie zaprezentowana strategia dydaktyczna oparta właśnie na tym zjawisku. Ponieważ pojęcie disengagement jest nieprecyzyjne, zostanie zamodelowane przy pomocy sterowników rozmytych bazujących na teorii zbiorów rozmytych [102, 180].

## 2.6. STEROWANIE ROZMYTE W PROBLEMIE KLASYFIKACJI

W zaprezentowanych powyżej badaniach zjawisko disengagement było modelowane metodami uczenia maszynowego, jednak ze względu na nieprecyzyjny charakter tego

zjawiska w niniejszej rozprawie zostanie ono definiowane w oparciu o sterownik rozmyty oparty na zbiorze reguł eksperckich. W literaturze takie podejście nie zostało opisane. Reguły pozyskuje się od ekspertów, czyli w tym wypadku doświadczonych dydaktyków. W ramach implementacji strategii dydaktycznej opartej na tym zjawisku, można zbudować bazę reguł opisujących sposób adaptacji w zależności od stopnia występowania zjawiska disengagement. Na przykład disengagement danego studenta mógłby być opisywany jako *niski*, *średni* czy *wysoki*. Eksperti mogą zdefiniować reguły w postaci prostych implikacji, w dodatku bez używania konkretnych liczb. Na przykład: *jeśli student mało pracuje w kursie, to jego disengagement jest wysoki*. Tak określone reguły są zrozumiałe dla człowieka, ponieważ opierają się na nieprecyzyjnej naturze języka. Dzięki temu nie tylko informatycy ale i eksperci w dziedzinie (w tym wypadku — dydaktycy) mogą zamodelować zjawisko disengagement. Metodą, która pozwala w prosty sposób wykorzystać takie eksperckie reguły jest sterowanie rozmyte [102] bazujące na zbiorach rozmytych [180], które można wykorzystać do opisu danych dydaktycznych, które są wykorzystywane w regułach eksperckich.

### 2.6.1. Wykorzystanie sterowania rozmytego w problemie klasyfikacji

Sterowanie rozmyte jest metodą, która umożliwia rozwiązywanie problemów, w których charakterystyki są trudne do opisanego w sposób precyzyjny (zatem nie jest możliwe zbudowanie zwykłego systemu regułowego) [2]. Sterowanie rozmyte wykorzystuje koncepcję zbiorów rozmytych i reguł, co pozwala na bardziej elastyczne i przybliżone modelowanie wielu zjawisk. Zamiast używać wartości binarnych (*prawda/falsz*), jak w tradycyjnym sterowaniu, sterowanie rozmyte korzysta z wartości nieprecyzyjnych (np. *bardzo niski*, *niski*, *średni*, *wysoki*, *bardzo wysoki*). Kluczowym elementem sterowania rozmytego jest zastosowanie reguł rozmytych, które opisują relacje między wejściami a wyjściami. Te reguły są zazwyczaj wyrażane w formie implikacji postaci „*jeśli... to...*” i opisują, jak należy dostosować wyjścia systemu w zależności od wartości wejściowych. Proces podejmowania decyzji w sterowaniu rozmytym jest bardziej elastyczny i odporny na nieliniowości oraz zmienność warunków [80].

Sterowanie rozmyte jest szeroko stosowane w różnych dziedzinach, takich jak automatyka przemysłowa, systemy kontroli ruchu, robotyka, systemy grzewcze i klimatyzacyjne, oraz wiele innych obszarów, gdzie konieczne jest skuteczne sterowanie w warunkach niepewności i zmienności [90].

W problemie klasyfikacji sterownik rozmyty może zostać potraktowany jako algorytm, który będzie obliczał wartość funkcji decyzyjnej w klasyfikatorze, analogicznie jak np. regresja logistyczna. W literaturze pojawiają się różne zastosowania sterowników rozmytych w problemie klasyfikacji [61, 85, 92, 128, 161, 183]. Jednym z przykładów jest system z 2021 roku zaproponowany przez Arasha Geramiana i Ajitha Abrahama [61]. Autorzy opracowali sterownik rozmyty, który został wykorzystany do klasyfikacji klientów na różnych rynkach. Natomiast rok wcześniej Yori Kurniasari wraz z innymi naukowcami opracował klasyfikator bazujący na sterowaniu rozmytym w celu klasyfikacji różnych miejsc w Indonezji pod względem poziomu bezrobocia [92]. W innym zastosowaniu zaproponowano natomiast rozwiązanie bazujące na sterowaniu rozmytym w problemie wykrywania raka jajnika [161, 183]. Zaś w 2022 roku Tomasz Piłka

wraz z innymi, zaproponował rozwiązanie wykorzystujące ekspercki sterownik rozmyte w procesie oceniania ryzyka kontuzji piłkarzy [128]. Sterowanie rozmyte było już także wykorzystywane w kontekście edukacji. Na przykład w 2020 roku Akrom Khomeiny wraz z innymi osobami zaproponował system oparty o sterowanie rozmyte, który wspomagał nauczycieli w procesie wystawiania ocen [85].

### 2.6.2. Sterowniki rozmyte

Metoda sterowania rozmytego polega na budowaniu sterowników rozmytych. W niniejszej rozprawie będą wykorzystywane dwa rodzaje sterowników rozmytych: sterowniki Mamdaniego [102] oraz sterowniki Takagi-Sugeno [165]. Na sterownik rozmyty składają się cztery elementy:

- blok rozmywania (fuzyfikacja),
- baza reguł,
- blok wnioskowania (w oparciu o bazę reguł),
- blok wyostrzania (defuzyfikacja).

#### **Blok rozmywania**

Na wejściu sterownik przyjmuje wartości liczbowe, które są rozmywane w bloku rozmywania w oparciu o zdefiniowane zmienne lingwistyczne w tym bloku. W przypadku sterowników Mamdaniego w tym bloku definiuje się także zmienne lingwistyczne modelujące wyjście sterownika. Poniższy przykład prezentuje działanie bloku rozmywania:

**Przykład 2.4.** Niech sterownik rozmyty przyjmuje na wejściu dwie zmienne lingwistyczne: liczba odwiedzonych elementów w kursie oraz czas spędzony w kursie. Dla każdej zmiennej zdefiniowano trzy terminy: low, medium, high, których interpretacje stanowią zbiory rozmyte o funkcjach przynależności zaprezentowanych na rysunku 2.2 (zob. następna strona).

Do wejścia sterownika zostały wysłane sygnały o znormalizowanych wartościach: liczba odwiedzonych elementów w kursie wynosząca 0,30 oraz czas spędzony w kursie wynoszący 0,80. W ramach procesu rozmywania, dla każdego terminu należy określić stopień przynależności tych sygnałów:

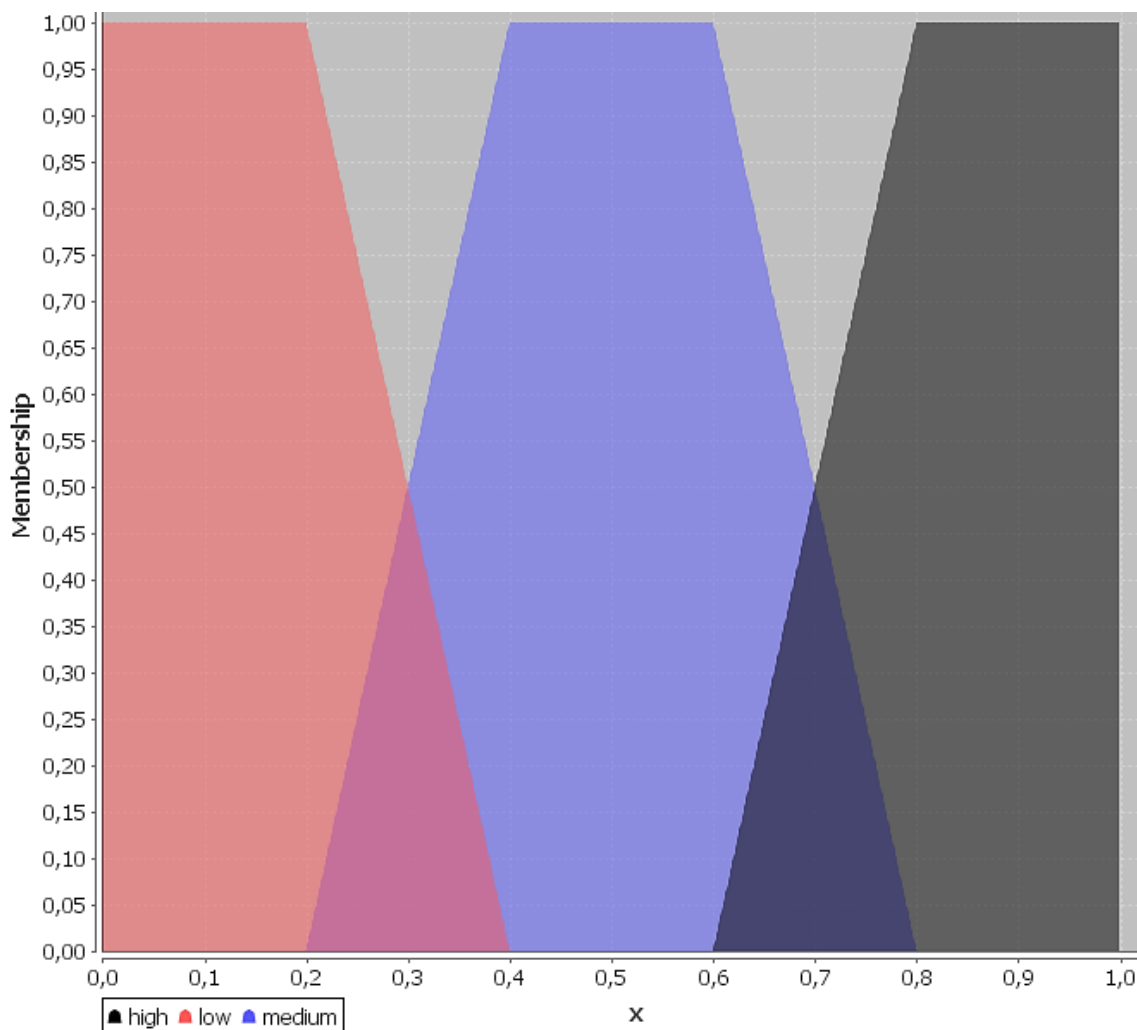
#### **Liczba odwiedzonych elementów w kursie:**

- $\text{low}(0,30) = 0,50$
- $\text{medium}(0,30) = 0,50$
- $\text{high}(0,30) = 0,00$

#### **Czas spędzony w kursie:**

- $\text{low}(0,80) = 0,00$

- $\text{medium}(0,80) = 0,00$
- $\text{high}(0,80) = 1,00$



**Rysunek 2.2.** Model zbioru wartości obu zmiennych wejściowych definiowanych w przykładowym sterowniku

### Baza reguł i blok wnioskowania

Kolejnym elementem tworzącym sterownik rozmyty są baza reguł i oparty o nią blok wnioskowania. Reguły w sterownikach rozmytych mają postać implikacji *IF...THEN...* — w przypadku sterowników Mamdaniego reguły mają postać *jeśli x jest A, to y jest B*, gdzie  $x, y$  to wartości sygnałów wejściowych modelowanych pojęć, natomiast  $A$  i  $B$ , to wartości lingwistyczne (termy) modelowane przy pomocy zbiorów rozmytych [102].

Natomiast w przypadku sterowników Takagi-Sugeno reguły mają postać *jeśli x jest A, to y = f(x)*, gdzie  $x, y$  to wartości sygnałów wejściowych modelowanych pojęć,  $A$  to wartość lingwistyczna modelowana przy pomocy zbioru rozmytego, zaś  $f$  to pewna funkcja zależna od wartości sygnałów wejściowych sterownika [165].

Reguły mogą być pozyskane od ekspertów lub wygenerowane automatycznie na podstawie wcześniej zebranych danych. W tym drugim przypadku cały sterownik jest

generowany w oparciu o dane, natomiast w odróżnieniu od metod uczenia maszynowego po wygenerowaniu sterownika, dostępny jest wgląd w zamodelowane zmienne oraz zdefiniowane reguły.

W przypadku automatycznego generowania sterowników często korzysta się z algorytmu klastrowania **Fuzzy C-Means (FCM)** [18]. W tym wypadku klastrami będą termy będące wartościami lingwistycznymi, jakie mogą przyjmować zmienne wykorzystywane w sterowniku. Liczbę klastrów można zdefiniować samodzielnie lub wyznaczyć ją np. przy pomocy metody różnicowej (ang. *subtractive clustering*) [31]. Często nie wiadomo, ile powinno być klastrów, więc ich liczbę wyznacza się automatycznie.

Algorytm Fuzzy C-Means jest popularną metodą grupowania danych stosowaną w analizie danych i uczeniu maszynowym. Jest to wariant klasycznego algorytmu k-średnich, ale zamiast przypisywać każdą próbkę danych do jednego klastra, algorytm FCM przypisuje każdą próbkę do każdego klastra z pewnym stopniem przynależności. Oznacza to, że dana próbka może należeć do więcej niż jednego klastra z różnym stopniem przynależności, co sprawia, że algorytm jest bardziej elastyczny i lepiej radzi sobie z danymi o niejasnych granicach między klastrami. Schemat działania algorytmu jest następujący:

#### 1. Inicjalizacja:

- Wybierz liczbę klastrów  $c$ .
- Zainicjuj macierz przynależności  $U$  losowymi wartościami tak, aby suma przynależności dla każdej próbki wynosiła 1.

#### 2. Obliczanie centroidów:

- Oblicz centroidy klastrów na podstawie obecnej macierzy przynależności  $U$ . Centroid dla klastra  $j$  oblicza się według wzoru:

$$v_j = \frac{\sum_{i=1}^n u_{ij}^m \cdot x_i}{\sum_{i=1}^n u_{ij}^m}$$

gdzie:

- $v_j$  to centroid klastra  $j$ ,
- $u_{ij}$  to stopień przynależności próbki  $i$  do klastra  $j$ ,
- $x_i$  to próbka danych,
- $m$  to parametr fuzyfikacji ( $m > 1$ ).

#### 3. Aktualizacja macierzy przynależności:

- Zaktualizuj macierz przynależności  $U$  na podstawie obliczonych centroidów. Przynależność próbki  $i$  do klastra  $j$  oblicza się według wzoru:

$$u_{ij} = \frac{1}{\sum_{k=1}^c \left( \frac{\|x_i - v_j\|}{\|x_i - v_k\|} \right)^{\frac{2}{m-1}}}$$

gdzie:

—  $\|x_i - v_j\|$  to odległość między próbką  $x_i$  a centroidem  $v_j$ .

#### 4. Sprawdzenie warunku stopu:

- Sprawdź, czy różnica między nową i starą macierzą przynależności jest mniejsza od zadanego progu  $\epsilon$  (np. normy macierzy  $U^{(k+1)} - U^{(k)}$ ). Jeśli tak, zakończ algorytm. Jeśli nie, wróć do kroku 2.

Poza określeniem klastrów, czyli w kontekście sterowników rozmytych termów, należy określić ich interpretacje, czyli zbiory rozmyte. Funkcje przynależności tych zbiorów definiowane są różnie, najczęściej jednak wykorzystywana jest krzywa Gaussa, gdzie jako średnią przyjmuje się centroid danego klastra. Natomiast na bazę reguł składa się tyle reguł, ile jest klastrów — po jednej regule dla każdego klastra a reguły są postaci:

*IF input\_1 IS cluster\_K AND input\_2 IS cluster\_K AND ... AND input\_N IS cluster\_K THEN  
output IS cluster\_K*

Algorytm FCM jest na przykład zaimplementowany w narzędziu Fuzzy Logic Designer [112] dostępnym w środowisku MATLAB [111]. Narzędzie to pozwala na generowanie sterowników Mamdaniego i Takagi-Sugeno. W przypadku tych drugich, istnieje natomiast inne narzędzie, które również pozwala na ich generowanie, jednak dodatkowo umożliwia automatyczny wybór najbardziej odpowiednich zmiennych wejściowych, które zostaną uwzględnione w wynikowym sterowniku. Narzędziem tym jest biblioteka pyFume języka Python [57]. Wynikiem działania narzędzia jest sterownik rozmyty Takagi-Sugeno zapisany w języku Python z wykorzystaniem biblioteki Simpsful pozwalającej na modelowanie zbiorów rozmytych i wnioskowanie w logice rozmytej [156].

W momencie, gdy baza reguł jest już zdefiniowana możliwe jest dokonanie procesu wnioskowania w oparciu o wcześniej rozmyte sygnały wejściowe sterownika. Przykłady 2.5 i 2.6 prezentują sposób działania bloku wnioskowania w sterownikach Mamdaniego i Takagi-Sugeno.

**Przykład 2.5.** Niech baza reguł sterownika Mamdaniego przyjmuje dwie reguły:

1. IF liczba\_odwiedzonych\_elementow IS low AND czas\_w\_kursie IS medium THEN  
wyjscie\_sterownika IS medium
2. IF liczba\_odwiedzonych\_elementow IS medium AND czas\_w\_kursie IS high THEN  
wyjscie\_sterownika IS high

W bloku rozmywania zdefiniowano pojęcie wyjście\_sterownika z trzema termami: low, medium, high, których interpretacje stanowią zbiory rozmyte o funkcjach przynależności zaprezentowanych na wcześniej przedstawionym rysunku 2.2.

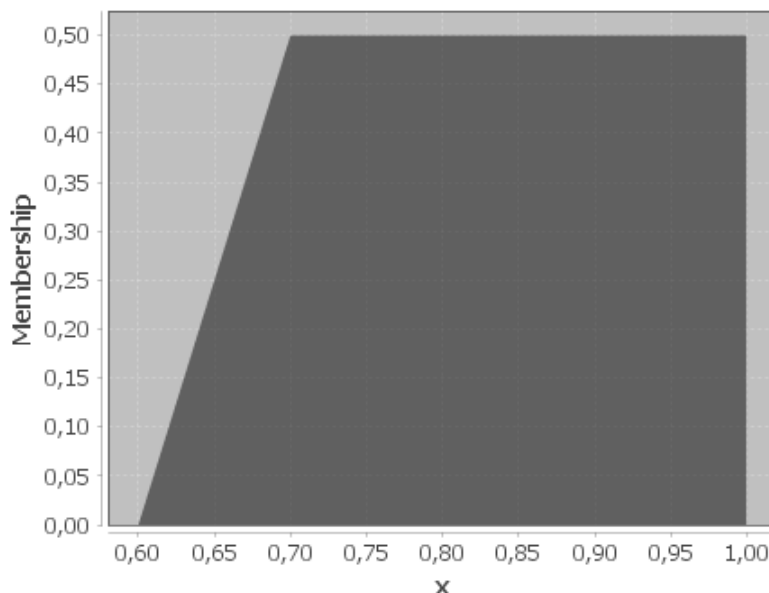
W poprzednikach reguł obie zmienne wejściowe są połączone ze sobą koniunkcją (mogą być też połączone alternatywą). Najczęściej, w sterownikach rozmytych koniunkcję realizuje się poprzez operację minimum (analogicznie alternatywę przez operację maksimum). Wynikiem każdej reguły jest zbiór rozmyty będący iloczynem zbioru określonego w następniku reguły oraz zbioru, którego funkcja przynależności zawsze przyjmuje wartość obliczoną w poprzedniku reguły. Następnie zbiory rozmyte otrzymane we wszystkich regułach są sumowane ze sobą i w wyniku otrzymywany jest jeden zbiór rozmyty, który zostanie wyostrzony w procesie defuzyfikacji.

Dla powyższych reguł oraz wartości z poprzedniego przykładu działanie sterownika będzie następujące:

**Reguła 1.**  $0,50 \wedge 0,00 = 0,00$

**Reguła 2.**  $0,50 \wedge 0,50 = 0,50$

Niezerowy stopień spełnienia poprzednika reguły ma tylko druga reguła, zatem tylko ją należy dalej rozpatrywać (ponieważ dla pierwszej reguły wynikiem będzie zbiór rozmyty, do którego nie należy żaden element — stopień przynależności zawsze jest zerowy). Wynikiem działania bloku wnioskowania będzie zatem zbiór będący iloczynem zbioru, w którym stopień przynależności zawsze wynosi 0,50 oraz zbioru będącego interpretacją pojęcia high zmiennej wyjście\_sterownika (zob. rysunek 2.3).



**Rysunek 2.3.** Wynik działania bloku wnioskowania w przykładowym sterowniku Mamdaniego



**Przykład 2.6.** Niech baza reguł sterownika Takagi-Sugeno przyjmuje dwie reguły:

1. IF liczba\_odwiedzonych\_elementow IS low AND czas\_w\_kursie IS medium THEN  
 $y = 2 \cdot \text{liczba\_odwiedzonych\_elementow} + \text{czas\_w\_kursie}$
2. IF liczba\_odwiedzonych\_elementow IS medium AND czas\_w\_kursie IS low THEN  
 $y = 3 \cdot \text{liczba\_odwiedzonych\_elementow} - \text{czas\_w\_kursie}$

Sterownik przyjmuje na wejściu zmienne oraz sygnały wykorzystywane w poprzednich przykładach. Poprzednik każdej reguły obliczany jest tak samo jak w przypadku sterowników Mamdaniego, zatem na podstawie poprzedniego przykładu stopień spełnienia poprzednika pierwszej reguły wynosi 0,00 a drugiej 0,50. W następniku każdej reguły należy obliczyć wartość funkcji  $y$  w oparciu o sygnały wejściowe sterownika:

**Reguła 1.**  $y = 2 \cdot 0,30 + 0,80 = 1,40$

**Reguła 2.**  $y = 3 \cdot 0,30 - 0,80 = 0,10$

Na tym proces wnioskowania się kończy. Obliczone poprzedniki i następniki reguł zostaną wykorzystane w bloku wyostrzania.

### Blok wyostrzania

Wynik działania bloku wnioskowania na końcu jest wyostrzony. W przypadku sterowników Mamdaniego wynikiem działania bloku wnioskowania jest zbiór rozmyty. Najczęściej do wyostrzania zbioru rozmytego wykorzystuje się metodę środka ciężkości (COG). W przypadku sterowników Takagi-Sugeno oblicza się średnią ważoną następników reguł z wagami odpowiadającymi poprzednikom reguł. Poniższy przykład prezentuje wartości obliczone przez sterowniki rozpatrywane w poprzednich przykładach tego podrozdziału:

**Przykład 2.7.** Kontynuując rozważanie sterowników z poprzednich przykładów, po wyostrzeniu metodą COG zbioru rozmytego będącego wynikiem działania bloku wnioskowania w sterowniku Mamdaniego zostanie zwrócona wartość wynosząca w przybliżeniu 0,82. Natomiast dla sterownika Takagi-Sugeno po obliczeniu średniej ważonej zostanie zwrócona wartość 0,10.

W niniejszej rozprawie zostanie opracowany sterownik Mamdaniego bazujący na regułach eksperckich a także sterowniki Mamdaniego i Takagi-Sugeno wygenerowane automatycznie na pozyskanych danych pozyskanych podczas pracy studentów z kursem. W przypadku sterownika eksperckiego nie zostanie opracowany sterownik Takagi-Sugeno ze względu na mniej zrozumiałą dla eksperta postać reguł (dokładnie ich następników).

### 2.6.3. Ewaluacja algorytmów klasyfikujących dane

W niniejszej rozprawie zostanie dokonane porównanie różnych metod uczenia maszynowego ze sterownikami rozmytymi w problemie klasyfikacji. W celu ewaluacji wszystkich metod zostaną wyznaczone wartości następujących metryk powszechnie wykorzystywanych przy ewaluacji algorytmów klasyfikujących:

- dokładność,
- precyzja,
- czułość,
- miara F1.

Przy ocenie efektywności klasyfikatorów przyjmuje się, że jeśli obiekt należy do danej klasy, to tę klasę nazywa się **pozytywną** a pozostałe klasy klasami **negatywnymi**. W celu zdefiniowania wcześniej wymienionych metryk wprowadza się następujące oznaczenia:

- liczbę obiektów przypisanych do klasy pozytywnej w zbiorze danych oczekiwanych oznacza się jako  $P$  (*positive*),
- liczbę obiektów przypisanych do klasy negatywnej w zbiorze danych oczekiwanych oznacza się jako  $N$  (*negative*),
- liczbę dobrze przypisanych obiektów do klasy pozytywnej oznacza się jako  $TP$  (*true positive*),
- liczbę dobrze przypisanych obiektów do klasy negatywnej oznacza się jako  $TN$  (*true negative*),
- liczbę źle przypisanych obiektów do klasy pozytywnej oznacza się jako  $FP$  (*false positive*),
- liczbę źle przypisanych obiektów do klasy negatywnej oznacza się jako  $FN$  (*false negative*).

W oparciu o powyższe oznaczenia wykorzystane metryki definiuje się następująco:

**Definicja 2.7. Dokładność** (ang. *accuracy*) oznacza stosunek liczby prawidłowych przypisań do wszystkich klas do liczby wszystkich przypadków:

$$Accuracy = \frac{TP + TN}{P + N}$$

**Definicja 2.8. Precyzja** (ang. *precision*) oznacza stosunek liczby prawidłowych przypisań do klasy pozytywnej do liczby wszystkich przypisań do tej klasy:

$$Precision = \frac{TP}{TP + FP}$$

**Definicja 2.9.** **Czułość** (ang. *recall*) oznacza stosunek liczby prawidłowych przypisań do klasy pozytywnej do liczby wszystkich przypisań do tej klasy w zbiorze danych oczekiwanych:

$$Recall = \frac{TP}{P}$$

**Definicja 2.10.** **Miara F1** (ang. *F-score*, *F-measure*) jest średnią harmoniczną precyzji i czułości:

$$F1 = \frac{2}{Precision^{-1} + Recall^{-1}} = 2 \cdot \frac{Precision \cdot Recall}{Precision + Recall}$$

Precyzję, czułość oraz miarę F1 dla całego zbioru danych w skali *macro* oblicza się średnią arytmetyczną oraz średnią ważoną liczbą przypadków dla wszystkich klas.

W oparciu o powyższe definicje, w ramach porównania algorytmów klasyfikujących zostaną wykorzystane wymienione wcześniej metryki: dokładność, precyzja, czułość i miara F1.

Zaletą sterowania rozmytego jest to, że wykorzystuje się dane nieprecyzyjne a także to, że opiera się ono na eksperckiej bazie rozmytych reguł i nie jest konieczne posiadanie wcześniej zebranych i przetworzonych danych do trenowania klasyfikatora. Takie reguły pozyskuje się od ekspertów w danej dziedzinie, którzy nie muszą posiadać technicznych i programistycznych umiejętności. Wykorzystują język naturalny, definiując reguły w postaci prostych implikacji w oparciu o nieprecyzyjne pojęcia. Sama nieprecyzyjność jest jedną z charakterystycznych cech języka ludzkiego. Sterowanie rozmyte może być wykorzystywane na przykład w problemie klasyfikacji. W dalszej części pracy zostanie zaprezentowany ekspercki sterownik rozmyty modelujący zjawisko disengagement, które będzie dalej wykorzystywane w realizacji założonej strategii dydaktycznej. Następnie zostaną zaprezentowane sterowniki wygenerowane automatycznie w oparciu o pozyskane dane a także przeprowadzone porównanie sterowników rozmytych z popularnymi metodami uczenia maszynowego wykorzystywanymi w problemie klasyfikacji.



## ROZDZIAŁ 3

# Strategia adaptacji z wykorzystaniem sterowania rozmytego

W dotychczasowych podejściach przy budowie kursów e-learningowych z adaptacyjną zawartością pojawiały się rozwiązania wykorzystujące różne strategie adaptacji jednak najczęściej wykorzystujące modele opisujące style uczenia się, takie jak model Feldera-Silverman [51] czy model VARK [54] (patrz rozdziały 2.2 i 2.3). W podejściu proponowanym w rozprawie zostanie zamodelowane zjawisko disengagement w oparciu o wybrane nieprecyzyjne dane o postępach studenta w kursie. W innych pracach zjawisko to było modelowane metodami uczenia maszynowego, natomiast w proponowanym rozwiązaniu wykorzystany zostanie ekspercki sterownik rozmyty Mamdaniego ze względu na nieprecyzyjny charakter modelowanego zjawiska (zob. rozdział 2.6.2). W oparciu o poziom występowania zjawiska disengagement u danego studenta, zostanie zaprezentowany algorytm adaptacji wykorzystujący zaproponowaną ogólną architekturę adaptacyjnego kursu e-learningowego. W rozdziale zostanie także przedstawiony proces implementacji kursu z modułem adaptacji treści oraz narzędzia, które trzeba było przygotować, aby taka implementacja była możliwa.

### 3.1. PRZYGOTOWANIE DO BADAŃ

Aby możliwa była realizacja prezentowanych w rozprawie badań, niezbędne było opracowanie kursu e-learningowego, w którym została później wdrożona strategia dydaktyczna. Konieczna była także implementacja narzędzi, które musiały wpisać się w istniejące środowisko techniczne pozwalających na zbieranie danych a także wykorzystanie sterowników rozmytych w kursie. Dane zebrane podczas pracy studentów w kursie musiały zostać poddane przetworzeniu i normalizacji, więc niezbędne było także przygotowanie narzędzia umożliwiającego realizację tych zadań.

#### 3.1.1. Kurs e-learningowy „Zastosowania zbiorów rozmytych”

Do realizacji badań został wykorzystany autorski kurs „Zastosowania zbiorów rozmytych” [93], który następnie uzupełniono o moduł adaptacyjności. Celem kursu jest zapoznanie studentów z elementami zbiorów i logiki rozmytej, zastosowaniami w kon-

tekście nieprecyzyjności informacji a także z narzędziami, jakie oferują zbiory rozmyte do rozwiązywania wybranych problemów sztucznej inteligencji. W trakcie pracy z kursem studenci zapoznają się z matematycznymi podstawami teorii zbiorów rozmytych oraz elementami logiki rozmytej. Dalej studenci uczą się rozwiązywania problemów takich jak rozmyte zapytania do baz danych, podejmowanie decyzji czy rekomendowanie obiektów na podstawie preferencji użytkownika.

Kurs składa się z siedmiu części, które są przygotowane w formacie SCORM 2004 [153], przy użyciu logicznego podziału zrealizowanego w ramach taksonomii UCTS [104]. Standard SCORM pozwala na konstrukcję materiałów opartych na hierarchicznej strukturze, w której SCO (ang. *Sharable Content Object*) stanowią liście. Technicznie rzecz biorąc, SCO to plik HTML z dowolną liczbą zasobów (mediów, plików). W niektórych badaniach obiekty SCO są interpretowane jako *learning objects* [13]. Układ treści kursu w standardzie SCORM zawsze zależy od jego autora. Oznacza to, że zasoby o tej samej wielkości i porównywalnym czasie zaangażowania studentów mogą mieć różną strukturę. Mogą być podzielone na kilka większych części lub kilkadziesiąt mniejszych obiektów SCO. Logiczny podział tych treści, bez zagłębiania się w ich postać techniczną, jest zapewniany przez wspomnianą wcześniej taksonomię UCTS (ang. *Universal Curricular Taxonomy System*). UCTS umożliwia podział zasobów na cztery poziomy:

- UCTS curriculum (program nauczania),
- UCTS module (moduł programu),
- UCTS unit (rozdział modułu),
- jednostki atomowe takie jak:
  - learning objects (pojedyncze obiekty SCO z treścią),
  - assessment (testy),
  - exercise (ćwiczenia),
  - self-assessment (testy pozwalające na weryfikację postępów przez samego studenta, mogą być zazwyczaj wielokrotnie rozwiązywane),
  - references (literatura).

Dzięki takiemu podziałowi nie trzeba odnosić się do samego procesu edukacyjnego (tzn. nie jest potrzebne używanie określeń takich jak *lekcja*, *wykład* czy *semestr*) a same treści mogą być wykorzystywane niezależnie od przeznaczenia danych zasobów edukacyjnych.

Podział logiczny kursu „Zastosowania zbiorów rozmytych” zgodnie z taksonomią UCTS przedstawia tabela 3.1. Zgodnie z podziałem prezentowanym w tabeli, na kurs składało się siedem rozdziałów (UCTS Unit), które zgrupowano w trzy moduły (UCTS Module). Każdy rozdział składał się od czterech do siedmiu obiektów SCO, wśród których były zawarte treści do nauki wraz z interaktywnymi ćwiczeniami a także testy zaliczeniowe.

Zaproponowany podział pozwala na osiągnięcie poziomu granularności, w którym każdy komponent stanowi logicznie zamkniętą całość. Dodatkowo, wprowadzenie mo-

dułów pozwala traktować kurs jak wykład, gdzie jeden moduł odpowiada 1,5 godziny tradycyjnego wykładu, wraz z określoną ilością pracy własnej studenta. Zdefiniowanie rozdziałów umożliwia dostarczanie studentom mniejszych porcji materiału, co ułatwia spełnienie wymogu systematycznej pracy. Taka struktura kursu pozwala na wygodne zarządzanie treściami oraz proste jej wyszukiwanie [107]. W pracy wykorzystano takie podejście, gdyż kurs jest wykorzystany w realnym zastosowaniu, ponieważ prowadzone badania są wykorzystane w procesie dydaktycznym.

Część kursu	Liczba SCO	Poziom agregacji
Podstawy wnioskowania przybliżonego		UCTS Module
Wprowadzenie do teorii zbiorów rozmytych	7	UCTS Unit
Elementy wnioskowania przybliżonego	6	UCTS Unit
Modelowanie danych nieprecyzyjnych		UCTS Module
Liczby rozmyte	7	UCTS Unit
Zmienne lingwistyczne	5	UCTS Unit
Zastosowania zbiorów rozmytych		UCTS Module
Operatory agregacji	4	UCTS Unit
Model Bellmana-Zadeha	5	UCTS Unit
Systemy rekomendacyjne	5	UCTS Unit

**Tabela 3.1.** Struktura kursu z podziałem logicznym wg UCTS

Każdy rozdział został przygotowany wcześniej w narzędziu Eduexe [44]. Fragmenty kursu zbudowanego w tym narzędziu przedstawiają rysunki 3.1 oraz 3.2.

Kurs „Zastosowania zbiorów rozmytych” był już wykorzystywany przed rozpoczęciem badań prezentowanych w tej rozprawie w ramach zajęć ze sztucznej inteligencji dla studentów informatyki. Pierwsza edycja miała miejsce w momencie rozpoczęcia pandemii COVID-19 w roku akademickim 2019/2020, kiedy zajęcia musiały być prowadzone zdalnie. Kurs był wykorzystywany w każdym kolejnym roku akademickim. W ramach badań poprzedzających badania prezentowane w niniejszej rozprawie były wdrażane strategie dydaktyczne, które przewidywały, że studenci będą pracować według różnych harmonogramów określających terminy, w których należało zdawać testy zaliczeniowe po każdej części kursu a także terminy określające daty otwarcia dostępu do kolejnych części [107]. Doświadczenia z tych badań pozwoliły na lepsze zrozumienie danych, jakie można zbierać na temat postępów studentów podczas pracy z kursem a także na analizę, jakie narzędzia są potrzebne, aby możliwe było realizowanie bardziej złożonych strategii w celu adaptacji treści w kursie. Doświadczenia te stanowiły podstawę do prac przeprowadzonych na potrzeby badań opisanych w tej rozprawie. W dalszej części pracy (patrz rozdział 4.1.1) zaprezentowane zostanie, w jaki sposób dane były zbierane i wykorzystane do budowy sterowników rozmytych.

Wskaż wypowiedzi, których interpretacja wymaga rozmytości:

Urodziłem się 24.02.1995r.

Kilka dni temu widziałem się z moim kumplem.

Mam już prawie 25 lat.

Idziemy do kina za pół godziny.

Przymierz trochę mniejsze spodnie.

Trochę historii...

Szukając informacji o logice rozmytej na pewno trafisz na nazwisko *Zadeh*. Uważa się, że to właśnie **Lotfi A. Zadeh** jest twórcą logiki rozmytej i teorii zbiorów rozmytych (o których powiemy sobie później). Właśnie on w 1965 roku opublikował artykuł pt. "Fuzzy Sets".

Prawdę powiedziawszy, o logice rozmytej myślano już wcześniej. Już **Platon** próbował wyjść poza logikę dwuwartościową, mówiąc że istnieje pewna dodatkowa przestrzeń między prawdą i fałszem.

Na początku XX wieku, **Jan Łukasiewicz** - polski uczony - zaproponował system logiki z trzema wartościami, co stanowiło bazę dla logiki rozmytej.

Zastosowania

Zastanawiasz się pewnie, na co tak w sumie potrzebna jest Ci wiedza o logice rozmytej. Do tej pory żyłeś tylko z prawdą i fałszem i było dobrze... No właśnie nie całkiem ;)

Systemy, w których wykorzystuje się pojęcie rozmytości pozwalają opisywać zjawiska o charakterze wieloznacznym, których nie opiszemy korzystając z logiki klasycznej. Systemy rozmyte znajdują zastosowanie wszędzie tam, gdzie nie mamy odpowiedniej wiedzy o modelu matematycznym, który ujmuje dane zjawisko lub tam, gdzie odtworzenie takiego modelu jest nieopłacalne czy po prostu niemożliwe. Zatem znajdziemy je w bazach danych, sterowaniu czy przy przetwarzaniu języka naturalnego. Logika rozmyta przydaje się w elektronice, systemach sterowania, medycynie czy w przemyśle.

Sprawdź się!

Logika dwuwartościowa jest szczególnym przypadkiem logiki...

Rysunek 3.1. Fragment materiałów do nauki w kursie „Zastosowania zbiorów rozmytych”



## Test

Test: Podstawy systemów rekomendacyjnych

Dane są trzy atrybuty: **a1**, **a2**, **a3** oraz obiekty, które spełniają te atrybuty odpowiednio:

**o1**: 0.4, 0.7, 0.5

**o2**: 0.8, 0.5, 0.6

**o3**: 0, 0.9, 0.6

Preferencje użytkownika **U** są następujące: 0.5, 0.5, 0.5.

Szukamy rekomendacji w systemie typu content-based.

Które zdanie jest **prawdziwe**?

- Jeżeli wybierzemy unormowaną metrykę Minkowskiego definiowaną przez odległość Hamminga do wyznaczenia podobieństwa, to najgorzej preferencje użytkownika oddaje obiekt **o3**.
- Jeżeli wybierzemy unormowaną metrykę Minkowskiego definiowaną przez odległość Hamminga do wyznaczenia podobieństwa, to najlepiej preferencje użytkownika oddaje obiekt **o1**.
- Jeżeli wybierzemy uogólniony współczynnik Jaccarda z funkcją wagową  $f_{1,t=0.5}$  do wyznaczenia podobieństwa, to najlepiej preferencje użytkownika oddaje obiekt **o1**.

Dane jest uniwersum  $\{x_1, x_2, x_3\}$  oraz dwa zbiory rozmyte:

$$A = \frac{0.2}{x_1} + \frac{0.7}{x_2} + \frac{0.5}{x_3}$$

$$B = \frac{0.4}{x_1} + \frac{0.5}{x_2} + \frac{0.6}{x_3}$$

Ich podobieństwo wynosi w przybliżeniu 0.66.

Jaką metryką wyznaczono podobieństwo tych zbiorów?

- unormowana metryka Minkowskiego definiowana przez odległość Hamminga
- uogólniony współczynnik Jaccarda z funkcją wagową  $f_{1,t=0.4}$
- uogólniony współczynnik dopasowania z funkcją wagową  $f_{1,t=0.4}$

Weźmy pod uwagę funkcje wagowe  $f_{1,t=0.2}$ ,  $f_{2,t=0.3}$ ,  $f_{3,p=2}$ .

Jaką zatem moc skalarną może mieć zbiór rozmyty:

$$\frac{0.2}{x_1} + \frac{0.3}{x_2} + \frac{0.1}{x_3} + \frac{0.2}{x_5} + \frac{0.4}{x_6} ?$$

**Rysunek 3.2.** Fragment testu zaliczeniowego w kursie „Zastosowania zbiorów rozmytych”

### 3.1.2. Konwerter *fcl2js*

W ramach przeprowadzonych badań opracowano kilka sterowników Mamdaniego, z czego dwa z nich zostały wykorzystane w kursie „Zastosowania zbiorów rozmytych”. Sterowniki te są zapisywane w języku FCL [41]. Aby możliwe było osadzenie sterownika w kursie, czyli technicznie rzecz biorąc zbiorze plików HTML/CSS/JavaScript, sterownik musiał być przekonwertowany do postaci kodu języka JavaScript. W tym celu zostało opracowane przez autora niniejszej rozprawy rozszerzenie biblioteki jFuzzyLogic o nazwie *fcl2js* [77, 103].

Biblioteka jFuzzyLogic stanowi pełną implementację języka FCL zgodnie ze specyfikacją IEC 61131 cz. 7. Wykorzystana biblioteka jest pakietem do języka Java natomiast domyślnie posiada możliwość nie tylko uruchamiania sterownika ale także konwertowania kodu FCL do postaci sterownika w języku C++. W ramach innego projektu realizowanego w ramach grantu, polegającego na przygotowaniu i zaimplementowaniu kursu z adaptacyjną zawartością na temat efektywnej współpracy z chińskim partnerem biznesowym [108], został zaprojektowany i zaimplementowany przez autora rozprawy moduł *fcl2js* do biblioteki jFuzzyLogic, który umożliwia konwertowanie sterowników zapisanych w języku FCL do języka JavaScript [103].

Kod źródłowy biblioteki jFuzzyLogic z dodatkowym modułem konwertującym FCL do języka JavaScript jest dostępny na platformie GitHub pod adresem: <https://github.com/marcin-szczepanski/jFuzzyLogic>.

Konwerter obsługuje sterowniki rozmyte bazujące na zbiorach rozmytych opisywanych przez funkcje przedziałami liniowe oraz funkcje Gaussa. Aby przekonwertować sterownik zapisany w języku FCL należy z repozytorium pobrać plik *jFuzzyLogic.jar* a następnie wywołać polecenie postaci:

```
java -jar jFuzzyLogic.jar -j controller.fcl > controller.js
```

Działanie konwertera zostanie zaprezentowane w rozdziale 3.5.1 na przykładzie eksperckiego sterownika rozmytego modelującego pojęcie disengagement.

Aktualnie dostępna wersja oprogramowania to v4.0.0. W jej skład wchodzi konwerter *fcl2js*, poprawki niektórych błędów, możliwość rozszerzania okien z wykresami funkcji przynależności zbiorów rozmytych zdefiniowanych w sterowniku oraz możliwość testowania sterowników w problemie klasyfikacji binarnej. Najnowsza wersja pliku *jFuzzyLogic.jar* na stronie <https://github.com/marcin-szczepanski/jFuzzyLogic/releases/tag/v4.00>.

### 3.1.3. Biblioteka *its-student-progress*

Wszystkie prezentowane w rozprawie sterowniki rozmyte bazują na danych o bieżących postępach studentów w kursie. Aby możliwe było przekazywanie takich danych do wejścia sterownika rozmytego konieczne było przygotowanie biblioteki języka JavaScript, która umożliwi zbieranie konkretnych informacji (które nie są obsługiwane jawnie przez standard SCORM 2004) a także ich pobieranie w celu dalszego ich przekazania do sterownika.

W ramach przygotowania do badań, została napisana przez autora rozprawy biblioteka języka JavaScript o nazwie *its-student-progress* [19]. Jest to para skryptów JavaScript - jeden implementuje proces śledzenia aktywności studentów w kursie, drugi pozwala na odczyt tych informacji. Narzędzie pozwala na zapisywanie informacji o interakcjach z ćwiczeniami typu Quiz (pytania wyboru), Fill-in (pytania krótkiej odpowiedzi), Fill-in Numeric (pytania o liczbę) czy Matching (łączenie elementów w pary). Poza tym śledzone są także kliknięcia w elementy kursu takie jak galerie zdjęć, linki, filmy czy dźwięki. Dane są zapisywane w formacie zdefiniowanym dla `cmi.interactions.n.objectives` w standardzie SCORM 2004 [153]. Dzięki temu informacje będą widoczne dla nauczyciela w dowolnym systemie LMS obsługującym standard SCORM 2004. W obszarze odczytu informacji o studencie, poza tym, że możliwe jest pobieranie danych o śledzonych przez bibliotekę aktywnościach, możliwe jest także pobieranie informacji dostępnych w ramach zmiennych zdefiniowanych w standardzie SCORM 2004. Biblioteka zatem stanowi nakładkę na format SCORM 2004, upraszczając pracę osobom, które nie mają dużego doświadczenia z tym formatem. Biblioteka jest dostępna na repozytorium <https://git.wmi.amu.edu.pl/jacekmar/its-student-progress>.

Aktualnie biblioteka może być wykorzystana do zbierania informacji o pracy studenta takie jak:

- historia kliknięć w zadania typu Quiz (pytania wyboru);
- historia kliknięć w zadania typu Fill-in (pytania krótkiej odpowiedzi);
- historia kliknięć w zadania typu Fill-in Numeric (pytania o liczbę);
- historia kliknięć w zadania typu Matching (łączenie elementów w pary);
- historia kliknięć w zadania typu Grouping (grupowanie elementów);
- historia kliknięć w linki;
- historia kliknięć w filmy;
- historia kliknięć w dźwięki;
- historia kliknięć w element typu slider (porównywanie dwóch obrazów poprzez wykorzystanie suwaka);
- historia kliknięć w spis treści (lista z linkami do zdefiniowanych wcześniej części kursu);
- historia kliknięć w galerie zdjęć;
- historia kliknięć w galerie zdjęć z linkami do konkretnego zdjęcia;
- historia kliknięć w galerie zdjęć w postaci kolażu;
- historia kliknięć w ramki przyciskami zmieniającymi jej treść;
- historia kliknięć w przyciski rozpoczęcia i zakończenia testu/ankiety;

W celu wykorzystania tej funkcjonalności biblioteki, należy dołączyć do kursu plik `saveStudentInteractions.js` dostępny na repozytorium. Każda z powyższych informacji obsługiwanych przez bibliotekę jest zapisywana jako obiekt `cmi.interactions` formatu SCORM 2004.

Drugi plik biblioteki - `students.js` to zbiór funkcji języka JavaScript umożliwiający odczyt różnych informacji o postępach studenta w kursie. Część funkcji pozwala na odczyt informacji zapisywanych przez część biblioteki odpowiedzialną za zapis

danych. Inne funkcje służą do odczytywania informacji o budowie kursu (np. informacja o liczbie obiektów SCO, na które składa się kurs czy liczba dostępnych interakcji/ćwiczeń w kursie) i jego konkretnym wdrożeniu (np. ile czasu upłynęło od momentu otwarcia dostępu do kursu czy ile czasu pozostało do zamknięcia dostępu). Kolejna grupa funkcji umożliwia odczyt danych zapisywanych przez sam kurs w formacie SCORM (np. liczba wejść do kursu, spędzony czas, wyniki testów, status zaliczenia). Pełna lista funkcji znajduje się na repozytorium projektu.

### 3.1.4. Biblioteka *its-raport-generator*

Kurs „Zastosowania zbiorów rozmytych” był wielokrotnie wykorzystywany przed wdrożeniem zaprezentowanej w niniejszej rozprawie strategii adaptacji, w ramach zajęć dydaktycznych, podczas których stosowano różne, proste strategie dydaktyczne. Każda edycja kursu wiązała się z gromadzeniem danych dydaktycznych, które jednak często zawierały błędne lub niekompletne informacje. Wraz z kolejnymi edycjami podejmowano decyzje o rozszerzeniu zakresu zbieranych danych, co pozwalało na bardziej szczegółowe monitorowanie aktywności studentów.

W pierwszych edycjach kursu brakowało np. zapisów dotyczących liczby rozwiązywanych ćwiczeń, a problemy techniczne oraz luki w logice działania kursu powodowały trudności w dokładnym pomiarze czasu spędzonego przez studentów w poszczególnych elementach kursu oraz całkowitego czasu spędzonego w kursie. Proces gromadzenia danych był złożony, a pełne wykorzystanie zebranych danych do celów badawczych stało się możliwe dopiero po kilku edycjach kursu. Dzięki wykorzystaniu cząstkowych danych możliwa była wstępna analiza, np. w kontekście opracowywania eksperckiego sterownika rozmytego, jednak pełne wykorzystanie danych do automatycznego generowania sterowników było wcześniej niemożliwe ze względu na brak ich pełności i poprawności.

Zebrane dane wymagały odpowiedniego przetworzenia przed ich dalszym wykorzystaniem w badaniach. Dane pobierane były w postaci raportów dostępnych w systemie Edumatic ITS używanym w badaniach [45]. Dodatkowo gromadzono podstawowe informacje o studentach, przechowywane poza systemem. Konieczne było połączenie tych danych i wygenerowanie plików z podzbiórami danych w różnych kontekstach, zgodnie z potrzebami w ramach zajęć dydaktycznych oraz badań naukowych. Na przykład, w jednym przypadku wymagany był raport z wynikami testów zaliczeniowych wszystkich studentów w formacie umożliwiającym import danych do systemu Moodle lub USOS, w innym natomiast generowano raport z odpowiedziami w ankietach umieszczanych w kursie. W kontekście generowania sterownika rozmytego niezbędne były dane w formacie CSV poddane normalizacji.

W odpowiedzi na te wyzwania, w ramach projektu wymienionego w rozdziale 3.1.2 realizowanego w ramach grantu, zostało opracowane przez autora rozprawy narzędzie *its-raport-generator* [60]. Narzędzie to umożliwia generowanie raportów na podstawie surowych danych pobranych z systemu Edumatic ITS oraz innych źródeł danych, tworzonych przez dydaktyków. Generator raportów *its-raport-generator* bazuje na danych SCORM generowanych przez system, wykorzystywany w badaniach dydaktycznych dla kursów budowanych w narzędziu Eduexe. Narzędzie umożliwia także łączenie gotowych raportów z różnych źródeł (np. Moodle, USOS). Generator został zaimplemen-

towany w języku Python z wykorzystaniem bibliotek *pandas*, *numpy* oraz *beautifulsoup4*. Na podstawie pliku JSON opisującego kurs oraz danych z systemu Edumatic ITS, możliwe jest wygenerowanie 14 różnych raportów, w tym raportów dotyczących postępów w testach, aktywności w ramach ćwiczeń interaktywnych oraz odpowiedzi w ankietach. Narzędzie pozwala także na tworzenie podsumowań dla całego kursu oraz generowanie pełnego zbioru wszystkich danych. Dodatkowo, narzędzie oferuje tryb łączenia danych z różnych źródeł, co umożliwia, np. integrację wyników testów z danymi o aktywnościach studentów w systemie Moodle (w niektórych edycjach kursu był on uruchamiany w tym systemie) oraz ocenami w systemie USOS.

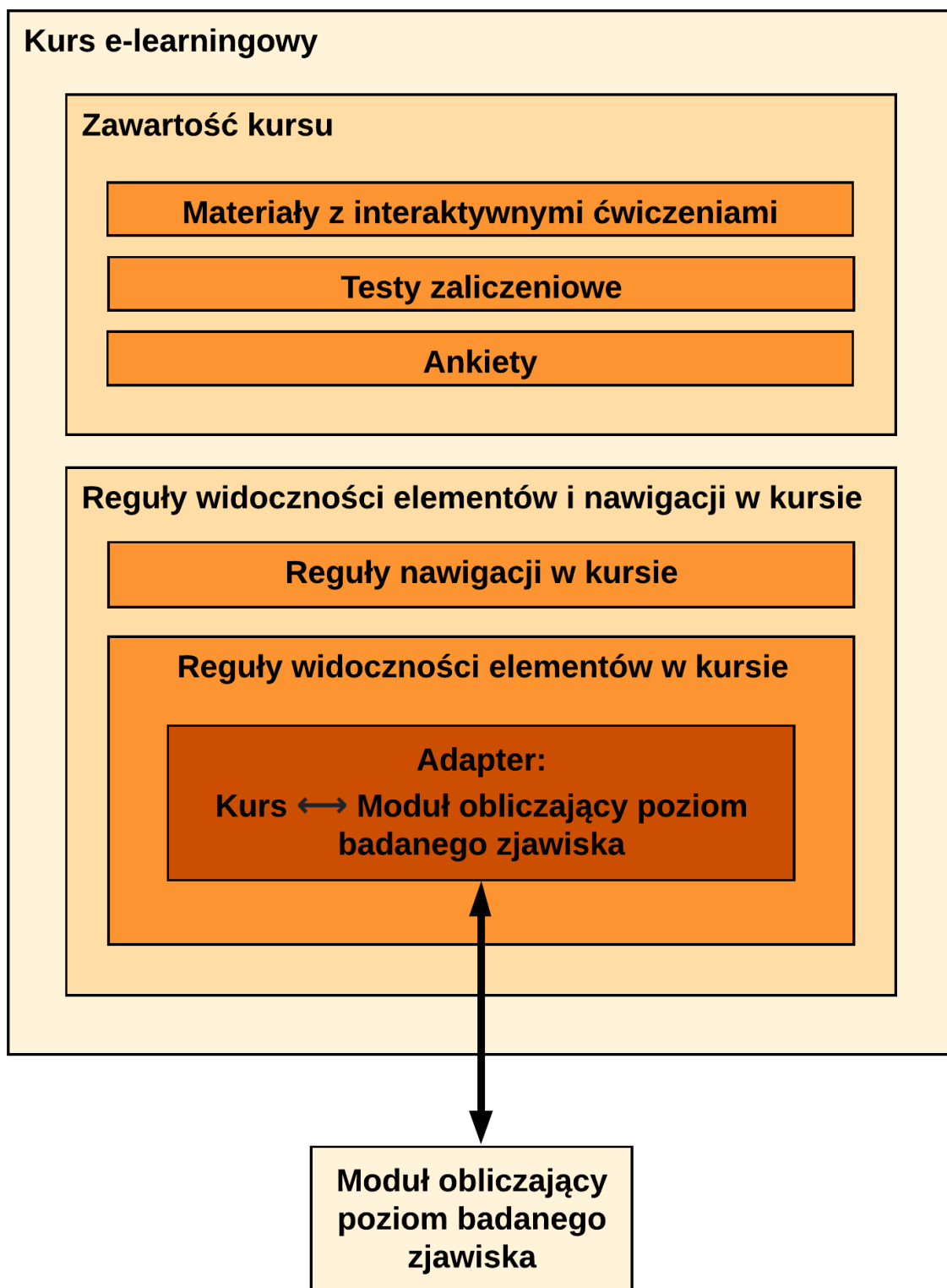
Opracowanie narzędzia *its-raport-generator* znacząco usprawniło proces przetwarzania i analizy danych dydaktycznych, umożliwiając bardziej efektywne wykorzystanie zgromadzonych informacji do celów badawczych oraz dydaktycznych. Dzięki temu możliwe jest generowanie szczegółowych raportów, które wspierają zarówno nauczycieli, jak i badaczy w lepszym zrozumieniu przebiegu kursu i aktywności studentów. Dane wykorzystane do budowy sterowników generowanych automatycznie prezentowanych w niniejszej rozprawie oraz do trenowania algorytmów uczenia maszynowego zostały przygotowane z wykorzystaniem generatora *its-raport-generator* (zob. rozdział 4).

Kod źródłowy narzędzia dostępny jest na repozytorium <https://git.wmi.amu.edu.pl/jacekmar/its-raport-generator>.

## 3.2. ARCHITEKTURA KURSU E-LEARNINGOWEGO Z ADAPTACYJNĄ ZAWARTOŚCIĄ POZWALAJĄCA NA STOSOWANIE RÓŻNYCH METOD ADAPTACJI

Aby możliwe było realizowanie różnych, nawet bardziej złożonych strategii dydaktycznych w kursie z adaptacyjną zawartością, konieczna jest taka architektura kursu, która będzie maksymalnie ogólna, nienastawiona na konkretną strategię dydaktyczną. W rozprawie zaproponowano architekturę, która dzięki swojej ogólności pozwala na wdrażanie różnych strategii. W proponowanym rozwiązaniu zostanie ona wykorzystana do wdrożenia strategii bazującej na poziomie zjawiska disengagement u danego studenta ale z powodzeniem może być ona stosowana także do implementacji innych strategii, np. bazujących na stylu uczenia się czy na kilku zjawiskach dydaktycznych znanych w e-learningu (nie tylko disengagement ale także poziom satysfakcji studenta czy poziom uwagi).

Ogólna architektura kursu e-learningowego z adaptacyjną zawartością jest zaprezentowana na rysunku 3.3. Rysunek przedstawia techniczną strukturę kursu z podziałem na treść oraz reguły widoczności elementów i nawigacji w kursie. Kurs może składać się z wielu komponentów (SCO zgodnie ze standardem SCORM) takich jak materiały do nauki z interaktywnymi ćwiczeniami, testy zaliczeniowe czy ankiety. Dla każdego komponentu autor kursu ma możliwość określenia reguł definiujących stan kursu u danego studenta, w którym komponent może zostać wyświetlony. Głównie są to reguły bazujące na danych takich jak stan całego kursu a także stan wybranych komponentów (na stan składają się: status ukończenia, status zaliczenia i zdobyte punkty).



**Rysunek 3.3.** Architektura kursu z modułem obliczającym poziom badanego zjawiska wykorzystywanego w ramach strategii adaptacji

Zgodnie z zaprezentowaną architekturą w kursie może zostać osadzony dodatkowy moduł uzupełniający blok reguł widoczności elementów w kursie. Reguły poza takimi bazującymi na stanie kursu lub komponentu mogą wykorzystywać zdefiniowane przez autora kursu zmienne. Zmienne te przyjmują wartości liczbowe. Na regułę może się składać wiele zmiennych połączonych ze sobą operatorami logicznymi. Przykładowa reguła określająca, czy dany element powinien zostać wyświetlony może być następująca:

```
IF zmienna1 == 1 AND zmienna2 > 3 OR zmienna3 != 7 THEN return true;
```

Wartość każdej ze zmiennych może być obliczana przez inny dołączony do kursu moduł a nawet przez wykonanie zapytania do zewnętrznego serwera. Zmienne mogą też przyjmować postać efektów uczenia zdefiniowanych w kursie poprzez ich przypisanie do pytań w testach zaliczeniowych. Komunikacja między modułami a blokiem sterującym widocznością elementów w kursie odbywa się poprzez tzw. adapter. Wszystkie zmienne deklarowane są w logice kursu (np. w przypadku zjawiska disengagement należy zadeklarować zmienną `disengagement` przyjmującą wartości liczbowe) natomiast obliczanie ich wartości odbywa się w zewnętrznym module. Działanie adaptera polega zatem na nadpisaniu domyślnych definicji metod z bloku sterującego widocznością elementów umożliwiających obliczanie wartości zmiennych w sposób niestandardowy. Schemat postępowania w przypadku wykorzystania niestandardowych zmiennych w regułach jest zatem następujący:

1. Zadeklaruj niestandardowe zmienne w kursie.
2. Zdefiniuj reguły widoczności elementów w kursie wykorzystujące niestandardowe zmienne w kursie.
3. W zewnętrznym module zdefiniuj metody obliczające wartości zadeklarowanych zmiennych jako zbiór funkcji języka JavaScript.
4. Przygotuj adapter (czyli kod języka JavaScript), w którym domyślne metody kursu odpowiadające za obliczenie wartości niestandardowych zmiennych zostaną nadpisane metodami wywołującymi kod z zewnętrznego modułu.
5. Osadź kod adaptera i zewnętrznego modułu w kursie.
6. Blok sterujący widocznością elementów w kursie w momencie natrafienia na zmienną, której sposób obliczania został nadpisany, wywoła poprzez adapter kod z zewnętrznego modułu a następnie w oparciu o otrzymaną wartość tej zmiennej zostanie sprawdzona spełnialność reguły widoczności danego elementu.

Przykład 3.1 pokazuje zastosowanie powyższej architektury oraz schematu postępowania w kursie, w którym dostępność komponentu zależy od liczby odwiedzonych elementów.

**Przykład 3.1.** Niech kurs e-learningowy z adaptacyjną zawartością składa się z pięciu komponentów:

1. Materiał do nauki A — dostępny zawsze
2. Materiał do nauki B — dostępny zawsze
3. Materiał do nauki C — dostępny zawsze
4. Test — dostępny w przypadku, gdy student odwiedził wszystkie poprzednie komponenty
5. Informacja — komponent dostępny w przypadku, gdy test jest niedostępny

Kurs jest zbudowany w narzędziu Eduexe [44]. Informację o liczbie odwiedzonych elementów można uzyskać poprzez wykorzystanie biblioteki *its-student-progress* (patrz rozdział 3.1.3).

W kursie zdefiniowano niestandardową zmienną `visited_elements` — liczba odwiedzonych elementów (materiałów do nauki). Dla testu zbudowano warunek widoczności:

```
IF visited_elements == 3 THEN return true;
```

Natomiast dla komponentu „Informacja” zbudowano warunek przeciwny:

```
IF visited_elements < 3 THEN return true;
```

W kursie osadzono plik `students.js` biblioteki *its-student-progress*. Będzie on pełnił funkcję zewnętrznego modułu. Natomiast w pliku, który będzie pełnił funkcję adaptera zdefiniowano metody nadpisujące te domyślne stosowane w kursie — przedstawia to wyciąg 3.1.

```
1 var tagsAndRulesRef = null;
2 var studentsjs = null;
3
4 if (window.parent.EduexeBrowserCore == undefined) {
5     tagsAndRulesRef = window.TagsAndRules;
6     studentsjs = window.StudentsJS;
7 } else {
8     tagsAndRulesRef = window.parent.TagsAndRules;
9     studentsjs = window.parent.StudentsJS;
10 }
11
12 tagsAndRulesRef.customCalculatedTagsValues = [0];
13 tagsAndRulesRef.customCalculatedTags = ["visited_elements"];
14
15 tagsAndRulesRef.getCustomCalculatedTagCurrentValue = function(tag) {
16     if (tag == "visited_elements") {
17         return studentsjs.getNumberOfVisitedSCOsInCourse([4, 5]);
18     }
19 }
```



```
20     return 0;
21 }
22
23 tagsAndRulesRef.getCustomCalculatedTagMaxValue = function(tag) {
24     if (tag == "visited_elements") {
25         return 3;
26     }
27
28     return 1;
29 }
```

**Wyciąg 3.1.** Kod przykładowego adaptera w kursie z adaptacyjną zawartością

W liniach 1-10 następuje zdefiniowanie zmiennych odpowiadających za komunikację między plikiem kursu odpowiedzialnego za reguły (zmienna `tagsAndRulesRef`) oraz biblioteki *its-student-progress* (zmienna `studentsjs`). W liniach 12-13 następuje wskazanie zmiennej wyliczanej niestandardowo i jej domyślnej wartości. Linie 15-21 stanowią definicję metody `getCustomCalculatedTagCurrentValue()` wykorzystywanej w kursie do obliczenia wartości zmiennej występującej w danej regule — tutaj zostaje wykorzystana funkcja biblioteki *its-student-progress* zwracająca informację o liczbie odwiedzonych elementów (linia nr 17; w wykorzystanej funkcji zgodnie z dokumentacją należy wskazać komponenty, których nie należy brać pod uwagę). W liniach 23-29 nadpisano metodę `getCustomCalculatedTagMaxValue()` informującą o tym, jaką maksymalną wartość może przyjąć niestandardowa zmienna — funkcja ta jest również wykorzystywana w kursie w procesie sprawdzania reguł.

Po osadzeniu adaptera i biblioteki *its-student-progress* w kursie, moduł adaptacyjności jest gotowy. W procesie sprawdzania reguł widoczności elementów w kursie, jeżeli funkcja `getNumberOfVisitedSCOsInCourse()` biblioteki *its-student-progress* zwróci liczbę 3, w kursie zostanie wyświetlony link do komponentu zawierającego test. Natomiast dopóki ta wartość nie zostanie osiągnięta, student będzie miał dostępny komponent „Informacja”.

Zgodnie z powyższym przykładem, w kursie opartym na prezentowanej architekturze można zbudować dowolną strategię adaptacji opartą na regułach widoczności elementów i niestandardowych zmiennych, których wartości obliczane są za pomocą zewnętrznych funkcji.

W ramach proponowanego w rozdziale 3.3 algorytmu adaptacji zostanie zdefiniowana niestandardowa zmienna, której wartość będzie oznaczała natężenie zjawiska disengagement u danego studenta. Proponowanym modułem obliczającym to natężenie będzie ekspercki sterownik rozmyty Mamdaniego [102].

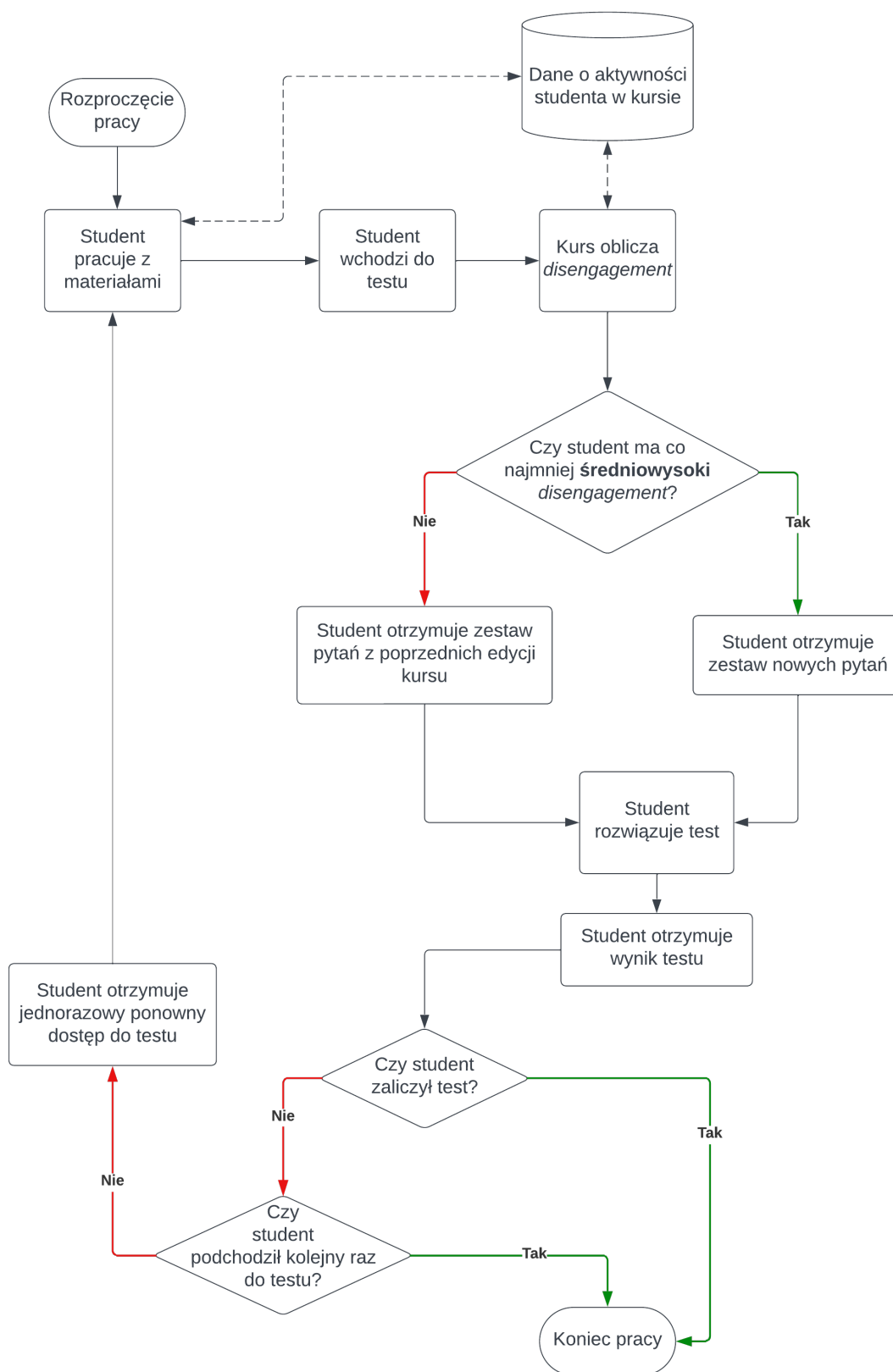
### 3.3. STRATEGIA DYDAKTYCZNA WYKORZYSTUJĄCA ZJAWISKO DISENGAGEMENT

W oparciu o prezentowaną architekturę kursu z adaptacyjną zawartością została zaproponowana strategia adaptacji treści oparta na zjawisku disengagement. Celem tej strategii jest modyfikacja zawartości testu zaliczeniowego w kursie w momencie, kiedy student nie jest wystarczająco zaangażowany w pracę z kursem. Dzięki takiej strategii, zminimalizowane zostanie ryzyko podchodzenia do testów bez wcześniejszej nauki. Osoby, które od razu podejść do testu, otrzymają zestaw pytań, które nigdy nie były wykorzystywane w poprzednich wersjach kursu. Oznacza to, że taki student nie będzie mógł zaliczyć testu, bo nie zna odpowiedzi na pytania. Dzięki temu będzie mieć możliwość nauki i ponownego podejścia do testu w momencie, gdy student rzeczywiście będzie odpowiednio do tego przygotowany.

Strategię adaptacji przedstawia rysunek 3.4. Na rysunku pokazano przepływ pracy studenta w kursie składającym się z pewnej liczby interaktywnych materiałów do nauki oraz testu zaliczeniowego na koniec kursu. W pierwszej fazie procesu dydaktycznego student pracuje z materiałami do nauki — postępy studenta są na bieżąco zapisywane. W momencie, kiedy student stwierdzi, że zakończył naukę, to podchodzi do testu zaliczeniowego. Należy tutaj zwrócić uwagę, że studenci pracowali bez nadzoru nauczyciela w określonym oknie czasowych oraz, że mogą przyjmować różne strategie pracy, np. jeden student może sumiennie pracować ze wszystkimi dostępnymi materiałami, natomiast inny student może pominąć materiały i od razu podejść do testu.

W proponowanej strategii dydaktycznej, obliczany jest poziom zjawiska disengagement w oparciu o dane o dotychczasowej aktywności studenta w kursie. Jeżeli poziom zjawiska disengagement jest co najmniej **średniowysoki**, to student otrzyma dostęp do testu z pytaniami, które nie były wykorzystywane we wcześniejszych edycjach kursu — zakłada się, że kurs był wykorzystywany w innych grupach zajęciowych, więc część pytań może być znana kolejnym studentom, ponieważ takie pytania mogły być udostępnione w Internecie. Jeżeli taki kurs jest wykorzystywany po raz pierwszy, to taki test może zawierać po prostu bardzo podobne pytania ale różniące się w stosunku do zestawu pytań w teście, który jest widoczny u studentów, których poziom zjawiska disengagement jest niższy.

Zgodnie z prezentowanym algorytmem adaptacji, po rozwiązaniu testu w zależności od jego wyniku, system uznaje test za zaliczony i wtedy praca studenta się kończy, lub w przypadku porażki student ma jednorazowy ponowny dostęp do testu. Wtedy student może dalej pracować z materiałami do nauki a później rozwiązać test. I tak jak w przypadku pierwszego podejścia, znowu obliczany jest poziom zjawiska disengagement i podejmowana jest decyzja o dostarczeniu studentowi odpowiedniej wersji testu.



Rysunek 3.4. Algorytm adaptacji treści w kursie z adaptacyjną zawartością oparty na zjawisku disengagement

W prezentowanym algorytmie adaptacji występuje pojęcie „średniowysoki poziom zjawiska disengagement”. Pojawia się pytanie, jak zdefiniować w regule widoczności komponentu wartość „średniowysoki”. Należy ustalić wartość, po przekroczeniu której student uzyska dostęp do odpowiedniej wersji testu. Przy założeniu, że disengagement jest mierzony w przedziale [0; 1], to autor kursu może wybrać jakąś konkretną wartość z tego przedziału, np. 0,5 czy inną. Jeżeli by spróbować w nieprecyzyjny sposób opisać wartości jakie może przyjmować zmienna opisująca poziom zjawiska disengagement, to można by było zastosować gradację wartości zgodnie z pięciostopniową skalą Likerta: *bardzo niski, niski, umiarkowany/średni, wysoki, bardzo wysoki*. W badaniach najczęściej wykorzystuje się właśnie pięciostopniową lub siedmiostopniową skalę Likerta [78]. Pięciostopniowa skala jest jednak prostsza w rozumieniu a co więcej przedział [0; 1] można podzielić na pięć równych części, z których każda będzie odpowiadała kolejnemu punktowi na skali.

W 2021 roku zostały przeprowadzone badania nad rozumieniem pięciostopniowej skali Likerta w porównaniu do skali procentowej [113]. Co prawda, pokazano, że są istotne statystycznie różnice w rozumieniu i co do zasady punkty na skali Likerta nie odzwierciedlają w stosunku 1:1 podziału na pięć równych części na skali procentowej, natomiast uśredniając, cały czas liczba 100 na skali procentowej jest dzielona przez liczbę zbliżoną do 20. Zatem, gdyby jednak podzielić przedział [0; 1] na pięć równych części (tzn. [0; 0,2), [0,2; 0,4), [0,4; 0,6), [0,6; 0,8) i [0,8; 1]), to każda z nich oznaczałaby kolejny punkt na skali Likerta. Czwarty i piąty punkt na skali opisują wysoki lub bardzo wysoki poziom zjawiska disengagement. Trzeci punkt natomiast oznacza średni poziom. Zatem można przyjąć założenie, że średniowysoki poziom zjawiska disengagement zaczyna się na granicy przedziałów opisujących trzeci i czwarty punkt skali Likerta, czyli na wartości 0,6. Więc w proponowanym rozwiązaniu, studenci, których disengagement jest co najmniej na poziomie 0,6, otrzymają dostęp do testu zaliczeniowego z nowymi pytaniami, ponieważ poziom zjawiska disengagement jest co najmniej średniowysoki.

Zaprezentowana metoda obliczania poziomu zjawiska disengagement pokazuje, w jaki sposób dydaktyk podchodzi do tego modelowania zagadnienia. Przyjęte w pracy rozwiązanie pozwala na uchwycenie takich problemów. W kontekście zjawiska disengagement, jego wartość może być różna, jeżeli dydaktyk tak uzna, ale architektura rozwiązania powinna pozwalać na natychmiastowe wprowadzenie takiej zmiany bez potrzeby przebudowywania technicznego całego kursu. Rozwiązanie prezentowane w rozprawie na to pozwala.

### 3.4. MODELOWANIE ZJAWISKA DISENGAGEMENT Z WYKORZYSTANIEM STEROWANIA ROZMYTEGO

W rozprawie przyjęto założenie, że ze względu na swój nieprecyzyjny charakter, poziom zjawiska disengagement będzie opisywany z wykorzystaniem wartości lingwistycznych — np. *bardzo niski, niski, umiarkowany/średni, wysoki, bardzo wysoki* (zob. rozdział 3.3). Są to pojęcia nieprecyzyjne, zatem należy wybrać metodę ich formalnego opisu, aby móc takie

wartości wykorzystać podczas obliczeń. W proponowanym rozwiązaniu do tego celu są wykorzystane zbiory rozmyte [180].

Stan kursu, który determinuje nasilenie zjawiska disengagement w prezentowanym rozwiązaniu będzie opisywany poprzez bazę eksperckich reguł. Mogłyby to być ostre reguły, tzn. takie, które wykorzystują liczby a nie nieprecyzyjne wartości lingwistyczne, jednak zbudowanie bazy ostrych reguł jest trudne — reguł może być dużo a ponadto praktycznie niemożliwe jest objęcie wszystkich możliwych schematów pracy studentów z kursem. Zatem takie rozwiązanie zostało odrzucone i wybrano zastosowanie rozmytych reguł. Logika rozmyta ułatwia bowiem konstruowanie systemów wnioskowania, które charakteryzują się nieciągłym i nieliniowym podejmowaniem decyzji, przez co bardzo przypominają ludzkie zachowanie w przeciwieństwie do logiki klasycznej [2]. W efekcie baza reguł w ramach konstruowanego systemu może być znacząco mniejsza. Poza tym, w kontekście e-learningu, objęcie wszystkich możliwych schematów pracy uczniów tak precyzyjnymi regułami może być trudne, ponieważ każdy uczeń pracuje inaczej. Logika rozmyta pokonuje tę trudność, oferując podejście, które upraszcza proces dostrajania reguł, ułatwiając uwzględnienie różnorodnych wzorców pracy studentów.

W proponowanym rozwiązaniu zostanie wykorzystany ekspercki sterownik rozmyty Mamdaniego. Zostaną zdefiniowane zmienne lingwistyczne będące wejściem sterownika a także baza reguł.

#### 3.4.1. Zmienne wejściowe sterownika

Przyjęto założenie, że ekspercki sterownik rozmyty Mamdaniego obliczający poziom zjawiska disengagement przyjmuje na wejściu dwie zmienne lingwistyczne:

1. **jakość nauki** (*learning\_quality*),
2. **czas pozostały do zamknięcia testu zaliczeniowego, liczony w momencie pierwszego wejścia do kursu** (*remaining\_time*).

W założeniach strategii adaptacji (patrz rozdział 3.3) przyjęto założenie, że baza reguł powinna być jak najmniejsza, aby była zrozumiała przy analizie działania sterownika dla konkretnych danych wejściowych. Dlatego zdecydowano się na wprowadzenie jedynie dwóch zmiennych lingwistycznych i właśnie z tego powodu wprowadzono pojęcie jakości nauki, które agreguje trzy typy informacji o aktywnościach studenta w kursie zamiast każdą z nich modelować jako osobną zmienną wejściową. Dzięki zastosowaniu takiego podejścia, sterownik zamiast przyjmować cztery zmienne — i przy założeniu, że każda jest określona na trzech termach, baza reguł może przyjąć do 81 reguł (cztery zmienne po trzy termy, czyli  $3^4$  kombinacji zmiennych i ich termów) — przyjmuje tylko dwie zmienne i baza reguł może już wtedy mieć maksymalnie zaledwie dziewięć reguł (dwie zmienne po trzy termy, czyli  $3^2$  kombinacji zmiennych i ich termów).

Z punktu widzenia dydaktyka, ważne jest, czy osoba ucząca się w kursie rozwiązuje ćwiczenia (czyli realnie się uczy), czy spędza odpowiednią ilość czasu oraz czy nie pomija

ważnych treści. Takie właśnie informacje zostały wybrane do zdefiniowania w sterowniku zmiennej opisującej jakość nauki studenta. Zatem na jakość nauki składają się:

- **średni czas spędzony w jednym elemencie kursu** (*time*) — w proponowanym rozwiązaniu ze względu na uwarunkowania techniczne (czas spędzony w danym komponencie kursu był odnotowywany co pół minuty) opracowano algorytm normalizacji czasu, w którym każde pół minuty spędzone w kursie to jeden punkt — przyjęto czas maksymalny na 40 punktów (w zależności od kursu, autor może określić inny górny limit punktów), czyli 20 minut — jeśli student spędził więcej czasu, to otrzymywał również 40 punktów — w przypadku kursu „Zastosowania zbiorów rozmytych” autor przyjął założenie, że maksymalnie w jednym elemencie kursu student powinien spędzić nie więcej niż 20 minut — tak zdefiniowany czas został znormalizowany do przedziału [0; 1],
- **liczba rozwiązanych ćwiczeń w kursie** (*interactions*) — wartość została znormalizowana do przedziału [0; 1],
- **liczba elementów, których student nie odwiedził** (*not\_visited*) — jeśli student w danym komponencie spędził mniej niż pół minuty (czyli nie zdobył punktów dla zmiennej *time*), to komponent był oznaczany jako nieodwiedzony — wartość została także znormalizowana do przedziału [0; 1].

Jakość nauki została określona wzorem [163]:

$$learning\_quality = \frac{interactions + 2 \cdot time - not\_visited}{3}$$

Wzór został opracowany poprzez wielokrotne iteracje, w których były zmieniane współczynniki przy każdej zmiennej wchodzącej w skład definicji pojęcia jakości nauki. Bazowano na analizie danych zebranych we wcześniejszych edycjach kursu „Zastosowania zbiorów rozmytych”. Ostatecznie wybrany został powyższy wzór, w którym średni czas spędzony w jednym elemencie kursu ma wyższy współczynnik, ponieważ jest to informacja o tym, ile łącznie czasu zajęło studentowi zapoznanie się z materiałami oraz rozwiązanie ćwiczeń — niektórzy studenci mogli przyjąć strategię pracy, w której nie czytali materiałów do nauki a tylko podchodzili do ćwiczeń, żeby sprawdzić, jakiego typu pytania mogą się pojawić na teście zaliczeniowym.

Czas pozostały do zamknięcia testu zaliczeniowego, liczony w momencie pierwszego wejścia do kursu został natomiast znormalizowany do przedziału [0; 1] zgodnie ze wzorem [163]:

$$remaining\_time = \frac{end - student\_start}{end - start}$$

gdzie:

- *start* to data otwarcia dostępu do kursu (reprezentowana jako *timestamp*),
- *end* to data zamknięcia dostępu do testu zaliczeniowego (reprezentowana jako *timestamp*),
- *student\_start* to data pierwszego wejścia studenta do kursu (reprezentowana jako *timestamp*).

Wzór ten wynika z metody normalizacji min-max, w której wykorzystuje się następującą formułę:

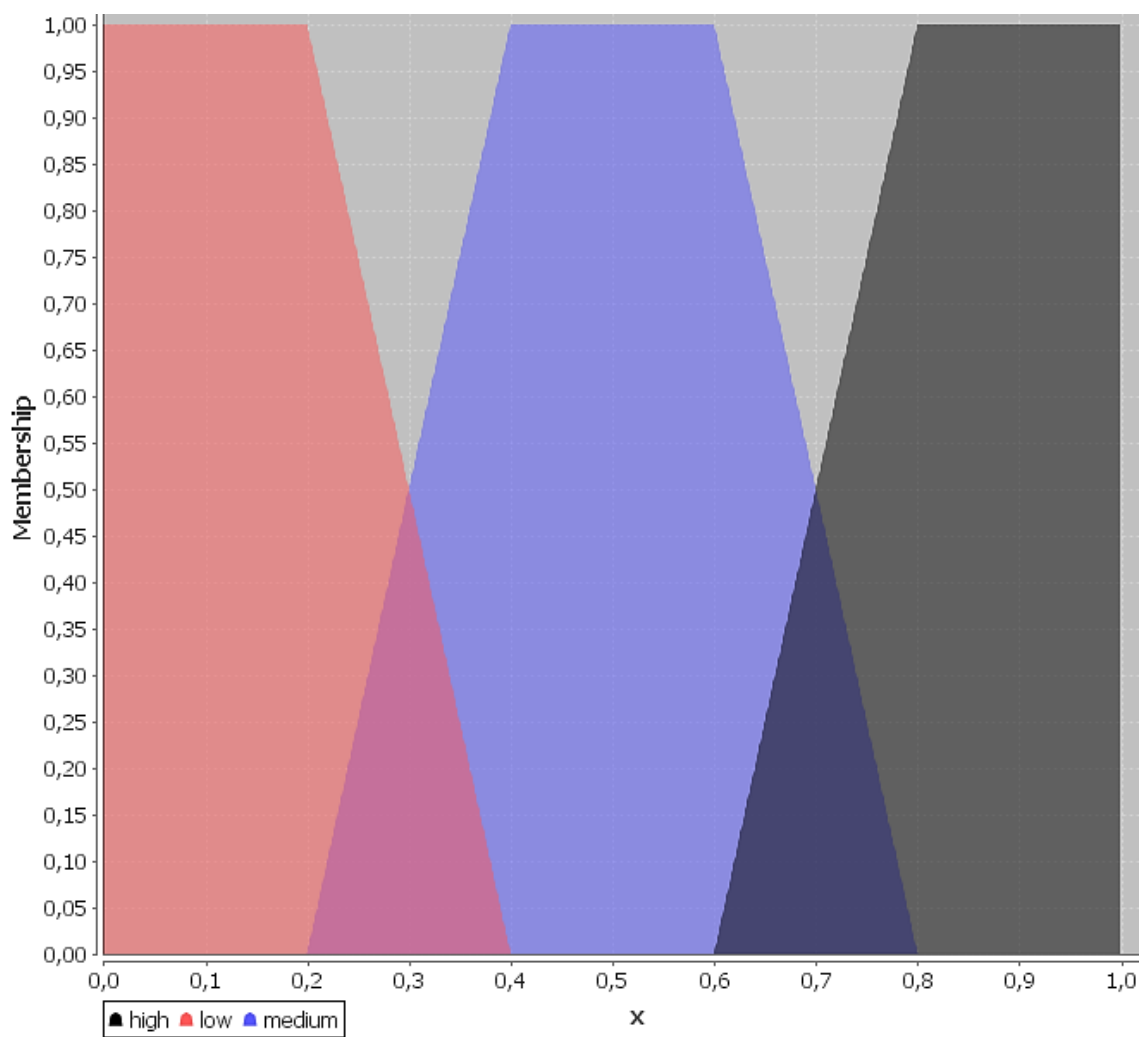
$$x' = \frac{x - \min(x)}{\max(x) - \min(x)}$$

Gdyby przyjąć ten wzór, to data pierwszego wejścia studenta do kursu oznaczałaby czas, który upłynął od momentu otwarcia dostępu do kursu. Przyjęto założenie, że data zamknięcia dostępu do testu zaliczeniowego jest ważniejsza, więc normalizacja została przeprowadzona nie w stosunku do wartości minimalnej a maksymalnej, co przedstawia wzór, który jest tożsamy ze wzorem pozwalającym obliczenie czasu pozostałego do zamknięcia testu zaliczeniowego:

$$x' = \frac{\max(x) - x}{\max(x) - \min(x)}$$

Obie zmienne wejściowe mogą przyjmować jedną z trzech wartości lingwistycznych: *niski*, *średni*, *wysoki*. Przyjęto założenie, że zmienne będą przyjmować tylko trzy wartości, aby było jak najmniej reguł w bazie (gdyby było ich np. pięć a nie trzy, to baza mogłaby mieć już maksymalnie  $5^2$  reguł, czyli 25, zamiast  $3^2$ , czyli dziewięć).

Jakość nauki została zdefiniowana na przedziale  $[-1; 1]$ , natomiast czas pozostały do zamknięcia testu zaliczeniowego jest znormalizowany do przedziału  $[0; 1]$ . W proponowanym rozwiązaniu przyjęto, że w przypadku, gdy jakość nauki przyjmuje wartość poniżej zera, to disengagement jest równy 1. Zatem w samym sterowniku jakość nauki także jest rozpatrywana na przedziale  $[0; 1]$ . Ponieważ dla obu zmiennych uniwersum jest takie samo (czyli zbiór  $[0; 1]$ ), to ich wartości lingwistyczne zostały zamodelowane w taki sam sposób. Model dla obu zmiennych przedstawiony jest na rysunku 3.5. Na rysunku przedstawione są wykresy funkcji przynależności do zbiorów rozmytych będących interpretacjami wartości lingwistycznych określonych dla obu zmiennych wejściowych sterownika (funkcja opisana jako *low* odpowiada wartości *niski*, *medium* — *średni*, *high* — *wysoki*). Funkcje te są funkcjami trapezowymi, przy czym funkcje *low* i *high* są do siebie symetryczne względem prostej  $x = 0,5$ , czyli prostej przecinającej w połowie rozpatrywane uniwersum. Natomiast funkcja *medium* przecina funkcje *low* i *high* dokładnie w ich połowach oraz wypełnia przestrzeń uniwersum niepokrytą funkcjami *low* i *high*. Zatem całe uniwersum jest pokryte wartościami lingwistycznymi, co jest zgodne ze sztuką modelowania zmiennych lingwistycznych.



**Rysunek 3.5.** Model zbioru wartości obu zmiennych wejściowych definiowanych w proponowanym sterowniku



### 3.4.2. Baza reguł eksperckich i proces defuzyfikacji

Kolejnym krokiem w procesie budowania sterownika rozmytego jest zdefiniowanie bazy reguł. Zgodnie z modelem bazy reguł sterownika rozmytego przedstawionym w rozdziale 2.6.2, w poprzedniku każdej reguły wykorzystywane są zmienne wejściowe zdefiniowane w bloku rozmywania. W następnikach reguł znajduje się zawsze odwołanie do zmiennej wyjściowej. W prezentowanym rozwiązaniu modelowany jest poziom zjawiska disengagement, zatem to właśnie będzie zmienna wyjściowa sterownika. Zmienna wyjściowa została tak jak zmienne wejściowe zdefiniowana na przedziale  $[0; 1]$  w oparciu o trzy wartości lingwistyczne. Reprezentuje ona znormalizowaną wartość poziomu zjawiska disengagement, zatem do jej rozmycia został zastosowany ten sam model, co w przypadku zmiennych wejściowych — czyli model przedstawiony na rysunku 3.5.

Modelowany sterownik rozmyty został zbudowany w oparciu o bazę reguł składającą się z dziewięciu reguł — jest to maksymalna liczba reguł, jakie można zbudować w oparciu o dwie zmienne wejściowe o trójelementowym zbiorze wartości. Zmienna wyjściowa również może przyjmować jedynie trzy wartości lingwistyczne, zatem dla każdej z tych wartości zbudowano trzy reguły z różnymi kombinacjami wartości zmiennych wejściowych. Baza reguł została zaprezentowana w tabeli 3.2. Tabela przedstawia wszystkie reguły wykorzystane w sterowniku.

Reguły są wykorzystywane w bloku wnioskowania sterownika rozmytego. W poprzednikach reguł obie zmienne wejściowe są połączone ze sobą koniunkcją. Najczęściej, w sterownikach rozmytych koniunkcję realizuje się poprzez operację minimum i w tym rozwiązaniu także został wybrany ten sposób. Ze względu na charakter tej operacji przyjęto założenie, że będzie ona lepsza niż alternatywa (analogicznie obliczana jako maksimum dwóch liczb), ponieważ obie zmienne powinny mieć realny wpływ w procesie podejmowania decyzji o wartości poziomu zjawiska disengagement. W przypadku alternatywy w przypadku, gdy jedna ze zmiennych ma wysoką wartość a druga niską, to ta druga nie miałaby praktycznie żadnego znaczenia.

Zgodnie z modelem sterownika Mamdaniego zaproponowanym do obliczenia poziomu zjawiska disengagement, wynikiem każdej reguły jest zbiór rozmyty będący iloczynem zbioru, który jest interpretacją termu zmiennej wyjściowej, znajdującego się w następniku reguły oraz zbioru, dla którego funkcja przynależności przyjmuje zawsze wartość obliczoną w poprzedniku reguły. Wszystkie tak powstałe zbiory rozmyte są agregowane przy użyciu operacji maksimum (co jest częstą praktyką), a wyjściowy zbiór rozmyty jest wyostrzany z wykorzystaniem metody środka ciężkości (COG) — również najczęściej wykorzystywanej w procesie wyostrzania w sterownikach rozmytych. Sposób działania prezentowanego sterownika przedstawia przykład 3.2.

Lp.	Reguła
1.	IF <b>learning_quality</b> IS <i>low</i> AND <b>remaining_time</b> IS <i>low</i> THEN <b>disengagement</b> IS <i>high</i>
2.	IF <b>learning_quality</b> IS <i>low</i> AND <b>remaining_time</b> IS <i>medium</i> THEN <b>disengagement</b> IS <i>medium</i>
3.	IF <b>learning_quality</b> IS <i>low</i> AND <b>remaining_time</b> IS <i>high</i> THEN <b>disengagement</b> IS <i>medium</i>
4.	IF <b>learning_quality</b> IS <i>medium</i> AND <b>remaining_time</b> IS <i>low</i> THEN <b>disengagement</b> IS <i>medium</i>
5.	IF <b>learning_quality</b> IS <i>medium</i> AND <b>remaining_time</b> IS <i>medium</i> THEN <b>disengagement</b> IS <i>medium</i>
6.	IF <b>learning_quality</b> IS <i>medium</i> AND <b>remaining_time</b> IS <i>high</i> THEN <b>disengagement</b> IS <i>low</i>
7.	IF <b>learning_quality</b> IS <i>high</i> AND <b>remaining_time</b> IS <i>low</i> THEN <b>disengagement</b> IS <i>medium</i>
8.	IF <b>learning_quality</b> IS <i>high</i> AND <b>remaining_time</b> IS <i>medium</i> THEN <b>disengagement</b> IS <i>low</i>
9.	IF <b>learning_quality</b> IS <i>high</i> AND <b>remaining_time</b> IS <i>high</i> THEN <b>disengagement</b> IS <i>low</i>

Tabela 3.2. Baza reguł wykorzystanych w proponowanym sterowniku

**Przykład 3.2.** Moduł odpowiedzialny za obliczenie wartości poziomu zjawiska disengagement (zgodnie ze strategią adaptacji prezentowaną na rysunku 3.4 w rozdziale 3.3) wywołał kod sterownika rozmytego. W oparciu o następujące dane wejściowe należy obliczyć disengagement:

- $learning\_quality = 0,30$ ,
- $remaining\_time = 0,65$ .

W pierwszej fazie działania sterownika otrzymane wartości zostaną rozmyte zgodnie ze wcześniej zdefiniowanym modelem (zob. rysunek 3.5):

**learning\_quality:**

- $low(0,30) = 0,50$
- $medium(0,30) = 0,50$
- $high(0,30) = 0,00$

**remaining\_time:**

—  $low(0,65) = 0,00$

—  $medium(0,65) = 0,75$

—  $high(0,65) = 0,25$

W drugiej fazie działania sterownika dokonywany jest proces wnioskowania w oparciu o bazę reguł (zob. tabela 3.2). Dla każdej reguły na podstawie wcześniej zfuzyfikowanych wartości zmiennych wejściowych należy wyznaczyć stopień spełnienia poprzednika reguły:

**Reguła 1.**  $0,50 \wedge 0,00 = 0,00$

**Reguła 2.**  $0,50 \wedge 0,75 = 0,50$

**Reguła 3.**  $0,50 \wedge 0,25 = 0,25$

**Reguła 4.**  $0,50 \wedge 0,00 = 0,00$

**Reguła 5.**  $0,50 \wedge 0,75 = 0,50$

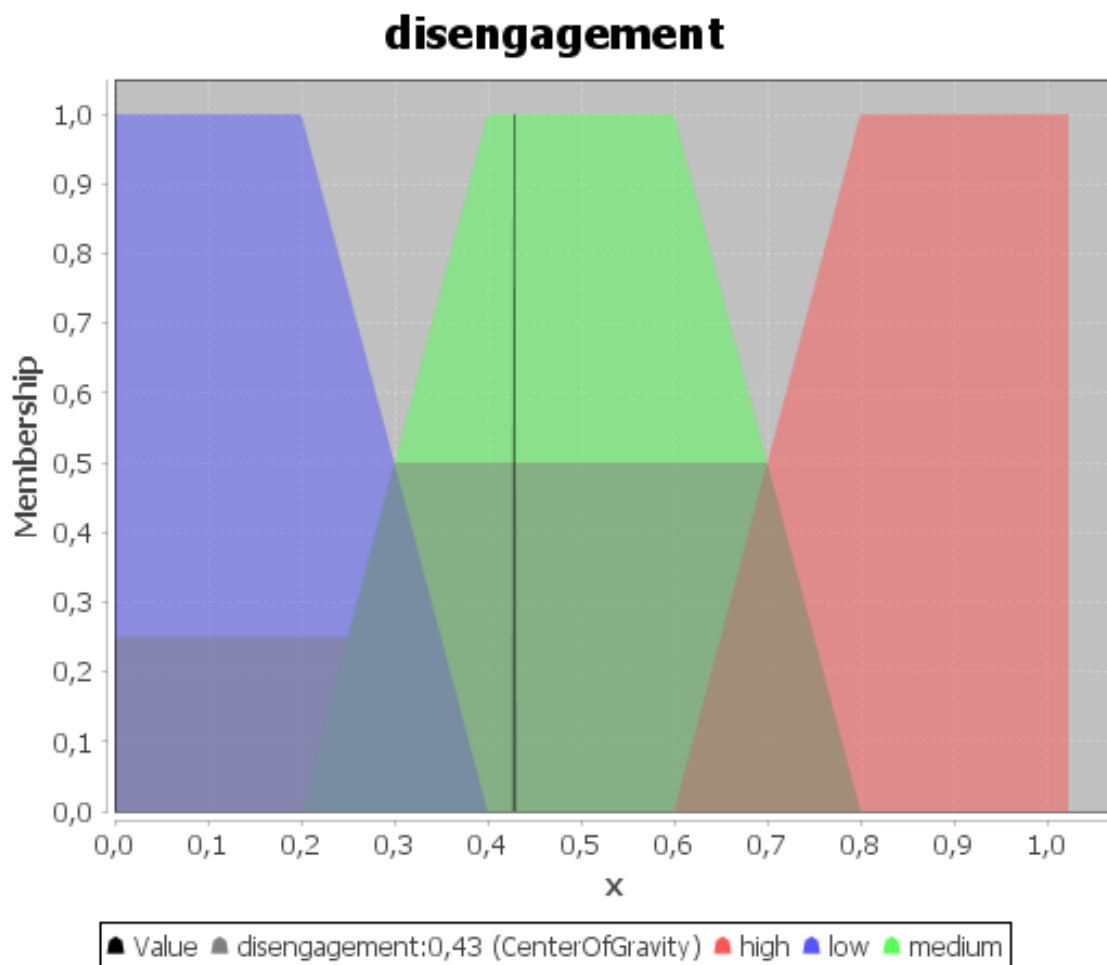
**Reguła 6.**  $0,50 \wedge 0,25 = 0,25$

**Reguła 7.**  $0,00 \wedge 0,00 = 0,00$

**Reguła 8.**  $0,00 \wedge 0,75 = 0,00$

**Reguła 9.**  $0,00 \wedge 0,25 = 0,00$

Zatem niezerowy stopień spełnienia poprzednika mają reguły nr 2, 3, 5 i 6. Kolejnym krokiem w działaniu sterownika jest zsumowanie zbiorów rozmytych będących wynikiem każdej z reguł zgodnie z algorytmem postępowania w przypadku działania sterownika Mamdaniego omawianym w rozdziale 2.6.2. Wynikowy zbiór wyjściowy został zaprezentowany na rysunku 3.6. Na koniec ten zbiór należy jeszcze wyostrzyć proponowaną w rozprawie metodą defuzyfikacji, czyli w tym przypadku COG — wartość po wyostrzeniu tego zbioru wynosi w przybliżeniu 0,43. Zatem zgodnie z prezentowaną strategią adaptacji, student otrzyma test z pytaniami, które zostały wykorzystane we wcześniejszych edycjach kursu, ponieważ otrzymana wartość jest niższa od 0,6.



**Rysunek 3.6.** Zbiór wyjściowy będący sumą zbiorów powstałych w oparciu o bazę reguł w proponowanym sterowniku

Zaprezentowany powyżej sterownik został wykorzystany w ramach implementacji zaproponowanej strategii adaptacji, w której w zależności od poziomu zjawiska disengagement, student będzie kierowany do testu zaliczeniowego z pytaniami, które były wykorzystywane we wcześniejszych edycjach kursu lub do testu z nowymi zestawami pytań.

### 3.5. IMPLEMENTACJA KURSU Z ADAPTACYJNĄ ZAWARTOŚCIĄ WYKORZYSTUJĄCEGO ZJAWISKO DISENGAGEMENT Z WYKORZYSTANIEM EKSPERCKIEGO STEROWNIKA ROZMYTEGO

Zaprezentowana w rozdziale 3.3 strategia dydaktyczna oparta na zjawisku disengagement modelowanym poprzez sterownik rozmyty Mamdaniego zaprezentowany w rozdziale 3.4 została wdrożona w kursie „Zastosowania zbiorów rozmytych”. W rozdziale zostanie

omówiony proces implementacji kursu z modułem adaptacji treści z wykorzystaniem architektury adaptacyjnego kursu e-learningowego wprowadzonej w rozdziale .

### 3.5.1. Wykorzystanie eksperckiego sterownika rozmytego

Wprowadzony w rozdziale 3.3 sterownik rozmyty obliczający poziom zjawiska disengagement został zapisany w języku FCL [41] a następnie przekonwertowany do języka JavaScript przy pomocy opracowanego wcześniej modułu *fcl2js* biblioteki jFuzzyLogic [77, 103] (zob. rozdział 3.1.2). Omówiony w poprzednim rozdziale sterownik został zapisany w postaci kodu języka FCL, co zostało zaprezentowane na wyciągu 3.2.

```
1 FUNCTION_BLOCK disengagementFCL
2   VAR_INPUT
3     learning_quality : REAL;
4     remaining_time : REAL;
5   END_VAR
6
7   VAR_OUTPUT
8     disengagement : REAL;
9   END_VAR
10
11  FUZZIFY learning_quality
12    TERM low := (0, 1) (0.2, 1) (0.4, 0);
13    TERM medium := (0.2, 0) (0.4, 1) (0.6, 1) (0.8, 0);
14    TERM high := (0.6, 0) (0.8, 1) (1, 1);
15  END_FUZZIFY
16
17  FUZZIFY remaining_time
18    TERM low := (0, 1) (0.2, 1) (0.4, 0);
19    TERM medium := (0.2, 0) (0.4, 1) (0.6, 1) (0.8, 0);
20    TERM high := (0.6, 0) (0.8, 1) (1, 1);
21  END_FUZZIFY
22
23  DEFUZZIFY disengagement
24    TERM low := (0, 1) (0.2, 1) (0.4, 0);
25    TERM medium := (0.2, 0) (0.4, 1) (0.6, 1) (0.8, 0);
26    TERM high := (0.6, 0) (0.8, 1) (1, 1);
27
28    METHOD : COG;
29    DEFAULT := 0.5;
30  END_DEFUZZIFY
31
32  RULEBLOCK No1
33    AND : MIN;
34    ACT : MIN;
35    ACCU : MAX;
36
37    RULE 1 : IF learning_quality IS low AND remaining_time IS low
38      THEN disengagement IS high;
```

```
39     RULE 2 : IF learning_quality IS low AND remaining_time IS medium
40         THEN disengagement IS medium;
41     RULE 3 : IF learning_quality IS low AND remaining_time IS high
42         THEN disengagement IS medium;
43     RULE 4 : IF learning_quality IS medium AND remaining_time IS low
44         THEN disengagement IS medium;
45     RULE 5 : IF learning_quality IS medium AND remaining_time IS medium
46         THEN disengagement IS medium;
47     RULE 6 : IF learning_quality IS medium AND remaining_time IS high
48         THEN disengagement IS low;
49     RULE 7 : IF learning_quality IS high AND remaining_time IS low
50         THEN disengagement IS medium;
51     RULE 8 : IF learning_quality IS high AND remaining_time IS medium
52         THEN disengagement IS low;
53     RULE 9 : IF learning_quality IS high AND remaining_time IS high
54         THEN disengagement IS low;
55     END_RULEBLOCK
56 END_FUNCTION_BLOCK
```

### Wyciąg 3.2. Ekspercki sterownik Mamdaniego zapisany w języku FCL

Linie 2-9 prezentowanego kodu stanowią deklaracje zmiennych wykorzystywanych w sterowniku. Zatem dwie zmienne wejściowe - jakość nauki (*learning\_quality*) oraz czas pozostały do zamknięcia dostępu do testu zaliczeniowego (*remaining\_time*) - a także zmienna wyjściowa, czyli poziom zjawiska *disengagement*. Linie 11-26 stanowią definicję bloku rozmywania sterownika, czyli informacje o tym, jakie wartości lingwistyczne przyjmują zmienne oraz jakie są funkcje przynależności zbiorów rozmytych będących interpretacjami tych wartości (w tym wypadku są to funkcje przedziałami liniowe a dokładniej mówiąc funkcje trapezoidalne ograniczone prostymi  $x = 0$ ,  $y = 0$  i  $y = 1$ ). Linia 28 wskazuje metodę defuzyfikacji zmiennej wyjściowej (w prezentowanym sterowniku jest to metoda środka ciężkości - COG). W linii 29 określono wartość domyślną zmiennej wyjściowej wykorzystywaną w przypadku, gdy żadna z reguł sterownika nie zostałaby spełniona. Linie 33-35 określają sposób realizacji operacji koniunkcji i metody agregacji zbiorów będących wynikiem działania każdej reguły. Linie 37-54 to baza reguł - każda reguła postaci: *IF zmienna\_wejsciova1 IS wartosc1 AND zmienna\_wejsciova2 IS wartosc2 THEN zmienna\_wyjsciowa IS wartosc3*.

Aby zbudowany sterownik osadzić w kursie e-learningowym tak przygotowany kod FCL sterownika jest następnie konwertowany z wykorzystaniem modułu *fcl2js* biblioteki *jFuzzyLogic* do postaci skryptu języka JavaScript, który osadza się w kursie. Pełny kod sterownika przekonwertowanego do języka JavaScript znajduje się w załączniku A.

W proponowanym podejściu sterownik w postaci kodu języka JavaScript jest uruchamiany bezpośrednio w przeglądarce studenta podczas jego pracy z kursem. Jest to realizowane zgodnie z architekturą kursu zaproponowaną w rozdziale 3.5 — sterownik rozmyty zapisany w języku JavaScript stanowi zewnętrzny moduł obliczający poziom zjawiska *disengagement*.

### 3.5.2. Realizacja architektury kursu e-learningowego z adaptacyjną zawartością

W kursie „Zastosowania zbiorów rozmytych” zastosowano prezentowaną w rozdziale 3.3 strategię adaptacji wykorzystującą sterownik rozmyty. Zgodnie z algorytmem adaptacji treści (zob. rysunek 3.4), o tym, jaki test (tzn. z jakim zestawem pytań) dostarczyć studentowi w kursie, decydują reguły widoczności komponentów bazujące na obliczonym przez sterownik rozmyty poziomie zjawiska disengagement, a także na informacji, jaki jest aktualny stan ukończenia danego testu. Ta właśnie informacja jest istotna, ponieważ nie może dojść do sytuacji, w której student straci dostęp do już rozwiązanego testu. Taka sytuacja może się zdarzyć w momencie, gdy student rozwiąże test a później wróci do pracy z materiałami. Przy ponownej pracy z materiałami stan kursu zmienia się, ponieważ student więcej czasu spędzi w całym kursie i może także wykonać większą liczbę ćwiczeń, co ma wpływ na działanie prezentowanego w rozprawie sterownika rozmytego.

Przy ponownym sprawdzeniu, które elementy kursu powinny być aktualnie widoczne, zostaną sprawdzone reguły widoczności w oparciu o poziom zjawiska disengagement. Po przeliczeniu jego wartości, uwzględniając aktualny postęp studenta, może się okazać, że poziom disengagement spadł poniżej średniowysokiego. W takiej sytuacji student, który wcześniej miał dostęp do testu przeznaczanego dla osób z wysokim poziomem disengagement, może stracić do niego dostęp. Zamiast tego kurs może udostępnić wersję testu dla osób o niskim poziomie disengagement, co oznacza, że student zobaczy komunikat sugerujący, że test nie został jeszcze rozwiązany. Aby zapobiec takim sytuacjom, reguły widoczności bazujące na disengagement zostały rozszerzone o warunek sprawdzający status ukończenia testu. Zatem dla komponentu zawierającego test z pytaniami, które były wykorzystywane wcześniej, została zbudowana reguła widoczności postaci prezentowanej na wyciągu 3.3 — narzędzie Eduexe umożliwia budowanie złożonych reguł w alternatywnej postaci normalnej):

```
1  [
2      (disengagement < 0.6)
3  AND
4      (Test_new_questions.completion_status == unknown)
5  ]
6  OR
7  [
8      (Test_old_questions.completion_status == completed)
9  AND
10     (Test_new_questions.completion_status == unknown)
11 ]
12 OR
13 [
14     (disengagement < 0.6)
15 AND
16     (Test_new_questions.completion_status == not_attempted)
17 ]
18 OR
19 [
```

```

20 |     (Test_old_questions.completion_status == completed)
21 |     AND
22 |     (Test_new_questions.completion_status == not_attempted)
23 | ]

```

### Wyciąg 3.3. Reguła widoczności testu z pytaniami dla osób o niższym niż średniowysoki poziomie zjawiska disengagement

Reguła przedstawiona na wyciągu 3.3 uwzględnia stany ukończenia (o zbiorze wartości: *not\_attempted* — student nie podszedł do testu; *incomplete* — student aktualnie rozwiązuje test; *completed* — student rozwiązał test) zarówno testu ze wcześniej wykorzystywanymi pytaniami (disengagement niższy niż średniowysoki) — *Test\_old\_questions* — jak i testu z nowymi pytaniami (disengagement co najmniej średniowysoki) — *Test\_new\_questions*. O analogiczną regułę widoczności musi być także uzupełniony test dostarczany osobom, które mają co najmniej średniowysoki poziom zjawiska disengagement. Wartość zmiennej *disengagement* jest obliczana przez sterownik rozmyty przekonwertowany do postaci kodu języka JavaScript, i osadzony w kursie w ramach komunikacji kurs – adapter – sterownik zgodnie z zaproponowaną ogólną architekturą kursu z adaptacyjną zawartością (patrz rozdział 3.5). Zatem poza implementacją sterownika należy jeszcze odpowiednio przygotować adapter, aby mógł być osadzony w kursie zgodnie z zaproponowaną architekturą.

Zgodnie z architekturą kursu z adaptacyjną zawartością przedstawioną w rozdziale 3.5, działanie adaptera polega na nadpisaniu domyślnych definicji metod metod pozwalających na obliczenie niestandardowych zmiennych wykorzystywanych w regułach widoczności, zbudowanych z wykorzystaniem wbudowanych mechanizmów narzędzia, w którym zbudowano kurs tzn. mechanizmów narzędzia Eduexe [44]. Analogicznie jak pokazano w przykładzie 3.1 przedstawionym w rozdziale 3.5, w logice kursu zostały zdefiniowane do tego celu dwa obiekty — definiujące listę niestandardowych zmiennych oraz ich wartości początkowe, a także dwie metody — jedna zwraca informację o maksymalnej wartości, jaką może przyjąć zmienna, natomiast druga oblicza wynik danego studenta w obrębie danej zmiennej. Kod źródłowy adaptera wykorzystanego w kursie stanowi wyciąg 3.4. Tak zbudowany sterownik osadzony jest w kursie adaptacyjnym z wykorzystaniem mechanizmów technicznych narzędzia Eduexe.

Linie nr 1-13 prezentowanego kodu stanowią określenie obiektu klasy implementującej część logiki działania kursu w obszarze reguł widoczności komponentów. Dzięki temu będzie możliwa komunikacja z kursem. Definiowany jest także obiekt klasy będącej implementacją sterownika rozmytego — dzięki temu możliwa jest komunikacja ze sterownikiem. Natomiast samo obliczenie wartości zmiennej *disengagement* odbywa się poprzez wywołanie metody z kodu sterownika (linia nr 17).

```

1 | var tagsAndRulesRef = null;
2 | var controllerjs = null;
3 |
4 | if (window.parent.EduexeBrowserCore == undefined) {
5 |     tagsAndRulesRef = window.TagsAndRules;
6 |     controllerjs = window.ControllerJS;

```



```
7 } else {
8     tagsAndRulesRef = window.parent.TagsAndRules;
9     controllerjs = window.parent.ControllerJS;
10 }
11
12 tagsAndRulesRef.customCalculatedTagsValues = [0];
13 tagsAndRulesRef.customCalculatedTags = ["disengagement"];
14
15 tagsAndRulesRef.getCustomCalculatedTagCurrentValue = function(tag) {
16     if (tag == "disengagement") {
17         return controllerjs.runControllerAndGetResult();
18     }
19
20     return 0;
21 }
22
23 tagsAndRulesRef.getCustomCalculatedTagMaxValue = function(tag) {
24     if (tag == "disengagement") {
25         return 1;
26     }
27     return 1;
28 }
```

**Wyciąg 3.4.** Kod źródłowy adaptera pozwalającego na komunikację pomiędzy kursem a sterownikiem rozmytym

Poza realizacją komunikacji kurs – adapter – sterownik konieczne było także opracowanie logiki śledzenia postępów studenta w kursie na potrzeby późniejszego dostępu do informacji o rozwiązanych ćwiczeniach w momencie uruchomienia sterownika rozmytego. Częściowo realizuje to logika działania kursu oferowana przez wykorzystane narzędzie Eduexe, jednakże narzędzie nie udostępnia możliwości śledzenia aktywności takich jak interakcje studenta z ćwiczeniami w materiałach do nauki. W trakcie prac implementacyjnych autor przyjął rozwiązanie, w którym informacja o takich aktywnościach jest składową zmiennej wejściowej *learning\_quality* w sterowniku rozmytym. W tym celu w kursie „Zastosowania zbiorów rozmytych” osadzono bibliotekę *its-student-progress* (zob. rozdział 3.1.3), która została zbudowana, aby śledzić postępy studentów w obrębie interakcji z ćwiczeniami dostępnymi w kursie.

Tak przygotowany kurs z modułem adaptacji treści w oparciu o poziom zjawiska disengagement obliczany przy pomocy sterownika rozmytego zaproponowanego w rozdziale 3.4 został wdrożony na zajęciach dydaktycznych ze sztucznej inteligencji. Dane zebrane w tym kursie posłużyły później w procesie automatycznego generowania sterowników oraz przy porównywaniu ich z metodami uczenia maszynowego.



## ROZDZIAŁ 4

# Sterowanie rozmyte a inne metody klasyfikacji studentów

W rozdziale 3 zaprezentowano strategię adaptacji treści w kursie e-learningowym w oparciu o zjawisko disengagement modelowane przy pomocy eksperckiego sterownika rozmytego. Strategia ta wraz ze sterownikiem została wdrożona w kursie e-learningowym a sam kurs wykorzystany na zajęciach dydaktycznych po to, aby pozyskać rzeczywiste dane o postępach studentów w tym kursie. Sterowniki rozmyte mogą być także generowane na podstawie zbioru danych [48, 57, 64, 75]. W ramach wdrożonego kursu „Zastosowania zbiorów rozmytych” z modułem adaptacji treści zebrano dane o postępach studentów w tym kursie. Dane te zostały przetworzone przy użyciu biblioteki *its-raport-generator* (zob. rozdział 3.1.4) a następnie poanotowane pod kątem występowania zjawiska disengagement przez grupę anotatorów. Tak przygotowane dane posłużyły do wygenerowania nowych sterowników rozmytych. Celem tych prac było porównanie sterowników wygenerowanych automatycznie ze sterownikiem eksperckim wprowadzonym w rozdziale 3.4, który nie był przygotowywany w oparciu o żadne zbiory danych a także z najpopularniejszymi metodami klasyfikacji wykorzystywanymi w uczeniu maszynowym. W niniejszym rozdziale będzie przedstawiony proces przygotowania danych, generowania sterowników Mamdaniego i Takagi-Sugeno opartych na danych a następnie zostanie przeprowadzone porównanie wybranych metod klasyfikacji ze sterownikami rozmytymi.

### 4.1. PRZYGOTOWANIE DANYCH

Przygotowanie danych do generowania sterowników rozmytych oraz trenowania algorytmów uczenia maszynowego w kontekście dydaktycznym wymaga pozyskania danych z rzeczywistej interakcji studentów z kursem podczas nauki i nadania tym danym interpretacji dydaktycznej. W przypadku danych zebranych w ramach pracy studentów z kursem e-learningowym „Zastosowania zbiorów rozmytych”, przyjęto założenie, że aby sterowniki oraz klasyfikatory wykorzystywane w uczeniu maszynowym mogły być porównywane później ze sterownikiem eksperckim (zob. rozdział 3.4), to cechy opisujące pracę studenta w kursie, w oparciu o które będą przygotowywane sterowniki i trenowane algorytmy, powinny być takie same jak w przypadku informacji wykorzystywanych w sterowniku eksperckim (patrz rozdział 3.4.1). Zatem wykorzystano następujące informacje:

- jakość nauki, na którą składają się:
  - średni czas spędzony w jednym elemencie kursu,
  - liczba rozwiązanych ćwiczeń w kursie,
  - liczba elementów, których student nie odwiedził,
- czas pozostały do zamknięcia testu zaliczeniowego, liczony w momencie pierwszego wejścia do kursu.

Zadaniem sterowników generowanych na podstawie zbioru danych i algorytmów uczenia maszynowego jest także jak w przypadku sterownika eksperckiego, zamodelowanie zjawiska disengagement. Konieczne jest zatem opisanie każdego rekordu zbioru danych informacją, czy dany student miał znaczący poziom zjawiska disengagement czy też nie. Aby to zrobić niezbędni są anotatorzy, którzy mają doświadczenie dydaktyczne, czyli nauczyciele. Ich zadaniem będzie podjęcie decyzji dla każdego pojedynczego przypadku, jak wysoki był poziom zjawiska disengagement dla każdego studenta uczestniczącego w kursie.

#### 4.1.1. Wdrożenie kursu na zajęciach dydaktycznych i zbieranie danych z kursu

W edycji kursu „Zastosowania zbiorów rozmytych”, w której wykorzystano wprowadzony w rozdziale 3.4 sterownik ekspercki, zapisanych było 97 studentów. Kurs składał się z 7 rozdziałów a każdy z nich kończył się testem zaliczeniowym (zob. rozdział 3.1.1), co daje 679 sytuacji, w których sterownik mógł zostać uruchomiony. Niektórzy studenci jednak nie podeszli do kursu a część z nich zrezygnowała z nauki w trakcie, a co za tym idzie sterownik został uruchomiony mniejszą liczbą razy — 479. Tabela 4.1 przedstawia reprezentatywną próbkę zebranych danych z tej edycji kursu, reprezentujących różne wzorce pracy studentów. Informacje przedstawione w tabeli, to cechy wykorzystywane na wejściu sterownika (zob. rozdział 3.4.1).

Pierwsza kolumna prezentowanej tabeli to znormalizowany średni czas spędzony w jednym elemencie kursu, druga to znormalizowana liczba odwiedzonych elementów kursu, trzecia to znormalizowana liczba interakcji, czyli ćwiczeń, które rozwiązał student. Kolejna kolumna to jakość nauki, czyli agregacja tych wcześniejszych danych zgodnie ze wzorem prezentowanym w rozdziale 3.4.1. Piąta kolumna oznacza znormalizowany czas pozostały do zamknięcia testu zaliczeniowego, liczony w momencie pierwszego wejścia do kursu. Przedostatnia kolumna prezentuje wartość obliczoną przez sterownik ekspercki — nasilenie zjawiska disengagement. Ostatnia kolumna informuje o zestawie pytań w teście zaliczeniowym, jaki został przydzielony studentowi w oparciu o prezentowaną w rozdziale 3.3 strategią adaptacji treści i poziom zjawiska disengagement — jeśli sterownik zwrócił wartość disengagement na poziomie 0,6 lub wyższą, to student otrzymywał test z pytaniami z nowego zestawu (wartość w tabeli: *nowy*). Pozostali studenci otrzymywali pytania z zestawów, które były wykorzystane w poprzednich edycjach kursu (wartość w tabeli: *stary*).

W trakcie projektowania eksperckiego sterownika rozmytego przyjęto założenie, że niewielka aktywność studentów w kursie stanowi sygnał niedostatecznego zaangażowania w proces nauki. Brak aktywności może także wskazywać na ograniczoną samodzielność w działaniu, na przykład skłonność do podejścia do testów mimo posia-

dania wcześniejszej wiedzy na temat pytań. Oczywiście, brak aktywności może mieć różnorodne przyczyny, takie jak choroba czy po prostu brak zainteresowania danym przedmiotem.

Z innej strony, zastosowanie sterownika rozmytego w ramach kursu mogło przyczynić się do zwiększenia zaangażowania niektórych studentów. Po przystąpieniu do pierwszego testu studenci mogli zauważyć, że system wybiera inne pytania niż te, które wcześniej pojawiły się u innych studentów zdających test wcześniej. W efekcie taka obserwacja mogła skłonić niektórych z nich do większego wysiłku w nauce przed kolejnymi testami w trakcie trwania kursu.

<i>Czas w elemencie</i>	<i>Elementy nie-odwiedzone</i>	<i>Interakcje</i>	<i>Jakość nauki</i>	<i>Czas do zamknięcia testu</i>	<i>Disengagement</i>	<i>Zestaw pytań w teście</i>
0,100	0,000	0,280	0,160	0,155	0,844	nowy
0,100	0,000	0,000	0,067	0,214	0,804	nowy
0,300	0,000	0,111	0,237	0,162	0,744	nowy
0,200	0,000	0,500	0,300	0,227	0,624	nowy
0,025	0,333	0,500	0,072	0,266	0,683	nowy
0,175	0,000	0,520	0,290	0,104	0,640	nowy
0,150	0,000	0,632	0,311	0,139	0,608	nowy
0,225	0,000	0,857	0,352	0,295	0,599	stary
0,025	0,500	1,000	0,183	0,360	0,544	stary
0,525	0,200	1,000	0,617	0,627	0,472	stary
0,600	0,000	0,920	0,707	0,514	0,365	stary
1,000	0,000	0,286	0,762	0,449	0,258	stary
0,675	0,000	1,000	0,783	0,779	0,206	stary
0,600	0,000	0,857	0,686	0,973	0,173	stary
0,950	0,000	1,000	0,967	0,971	0,155	stary

**Tabela 4.1.** Wybrane wyniki działania eksperckiego sterownika Mamdaniego

Po zakończeniu pracy studentów w kursie pobrano dwa raporty z systemu Edumatic ITS [45], w którym ten kurs został osadzony — raport postępów studenta w poszczególnych elementach kursu oraz raport prezentujący aktywności studenta we wszystkich ćwiczeniach, które były przygotowane w materiałach do nauki. Sam kurs „Zastosowania zbiorów rozmytych” był podzielony na siedem części (zob. opis kursu w rozdziale 3.1.1) — każda z nich stanowiła osobną jednostkę do nauki w systemie — dla każdej części pobrane zostały osobne raporty. Na pobrany raport składa się wiele plików CSV - po jed-

nym pliku dla każdego studenta. Ponieważ było zapisanych 97 studentów, to łącznie pobrano z systemu 194 pliki CSV na każdą z siedmiu części kursu — łącznie 1358 plików.

Na raport postępów studenta w poszczególnych elementach (obiektach SCO) kursu (czyli w tym wypadku konkretnej części kursu „Zastosowania zbiorów rozmytych”) składa się 25 kolumn:

1. Nazwa SCO
2. Blok/SCO — informacja, czy dany obiekt SCO jest jednostką do nauki, czy grupuje kilka obiektów SCO w logiczną całość (blok)
3. Identyfikator — identyfikator obiektu SCO
4. Identyfikator przodka — identyfikator bloku, w którym znajduje się SCO
5. Czas spędzony w SCO — łączny czas spędzony przez studenta w SCO
6. Czas bezczynności — łączny czas bezczynności w SCO, tzn. momentów, w których student nie wykonywał żadnych czynności w kursie/kurs miał otwarty w innej karcie/oknie przeglądarki
7. Liczba wejść (session) — liczba wszystkich wejść do SCO
8. Liczba prób (attempt) — liczba wszystkich podejść (czyli pracy od początku) - w kursie było dostępne jedno podejście dla każdego obiektu SCO z możliwością przerywania (wiele wejść)
9. Historia wejść (attempts) — obiekt JSON z listą dat wejść do SCO
10. Historia wejść (wsad) — lista dat wejść do SCO w postaci surowej, tzn. w formacie zapisywanym przez system
11. Pierwsze wejście — data pierwszego wejścia do SCO
12. Ostatnie wejście — data ostatniego wejścia do SCO
13. Tryb testu (cmi.mode) — stan obiektu SCO (jeżeli obiekt zawierał test, to możliwe jest sterowanie stanem testu — np. możliwe było ponowne otwarcie dostępu do testu w uzasadnionych przypadkach)
14. Status ukończenia (cmi.completion\_status) — status ukończenia pracy w obiekcie SCO (wg standardu SCORM 2004: *unknown* — nieznan; *not attempted* — nierozpoczęty; *incomplete* — w trakcie pracy; *completed* - zakończony)
15. Status ukończenia ustawiono na zakończony (cmi.completion\_status = completed) — data uzyskania przez obiekt SCO wartości *completed* dla stanu ukończenia
16. Próg ukończenia cmi (cmi.completion\_threshold) — dodatkowy warunek osiągnięcia stanu *completed* dla statusu ukończenia — w tym kursie nieustawiony
17. Miara postępu (cmi.progress\_measure) — miara opisująca sposób obliczania postępu w obiekcie SCO — w tym kursie nieustawiona
18. Status zaliczenia (cmi.success\_status) — status zaliczenia obiektu SCO — istotne w przypadku testów (wg standardu SCORM 2004 stany: *unknown* — nieznan; *failed* — niezaliczony; *passed* — zaliczony)
19. Próg zaliczenia (cmi.scaled\_passing\_score) — znormalizowana liczba punktów, jaką trzeba zdobyć w SCO, aby stan zaliczenia osiągnął wartość *passed*
20. Punkty znormalizowane (cmi.score.scaled) — znormalizowana liczba punktów zdobytych przez studenta w SCO
21. Punkty (cmi.score.raw) — liczba punktów zdobytych przez studenta w SCO

22. Wartość minimalna punktów (cmi.score.min) — minimalna liczba punktów, którą można uzyskać w SCO
23. Wartość maksymalna punktów (cmi.score.max) — maksymalna liczba punktów, którą można uzyskać w SCO
24. Dane tymczasowe (cmi.suspend\_data) — obiekt JSON zapisywany przez kurs, zawierający dodatkowe informacje, które normalnie nie są zapisywane w SCORM 2004
25. Komentarz do testu (cmi.comments\_from\_lms) — student może otrzymać informację zwrotną od nauczyciela — nie wykorzystane w kursie

Każdy wiersz w raporcie odpowiada jednemu obiektowi SCO. Cały kurs „Zastosowania zbiorów rozmytych” zawierał 39 obiektów SCO, zatem w jednym pliku CSV w zależności od postępów studenta znajdowało się do 39 rekordów. Co daje dla wszystkich studentów maksymalnie 3783 rekordów.

Na raport prezentujący aktywności studenta we wszystkich ćwiczeniach (interakcjach), które były przygotowane w materiałach do nauki natomiast składa się 16 kolumn:

1. Nazwa SCO
2. Identyfikator — identyfikator obiektu SCO, w którym znajduje się dana interakcja/ćwiczenie
3. Punkty (cmi.score.raw) — liczba punktów osiągnięta po rozwiązaniu ćwiczenia
4. Punkty znormalizowane (cmi.score.scaled) — znormalizowana liczba punktów osiągnięta po rozwiązaniu ćwiczenia
5. Wartość maksymalna punktów (cmi.score.max) — maksymalna liczba punktów, jaką można było uzyskać po rozwiązaniu ćwiczenia
6. Id interakcji (cmi.interactions.n.id) — identyfikator interakcji/ćwiczenia
7. Typ interakcji (cmi.interactions.n.type) — typ ćwiczenia (np. *choice* — pytanie wyboru, *fill-in* — pytanie krótkiej odpowiedzi czy *matching* — zadanie na dopasowania)
8. Objectives (cmi.interactions.n.objectives) — dodatkowe informacje o postępach w ramach interakcji zapisywane przez kurs (np. historia pracy studenta z danym ćwiczeniem)
9. Czas uzyskania dostępu do interakcji (cmi.interactions.n.timestamp) — informacja o tym, kiedy student pierwszy raz uzyskał dostęp do ćwiczenia
10. Poprawne odpowiedzi (cmi.interactions.n.correct\_responses) — wzorzec prawidłowych odpowiedzi
11. Waga interakcji (cmi.interactions.n.weighting) — możliwe jest zdecydowanie, czy jakieś ćwiczenia są ważniejsze od innych — nie wykorzystane w kursie
12. Odpowiedź studenta (cmi.interactions.n.learner\_response) — odpowiedź studenta w ramach ćwiczenia
13. Rezultat (cmi.interactions.n.result) — wynik studenta (wg standardu SCORM 2004: *correct* — rozwiązano prawidłowo; *incorrect* — rozwiązano nieprawidłowo; *real* — rozwiązano częściowo prawidłowo)
14. Punkty (jeżeli rezultat = real) — jeżeli odpowiedź studenta jest częściowo prawidłowa, to określa się procentowo, jak bardzo rozwiązanie jest bliskie poprawnej odpowiedzi

15. Czas pierwszego wyboru odpowiedzi (cmi.interactions.n.latency) — data pierwszego podejścia do ćwiczenia
16. Opis interakcji (cmi.interactions.n.description) — treść pytania w ramach ćwiczenia

Każdy wiersz w raporcie odpowiada jednej interakcji/ćwiczeniu. Cały kurs zawierał 87 ćwiczeń, co daje dla wszystkich studentów maksymalnie 8439 rekordów.

Prezentowane raporty zawierają dużo różnych danych. Aby uzyskać zbiór wyłącznie informacji potrzebnych do generowania sterowników rozmytych, pobrane dane zostały poddane przetworzeniu. Wykorzystano w tym celu bibliotekę *its-report-generator* (zob. rozdział 3.1.4). Schemat postępowania był następujący:

1. Pobrać raporty z systemu
2. Umieścić raporty w katalogu z plikami biblioteki
3. Opracować plik JSON z opisem kursu (zgodnie z dokumentacją biblioteki)
4. Uruchomić skrypt generujący raporty biblioteki (skrypt najpierw przetwarza pliki wejściowe pod kątem ewentualnych problemów z kodowaniem znaków a następnie generuje zbiorcze raporty ze znormalizowanymi metodą min-max danymi)
5. Wybrać z wygenerowanych raportów potrzebne kolumny

W ten sposób uzyskano plik CSV wyłącznie z informacjami niezbędnymi do wygenerowania sterowników i wytrenowania algorytmów uczenia maszynowego, czyli dane, których fragment zaprezentowano w pierwszej, drugiej, trzeciej i piątej kolumnie tabeli 4.1 — czwarta kolumna, czyli jakość nauki to agregacja danych z pierwszych trzech kolumn zgodnie ze wzorem przedstawionym w rozdziale 3.4.1. Jeden rekord w tym zbiorze danych oznaczał jednego studenta, który rozpoczął pracę w danej części kursu. Zebrano 879 rekordów. Zebrane dane posłużyły do wygenerowania sterowników oraz do porównania ich działania z metodami uczenia maszynowego.

#### 4.1.2. Anotowanie pozyskanych danych dydaktycznych

W oparciu o zebrane dane z kursu „Zastosowania zbiorów rozmytych” zostały wygenerowane nowe sterowniki rozmyte a także w oparciu o te dane zostały wytrenowane najpopularniejsze algorytmy klasyfikujące wykorzystywane w uczeniu maszynowym w celu porównania wszystkich tych metod klasyfikacji studentów pod kątem zjawiska disengagement. Aby było to możliwe, opracowane dane podlegały procesowi anotacji. W ramach realizacji tego zadania przyjęto, że należy otrzymać zbiory pozwalające na klasyfikację binarną (czyli tak, jak wykorzystano to w prezentowanej w rozprawie strategii dydaktycznej — student otrzymywał jedną z dwóch wersji testu zaliczeniowego w oparciu o disengagement — zob. rozdział 3.3) oraz klasyfikację pięciostopniową w oparciu o skalę Likerta. Wybrano klasyfikację pięciostopniową, aby możliwe było bardziej szczegółowe opisanie zjawiska disengagement na skali o wartościach: *bardzo niski, niski, umiarkowany/średni, wysoki, bardzo wysoki* (zob. opis zjawiska disengagement w rozdziale 3.3).



Każdy rekord zbioru danych należało opisać informacją, jaki był poziom zjawiska disengagement w przypadku danego studenta i danej części kursu „Zastosowania zbiorów rozmytych”. Pierwszą próbą było wykonanie tego zadania przez autora rozprawy. Rekordy były opisywane jedną z dwóch etykiet: *u studenta występuje znaczący disengagement* (wartość 1)/*u studenta nie występuje znaczący disengagement* (wartość 0). Zatem tak poanotowany zbiór danych był przygotowany do generowania sterowników rozmytych oraz trenowania algorytmów uczenia maszynowego w ramach klasyfikacji binarnej. Fragment tego zbioru przedstawia tabela 4.2. Pierwsze pięć kolumn oznacza kolejno te same informacje co kolumny w tabeli 4.1. W ostatniej kolumnie natomiast znajduje się przypisana wartość przez autora rozprawy — tak jak napisano powyżej 0 — kiedy autor uważał, że u danego studenta nie występuje znaczący disengagement oraz 1 — w przeciwnym wypadku. Cały zbiór danych zawiera 879 rekordów.

<i>Czas w elemencie</i>	<i>Elementy nieodwiedzone</i>	<i>Interakcje</i>	<i>Jakość nauki</i>	<i>Czas do zamknięcia testu</i>	<i>Przypisana wartość</i>
1	0	0,714285714	0,904761905	0,27434127	0
0,85	0	0,68	0,793333333	0,224833333	0
0	0	0	0	0,890097518	1
0,95	0	1	0,966666667	0,970998818	0
0,175	0	0,857142857	0,402380952	0,238793651	0
0	1	0	-0,333333333	0,433158983	1
0,45	0	1	0,633333333	0,10608156	0
0	0	0	0	0,491507937	1
0,975	0	0,777777778	0,909259259	0,481675532	0
0,3	0	0,44	0,346666667	0,137448413	0
0,15	0	1	0,433333333	0,637904846	0
0,8	0	0,944444444	0,848148148	0,641365248	0

**Tabela 4.2.** Fragment zbioru danych anotowanego przez autora rozprawy

Ten sam zbiór danych został później ponownie poanotowany przez 11 studentów piątego roku kierunku „Nauczanie matematyki i informatyki” w ramach zajęć z przedmiotu „Sztuczna inteligencja i jej zastosowania”. Studenci ci są nauczycielami i byli przedstawicielami docelowej grupy odpowiedzialnej za anotację zasobów tzn. byli dydaktykami rozumiejącymi zagadnienia zaangażowania lub jego braku w naukę. W ramach zadania ci studenci anotowali dane w oparciu o pięcioelementowy zbiór możliwych etykiet przypisanych zachowaniu danego studenta w kursie (a nie jak wcześniej dwóch etykiet - dzięki temu disengagement został opisany dokładniej).

Anotatorzy określali w skali Likerta poziom zaangażowania w pracę z kursem,

gdzie „1” oznaczało brak zaangażowania a „5” wysokie zaangażowanie. Tak zebrane dane zostały przetworzone tak, że odwrócono etykiety anotatorów po to, aby osiągnąć ocenę zjawiska disengagement (które jest bliższe brakowi zaangażowania; zatem skala: „1” — nie występuje disengagement lub występuje bardzo niski jego poziom; „5” — występuje bardzo wysoki disengagement). Dzięki tak poanotowanemu zbiorowi danych możliwa jest pięcioklasowa klasyfikacja studentów pod kątem występowania zjawiska disengagement w kursie.

W pierwszym etapie anotowania każdy anotator otrzymywał losowy podzbiór danych zawierający 100 rekordów. Za to zadanie otrzymywali oni punkty w ramach zajęć. Natomiast osoby, które chciały otrzymać więcej punktów, musiały dokonać anotacji całego zbioru danych. Otrzymano w ten sposób zbiór zawierający łącznie 4600 rekordów. Fragment otrzymanego zbioru danych przedstawia tabela 4.3. Pierwsze pięć kolumn oznacza kolejno te same informacje co kolumny w tabeli 4.2. W ostatniej kolumnie natomiast znajduje się przypisana wartość przez danego anotatora.

<i>Czas w elemencie</i>	<i>Elementy nieodwiedzone</i>	<i>Interakcje</i>	<i>Jakość nauki</i>	<i>Czas do zamknięcia testu</i>	<i>Przypisana wartość</i>
1	0	0,714285714	0,904761905	0,27434127	2
0,85	0	0,68	0,793333333	0,224833333	2
0	0	0	0	0,890097518	5
0,95	0	1	0,966666667	0,970998818	1
0,175	0	0,857142857	0,402380952	0,238793651	3
0	1	0	-0,333333333	0,433158983	5
0,45	0	1	0,633333333	0,10608156	3
0	0	0	0	0,491507937	5
0,975	0	0,777777778	0,909259259	0,481675532	2
0,3	0	0,44	0,346666667	0,137448413	3
0,15	0	1	0,433333333	0,637904846	3
0,8	0	0,944444444	0,848148148	0,641365248	1

**Tabela 4.3.** Fragment zbioru danych anotowanego przez grupę anotatorów po odwróceniu etykiet

Ponieważ było wielu anotatorów, to w danych mogły pojawiać się przypadki, gdy różne osoby anotowały to samo zachowanie konkretnego studenta. Każdy z anotatorów mógł inaczej takie zachowanie ocenić. W kontekście tej sytuacji przygotowano zostały dwie wersje otrzymanego zbioru danych — zbiór, w którym ocena anotatorów została uśredniona

oraz zbiór oryginalny, czyli taki, w którym dane nie zostały już dalej zmodyfikowane, tzn. zezwolono na powtórzenia w zbiorze danych z ryzykiem różnej oceny anotatorów.

Zbiór danych anotowany przez autora rozprawy zawierał 879 rekordów, natomiast zbiór anotowany przez grupę studentów-nauczycieli zawierał 4600 rekordów. Należy tutaj pamiętać, że studenci-nauczyciele anotowali dane ze zbioru anotowanego przez autora rozprawy, zatem dla zbioru z uśrednionymi danymi będzie również 879 rekordów.

Zgodnie z przyjętym celem, tzn. pozyskaniem zbioru do klasyfikacji binarnej, zbiory otrzymane w ramach procesu anotowania przez grupę studentów przetworzono do problemu klasyfikacji binarnej — etykiety „4” i „5” zastąpiono etykietą *u studenta występuje znaczący disengagement* a pozostałe etykiety zastąpiono etykietą *u studenta nie występuje znaczny disengagement*. Podział ten jest zgodny z koncepcją doboru punktu odcięcia studentów (tzn. oznaczenia średniowysokiego poziomu zjawiska disengagement, czyli etykiety „4” i „5”), którzy powinni otrzymać zestaw nowych pytań w teście zaliczeniowym, która była zaproponowana w rozdziale 3.3. Fragment otrzymanego zbioru danych przedstawia tabela 4.4. Pierwsze pięć kolumn oznacza kolejno te same informacje co kolumny w tabeli 4.3. W ostatniej kolumnie natomiast znajduje się przypisana wartość przez danego studenta przekonwertowana do klasyfikacji binarnej.

<i>Czas w elemencie</i>	<i>Elementy nieodwiedzone</i>	<i>Interakcje</i>	<i>Jakość nauki</i>	<i>Czas do zamknięcia testu</i>	<i>Przypisana wartość</i>
1	0	0,714285714	0,904761905	0,27434127	0
0,85	0	0,68	0,793333333	0,224833333	0
0	0	0	0	0,890097518	1
0,95	0	1	0,966666667	0,970998818	0
0,175	0	0,857142857	0,402380952	0,238793651	1
0	1	0	-0,333333333	0,433158983	1
0,45	0	1	0,633333333	0,10608156	1
0	0	0	0	0,491507937	1
0,975	0	0,777777778	0,909259259	0,481675532	0
0,3	0	0,44	0,346666667	0,137448413	1
0,15	0	1	0,433333333	0,637904846	1
0,8	0	0,944444444	0,848148148	0,641365248	0

**Tabela 4.4.** Fragment zbioru danych anotowanego przez grupę anotatorów po przetworzeniu do klasyfikacji binarnej

W wyniku anotowania zbiorów i ich przetwarzania otrzymano ostatecznie następujące zbiory danych:

1. Zbiór danych anotowanych przez autora rozprawy — 879 rekordów; klasyfikacja binarna
2. Zbiór danych anotowanych przez grupę anotatorów — 4600 rekordów; klasyfikacja binarna; rekordy powtarzają się z anotacjami różnych studentów-nauczycieli
3. Zbiór danych anotowanych przez grupę anotatorów — 4600 rekordów; klasyfikacja pięciostopniowa; rekordy powtarzają się z anotacjami różnych studentów-nauczycieli
4. Zbiór danych anotowanych przez grupę anotatorów — 879 rekordów; klasyfikacja binarna; rekordy nie powtarzają się a anotacje studentów-nauczycieli zostały uśrednione
5. Zbiór danych anotowanych przez grupę anotatorów — 879 rekordów; klasyfikacja pięciostopniowa; rekordy nie powtarzają się a anotacje studentów-nauczycieli zostały uśrednione

Zbiory te zostały wykorzystane w procesie generowania sterowników rozmytych a także przy trenowaniu algorytmów uczenia maszynowego oraz porównywaniu wszystkich metod klasyfikacji. Wszystkie zbiory zostały podzielone w stosunku 80:20 na zbiory do trenowania algorytmów uczenia maszynowego/generowania sterowników rozmytych oraz zbiory testowe wykorzystane później w ramach procesu porównywania wszystkich metod.

## 4.2. STEROWNIKI ROZMYTE WYGENEROWANE AUTOMATYCZNIE

Dla każdego ze zbiorów danych przedstawionych w poprzednim podrozdziale wygenerowano sterowniki Mamdaniego. Ponieważ zbiór danych anotowany przez autora rozprawy służył głównie testom, to kolejne sterowniki, tj. sterowniki Takagi-Sugeno zostały wygenerowane tylko na zbiorach otrzymanych z anotacji studentów-nauczycieli. Jednak wygenerowanie sterownika Mamdaniego na zbiorze danych anotowanych przez autora rozprawy było ważnym krokiem w realizacji zadania polegającego na wygenerowaniu różnych sterowników rozmytych, ponieważ dzięki temu zostały zidentyfikowane różne problemy (np. implementacyjne, czy dotyczące samej metody generowania sterowników), które mogły zostać rozwiązane przed wygenerowaniem kolejnych sterowników na zbiorach danych pozyskanych później. W efekcie generowania sterowników otrzymano pięć sterowników Mamdaniego i cztery sterowniki Takagi-Sugeno.

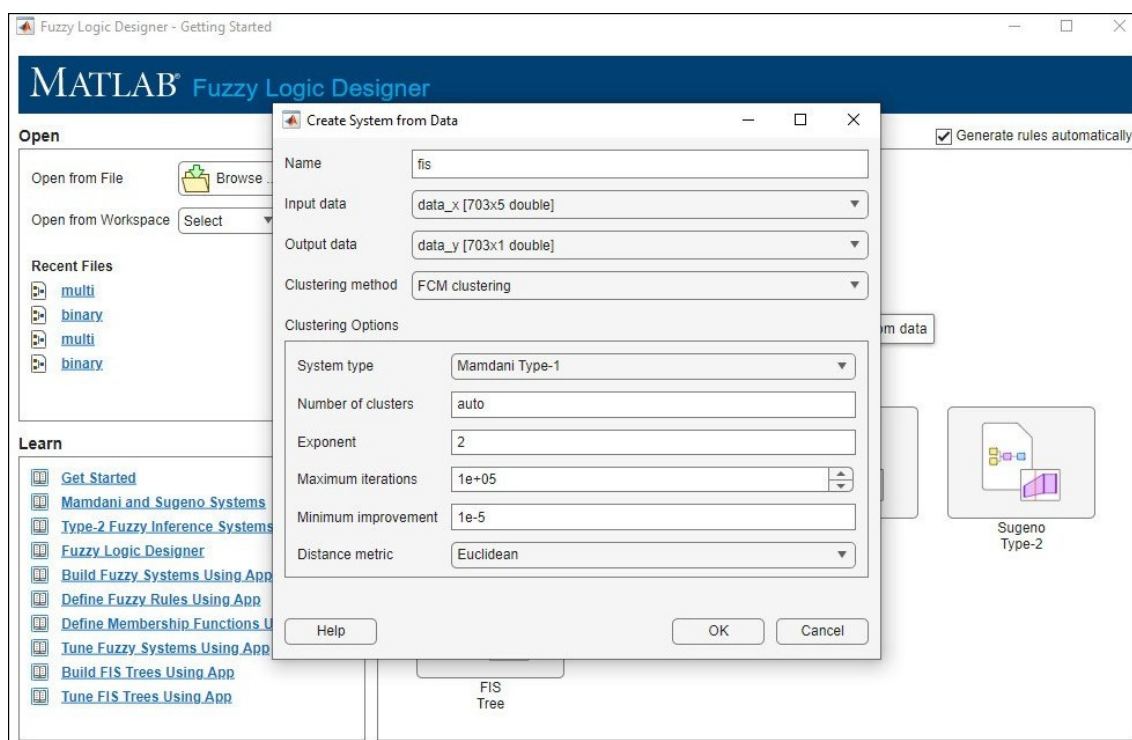
### 4.2.1. Sterowniki Mamdaniego

W przypadku sterowników Mamdaniego wykorzystano algorytm klastrowania Fuzzy C-Means (zob. rozdział 2.6.2) zaimplementowany w narzędziu Fuzzy Logic Designer [112]

dostępnym w środowisku MATLAB [111]. Sterowniki rozmyte zostały wygenerowane przy użyciu tego narzędzia z następującymi parametrami dla algorytmu Fuzzy C-Means:

- liczba klastrów  $c$  została ustawiona na wartość *auto* — liczba ta zostanie wyznaczona automatycznie metodą różnicową;
- parametr fuzyfikacji  $m$  wynosi dwa (zgodnie z algorytmem FCM musi być większy od jednego);
- maksymalna liczba iteracji w przypadku niespełnienia warunku stopu została ustawiona na 100000;
- odległość między próbką a centroidem jest obliczana poprzez odległość Euklidesową;
- próg  $\epsilon$  został ustawiony na wartość 0,00001.

Powyższe ustawienia zastosowane przy konfiguracji narzędzia Fuzzy Logic Designer przedstawia rysunek 4.1.



**Rysunek 4.1.** Konfigurowanie narzędzia Fuzzy Logic Designer oraz parametrów algorytmu FCM w środowisku MATLAB w celu wygenerowania sterownika

Wyciąg 4.1 przedstawia jeden ze sterowników w formie FIS [53] wygenerowanych w środowisku MATLAB na zbiorze danych anotowanych przez studentów-nauczycieli dla klasyfikacji binarnej, gdzie rekordy nie powtarzają się a anotacje studentów-nauczycieli zostały uśrednione. Zgodnie z budową sterownika w formie FIS, pierwsza część to informacje ogólne (linie nr 1-12) — wygenerowany sterownik zawiera cztery zmienne wejściowe (tyle było kolumn w zbiorze danych opisujących zachowanie studenta), jedna

zmienna wyjściowa (disengagement), 11 reguł (tyle klastrów wyznaczył algorytm różnicowy, po jednej regule na klaster), koniunkcja realizowana jest operacją minimum, implikacja również operacją minimum. Wyniki działania reguł są agregowane funkcją maksimum a wynikowy zbiór rozmyty wyostrzony metodą środka ciężkości.

Druga część to definicje zmiennych wejściowych (linie nr 14-76) — cztery zmienne wejściowe, każda ma 11 termów, których interpretacje to zbiory rozmyte z funkcją Gaussa (gaussmf) jako funkcją przynależności. W liniach 18-28 znajdują się interpretacje termów powstałych w ramach klastrowania, czyli funkcje przynależności zbiorów rozmytych. W linii nr 18 zdefiniowana jest interpretacja termu `in1cluster1` dla zmiennej `in1` — zbiór rozmyty o funkcji przynależności definiowanej wzorem:

$$\text{gaussmf}(x, \sigma, \mu) = e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

Analogicznie w kolejnych liniach definiowane są interpretacje dla kolejnych termów. Trzecia część sterownika w formacie FIS to definicje zmiennych wyjściowych — sterownik zawiera jedną zmienną wyjściową (linie nr 78-92) — definicja jest analogiczna jak dla zmiennych wejściowych. Baza reguł (linie nr 94-105) zawiera 11 reguł (po jednej na klaster). Reguły są postaci:

*IF in1 IS clusterK AND in2 IS clusterK AND in3 IS clusterK AND in4 IS clusterK THEN out1 IS clusterK*

gdzie *clusterK* oznacza K-ty klaster.

```

1  [System]
2  Name='disengagement_mamdani_from_matlab'
3  Type='mamdani'
4  Version=2.0
5  NumInputs=4
6  NumOutputs=1
7  NumRules=11
8  AndMethod='min'
9  OrMethod='max'
10 ImpMethod='min'
11 AggMethod='max'
12 DefuzzMethod='centroid'
13
14 [Input1]
15 Name='in1'
16 Range=[0 1]
17 NumMFs=11
18 MF1='in1cluster1':'gaussmf',[0.127344488632215 0.486365100300772]
19 MF2='in1cluster2':'gaussmf',[0.140221266161368 0.155262925821961]
20 MF3='in1cluster3':'gaussmf',[0.175305124714692 0.962176695738479]
21 MF4='in1cluster4':'gaussmf',[0.17564459707538 0.947673685608229]

```

```
22 MF5='in1cluster5':'gaussmf',[0.174013301029544 0.944208993236926]
23 MF6='in1cluster6':'gaussmf',[0.12669882582513 0.383151505069264]
24 MF7='in1cluster7':'gaussmf',[0.125350559025147 0.39120098329391]
25 MF8='in1cluster8':'gaussmf',[0.127319191646339 0.333455283503049]
26 MF9='in1cluster9':'gaussmf',[0.158427794193252 0.0288175139951597]
27 MF10='in1cluster10':'gaussmf',[0.131873800961864 0.618099917636989]
28 MF11='in1cluster11':'gaussmf',[0.145979748822329 0.115032833375835]
29
30 [Input2]
31 Name='in2'
32 Range=[0 1]
33 NumMFs=11
34 MF1='in2cluster1':'gaussmf',[0.0434158983393201 0.00944579159779164]
35 MF2='in2cluster2':'gaussmf',[0.0637780459251198 0.0570687363704181]
36 MF3='in2cluster3':'gaussmf',[0.0396608824522194 0.00231751838913958]
37 MF4='in2cluster4':'gaussmf',[0.0404283407283434 0.00310415404785033]
38 MF5='in2cluster5':'gaussmf',[0.0446049612488837 0.0132111230241344]
39 MF6='in2cluster6':'gaussmf',[0.0414700604921625 0.00333539633519379]
40 MF7='in2cluster7':'gaussmf',[0.0436938807109482 0.0068544680392823]
41 MF8='in2cluster8':'gaussmf',[0.040909746500398 0.0035773561458969]
42 MF9='in2cluster9':'gaussmf',[0.266721851038203 0.866247223244947]
43 MF10='in2cluster10':'gaussmf',[0.0413680963986715 0.00545277599898769]
44 MF11='in2cluster11':'gaussmf',[0.0919378739783544 0.181077004079729]
45
46 [Input3]
47 Name='in3'
48 Range=[0 1]
49 NumMFs=11
50 MF1='in3cluster1':'gaussmf',[0.128248544948931 0.924767703049198]
51 MF2='in3cluster2':'gaussmf',[0.14035637637282 0.457803353296698]
52 MF3='in3cluster3':'gaussmf',[0.126514361946808 0.930632679203546]
53 MF4='in3cluster4':'gaussmf',[0.127044434396676 0.918783397164936]
54 MF5='in3cluster5':'gaussmf',[0.131235436965259 0.571331353646802]
55 MF6='in3cluster6':'gaussmf',[0.129790599722538 0.929399710886623]
56 MF7='in3cluster7':'gaussmf',[0.134670237052343 0.527890640476874]
57 MF8='in3cluster8':'gaussmf',[0.126820423465926 0.947246655310078]
58 MF9='in3cluster9':'gaussmf',[0.204817811878333 0.0218392912177431]
59 MF10='in3cluster10':'gaussmf',[0.125517111600608 0.919619320971162]
60 MF11='in3cluster11':'gaussmf',[0.204254891101774 0.0711228893274988]
61
62 [Input4]
63 Name='in4'
64 Range=[0 1]
65 NumMFs=11
66 MF1='in4cluster1':'gaussmf',[0.121334289670493 0.484156282469016]
67 MF2='in4cluster2':'gaussmf',[0.112319743953623 0.168717994131479]
68 MF3='in4cluster3':'gaussmf',[0.179855731734113 0.91427720307009]
69 MF4='in4cluster4':'gaussmf',[0.117918532020257 0.245402765385144]
70 MF5='in4cluster5':'gaussmf',[0.114824482472781 0.295803836180168]
71 MF6='in4cluster6':'gaussmf',[0.121354991344536 0.1761139897947]
72 MF7='in4cluster7':'gaussmf',[0.115501336843843 0.217882158598592]
```

```

73 MF8='in4cluster8':'gaussmf',[0.177979526793363 0.909990302444314]
74 MF9='in4cluster9':'gaussmf',[0.107306969624356 0.222232997014977]
75 MF10='in4cluster10':'gaussmf',[0.171758439618748 0.870879576609811]
76 MF11='in4cluster11':'gaussmf',[0.107093279805772 0.276432850521015]
77
78 [Output1]
79 Name='out1'
80 Range=[0 1]
81 NumMFs=11
82 MF1='out1cluster1':'gaussmf',[0.106293207135751 0.0159445166337191]
83 MF2='out1cluster2':'gaussmf',[0.237654461896844 0.995772031235064]
84 MF3='out1cluster3':'gaussmf',[0.0984359852102391 0.00368713929069681]
85 MF4='out1cluster4':'gaussmf',[0.100971723365207 0.00606184098850582]
86 MF5='out1cluster5':'gaussmf',[0.103876178757675 0.0106826027837995]
87 MF6='out1cluster6':'gaussmf',[0.105252957498416 0.00881971633923944]
88 MF7='out1cluster7':'gaussmf',[0.109886673664818 0.0166955888128662]
89 MF8='out1cluster8':'gaussmf',[0.102703331091824 0.00863678571156374]
90 MF9='out1cluster9':'gaussmf',[0.223799103082429 0.998817597843834]
91 MF10='out1cluster10':'gaussmf',[0.102351610271653 0.0101627500617441]
92 MF11='out1cluster11':'gaussmf',[0.232024244397149 0.995893056978661]
93
94 [Rules]
95 1 1 1 1, 1 (1) : 1
96 2 2 2 2, 2 (1) : 1
97 3 3 3 3, 3 (1) : 1
98 4 4 4 4, 4 (1) : 1
99 5 5 5 5, 5 (1) : 1
100 6 6 6 6, 6 (1) : 1
101 7 7 7 7, 7 (1) : 1
102 8 8 8 8, 8 (1) : 1
103 9 9 9 9, 9 (1) : 1
104 10 10 10 10, 10 (1) : 1
105 11 11 11 11, 11 (1) : 1

```

**Wyciąg 4.1.** Sterownik Mamdaniego wygenerowany w środowisku MATLAB na zbiorze danych anotowanych przez studentów-nauczycieli dla klasyfikacji binarnej, gdzie rekordy nie powtarzają się a anotacje studentów-nauczycieli zostały uśrednione

Wszystkie wygenerowane sterowniki zostały uwzględnione w procesie porównania z metodami uczenia maszynowego pod kątem oceny działania w problemie klasyfikacji.

#### 4.2.2. Sterowniki rozmyte Takagi-Sugeno

W oparciu o algorytm klastrowania Fuzzy C-Means zostały wygenerowane także sterowniki rozmytych Takagi-Sugeno, które od sterowników Mamdaniego różnią się jedynie postacią reguł, tzn. w następniku implikacji nie znajduje się zbiór rozmyty lecz funkcja zależna od zmiennych znajdujących się w poprzedniku reguły (zob. rozdział 2.6.2). Funkcja



ta jest estymowana metodą najmniejszych kwadratów. Do wygenerowania sterowników wykorzystano algorytm klastrowania Fuzzy C-Means zaimplementowany w narzędziu pyFume (ze względu na dodatkową funkcjonalność polegającą na automatycznym wybo-  
rze najbardziej odpowiednich zmiennych wejściowych dla sterownika — zob. rozdział 2.6.2). Wynikiem działania tego narzędzia nie są jak w przypadku środowiska MATLAB pliki w formacie FIS a kod języka Python.

Kod języka Python generujący sterownik z wykorzystaniem biblioteki pyFume został przedstawiony w załączniku B. Jeden z wygenerowanych sterowników w języku Python z wykorzystaniem biblioteki Simful stanowi załącznik C. Wygenerowane sterowniki Takagi-Sugeno zostaną uwzględnione w procesie porównania z metodami uczenia ma-  
szynowego pod kątem oceny działania w problemie klasyfikacji.

### 4.3. PORÓWNANIE STEROWNIKÓW ROZMYTYCH Z METODAMI UCZENIA MASZYNOWEGO W PROBLEMIE KLASYFIKACJI STUDENTÓW

Zaprezentowana w rozdziale 3.3 strategia adaptacji opiera się na wykorzystaniu eksperckiego sterownika rozmytego, który aby go zbudować nie wymaga posiadania zbiorów uczących, co jest istotne w nauczaniu, bo rzadko kiedy dydaktycy posiadają takie zbiory. W celu zweryfikowania działania tej metody, w ramach prowadzonych prac zostało przygotowane porównanie z popularnymi metodami uczenia maszynowego oraz ze sterownikami rozmytymi wygenerowanymi automatycznie z danych. Zadanie to było możliwe do wykonania dzięki wcześniejszemu zebraniu danych pochodzących z rzeczywistych interakcji studentów z kursem z osadzonym sterownikiem eksperckim oraz nadaniu im etykiet przez grupę anotatorów z doświadczeniem dydaktycznym. Aby porównać różne metody klasyfikacji studentów pod kątem zjawiska disengagement, dane zaprezentowane w podrozdziale 4.1.1 zostały wykorzystane w procesie trenowania algorytmów uczenia maszynowego. W ramach porównania wybrano najważniejsze metryki ewaluacji wykorzystywane przy ocenie algorytmów klasyfikujących opisane w rozdziale 2.6.3: dokładność, precyzję, czułość i F1, co jest zgodne z metodologią badań prowadzonych w obszarze uczenia maszynowego [88, 94].

#### 4.3.1. Metody klasyfikacji zawarte w porównaniu

Celem porównania algorytmów jest zweryfikowanie działania metody wykorzystującej ekspercki sterownik rozmyty. Jest to ważne, ponieważ jak wspomniano wcześniej, do budowy takiego sterownika nie jest potrzebny żaden zbiór danych w przeciwieństwie do metod uczenia maszynowego. W dodatku w sterowniku widoczne są wszystkie reguły i sposób modelowania zmiennych, więc autor sterownika ma pełną kontrolę nad jego działaniem, co nie jest już tak oczywiste w uczeniu maszynowym.

Do trenowania algorytmów uczenia maszynowego oraz późniejszego porównania wszystkich metod wykorzystano zbiory otrzymane w wyniku anotowania przez grupę 11 anotatorów (zob. podrozdział 4.1.1):

1. Zbiór danych anotowanych przez grupę anotatorów — 4600 rekordów; klasyfikacja binarna; rekordy powtarzają się z anotacjami różnych studentów
2. Zbiór danych anotowanych przez grupę anotatorów — 4600 rekordów; klasyfikacja pięciostopniowa; rekordy powtarzają się z anotacjami różnych studentów
3. Zbiór danych anotowanych przez grupę anotatorów — 879 rekordów; klasyfikacja binarna; rekordy nie powtarzają się a anotacje studentów zostały uśrednione
4. Zbiór danych anotowanych przez grupę anotatorów — 879 rekordów; klasyfikacja pięciostopniowa; rekordy nie powtarzają się a anotacje studentów zostały uśrednione

W ramach porównania zostaną przygotowane cztery zestawienia (po jednym dla każdego z powyższych zbiorów danych) z wartościami metryk ewaluacji dla wszystkich porównywanych metod. W porównaniu nie wykorzystano zbioru danych anotowanych przez autora rozprawy, ponieważ zbiór ten służył jedynie testom w procesie generowania sterowników rozmytych. Zbiór ten w badaniach był wykorzystany tylko do wygenerowania pierwszego sterownika rozmytego Mamdaniego opartego na danych.

Dla każdego zbioru danych opisanego w podrozdziale 4.1.1 dokonano osobnego trenowania algorytmów uczenia maszynowego oraz późniejszego porównania wszystkich metod. Do porównania wybrano dziewięć popularnych algorytmów klasyfikujących:

1. naiwny klasyfikator Bayesa [175],
2. regresja logistyczna [20]/softmax [139],
3. algorytm  $k$  najbliższych sąsiadów (KNN) [118],
4. metoda wektorów nośnych (SVM) [35],
5. drzewo decyzyjne [58],
6. pochodne drzewa decyzyjnego:
  - a) bagging [49],
  - b) boosting — AdaBoost [55],
  - c) boosting — Gradient Boosting [56],
  - d) las losowy [73].

Większość tych algorytmów cechuje się prostotą i szybkością działania a także nie wymagają bardzo dużych zbiorów danych uczących jak w przypadku sieci neuronowych.

Do porównania zostały wybrane także wszystkie przygotowane sterowniki rozmyte:

- I ekspercki sterownik Mamdaniego,
- II sterowniki Mamdaniego wygenerowane automatycznie:
  - a) na danych anotowanych przez autora rozprawy,
  - b) w oparciu o zbiór danych wykorzystywany w danym zestawieniu,

III sterownik Takagi-Sugeno wygenerowany automatycznie w oparciu o zbiór danych wykorzystywany w danym zestawieniu.

Każdy z wykorzystywanych zbiorów danych osobno wykorzystano do procesu uczenia algorytmów klasyfikujących a później testowania w ramach porównania wszystkich metod. Zbiory zostały podzielone na zbiór uczący i zbiór testowy w stosunku 80:20. Do wyuczenia algorytmów klasyfikujących wybrano bibliotekę *scikit-learn* języka Python [151].

#### 4.3.2. Porównanie metod klasyfikacji studentów

Wszystkie algorytmy klasyfikujące oraz sterowniki rozmyte zostały porównane w tabelach: 4.5, 4.6, 4.7 oraz 4.8. Każda z tabel przedstawia wartości metryk ewaluacji dla wszystkich porównywanych metod z podziałem na zbiór danych, który został wykorzystany w procesie uczenia a później testowania algorytmów. I tak:

- tabela 4.5 przedstawia wyniki na zbiorze danych z etykietami *u studenta występuje znaczący disengagement* / *u studenta nie występuje znaczący disengagement* — klasyfikacja binarna;
- tabela 4.6 przedstawia wyniki na zbiorze danych z etykietami *u studenta występuje znaczący disengagement* / *u studenta nie występuje znaczący disengagement*, w którym uwzględniono fakt, że różni anotatorzy mogli przypisać inną etykietę — dla każdego rekordu w tym zbiorze danych została przypisana etykieta, która była wybierana przez anotatorów częściej — klasyfikacja binarna;
- tabela 4.7 przedstawia wyniki na zbiorze danych z pięcioma etykietami stopniującymi natężenie zjawiska disengagement w skali Likerta — klasyfikacja pięcioklasowa;
- tabela 4.8 przedstawia wyniki na zbiorze danych z pięcioma etykietami stopniującymi natężenie zjawiska disengagement w skali Likerta, w którym uwzględniono fakt, że różni anotatorzy mogli przypisać inną etykietę — dla każdego rekordu w tym zbiorze danych wartości etykiet (1-5) zostały uśrednione a następnie zaokrąglone do najbliższej liczby naturalnej i ostatecznie przypisana była etykieta o wartości równej tej liczbie — klasyfikacja pięcioklasowa.

W każdej tabeli umieszczono również wyniki dla sterownika eksperckiego oraz sterownika wygenerowanego na danych poanotowanych przez autora rozprawy.

Klasyfikator	Dokład- ność	Precyzja <i>macro</i>	Czułość <i>macro</i>	F1 <i>ma- cro</i>	Precyzja <i>macro</i> ( <i>weigh- ted avg</i> )	Czułość <i>macro</i> ( <i>weigh- ted avg</i> )	F1 <i>macro</i> ( <i>weigh- ted avg</i> )
st. ekspercki	0,61	0,77	0,65	0,58	0,79	0,61	0,56
st. Mamda- niego - dane autora	0,69	0,68	0,68	0,68	0,69	0,69	0,69
st. Mamda- niego	0,87	0,87	0,87	0,87	0,87	0,87	0,87
st. Takagi-Sugeno	0,81	0,84	0,83	0,81	0,86	0,81	0,81
Bayes	0,80	0,83	0,82	0,80	0,84	0,80	0,80
regresja logi- styczna	0,91	0,91	0,91	0,91	0,91	0,91	0,91
KNN	0,90	0,90	0,90	0,90	0,90	0,90	0,90
SVM	0,76	0,82	0,79	0,76	0,84	0,76	0,75
drzewo decy- zyjne	0,88	0,88	0,87	0,87	0,88	0,88	0,88
bagging	0,88	0,88	0,88	0,88	0,88	0,88	0,88
AdaBoost	0,91	0,90	0,90	0,90	0,91	0,91	0,91
Gradient Boosting	0,90	0,90	0,90	0,90	0,90	0,90	0,90
las losowy	0,88	0,89	0,88	0,88	0,88	0,88	0,88

**Tabela 4.5.** Porównanie algorytmów w problemie klasyfikacji binarnej na zbiorze danych bez uśredniania anotacji w przypadku występowania duplikatów

Kolorem zielonym oznaczono komórki z najlepszym wynikiem w ramach danej metryki (kolumny) natomiast kolorem czerwonym oznaczono komórki z najgorszym wynikiem.

Klasyfikator	Dokład- ność	Precyzja <i>macro</i>	Czułość <i>macro</i>	F1 <i>ma- cro</i>	Precyzja <i>macro</i> ( <i>weigh- ted avg</i> )	Czułość <i>macro</i> ( <i>weigh- ted avg</i> )	F1 <i>macro</i> ( <i>weigh- ted avg</i> )
st. ekspercki	0,62	0,77	0,66	0,60	0,80	0,62	0,59
st. Mamda- niego - dane autora	0,76	0,76	0,76	0,76	0,76	0,76	0,76
st. Mamda- niego	0,95	0,95	0,94	0,95	0,95	0,95	0,95
st. Takagi-Sugeno	0,95	0,95	0,94	0,95	0,95	0,95	0,95
Bayes	0,80	0,85	0,82	0,80	0,86	0,80	0,80
regresja logi- styczna	0,94	0,94	0,95	0,94	0,95	0,94	0,94
KNN	0,95	0,95	0,95	0,95	0,95	0,95	0,95
SVM	0,70	0,80	0,73	0,69	0,82	0,70	0,68
drzewo decy- zyjne	0,91	0,92	0,92	0,91	0,93	0,91	0,91
bagging	0,94	0,94	0,94	0,94	0,94	0,94	0,94
AdaBoost	0,95	0,95	0,96	0,95	0,96	0,95	0,95
Gradient Boosting	0,97	0,96	0,97	0,97	0,97	0,97	0,97
las losowy	0,95	0,95	0,96	0,95	0,95	0,95	0,95

**Tabela 4.6.** Porównanie algorytmów w problemie klasyfikacji binarnej na zbiorze danych z uśrednionymi anotacjami w przypadku występowania duplikatów

Kolorem zielonym oznaczono komórki z najlepszym wynikiem w ramach danej metryki (kolumny) natomiast kolorem czerwonym oznaczono komórki z najgorszym wynikiem.

Klasyfikator	Dokład- ność	Precyzja <i>macro</i>	Czułość <i>macro</i>	F1 <i>ma- cro</i>	Precyzja <i>macro</i> ( <i>weigh- ted avg</i> )	Czułość <i>macro</i> ( <i>weigh- ted avg</i> )	F1 <i>macro</i> ( <i>weigh- ted avg</i> )
st. ekspercki	0,49	0,54	0,47	0,44	0,54	0,49	0,46
st. Mamda- niego - dane autora	0,30	0,32	0,32	0,26	0,34	0,30	0,26
st. Mamda- niego	0,59	0,66	0,58	0,57	0,68	0,59	0,59
st. Takagi-Sugeno	0,71	0,71	0,69	0,70	0,73	0,71	0,71
Bayes	0,43	0,38	0,38	0,30	0,42	0,43	0,33
regresja logi- styczna	0,73	0,73	0,70	0,70	0,73	0,73	0,72
KNN	0,68	0,66	0,66	0,66	0,68	0,68	0,68
SVM	0,41	0,24	0,36	0,27	0,27	0,41	0,29
drzewo decy- zyjne	0,61	0,60	0,61	0,60	0,60	0,61	0,60
bagging	0,62	0,62	0,62	0,61	0,62	0,62	0,62
AdaBoost	0,64	0,61	0,62	0,61	0,63	0,64	0,63
Gradient Boosting	0,70	0,68	0,68	0,68	0,70	0,70	0,70
las losowy	0,64	0,64	0,61	0,59	0,64	0,64	0,62

**Tabela 4.7.** Porównanie algorytmów w problemie klasyfikacji pięcioklasowej na zbiorze danych bez uśredniania anotacji w przypadku występowania duplikatów

Kolorem zielonym oznaczono komórki z najlepszym wynikiem w ramach danej metryki (kolumny) natomiast kolorem czerwonym oznaczono komórki z najgorszym wynikiem.

Klasyfikator	Dokładność	Precyzja <i>macro</i>	Czułość <i>macro</i>	F1 <i>macro</i>	Precyzja <i>macro</i> (weighted avg)	Czułość <i>macro</i> (weighted avg)	F1 <i>macro</i> (weighted avg)
st. ekspercki	0,43	0,56	0,45	0,40	0,59	0,43	0,38
st. Mamdaniego - dane autora	0,36	0,32	0,37	0,29	0,35	0,36	0,29
st. Mamdaniego	0,78	0,80	0,79	0,79	0,81	0,78	0,79
st. Takagi-Sugeno	0,78	0,80	0,79	0,79	0,81	0,78	0,79
Bayes	0,42	0,31	0,34	0,26	0,35	0,42	0,31
regresja logistyczna	0,76	0,77	0,72	0,72	0,77	0,76	0,75
KNN	0,78	0,78	0,76	0,76	0,78	0,78	0,78
SVM	0,31	0,26	0,24	0,15	0,30	0,31	0,18
drzewo decyzyjne	0,64	0,66	0,65	0,64	0,66	0,64	0,64
bagging	0,69	0,70	0,69	0,69	0,71	0,69	0,69
AdaBoost	0,72	0,75	0,71	0,71	0,77	0,72	0,72
Gradient Boosting	0,80	0,79	0,78	0,78	0,80	0,80	0,80
las losowy	0,70	0,75	0,68	0,70	0,75	0,70	0,71

**Tabela 4.8.** Porównanie algorytmów w problemie klasyfikacji pięcioklasowej na zbiorze danych z uśrednionymi anotacjami w przypadku występowania duplikatów

Kolorem zielonym oznaczono komórki z najlepszym wynikiem w ramach danej metryki (kolumny) natomiast kolorem czerwonym oznaczono komórki z najgorszym wynikiem.

W przypadku klasyfikacji binarnej najlepsze wartości metryk uzyskiwane są przy zastosowaniu regresji logistycznej z F1 równym 0,91 (w przypadku zbioru bez uśredniania anotacji dla duplikatów — tabela 4.5) oraz algorytmu Gradient Boosting z F1 równym 0,97 (w przypadku zbioru z uśrednianiem anotacji dla duplikatów — tabela 4.6). Najniższe wartości metryk osiąga ekspercki sterownik Mamdaniego z F1 równym 0,60 — należy tutaj pamiętać, że ten sterownik nie został wyuczony na żadnym zbiorze danych (był sterownikiem eksperckim) i był pierwszym sterownikiem, który powstał, gdzie jeszcze nie było zbiorów danych z anotacjami. Mimo najłabszych wyników dla tego sterownika, warto podkreślić, że nie są to bardzo złe wyniki. Najniższe w całym porównaniu wyniki nie przekreślają sensowności wykorzystywania eksperckich sterowników rozmytych. Być może baza reguł lub interpretacje termów wykorzystanych w sterowniku zmiennych wymagają poprawienia.

Dla klasyfikacji pięcioklasowej na zbiorze bez uśredniania anotacji dla duplikatów (tabela 4.7) najlepsze efekty osiągane są przy zastosowaniu regresji logistycznej z F1 równym 0,70. Natomiast warto zwrócić uwagę, że przy wykorzystaniu sterownika Takagi-Sugeno F1 także wynosi 0,70, choć inne metryki są minimalnie niższe. Najniższe wartości metryk osiągnięto dla sterownika Mamdaniego wygenerowanego na zbiorze danych anotowanych przez autora sterownika eksperckiego (F1 na poziomie 0,26). Nie powinno to dziwić, ponieważ anotacje zbioru uczącego były mocno subiektywne a ponadto anotacji dokonywała tylko jedna osoba. Bardzo niskie wartości metryk uzasadnia dodatkowo fakt, że zbiór służący do wygenerowania tego sterownika był anotowany dla klasyfikacji binarnej a nie pięciostopniowej. Natomiast niskie wartości metryk dla eksperckiego sterownika Mamdaniego również nie powinny dziwić, ponieważ baza reguł była budowana z myślą o klasyfikacji binarnej, tzn. sterownik był budowany z myślą o strategii adaptacji, w której student w kursie e-learningowym był kierowany na jedną z dwóch ścieżek w kontekście podchodzenia do testów zaliczeniowych (tj. student albo miał znaczący disengagement albo nie — nie było tutaj gradacji w pięciostopniowej skali Likerta).

W przypadku klasyfikacji pięcioklasowej na zbiorze, w którym dokonano uśrednienia anotacji dla duplikatów (tabela 4.8) najlepsze wartości metryk osiągnięto dla sterownika Mamdaniego generowanego na zbiorze danych anotowanych przez grupę studentów-nauczycieli z F1 na poziomie 0,79 a także dla sterownika Takagi-Sugeno — z takimi samymi wynikami dla wszystkich metryk. Pokazuje to, że w przypadku dobrej bazy reguł i odpowiednio zamodelowanych zmiennych, sterowniki rozmyte mogą dawać najlepsze efekty w problemie klasyfikacji. Z drugiej strony, ciekawym faktem jest to, że dla metody wektorów nośnych osiągnięto najłabsze wyniki (F1 na poziomie 0,15; dla skali macro z wykorzystaniem średniej ważonej — 0,18).

### 4.3.3. Dyskusja

Wyniki porównania wybranych metod pokazują, że w przypadku posiadania zbioru danych do trenowania popularne metody uczenia maszynowego w ogólności mają porównywalne wyniki ze sterownikami rozmytymi.

Wartości metryk osiągnięte przez sterowniki rozmyte wygenerowane automatycznie są zadowalające, natomiast w przypadku eksperckiego sterownika rozmytego są niż-



sze, jednak należy pamiętać o tym, że ten sterownik nie został wyuczony na żadnym zbiorze danych (nie wymagał tego ze względu na swoją eksperckość). W prezentowanej w rozdziale 3.3 strategii dydaktycznej (nastawionej na klasyfikację binarną studenta) tak przygotowany sterownik działał zgodnie z oczekiwaniami. Cały czas znaczącą wartością eksperckich sterowników jest fakt, że nie jest wymagany żaden zbiór danych do uczenia a same sterowniki działają bardzo szybko. Co więcej, w prezentowanym w rozprawie rozwiązaniu, sterownik został wprost osadzony w kursie e-learningowym i kurs od razu działał a samo uruchomienie sterownika odbywało się w przeglądarce internetowej studenta — nie było potrzeby wcześniejszego zbierania i anotowania danych a następnie wyuczania modelu. Przy zastosowaniu takiego rozwiązania niepotrzebne było wykorzystywanie zewnętrznych serwerów, na których odbywałoby się obliczanie poziomu zjawiska disengagement np. w oparciu o metody uczenia maszynowego. Dzięki temu można zaoszczędzić czas a w przypadku poważniejszych problemów i dużo większych zbiorów danych — także koszty.

Przy ostatecznej ocenie efektywności wszystkich prezentowanych metod należy jednak zaznaczyć, że osiągnięte wyniki są obarczone niepewnością, ponieważ zbiór danych jest naznaczony subiektywnością przy anotowaniu. Subiektywny charakter anotowania danych wynika z tego, że anotowali go studenci-nauczyciele w ramach zajęć (patrz rozdział 4.1.2). Różne czynniki mogły wpłynąć na efekt anotowania. Na przykład styl prowadzenia zajęć przez prowadzącego czy sposób przedstawienia problemu studentom a także sama postać danych dydaktycznych mogły wpłynąć na ostateczny wynik anotowania. Dodatkowo, cechy charakteru anotatora (w tym wypadku — studenta kierunku nauczanie matematyki i informatyki) czy poziom jego skupienia, który może być zaburzony zmęczeniem, także mógł mieć wpływ na wyniki anotowania. Subiektywność anotatorów jest znanym problemem w kontekście metodologii badań w obszarze uczenia maszynowego [4, 142, 174, 178].

Pomimo wymienionych wyżej ograniczeń wynikających z problemów związanych z pozyskiwaniem zbioru danych należy stwierdzić, że sterowniki rozmyte dają dość dobre wyniki w problemie klasyfikacji przy szybkim działaniu i niskim zużyciu zasobów obliczeniowych. Brak konieczności posiadania zbioru danych do uczenia w procesie budowania sterowników rozmytych oraz łatwość implementacji i osadzania w różnych systemach (w tym wypadku w kursie e-learningowym jako kod języka JavaScript) sprawia, że w problemach klasyfikacji (takich jak omawiana w rozprawie klasyfikacja studentów na potrzeby adaptacji treści w e-learningu) wykorzystanie sterowników rozmytych będzie dobrym i uzasadnionym rozwiązaniem. Jednym z celów badań w kontekście budowania strategii dydaktycznych w kursach e-learningowych z adaptacyjną zawartością było potwierdzenie zasadności wykorzystania sterowników rozmytych w problemie klasyfikacji studentów. Wyniki działania sterowników rozmytych osiągnięte w porównaniu różnych metod klasyfikacji są w ogólności porównywalne z algorytmami uczenia maszynowego. W dodatku w sterownikach rozmytych użytkownik zawsze ma wgląd w postać reguł i modeli zmiennych, więc łatwiejsza jest analiza działania tej metody, w przeciwieństwie do algorytmów wykorzystywanych w uczeniu maszynowym. W związku z tym, opisany wyżej cel badań został spełniony.



## ROZDZIAŁ 5

# Rozszerzona strategia dydaktyczna wykorzystująca sterowanie rozmyte

Prezentowana w rozdziałach 3.3 i 4 strategia adaptacji treści w kursie e-learningowym jest oparta na zjawisku disengagement, którego poziom jest obliczany z wykorzystaniem sterowania rozmytego. W ramach dalszych badań w obszarze kursów z adaptacyjną zawartością strategia została rozszerzona o inne pojęcia również definiowane przez sterowniki rozmyte. Cały czas jednak wykorzystywane jest zjawisko disengagement i ekspercki sterownik rozmyty. W niniejszym rozdziale zostanie przedstawiona rozszerzona strategia dydaktyczna wykorzystująca architekturę kursu z adaptacyjną zawartością przedstawioną w rozdziale 3.5. W stosunku do strategii omówionej w rozdziale 3.3 rozszerzenie polega na wykorzystaniu:

- w innym zastosowaniu dydaktycznym i innym kursie e-learningowym z adaptacyjną zawartością,
- zjawiska disengagement i sterownika eksperckiego (zob. rozdział 3.4) w innym kontekście dydaktycznym,
- dwóch innych rozmytych sterowników eksperckich — zatem w kursie będą zastosowane łącznie trzy sterowniki rozmyte obliczające wartości trzech niestandardowych zmiennych (zob. rozdział 3.5) a nie jeden jak w przypadku strategii omawianej w rozdziale 3.3.

### 5.1. KURS E-LEARNINGOWY WYKORZYSTANY W BADANIACH

W ramach opracowania i weryfikacji rozszerzonej strategii dydaktycznej został wykorzystany kurs e-learningowy „Wstęp do uczenia maszynowego” [59]. Jest to kurs, który przybliży studentom podstawowe pojęcia związane z uczeniem maszynowym a także wprowadza ich w kwestie związane z konstruowaniem i analizowaniem modeli uczenia maszynowego. Kurs jest nastawiony na rozwijanie umiejętności w obszarze podstaw uczenia maszynowego. Zawiera wiele przykładów i ćwiczeń, co jest istotne przy ewentualnym wdrażaniu strategii dydaktycznych, ponieważ możliwe jest śledzenie

postępów studentów w takich aktywnościach. Kurs nie wymaga większych umiejętności w zakresie matematyki, więc może być także wykorzystywany w szkołach średnich w klasach matematyczno-informatycznych. Kurs został wdrożony na II roku studiów na kierunku informatyka w ramach zajęć ze sztucznej inteligencji jako materiał uzupełniający. Na pracę z kursem studenci otrzymali sześć dni.

Na kurs składają się cztery moduły:

1. Wprowadzenie do uczenia maszynowego
2. Dane w procesie uczenia
3. Podstawowe pojęcia uczenia maszynowego
4. Metryki

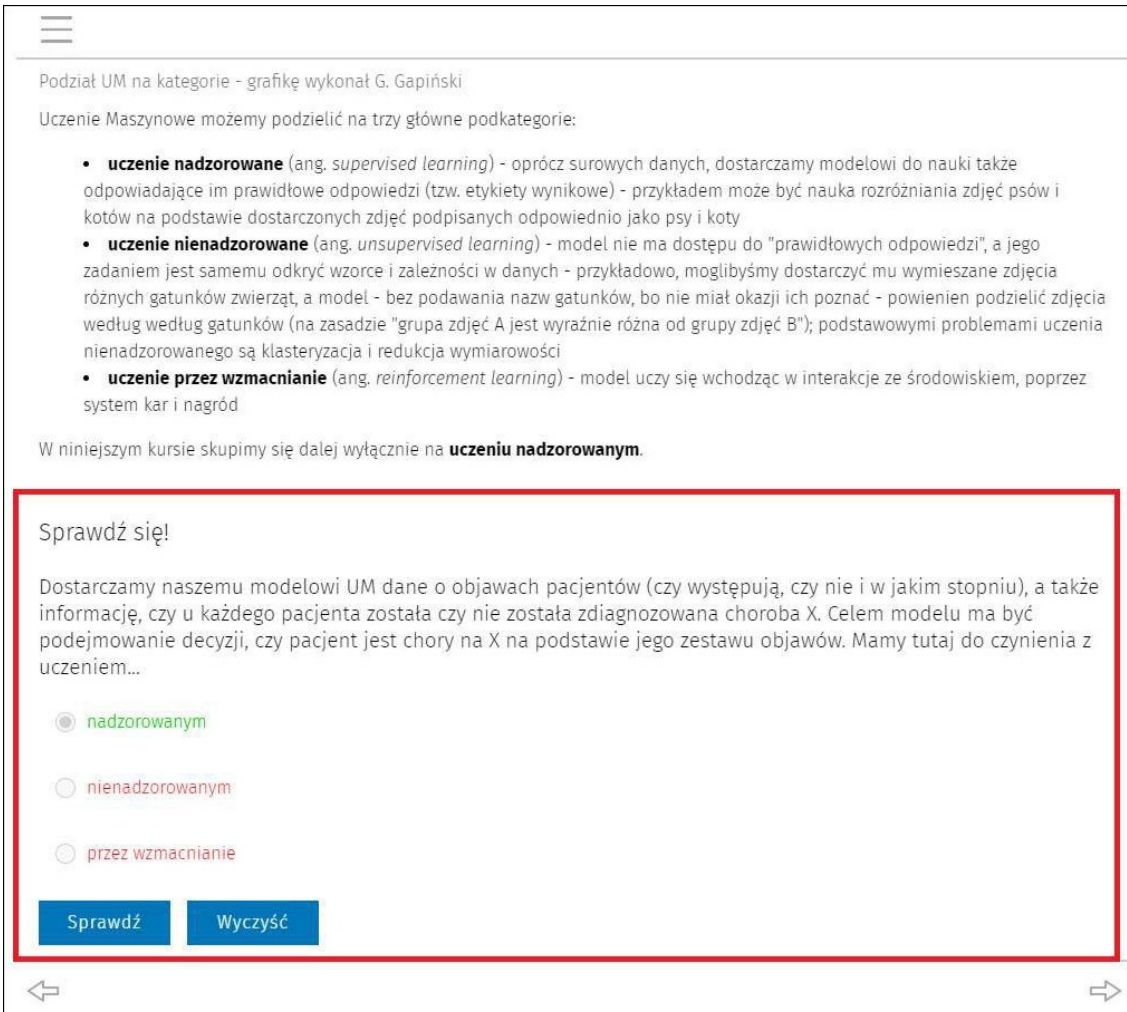
Kurs rozpoczyna się wstępnym testem wiedzy w zakresie uczenia maszynowego. W każdym module natomiast znajdują się materiały do nauki, przykłady i ćwiczenia a także krótkie testy sprawdzające poziom umiejętności po danym module. W przypadku niepowodzenia, studenci otrzymują materiały dodatkowe pozwalające uzupełnić braki w ramach danego modułu. Kurs kończy się testem weryfikującym ogólny poziom umiejętności, które student powinien nabyć w trakcie pracy z kursem. Fragmenty kursu przedstawiają rysunki 5.1 i 5.2:



The screenshot shows a test result page for the course 'Wstęp do uczenia maszynowego v1.0 Beta 1'. The page title is 'AT - Wprowadzenie do UM - Test'. The user's score is 5 / 5. A message states: 'Dziękuję za rozwiązanie testu. Jeżeli test jest zaliczony (uzyskano co najmniej 50% punktów), możesz kontynuować kurs. Jeżeli zaś nie, za dwie minuty udostępnione zostanie kolejne podejście. Możesz skorzystać z **materiałów powtórkowych**, do których odnośnik znajduje się poniżej, żeby lepiej opanować materiał z tego modułu.' Below this, there are links for 'Dodatkowe przykłady' and 'Kontynuuj kurs'. A red box highlights two lines of text: 'Masz swobodny dostęp do kursu. Sprawdź spis treści.' and 'Gratulacje! Twoje postępy w nauce pozwoliły na uzyskanie dostępu do wszystkich treści w kursie.'

**Rysunek 5.1.** Przykładowy test po jego rozwiązaniu przez studenta w kursie „Wstęp do uczenia maszynowego” z włączonym modułem adaptacji treści

(Opracował mgr inż. Grzegorz Gapiński [59])



Podział UM na kategorie - grafikę wykonał G. Gapiński

Uczenie Maszynowe możemy podzielić na trzy główne podkategorie:

- **uczenie nadzorowane** (ang. *supervised learning*) - oprócz surowych danych, dostarczamy modelowi do nauki także odpowiadające im prawidłowe odpowiedzi (tzw. etykiety wynikowe) - przykładem może być nauka rozróżniania zdjęć psów i kotów na podstawie dostarczonych zdjęć podpisanych odpowiednio jako psy i koty
- **uczenie nienadzorowane** (ang. *unsupervised learning*) - model nie ma dostępu do "prawidłowych odpowiedzi", a jego zadaniem jest samemu odkryć wzorce i zależności w danych - przykładowo, moglibyśmy dostarczyć mu wymieszane zdjęcia różnych gatunków zwierząt, a model - bez podawania nazw gatunków, bo nie miał okazji ich poznać - powinien podzielić zdjęcia według gatunków (na zasadzie "grupa zdjęć A jest wyraźnie różna od grupy zdjęć B"); podstawowymi problemami uczenia nienadzorowanego są klasteryzacja i redukcja wymiarowości
- **uczenie przez wzmacnianie** (ang. *reinforcement learning*) - model uczy się wchodząc w interakcje ze środowiskiem, poprzez system kar i nagród

W niniejszym kursie skupimy się dalej wyłącznie na **uczeniu nadzorowanym**.

**Sprawdź się!**

Dostarczamy naszemu modelowi UM dane o objawach pacjentów (czy występują, czy nie i w jakim stopniu), a także informację, czy u każdego pacjenta została czy nie została zdiagnozowana choroba X. Celem modelu ma być podejmowanie decyzji, czy pacjent jest chory na X na podstawie jego zestawu objawów. Mamy tutaj do czynienia z uczeniem...

nadzorowanym

nienadzorowanym

przez wzmacnianie

Rysunek 5.2. Przykładowe ćwiczenie w kursie „Wstęp do uczenia maszynowego”

(Opracował mgr inż. Grzegorz Gapiński [59])

## 5.2. STRATEGIA DYDAKTYCZNA ZASTOSOWANA W KURSIE

Kurs wykorzystany w badaniach realizuje architekturę kursu z adaptacyjną zawartością przedstawioną w rozdziale 3.5, jednak w przeciwieństwie do kursu „Zastosowania zbiorów rozmytych (zob. rozdział 3.1.1) zastosowana strategia wykorzystuje nie jeden a trzy sterowniki rozmyte. Wdrożona w kursie „Wstęp do uczenia maszynowego” strategia wykorzystuje, omawiane w rozprawie zjawisko disengagement a także dwa inne pojęcia: poziom kompetencji w zakresie podstaw uczenia maszynowego oraz tzw. *question exchange requirement*, czyli informacja o tym, czy końcowy test umiejętności w kursie powinien składać się z pytań, które już wcześniej w kursie wystąpiły w ramach częściowych testów po każdym module, czy też test powinien zawierać wyłącznie pytania, które w kursie nigdzie nie zostały wykorzystane. Wartość *question exchange requirement* jest obliczana na podstawie wartości poziomu disengagement oraz kompetencji w zakresie podstaw uczenia maszynowego.

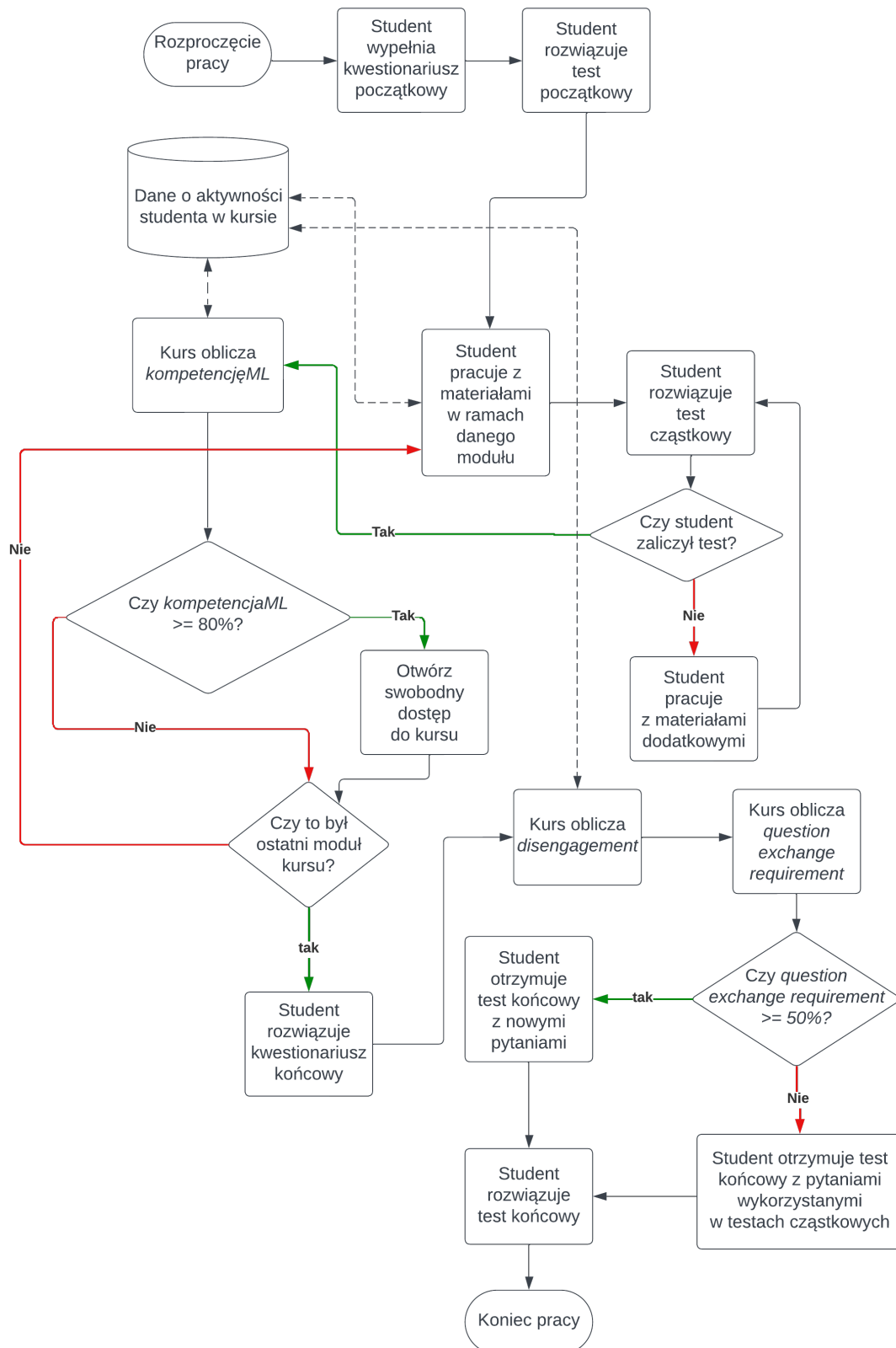
Pełną strategię dydaktyczną przedstawia rysunek 5.3. Na rysunku pokazano przepływ pracy studenta w omawianym kursie. Strategia ta została zaproponowana przez mgra inż. Grzegorza Gapińskiego i uzupełniona przez prof. UAM dr hab. Jacka Marciniaka oraz autora rozprawy a jej realizacja została zapewniona przez sterowniki rozmyte:

- disengagement — ekspercki sterownik rozmyty zaproponowany przez autora rozprawy (wykorzystany wcześniej w kursie „Zastosowania zbiorów rozmytych” — zob. rozdziały 3.1.1 i 3.4);
- kompetencja w zakresie podstaw uczenia maszynowego (*kompetencjaML*) — ekspercki sterownik rozmyty przygotowany wspólnie przez autora rozprawy i mgra inż. Grzegorza Gapińskiego;
- *question exchange requirement* — ekspercki sterownik rozmyty przygotowany wspólnie przez autora rozprawy i mgra inż. Grzegorza Gapińskiego.

Sterownik obliczający poziom zjawiska disengagement działa zgodnie z jego definicją omówioną w rozdziale 3.4 — bazuje on na pojęciu jakości nauki i czasie pozostałym do zamknięcia końcowego testu zaliczeniowego w momencie pierwszego wejścia do kursu.

Z kolei sterownik wyznaczający poziom kompetencji w zakresie podstaw uczenia maszynowego jest eksperckim sterownikiem Mamdaniego (który jest bardziej zrozumiały jeśli chodzi o bazę reguł), który na wejściu przyjmuje dwie informacje: liczbę punktów zdobytych w teście wiedzy dostępnym na początku kursu oraz aktualną sumę punktów zdobytych w testach częściowych dostępnych na końcu danego modułu. W oparciu o obliczoną wartość w kursie jest podejmowana decyzja, czy student powinien otrzymać swobodny dostęp do kursu czy powinien pracować sekwencyjnie.

Natomiast sterownik obliczający wartość *question exchange requirement*, która jest wykorzystywana w momencie podejmowania decyzji o zawartości końcowego testu zaliczeniowego, bazuje na wartościach obliczonych przez sterowniki rozmyte: *disengagement* i *kompetencjaML*.



Rysunek 5.3. Algorytm adaptacji treści w kursie z rozszerzoną strategią dydaktyczną

Zgodnie z prezentowaną na rysunku 5.3 strategią, student rozpoczyna pracę z kursem od rozwiązania krótkiego kwestionariusza i testu wiedzy. Test ten definiuje w kursie poziom kompetencji w zakresie podstaw uczenia maszynowego w obszarze wiedzy. Po rozwiązaniu testu student rozpoczyna pracę z pierwszym modułem kursu. W trakcie pracy student otrzymuje zestaw ćwiczeń do wykonania w oparciu o prezentowane w module przykłady oraz część teoretyczną. Po zakończeniu pracy student podchodzi do testu sprawdzającego poziom umiejętności, które student powinien nabyć w danym module — testy te wpływają na poziom kompetencji w zakresie podstaw uczenia maszynowego w obszarze umiejętności. Jeżeli test zakończy się powodzeniem (tzn. student otrzyma co najmniej połowę punktów możliwych do zdobycia), to zostaje otwarty dostęp do kolejnego modułu. W przeciwnym wypadku, student otrzymuje materiały dodatkowe pozwalające na uzupełnienie braków. Student może podchodzić do testu aż do momentu uzyskania zaliczenia. W ten sposób zaprojektowana została praca ze wszystkimi modułami. Natomiast jeżeli po rozwiązaniu danego testu częściowego kurs obliczy wysoki poziom kompetencji w zakresie podstaw uczenia maszynowego, to automatycznie zostaje otwarty swobodny dostęp do wszystkich pozostałych modułów. Studenci, którzy nie będą usatysfakcjonowani zdobytą liczbą punktów w ramach wszystkich testów częściowych mają możliwość podejścia do końcowego testu weryfikującego ogólny poziom umiejętności, które studenci powinni nabyć dzięki pracy z całym kursem. Zawartość tego testu zależy od poziomu kompetencji w zakresie podstaw uczenia maszynowego a także poziomu zjawiska *disengagement*. Te dwie informacje określają *question exchange requirement* i w przypadku, gdy ten współczynnik wynosi mniej niż 50%, to student otrzymuje zestaw pytań, które pojawiały się w testach częściowych (zakłada się tutaj, że taki student uczciwie pracował w kursie ale miał trochę problemów z materiałem). W przeciwnym wypadku student otrzymuje zestaw pytań, które jeszcze się nie pojawiły.

Aby możliwe było wdrożenie prezentowanej strategii dydaktycznej, w kursie wykorzystano bibliotekę *its-student-progress* (zob. rozdział 3.1.3) do śledzenia postępów studentów oraz przesyłania informacji do sterowników rozmytych. Same sterowniki rozmyte natomiast zostały zapisane w języku FCL [41] a następnie przekonwertowane do języka JavaScript z wykorzystaniem modułu *fcl2js* biblioteki *jFuzzyLogic* (zob. rozdział 3.1.2).

### 5.3. STEROWNIKI ROZMYTE W REALIZACJI STRATEGII DYDAKTYCZNEJ

Prezentowana w podrozdziale 5.2 strategia dydaktyczna wymaga obliczenia wartości trzech współczynników: *disengagement*, kompetencja w zakresie podstaw uczenia maszynowego oraz *question exchange requirement*. Wartości te są obliczane za pomocą eksperckich sterowników rozmytych. Pierwszy z nich, czyli ten obliczający poziom zjawiska *disengagement* został zaprezentowany w rozdziale 3.4 i wykorzystany w omawianym wcześniej kursie „Zastosowania zbiorów rozmytych”. Kolejne dwa sterowniki zostały



przygotowane wspólnie przez autora rozprawy oraz mgra inż. Grzegorza Gapińskiego będącego autorem kursu „Wstęp do uczenia maszynowego”.

### 5.3.1. Sterownik obliczający poziom kompetencji w zakresie podstaw uczenia maszynowego

Zgodnie ze algorytmem adaptacji treści w kursie przedstawionym na rysunku 5.3, częścią strategii jest wykorzystanie informacji o kompetencji w zakresie podstaw uczenia maszynowego. Poziom tej kompetencji jest obliczany z wykorzystaniem eksperckiego sterownika rozmytego Mamdaniego, który na wejściu przyjmuje dwie zmienne:

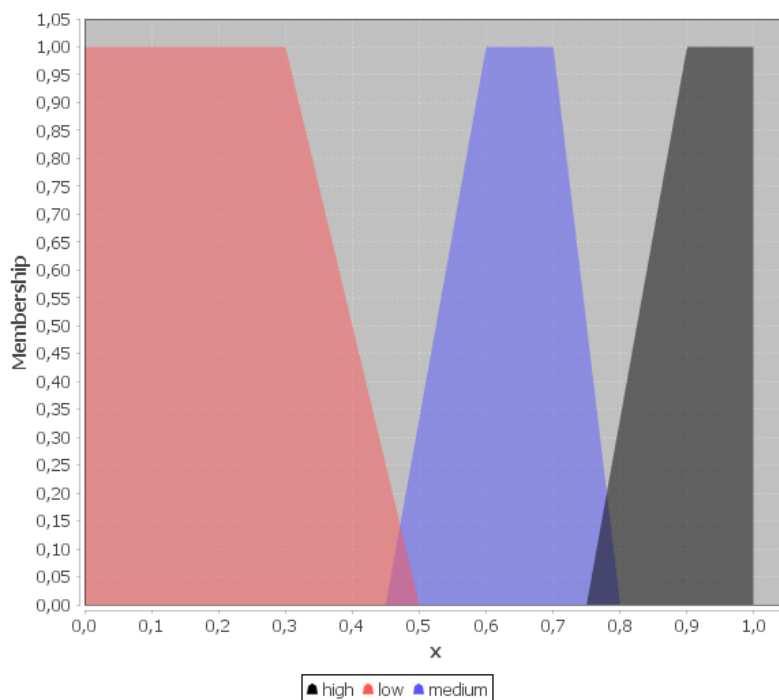
- **wiedza** — liczba punktów zdobytych w początkowym teście wiedzy (obszar wiedzy),
- **umiejętności** — bieżąca suma punktów zdobytych w testach cząstkowych (obszar umiejętności).

Dla każdej ze zmiennych zdefiniowano trzy termy (podobnie jak w przypadku sterownika obliczającego poziom zjawiska disengagement — zob. rozdział 3.4): *low*, *medium*, *high* — im mniej termów i zmiennych tym mniejsza i bardziej zrozumiała dla eksperta (w tym wypadku dydaktyka) będzie baza reguł. Na rysunku 5.4 przedstawiono funkcje przynależności zbiorów rozmytych będących interpretacjami tych termów.

Zgodnie z modelem sterownika rozmytego Mamdaniego (zob. rozdział 2.6.2), w bloku rozmywania należy jeszcze zdefiniować zmienną wyjściową, czyli opisującą poziom kompetencji w zakresie podstaw uczenia maszynowego. Ze względu na to, aby baza reguł była mniejsza, to zmienna ta została zdefiniowana również na trzech termach — tak samo jak zmienne wejściowe (zob. rysunek 5.4).

Przy definiowaniu zmiennych przyjęto założenie, że interpretacje termów powinny odzwierciedlać sposób oceniania studenta w ramach zajęć dydaktycznych. Przyjęto założenie, że student nie otrzymuje pozytywnej oceny poniżej 50% punktów a wysoka ocena to co najmniej 4,0 (od 75% punktów).

Kolejnym krokiem w opracowaniu sterownika rozmytego jest zbudowanie bazy reguł. Reguły zaprezentowane są w tabeli 5.1. Tabela przedstawia wszystkie reguły wykorzystane w sterowniku. Przy budowie bazy reguł sterownika przyjęto założenie, że w kontekście kompetencji w ramach podstaw uczenia maszynowego bardziej istotne są umiejętności, dlatego np. w trzeciej regule przy niskim poziomie wiedzy i wysokim poziomie umiejętności poziom kompetencji jest określany jako średni ale już odwrotnie (siódma reguła), czyli w sytuacji, gdy poziom wiedzy jest wysoki a poziom umiejętności niski, to kompetencja dalej pozostaje na niskim poziomie.



**Rysunek 5.4.** Model zbioru wartości obu zmiennych wejściowych definiowanych w sterowniku obliczającym poziom kompetencji w zakresie podstaw uczenia maszynowego

---

#### Lp. Reguła

---

1. IF **wiedza** IS *low* AND **umiejętności** IS *low* THEN **kompetencjaML** IS *low*
  2. IF **wiedza** IS *low* AND **umiejętności** IS *medium* THEN **kompetencjaML** IS *low*
  3. IF **wiedza** IS *low* AND **umiejętności** IS *high* THEN **kompetencjaML** IS *medium*
  4. IF **wiedza** IS *medium* AND **umiejętności** IS *low* THEN **kompetencjaML** IS *low*
  5. IF **wiedza** IS *medium* AND **umiejętności** IS *medium* THEN **kompetencjaML** IS *medium*
  6. IF **wiedza** IS *medium* AND **umiejętności** IS *high* THEN **kompetencjaML** IS *high*
  7. IF **wiedza** IS *high* AND **umiejętności** IS *low* THEN **kompetencjaML** IS *low*
  8. IF **wiedza** IS *high* AND **umiejętności** IS *medium* THEN **kompetencjaML** IS *medium*
  9. IF **wiedza** IS *high* AND **umiejętności** IS *high* THEN **kompetencjaML** IS *high*
- 

**Tabela 5.1.** Baza reguł wykorzystanych w sterowniku obliczającym poziom kompetencji w zakresie podstaw uczenia maszynowego

Baza reguł sterownika została opracowana wspólnie przez autora rozprawy i autora kursu — mgr inż. Grzegorza Gapińskiego.

W poprzednikach reguł obie zmienne wejściowe połączone są ze sobą spójnikiem koniunkcji. Zgodnie z powszechną praktyką spójnik ten w sterowniku został zdefiniowany jako operacja minimum (zob. rozdział 2.5). Podobnie jak operator THEN, czyli implikacja.

W wyniku działania bloku wnioskowania w oparciu o bazę reguł powstaje jeden zbiór rozmyty będący sumą zbiorów powstałych w wyniku działania danej reguły z bazy dla konkretnych wartości zmiennych wejściowych sterownika (zob. blok wnioskowania w rozdziale 2.6.2). Ostatnim krokiem jest wyostrzenie tego zbioru metodą środka ciężkości, czyli metodą najczęściej wykorzystywaną przy wyostrzaniu zbiorów rozmytych (zob. blok wyostrzania w rozdziale 2.6.2). Działanie prezentowanego sterownika rozmytego przedstawia przykład 5.1.

**Przykład 5.1.** Na wejściu sterownik otrzymuje następujące wartości:

- $wiedza = 0,40$ ,
- $umiejętności = 0,80$ .

W bloku rozmywania sterownika otrzymane wartości zostaną rozmyte zgodnie z definicjami zmiennych wejściowych (zob. rysunek 5.4):

**wiedza:**

- $low(0,40) = 0,50$
- $medium(0,40) = 0,00$
- $high(0,40) = 0,00$

**umiejętności:**

- $low(0,80) = 0,00$
- $medium(0,80) = 0,00$
- $high(0,80) = 0,30$

Po rozmyciu wartości wejściowych, sterownik przechodzi do bloku wnioskowania w oparciu o bazę reguł (zob. tabela 5.1). Dla każdej reguły na podstawie wcześniej zdefiniowanych wartości zmiennych wejściowych należy wyznaczyć stopień spełnienia poprzednika reguły:

**Reguła 1.**  $0,50 \wedge 0,00 = 0,00$

**Reguła 2.**  $0,50 \wedge 0,00 = 0,00$

**Reguła 3.**  $0,50 \wedge 0,30 = 0,30$

**Reguła 4.**  $0,00 \wedge 0,00 = 0,00$

**Reguła 5.**  $0,00 \wedge 0,00 = 0,00$

**Reguła 6.**  $0,00 \wedge 0,30 = 0,00$

**Reguła 7.**  $0,00 \wedge 0,00 = 0,00$

**Reguła 8.**  $0,00 \wedge 0,00 = 0,00$

**Reguła 9.**  $0,00 \wedge 0,30 = 0,00$

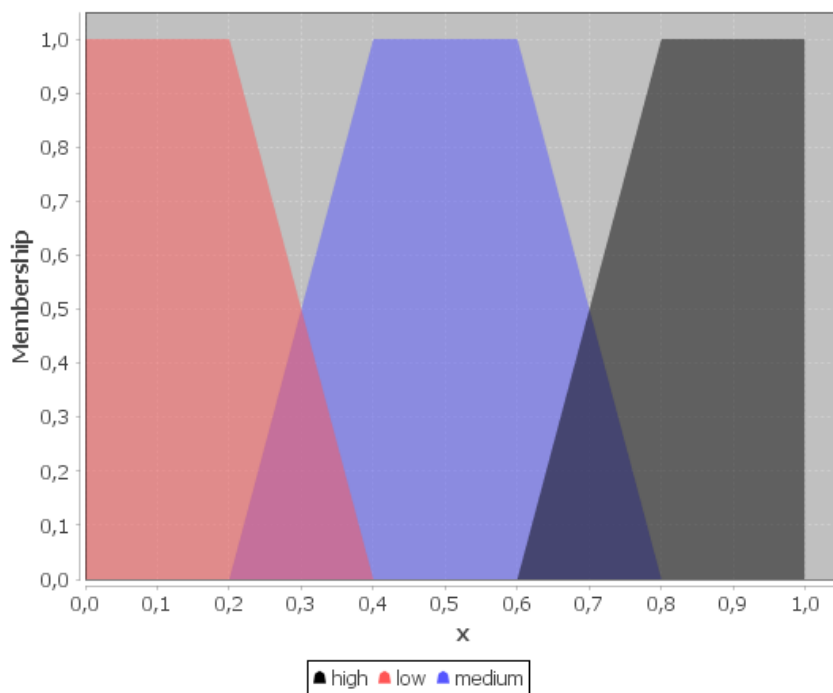
Zatem niezerowy stopień spełnienia poprzednika ma jedynie reguła nr 3. Następnym krokiem działania sterownika jest zbudowanie wyjściowego zbioru rozmytego w oparciu o regułę nr 3 i definicję zmiennej wyjściowej — iloczynu zbioru będącego interpretacją terminu *medium* dla zmiennej **kompetencjaML** (bo tak wynika z następnika reguły nr 3) oraz zbioru, którego funkcja przynależności zawsze przyjmuje wartość wynikającą z poprzednika reguły nr 3 — czyli wartości 0,30. Ostatnim elementem działania sterownika jest wyostrzenie tak powstałego zbioru rozmytego. Zgodnie z powyższym wykorzystana zostanie metoda środka ciężkości. Zatem wartość po wyostrzeniu tego zbioru wynosi w przybliżeniu 0,63. Oznacza to, że zgodnie z prezentowaną w rozdziale 5.2 strategią adaptacji, student przy bieżącym stanie punktów zdobytych w testach jeszcze nie może uzyskać swobodnego dostępu do całego kursu (należy uzyskać co najmniej wartość 0,8 zgodnie z algorytmem adaptacji prezentowanym na rysunku 5.3 w rozdziale 5.2).

### 5.3.2. Sterownik obliczający wartość *question exchange requirement*

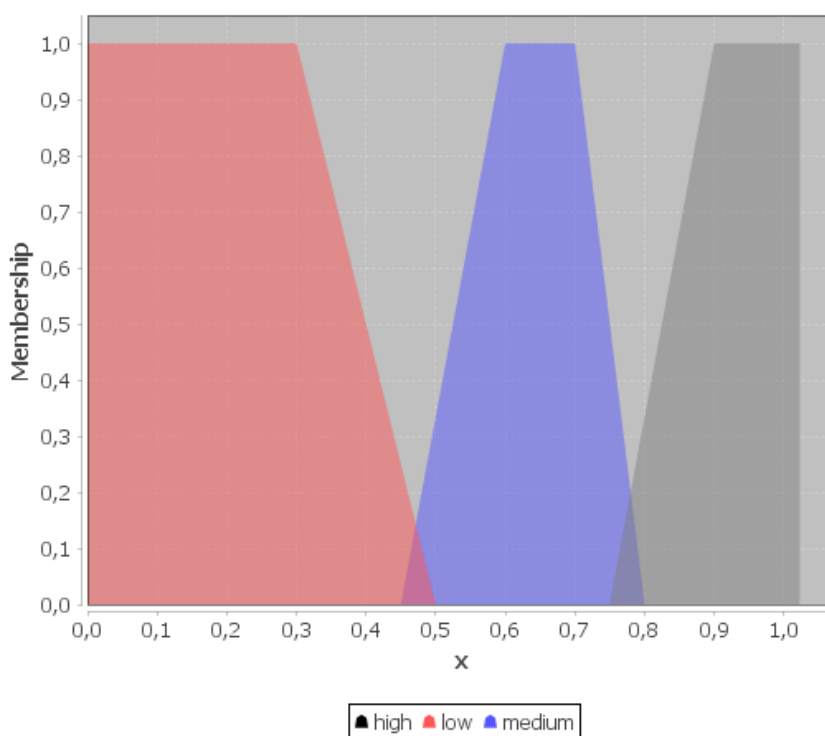
Zgodnie ze algorytmem adaptacji treści w kursie przedstawionym na rysunku 5.3, częścią strategii adaptacji w kursie „Wstęp do uczenia maszynowego” jest wykorzystanie wartości współczynnika *question exchange requirement* (**QER**). Wartość ta jest wykorzystywana przy podejmowaniu decyzji o tym, czy w końcowym teście zaliczeniowym student powinien otrzymać pytania, które już wcześniej pojawiały się w kursie czy też nie. Wartość jest obliczana z wykorzystaniem eksperckiego sterownika Mamdaniego w oparciu o dwie zmienne:

- **kompetencjaML** — poziom kompetencji w zakresie podstaw uczenia maszynowego
- wartość obliczana przez ekspercki sterownik prezentowany w podrozdziale 5.3.1,
- **disengagement** — poziom zjawiska disengagement — wartość obliczana przez ekspercki sterownik prezentowany w rozdziale 3.4.

Tak jak w przypadku sterownika *kompetencjaML* przyjęto założenie, że baza reguł w sterowniku obliczającym wartość *question exchange requirement* powinna być jak najmniejsza i jak najbardziej zrozumiała dla eksperta. W związku z tym sterownik opiera się tylko na dwóch zmiennych a każda z nich jest definiowana w oparciu o trzy terminy: (*low, medium, high*). Interpretacje terminów w przypadku obu zmiennych są identyczne. Przedstawia to rysunek 5.5. W bloku rozmywania zgodnie z modelem sterownika Mamdaniego (zob. rozdział 2.6.2) należy zdefiniować zmienną wyjściową. Została ona określona również na trzech terminach, co przedstawia rysunek 5.6.



Rysunek 5.5. Model zbioru wartości obu zmiennych wejściowych definiowanych w sterowniku obliczającym *question exchange requirement*



Rysunek 5.6. Model zbioru wartości zmiennej wyjściowej definiowanej w sterowniku obliczającym *question exchange requirement*

Kolejnym krokiem w opracowaniu sterownika jest zdefiniowanie bazy reguł. Reguły zaprezentowane są w tabeli 5.2. Tabela przedstawia wszystkie reguły wykorzystane w sterowniku. Przy budowie reguł przyjęto założenie, że bardziej na poziom współczynnika *question exchange requirement* wpływa *disengagement*. To, że student ma niską kompetencję w zakresie podstaw uczenia maszynowego nie musi wynikać z niskiej aktywności w kursie, zatem takiego studenta nie należy „karać” tak bardzo jak studenta, który rzeczywiście ma niską aktywność w kursie.

Lp.	Reguła
1.	IF <b>disengagement</b> IS <i>low</i> AND <b>kompetencjaML</b> IS <i>high</i> THEN <b>QER</b> IS <i>low</i>
2.	IF <b>disengagement</b> IS <i>low</i> AND <b>kompetencjaML</b> IS <i>medium</i> THEN <b>QER</b> IS <i>low</i>
3.	IF <b>disengagement</b> IS <i>low</i> AND <b>kompetencjaML</b> IS <i>low</i> THEN <b>QER</b> IS <i>medium</i>
4.	IF <b>disengagement</b> IS <i>medium</i> AND <b>kompetencjaML</b> IS <i>high</i> THEN <b>QER</b> IS <i>low</i>
5.	IF <b>disengagement</b> IS <i>medium</i> AND <b>kompetencjaML</b> IS <i>medium</i> THEN <b>QER</b> IS <i>medium</i>
6.	IF <b>disengagement</b> IS <i>medium</i> AND <b>kompetencjaML</b> IS <i>low</i> THEN <b>QER</b> IS <i>medium</i>
7.	IF <b>disengagement</b> IS <i>high</i> AND <b>kompetencjaML</b> IS <i>high</i> THEN <b>QER</b> IS <i>medium</i>
8.	IF <b>disengagement</b> IS <i>high</i> AND <b>kompetencjaML</b> IS <i>medium</i> THEN <b>QER</b> IS <i>high</i>
9.	IF <b>disengagement</b> IS <i>high</i> AND <b>kompetencjaML</b> IS <i>low</i> THEN <b>QER</b> IS <i>high</i>

**Tabela 5.2.** Baza reguł wykorzystanych w sterowniku obliczającym poziom współczynnika *question exchange requirement*

Baza reguł sterownika została opracowana wspólnie przez autora rozprawy i autora kursu — mgra inż. Grzegorza Gapińskiego.

Tak jak w przypadku sterownika przedstawionego w rozdziale 5.3.1, w poprzednikach reguł obie zmienne wejściowe połączone są ze sobą spójnikiem koniunkcji. Ten spójnik oraz operator THEN są realizowane operacją minimum. Zbiór powstały w wyniku sprawdzenia wszystkich reguł dla konkretnych wartości zmiennych wejściowych sterownika jest wyostrzony metodą środka ciężkości. Działanie prezentowanego sterownika prezentuje przykład 5.2.

**Przykład 5.2.** Na wejściu sterownik otrzymuje następujące wartości obliczone przez sterowniki *disengagement* (zob. rozdział 3.4) i *kompetencjaML* (zob. rozdział 5.3.1):

- *disengagement* = 0,30,
- *kompetencjaML* = 0,70.

W bloku rozmywania sterownika otrzymane wartości zostaną rozmyte zgodnie z definicjami zmiennych wejściowych (zob. rysunek 5.5):

**disengagement:**

- $low(0,30) = 0,50$
- $medium(0,30) = 0,50$
- $high(0,30) = 0,00$

**kompetencjaML:**

- $low(0,70) = 0,00$
- $medium(0,70) = 0,50$
- $high(0,70) = 0,50$

Po rozmyciu wartości wejściowych, sterownik przechodzi do bloku wnioskowania w oparciu o bazę reguł (zob. tabela 5.2). Dla każdej reguły na podstawie wcześniej zfuzyfikowanych wartości zmiennych wejściowych należy wyznaczyć stopień spełnienia poprzednika reguły:

**Reguła 1.**  $0,50 \wedge 0,50 = 0,50$

**Reguła 2.**  $0,50 \wedge 0,50 = 0,50$

**Reguła 3.**  $0,50 \wedge 0,00 = 0,00$

**Reguła 4.**  $0,50 \wedge 0,50 = 0,50$

**Reguła 5.**  $0,50 \wedge 0,50 = 0,50$

**Reguła 6.**  $0,50 \wedge 0,00 = 0,00$

**Reguła 7.**  $0,00 \wedge 0,50 = 0,00$

**Reguła 8.**  $0,00 \wedge 0,50 = 0,00$

**Reguła 9.**  $0,00 \wedge 0,00 = 0,00$

Zatem niezerowy stopień spełnienia poprzednika mają reguły nr 1, 2, 4 oraz 5. Następnym krokiem działania sterownika jest zbudowanie wyjściowego zbioru rozmytego w oparciu o te reguły i definicję zmiennej wyjściowej — sumy zbiorów wynikających z każdej reguły (czyli zbiorów będących iloczynami zbiorów będących interpretacją termu zmiennej wyjściowej występującego w następniku danej reguły oraz zbioru, którego funkcja przynależności zawsze przyjmuje wartość wynikającą z poprzednika danej reguły). Ostatnim krokiem jest wyostrenie tak otrzymanego zbioru rozmytego. Zgodnie z powyższym zostanie wykorzystana metoda środka ciężkości. Zatem wartość po wyostreniu tego zbioru wynosi w przybliżeniu 0,38. Oznacza to, że zgodnie z prezentowaną strategią adaptacji w rozdziale 5.2, student otrzyma test z pytaniami, które już wcześniej zostały wykorzystane w testach cząstkowych.

### 5.3.3. Dyskusja

Zaprezentowana w niniejszym rozdziale strategia dydaktyczna rozszerza zastosowanie sterownika eksperckiego obliczającego poziom zjawiska disengagement a pokazuje pełne wykorzystanie architektury kursu z adaptacyjną zawartością prezentowaną w rozdziale 3.5. Ta architektura jest na tyle ogólna, że umożliwiła wdrożenie kursu, którego strategia adaptacji opierała się na jednym pojęciu (zob. rozdział 3.3) oraz kursu ze strategią wykorzystującą trzy pojęcia (zob. podrozdział 5.2). W przypadku obu strategii wykorzystano eksperckie sterowniki rozmyte, dzięki czemu do wdrożenia tych strategii nie były potrzebne żadne zbiory danych. W realizacji rozszerzonej strategii dydaktycznej prezentowanej w podrozdziale 5.2 wykorzystany został omawiany w rozprawie sterownik obliczający poziom zjawiska disengagement. Jego wykorzystanie było możliwe dzięki temu, że nie został zbudowany z myślą o wyłącznie jednym kursie e-learningowym, zatem mógł być wykorzystany ponownie w kursie z zupełnie inną strategią dydaktyczną. To pokazuje, że eksperckie sterowniki rozmyte mogą być wykorzystane w różnych zastosowaniach, o ile modele zmiennych oraz baza reguł będą wystarczająco ogólne.



## ROZDZIAŁ 6

# Podsumowanie

W niniejszej rozprawie zaprezentowano dwie strategie adaptacji treści w kursach e-learningowych, które wykorzystują sterownik rozmyty jako główny element algorytmu klasyfikującego studentów [102]. Przedstawione w rozdziałach 3.3 i 5.2 strategie w połączeniu z ogólną architekturą kursu z adaptacyjną zawartością (zob. rozdział 3.5), umożliwiły identyfikację zjawiska disengagement [33], którego poziom wpływał na treści prezentowane studentom. Ze względu na nieprecyzyjny charakter tego zjawiska oraz parametrów opisujących aktywność studentów w kursie (zob. rozdziały 2.4.2 i 2.5.2), zostały one zamodelowane w oparciu o teorię zbiorów rozmytych (zob. rozdział 3.4).

W problemie klasyfikacji studentów pod kątem występowania zjawiska disengagement zastosowano sterownik rozmyty, co było uzasadnione tym, że w takim rozwiązaniu nie jest wymagany zbiór danych do uczenia (rzadko kiedy dydaktycy posiadają takie zbiory), a jedynie zbiór reguł definiowanych przez eksperta (w tym wypadku dydaktyka). Reguły te są tworzone w oparciu o język naturalny, który charakteryzuje się nieprecyzyjnością, co ułatwia ekspertom ich tworzenie. Zastosowanie teorii zbiorów rozmytych pozwala na uwzględnienie niepewności i subiektywnych ocen, co jest szczególnie ważne w kontekście edukacji, gdzie parametry takie jak zaangażowanie czy postępy w nauce są trudne do jednoznacznego zdefiniowania.

Sterowniki rozmyte mogą również być generowane automatycznie na podstawie posiadanego zbioru danych i metod klastrowania wykorzystywanych w uczeniu maszynowym. Proces ten został przeprowadzony zarówno dla sterowników rozmytych Mamdaniego, jak i Takagi-Sugeno. Przedstawiona strategia dydaktyczna została wdrożona w kursie e-learningowym, który został wykorzystany na zajęciach dydaktycznych, co umożliwiło zebranie danych do późniejszego generowania nowych sterowników. Odpowiednią postać danych do dalszego przetwarzania zapewniło wykorzystanie biblioteki *its-raport-generator* (zob. rozdział 3.1.4). Dane zostały następnie anotowane przez grupę anotatorów. Na potrzeby klasyfikacji studentów wszystkie sterowniki porównano z najpopularniejszymi algorytmami uczenia maszynowego, takimi jak regresja logistyczna, drzewa decyzyjne czy lasy losowe. Do ewaluacji wybrano metryki powszechnie stosowane przy ocenie algorytmów klasyfikujących, takie jak dokładność, precyzja, czułość i miara F1. Takie podejście jest zgodne z metodologią badań w obszarze uczenia maszynowego, gdzie standardy oceny muszą być jasno określone i umożliwiać porównywanie różnych metod na tych samych danych [88, 94].

Wyniki przeprowadzonego porównania metod klasyfikacji studentów na potrzeby identyfikacji zjawiska disengagement wskazują, że sterowniki rozmyte mogą być skutecz-

nie stosowane przy rozwiązywaniu problemów tego typu. Wartości metryk dla sterowników rozmytych są porównywalne z wartościami dla algorytmów uczenia maszynowego. Obiecujące są wyniki uzyskane dla eksperckiego sterownika rozmytego, który nie wymagał żadnego zbioru danych do uczenia. Ten wynik wskazuje, że możliwe jest zastosowanie sterowników rozmytych w sytuacjach, gdzie dostęp do danych jest ograniczony lub ich zebranie jest trudne. Należy jednakże w kolejnych badaniach sprawdzić, jak będzie działać ekspercki sterownik rozmyty obliczający nasilenie zjawiska disengagement zbudowany przez różnych doświadczonych dydaktyków. Być może wybór innych zmiennych wejściowych lub inna baza reguł znacząco wpłyną na działanie sterownika. Różne perspektywy i doświadczenia dydaktyków mogą prowadzić do opracowania bardziej uniwersalnych i efektywnych reguł, które lepiej uwzględniają różnorodność zachowań studentów.

W ramach badań, poza strategią adaptacji opartą jedynie na zjawisku disengagement, wdrożono również rozszerzoną strategię wykorzystującą disengagement oraz dwa inne pojęcia: poziom kompetencji w zakresie podstaw uczenia maszynowego i współczynnik *question exchange requirement* wyliczany w oparciu o zjawisko disengagement a także wspomnianą kompetencję w zakresie podstaw uczenia maszynowego, którego zadaniem jest wyznaczenie procentowej liczby pytań, które należy wymienić w teście zaliczeniowym w kursie. Pojęcia te również zamodelowane zostały z wykorzystaniem eksperckich sterowników rozmytych. Omawiane w rozprawie strategie dydaktyczne wykorzystujące wspomniane sterowniki rozmyte potwierdzają użyteczność zaproponowanej ogólnej architektury adaptacyjnego kursu prezentowanej w rozdziale 3.5 oraz jej produkcyjnego charakteru, charakteru tzn. zdolności do wykorzystania przy implementacji różnych strategii adaptacji w kursach o różnej postaci i kierowanych do różnych grup odbiorców.

W kontekście e-learningu, adaptacja treści może obejmować nie tylko reakcję na poziom zjawiska disengagement czy kompetencji w zakresie jakiejś dziedziny, ale także na różne style uczenia się, tempo przyswajania wiedzy, czy poziom trudności zadań. Kontynuacja badań nad sterowaniem rozmytym w kontekście kursów z adaptacyjną zawartością wymaga porównania sterowników, wykorzystanych w ramach rozszerzonej strategii adaptacji, z metodami uczenia maszynowego, podobnie jak w przypadku sterownika obliczającego poziom nasilenia zjawiska disengagement. Takie porównanie jest konieczne, aby potwierdzić (lub zaprzeczyć) zasadność stosowania sterowników rozmytych w tego typu problemach. Weryfikacja działania różnych metod w kontekście realnych kursów e-learningowych pozwoli na lepsze zrozumienie ich zalet i ograniczeń. Badania w tym obszarze wymagają zebrania i przygotowania danych, która powinny pochodzić z rzeczywistych kursów realizowanych przez studentów o różnym poziomie kompetencji i zaangażowania w naukę.

Istotne jest, że zaprezentowane strategie adaptacji treści mają charakter produkcyjny, tzn. można je stosować w kursach o różnej postaci, ponieważ zostały wykorzystane w dwóch różnych kursach e-learningowych wdrożonych na zajęciach dydaktycznych. Do ich realizacji użyto sterowników rozmytych, ze względu na ich przejrzystość, prostotę implementacji oraz łatwość integracji z zewnętrznymi systemami (w tym przypadku – kursami e-learningowymi). Wykorzystanie sterowników rozmytych w kursach było możliwe dzięki zastosowaniu opracowanych narzędzi — konwertera *fcl2js* oraz biblioteki *its-student-progress* (zob. rozdziały 3.1.2 i 3.1.3). Dzięki zastosowanej architekturze, wszelkie obliczenia, choć niewymagające dużych zasobów, były przeprowadzane

po stronie przeglądarki studenta, eliminując potrzebę posiadania zewnętrznego serwera do klasyfikacji studentów, ponieważ nigdzie nie trzeba przechowywać wytrenowanego modelu algorytmu klasyfikującego i wykonywać w oparciu o ten model obliczeń. Takie rozwiązania są szczególnie korzystne w kontekście dużych instytucji edukacyjnych, gdzie infrastruktura IT może być ograniczona, a jednocześnie istnieje potrzeba indywidualizacji kształcenia. Przeprowadzanie obliczeń po stronie przeglądarki studenta redukuje koszty operacyjne i umożliwia skalowalność systemu.

Indywidualizacja kształcenia staje się coraz ważniejsza w dobie masowej edukacji online, gdzie studenci różnią się pod względem umiejętności, zainteresowań i stylów uczenia się. Rozwiązania zaprezentowane w rozprawie otwierają drogę do tworzenia bardziej zaawansowanych strategii umożliwiających większą indywidualizację kształcenia, wykorzystując różne zjawiska opisujące sposób pracy studenta z kursem, szczególnie w sytuacji, gdy studenci pracują bez nadzoru nauczyciela, w wybranym przez siebie rytmie i czasie.

Przedstawione w rozprawie wykorzystanie sterowników rozmytych ukazuje kolejne zastosowanie sterowania rozmytego, podkreślając potrzebę dalszych badań nad wykorzystaniem tej metody w praktycznych problemach. Teoria zbiorów rozmytych oraz sterowanie rozmyte mogą znaleźć zastosowanie nie tylko w adaptacji treści edukacyjnych, ale także w innych obszarach, takich jak diagnostyka edukacyjna czy ocena postępów studentów. Wykorzystanie sterowników rozmytych w kontekście dydaktyki jest zasadne, ponieważ praktycznie natychmiastowo zwracają wyniki, ich działanie co wykazano w rozprawie jest porównywalne z metodami uczenia maszynowego a w dodatku są bardziej od nich przejrzyste, ponieważ w każdej chwili ekspert w danej dziedzinie ma wgląd w bazę reguł sterownika i może ją dostosowywać do konkretnych potrzeb. Oznacza to, że dalsze badania w obszarze sterowania rozmytego wykorzystywanego w edukacji pozwolą na opracowanie bardziej efektywnych i złożonych systemów e-learningowych, które lepiej odpowiadają na zróżnicowane potrzeby uczących się.



## ZAŁĄCZNIK A

# Ekspercki sterownik Mamdaniego przekonwertowany do języka JavaScript przy pomocy konwertera *fcl2js*

```
1 function ruleAccumulationMethod_max(defuzzifierValue, valueToAggregate) {
2   return (defuzzifierValue > valueToAggregate ? defuzzifierValue
3     : valueToAggregate);
4 }
5
6 function ruleActivationMethod_min(degreeOfSupport, membership) {
7   return (degreeOfSupport < membership ? degreeOfSupport : membership);
8 }
9
10 function ruleConnectionMethod_and(antecedent1, antecedent2) {
11   return (antecedent1 < antecedent2 ? antecedent1 : antecedent2);
12 }
13
14 class FunctionBlock_disenagegementFCL {
15   learning_quality;
16   remaining_time;
17
18   // VAR_OUTPUT
19   disengagement;
20
21   // FUZZIFY learning_quality
22   learning_quality_high;
23   learning_quality_low;
24   learning_quality_medium;
25
26   // FUZZIFY remaining_time
27   remaining_time_high;
28   remaining_time_low;
29   remaining_time_medium;
30
31   // DEFUZZIFY disengagement
32   defuzzify_disengagement;
33
34   // Constructor
35   constructor() {
```

```
36     this.disengagement = 0.5;
37     this.defuzzify_disengagement = new Array(1000);
38 }
39
40 // Calculate function block
41 calc() {
42     this.reset();
43     this.fuzzify();
44     this.calc_No1();
45     this.defuzzify();
46 }
47
48 // Print
49 print() {
50     console.log('Function block students:');
51     console.log('      Output %s: %f', 'disengagement',
52         this.disengagement);
53     console.log('      Input  %s: %f', 'learning_quality',
54         this.learning_quality);
55     console.log('      %s: %f', 'learning_quality_high',
56         this.learning_quality_high);
57     console.log('      %s: %f', 'learning_quality_low',
58         this.learning_quality_low);
59     console.log('      %s: %f', 'learning_quality_medium',
60         this.learning_quality_medium);
61     console.log('      Input  %s: %f', 'remaining_time',
62         this.remaining_time);
63     console.log('      %s: %f', 'remaining_time_high',
64         this.remaining_time_high);
65     console.log('      %s: %f', 'remaining_time_low',
66         this.remaining_time_low);
67     console.log('      %s: %f', 'remaining_time_medium',
68         this.remaining_time_medium);
69 }
70
71 // Defuzzify
72 defuzzify() {
73     let sum_out = 0.0;
74     let wsum_out = 0.0;
75     let x;
76
77     for (let i = 0; i < 1000 ; i++) {
78         x = 0.0 + i * 0.001;
79         sum_out += this.defuzzify_disengagement[i];
80         wsum_out += x * this.defuzzify_disengagement[i];
81     }
82
83     this.disengagement = wsum_out / sum_out;
84 }
85
86 // Fuzzify all variables
```

```
87     fuzzify() {
88         this.learning_quality_high =
89             this.membership_learning_quality_high(this.learning_quality);
90         this.learning_quality_low =
91             this.membership_learning_quality_low(this.learning_quality);
92         this.learning_quality_medium =
93             this.membership_learning_quality_medium(this.learning_quality);
94         this.remaining_time_high =
95             this.membership_remaining_time_high(this.remaining_time);
96         this.remaining_time_low =
97             this.membership_remaining_time_low(this.remaining_time);
98         this.remaining_time_medium =
99             this.membership_remaining_time_medium(this.remaining_time);
100     }
101
102     // Reset output
103     reset() {
104         for (let i=0 ; i < 1000 ; i++) {
105             this.defuzzify_disengagement[i] = 0.0;
106         }
107     }
108
109     // Membership functions
110     membership_out_high(x) {
111         if (x <= 0.6)         return 0.0;
112         if (x > 1.0)         return 1.0;
113         if (x <= 0.8)         return 0.0 + (1.0 - 0.0) * ((x - 0.6) / (0.8 - 0.6));
114         if (x <= 1.0)         return 1.0 + (1.0 - 1.0) * ((x - 0.8) / (1.0 - 0.8));
115     }
116
117     membership_out_low(x) {
118         if (x <= 0.0)         return 1.0;
119         if (x > 0.4)         return 0.0;
120         if (x <= 0.2)         return 1.0 + (1.0 - 1.0) * ((x - 0.0) / (0.2 - 0.0));
121         if (x <= 0.4)         return 1.0 + (0.0 - 1.0) * ((x - 0.2) / (0.4 - 0.2));
122     }
123
124     membership_out_medium(x) {
125         if (x <= 0.2)         return 0.0;
126         if (x > 0.8)         return 0.0;
127         if (x <= 0.4)         return 0.0 + (1.0 - 0.0) * ((x - 0.2) / (0.4 - 0.2));
128         if (x <= 0.6)         return 1.0 + (1.0 - 1.0) * ((x - 0.4) / (0.6 - 0.4));
129         if (x <= 0.8)         return 1.0 + (0.0 - 1.0) * ((x - 0.6) / (0.8 - 0.6));
130     }
131
132     membership_learning_quality_high(x) {
133         if (x <= 0.6)         return 0.0;
134         if (x > 1.0)         return 1.0;
135         if (x <= 0.8)         return 0.0 + (1.0 - 0.0) * ((x - 0.6) / (0.8 - 0.6));
136         if (x <= 1.0)         return 1.0 + (1.0 - 1.0) * ((x - 0.8) / (1.0 - 0.8));
137     }
```

```

138
139 membership_learning_quality_low(x) {
140     if (x <= 0.0)         return 1.0;
141     if (x > 0.4)         return 0.0;
142     if (x <= 0.2)         return 1.0 + (1.0 - 1.0) * ((x - 0.0) / (0.2 - 0.0));
143     if (x <= 0.4)         return 1.0 + (0.0 - 1.0) * ((x - 0.2) / (0.4 - 0.2));
144 }
145
146 membership_learning_quality_medium(x) {
147     if (x <= 0.2)         return 0.0;
148     if (x > 0.8)         return 0.0;
149     if (x <= 0.4)         return 0.0 + (1.0 - 0.0) * ((x - 0.2) / (0.4 - 0.2));
150     if (x <= 0.6)         return 1.0 + (1.0 - 1.0) * ((x - 0.4) / (0.6 - 0.4));
151     if (x <= 0.8)         return 1.0 + (0.0 - 1.0) * ((x - 0.6) / (0.8 - 0.6));
152 }
153
154 membership_remaining_time_high(x) {
155     if (x <= 0.6)         return 0.0;
156     if (x > 1.0)         return 1.0;
157     if (x <= 0.8)         return 0.0 + (1.0 - 0.0) * ((x - 0.6) / (0.8 - 0.6));
158     if (x <= 1.0)         return 1.0 + (1.0 - 1.0) * ((x - 0.8) / (1.0 - 0.8));
159 }
160
161 membership_remaining_time_low(x) {
162     if (x <= 0.0)         return 1.0;
163     if (x > 0.4)         return 0.0;
164     if (x <= 0.2)         return 1.0 + (1.0 - 1.0) * ((x - 0.0) / (0.2 - 0.0));
165     if (x <= 0.4)         return 1.0 + (0.0 - 1.0) * ((x - 0.2) / (0.4 - 0.2));
166 }
167
168 membership_remaining_time_medium(x) {
169     if (x <= 0.2)         return 0.0;
170     if (x > 0.8)         return 0.0;
171     if (x <= 0.4)         return 0.0 + (1.0 - 0.0) * ((x - 0.2) / (0.4 - 0.2));
172     if (x <= 0.6)         return 1.0 + (1.0 - 1.0) * ((x - 0.4) / (0.6 - 0.4));
173     if (x <= 0.8)         return 1.0 + (0.0 - 1.0) * ((x - 0.6) / (0.8 - 0.6));
174 }
175
176 // RULEBLOCK No1
177 calc_No1() {
178     // RULE 1 : IF (learning_quality IS low) AND (remaining_time IS low)
179     // THEN disengagement IS high;
180     let degreeOfSupport_1 = 1.0 * (ruleConnectionMethod_and(
181         this.learning_quality_low, this.remaining_time_low));
182
183     if (degreeOfSupport_1 > 0) {
184         for (let i = 0 ; i < 1000 ; i++) {
185             let x = 0.0 + i * 0.001;
186             let membership = this.membership_out_high(x);
187             let y = ruleActivationMethod_min(degreeOfSupport_1, membership);
188             this.defuzzify_disengagement[i] = ruleAccumulationMethod_max(

```



```
189         this.defuzzify_disengagement[i], y);
190     }
191 }
192
193 // RULE 2 : IF (learning_quality IS low) AND (remaining_time IS medium)
194 // THEN disengagement IS medium;
195 let degreeOfSupport_2 = 1.0 * (ruleConnectionMethod_and(
196     this.learning_quality_low, this.remaining_time_medium));
197 if (degreeOfSupport_2 > 0) {
198     for (let i = 0 ; i < 1000 ; i++) {
199         let x = 0.0 + i * 0.001;
200         let membership = this.membership_out_medium(x);
201         let y = ruleActivationMethod_min(degreeOfSupport_2, membership);
202         this.defuzzify_disengagement[i] = ruleAccumulationMethod_max(
203             this.defuzzify_disengagement[i], y);
204     }
205 }
206
207 // RULE 3 : IF (learning_quality IS low) AND (remaining_time IS high)
208 // THEN disengagement IS medium;
209 let degreeOfSupport_3 = 1.0 * (ruleConnectionMethod_and(
210     this.learning_quality_low, this.remaining_time_high));
211 if (degreeOfSupport_3 > 0) {
212     for (let i = 0 ; i < 1000 ; i++) {
213         let x = 0.0 + i * 0.001;
214         let membership = this.membership_out_medium(x);
215         let y = ruleActivationMethod_min(degreeOfSupport_3, membership);
216         this.defuzzify_disengagement[i] = ruleAccumulationMethod_max(
217             this.defuzzify_disengagement[i], y);
218     }
219 }
220
221 // RULE 4 : IF (learning_quality IS medium) AND (remaining_time IS low)
222 // THEN disengagement IS medium;
223 let degreeOfSupport_4 = 1.0 * (ruleConnectionMethod_and(
224     this.learning_quality_medium, this.remaining_time_low));
225 if (degreeOfSupport_4 > 0) {
226     for (let i = 0 ; i < 1000 ; i++) {
227         let x = 0.0 + i * 0.001;
228         let membership = this.membership_out_medium(x);
229         let y = ruleActivationMethod_min(degreeOfSupport_4, membership);
230         this.defuzzify_disengagement[i] = ruleAccumulationMethod_max(
231             this.defuzzify_disengagement[i], y);
232     }
233 }
234
235 // RULE 5 : IF (learning_quality IS medium) AND (remaining_time IS medium)
236 // THEN disengagement IS medium;
237 let degreeOfSupport_5 = 1.0 * (ruleConnectionMethod_and(
238     this.learning_quality_medium, this.remaining_time_medium));
239 if (degreeOfSupport_5 > 0) {
```

```
240     for (let i = 0 ; i < 1000 ; i++) {
241         let x = 0.0 + i * 0.001;
242         let membership = this.membership_out_medium(x);
243         let y = ruleActivationMethod_min(degreeOfSupport_5, membership);
244         this.defuzzify_disengagement[i] = ruleAccumulationMethod_max(
245             this.defuzzify_disengagement[i], y);
246     }
247 }
248
249 // RULE 6 : IF (learning_quality IS medium) AND (remaining_time IS high)
250 // THEN disengagement IS low;
251 let degreeOfSupport_6 = 1.0 * (ruleConnectionMethod_and(
252     this.learning_quality_medium, this.remaining_time_high));
253 if (degreeOfSupport_6 > 0) {
254     for (let i = 0 ; i < 1000 ; i++) {
255         let x = 0.0 + i * 0.001;
256         let membership = this.membership_out_low(x);
257         let y = ruleActivationMethod_min(degreeOfSupport_6, membership);
258         this.defuzzify_disengagement[i] = ruleAccumulationMethod_max(
259             this.defuzzify_disengagement[i], y);
260     }
261 }
262
263 // RULE 7 : IF (learning_quality IS high) AND (remaining_time IS low)
264 // THEN disengagement IS medium;
265 let degreeOfSupport_7 = 1.0 * (ruleConnectionMethod_and(
266     this.learning_quality_high, this.remaining_time_low));
267 if (degreeOfSupport_7 > 0) {
268     for (let i = 0 ; i < 1000 ; i++) {
269         let x = 0.0 + i * 0.001;
270         let membership = this.membership_out_medium(x);
271         let y = ruleActivationMethod_min(degreeOfSupport_7, membership);
272         this.defuzzify_disengagement[i] = ruleAccumulationMethod_max(
273             this.defuzzify_disengagement[i], y);
274     }
275 }
276
277 // RULE 8 : IF (learning_quality IS high) AND (remaining_time IS medium)
278 // THEN disengagement IS low;
279 let degreeOfSupport_8 = 1.0 * (ruleConnectionMethod_and(
280     this.learning_quality_high, this.remaining_time_medium));
281 if (degreeOfSupport_8 > 0) {
282     for (let i = 0 ; i < 1000 ; i++) {
283         let x = 0.0 + i * 0.001;
284         let membership = this.membership_out_low(x);
285         let y = ruleActivationMethod_min(degreeOfSupport_8, membership);
286         this.defuzzify_disengagement[i] = ruleAccumulationMethod_max(
287             this.defuzzify_disengagement[i], y);
288     }
289 }
290 }
```

```
291     // RULE 9 : IF (learning_quality IS high) AND (remaining_time IS high)
292     // THEN disengagement IS low;
293     let degreeOfSupport_9 = 1.0 * (ruleConnectionMethod_and(
294       this.learning_quality_high, this.remaining_time_high));
295     if (degreeOfSupport_9 > 0) {
296       for (let i = 0 ; i < 1000 ; i++) {
297         let x = 0.0 + i * 0.001;
298         let membership = this.membership_out_low(x);
299         let y = ruleActivationMethod_min(degreeOfSupport_9, membership);
300         this.defuzzify_disengagement[i] = ruleAccumulationMethod_max(
301           this.defuzzify_disengagement[i], y);
302       }
303     }
304   }
305 }
306
307 function calcDisengagement(learning_quality, remaining_time) {
308   // Create function blocks
309   const disenagementFCL = new FunctionBlock_disenagementFCL();
310
311   disenagementFCL.learning_quality = learning_quality;
312   disenagementFCL.remaining_time = remaining_time;
313   disenagementFCL.calc();
314   disenagementFCL.print();
315
316   return disenagementFCL.disengagement;
317 }
```

**Wyciąg A.1.** Ekspercki sterownik Mamdaniego przekonwertowany do języka JavaScript przy pomocy konwertera *fcl2js*



## ZAŁĄCZNIK B

# Kod języka Python wykorzystujący bibliotekę pyFume do generowania sterownika Takagi-Sugeno

```
1 from pyfume import *
2 import pandas as pd
3
4 def r():
5     # Set the path to the data and choose the number of clusters
6     path='.././train_binary.csv'
7     testPath = '.././test_binary.csv'
8     nr_clus=3
9
10    # Load and normalize the data using min-max normalization
11    dl=DataLoader(path,normalize='minmax',delimiter=';')
12    test=DataLoader(testPath,normalize='minmax',delimiter=';')
13    variable_names=dl.variable_names
14
15    x_train = dl.dataX
16    y_train = dl.dataY
17
18    x_test = test.dataX
19    y_test = test.dataY
20
21    # Select features relevant to the problem
22    fs=FeatureSelector(dataX=x_train, dataY=y_train, nr_clus=nr_clus,
23                       variable_names=variable_names)
24    selected_feature_indices, variable_names=fs.wrapper()
25
26    # Adapt the training and test input data after feature selection
27    x_train = x_train[:, selected_feature_indices]
28    x_test = x_test[:, selected_feature_indices]
29
30    # Cluster the training data (in input-output space) using FCM with default
31    # settings
32    cl = Clusterer(x_train=x_train, y_train=y_train, nr_clus=nr_clus)
33    cluster_centers, partition_matrix, _ = cl.cluster(method="fcm")
34
35    # Estimate the membership functions of the system (default: mf_shape
```

```
36 # = gaussian)
37 ae = AntecedentEstimator(x_train=x_train, partition_matrix=partition_matrix)
38 antecedent_parameters = ae.determineMF()
39
40 # Calculate the firing strength of each rule for each data instance
41 fsc=FireStrengthCalculator(antecedent_parameters=antecedent_parameters,
42                             nr_clus=nr_clus, variable_names=variable_names)
43 firing_strengths = fsc.calculate_fire_strength(data=x_train)
44
45 # Estimate the parameters of the consequent functions
46 ce = ConsequentEstimator(x_train=x_train, y_train=y_train,
47                           firing_strengths=firing_strengths)
48 consequent_parameters = ce.suglms()
49
50 # Build a first-order Takagi-Sugeno model using Simpful. Specify the optional
51 # 'extreme_values' argument to specify the universe of discourse of the input
52 # variables if you which to use Simpful's membership function plot
53 # functionalities.
54 simpbuilder = SugenoFISBuilder(antecedent_sets=antecedent_parameters,
55                                consequent_parameters=consequent_parameters,
56                                variable_names=variable_names,
57                                save_simpful_code=True)
58 model = simpbuilder.get_model()
59
60 # Calculate the mean squared error (MSE) of the model using the test data set
61 test=SugenoFISTester(model=model, test_data=x_test,
62                       variable_names=variable_names,
63                       golden_standard=y_test)
64 MSE = test.calculate_MSE()
65
66 print('The mean squared error of the created model is', MSE)
67
68
69 if __name__ == "__main__":
70     r()
```

**Wyciąg B.1.** Kod języka Python wykorzystujący bibliotekę pyFume do generowania sterownika Takagi-Sugeno

## ZAŁĄCZNIK C

# Kod języka Python prezentujący wygenerowany sterownik Takagi-Sugeno

```
1 from simpful import *
2
3 FS = FuzzySystem(show_banner=False)
4 RULE1 = '''IF (learningQuality IS cluster1)
5 AND (timeSpentPointsNormalized IS cluster1)
6 AND (numberOfStudentInteractionsNormalized IS cluster1)
7 AND (remainingTime IS cluster1)
8 THEN (OUTPUT IS fun1)'''
9 RULE2 = '''IF (learningQuality IS cluster2)
10 AND (timeSpentPointsNormalized IS cluster2)
11 AND (numberOfStudentInteractionsNormalized IS cluster2)
12 AND (remainingTime IS cluster2)
13 THEN (OUTPUT IS fun2)'''
14 RULE3 = '''IF (learningQuality IS cluster3)
15 AND (timeSpentPointsNormalized IS cluster3)
16 AND (numberOfStudentInteractionsNormalized IS cluster3)
17 AND (remainingTime IS cluster3)
18 THEN (OUTPUT IS fun3)'''
19
20 FS.add_rules([RULE1, RULE2, RULE3])
21
22 FS.set_output_function('fun1', '''7.133643e-01*learningQuality+
23 -9.671679e-01*timeSpentPointsNormalized
24 +-5.803277e-01*numberOfStudentInteractionsNormalized
25 +-1.752722e-01*remainingTime
26 +1.074011e+00''')
27 FS.set_output_function('fun2', '''-3.907557e-01*learningQuality
28 +-6.422901e-01*timeSpentPointsNormalized
29 +-6.280251e-01*numberOfStudentInteractionsNormalized
30 +-2.082751e-01*remainingTime
31 +1.709230e+00''')
32 FS.set_output_function('fun3', '''1.001960e+00*learningQuality
33 +-1.338701e+00*timeSpentPointsNormalized+
34 -7.754339e-01*numberOfStudentInteractionsNormalized
35 +-2.327558e-01*remainingTime+
```

```
36         1.220921e+00''')
37
38 FS_1 = FuzzySet(function=Gaussian_MF(0.142286, 0.414429), term='cluster1')
39 FS_2 = FuzzySet(function=Gaussian_MF(0.877904, 0.291787), term='cluster2')
40 FS_3 = FuzzySet(function=Gaussian_MF(0.619456, 0.316053), term='cluster3')
41 MF_learningQuality = LinguisticVariable([FS_1, FS_2, FS_3],
42                                         concept='learningQuality')
43 FS.add_linguistic_variable('learningQuality', MF_learningQuality)
44
45 FS_4 = FuzzySet(function=Gaussian_MF(-0.074683, 0.730690), term='cluster1')
46 FS_5 = FuzzySet(function=Gaussian_MF(0.768889, 0.510789), term='cluster2')
47 FS_6 = FuzzySet(function=Gaussian_MF(0.412550, 0.483888), term='cluster3')
48 MF_timeSpentPointsNormalized = LinguisticVariable([FS_4, FS_5, FS_6],
49                                                  concept='timeSpentPointsNormalized')
50 FS.add_linguistic_variable('timeSpentPointsNormalized',
51                            MF_timeSpentPointsNormalized)
52
53 FS_7 = FuzzySet(function=Gaussian_MF(0.002858, 0.534426), term='cluster1')
54 FS_8 = FuzzySet(function=Gaussian_MF(0.952167, 0.730268), term='cluster2')
55 FS_9 = FuzzySet(function=Gaussian_MF(0.881851, 0.572307), term='cluster3')
56 MF_numberOfStudentInteractionsNormalized = LinguisticVariable([FS_7, FS_8,
57                                                                FS_9],
58                                                                concept='numberOfStudentInteractionsNormalized')
59 FS.add_linguistic_variable('numberOfStudentInteractionsNormalized',
60                            MF_numberOfStudentInteractionsNormalized)
61
62 FS_10 = FuzzySet(function=Gaussian_MF(0.294346, 0.682729), term='cluster1')
63 FS_11 = FuzzySet(function=Gaussian_MF(0.616341, 0.651170), term='cluster2')
64 FS_12 = FuzzySet(function=Gaussian_MF(0.360132, 0.562283), term='cluster3')
65 MF_remainingTime = LinguisticVariable([FS_10, FS_11, FS_12],
66                                       concept='remainingTime')
67 FS.add_linguistic_variable('remainingTime', MF_remainingTime)
```

**Wyciąg C.1.** Kod języka Python prezentujący wygenerowany sterownik Takagi-Sugeno



## Spis tabel

3.1	Struktura kursu z podziałem logicznym wg UCTS . . . . .	47
3.2	Baza reguł wykorzystanych w proponowanym sterowniku . . . . .	66
4.1	Wybrane wyniki działania eksperckiego sterownika Mamdaniego . . . . .	77
4.2	Fragment zbioru danych anotowanego przez autora rozprawy . . . . .	81
4.3	Fragment zbioru danych anotowanego przez grupę anotatorów po odwróceniu etykiet . . . . .	82
4.4	Fragment zbioru danych anotowanego przez grupę anotatorów po przetworzeniu do klasyfikacji binarnej . . . . .	83
4.5	Porównanie algorytmów w problemie klasyfikacji binarnej na zbiorze danych bez uśredniania anotacji w przypadku występowania duplikatów . . . . .	92
4.6	Porównanie algorytmów w problemie klasyfikacji binarnej na zbiorze danych z uśrednionymi anotacjami w przypadku występowania duplikatów . . . . .	93
4.7	Porównanie algorytmów w problemie klasyfikacji pięcioklasowej na zbiorze danych bez uśredniania anotacji w przypadku występowania duplikatów . . . . .	94
4.8	Porównanie algorytmów w problemie klasyfikacji pięcioklasowej na zbiorze danych z uśrednionymi anotacjami w przypadku występowania duplikatów . . . . .	95
5.1	Baza reguł wykorzystanych w sterowniku obliczającym poziom kompetencji w zakresie podstaw uczenia maszynowego . . . . .	106
5.2	Baza reguł wykorzystanych w sterowniku obliczającym poziom współczynnika <i>question exchange requirement</i> . . . . .	110



## Spis rysunków

2.1	Interpretacje termów zmiennej: <i>liczba odwiedzonych elementów w kursie</i> . . . . .	30
2.2	Model zbioru wartości obu zmiennych wejściowych definiowanych w przykładowym sterowniku . . . . .	37
2.3	Wynik działania bloku wnioskowania w przykładowym sterowniku Mamdaniego . .	40
3.1	Fragment materiałów do nauki w kursie „Zastosowania zbiorów rozmytych” . . . .	48
3.2	Fragment testu zaliczeniowego w kursie „Zastosowania zbiorów rozmytych” . . . .	49
3.3	Architektura kursu z modułem obliczającym poziom badanego zjawiska wykorzystywanego w ramach strategii adaptacji . . . . .	54
3.4	Algorytm adaptacji treści w kursie z adaptacyjną zawartością oparty na zjawisku disengagement . . . . .	59
3.5	Model zbioru wartości obu zmiennych wejściowych definiowanych w proponowanym sterowniku . . . . .	64
3.6	Zbiór wyjściowy będący sumą zbiorów powstałych w oparciu o bazę reguł w proponowanym sterowniku . . . . .	68
4.1	Konfigurowanie narzędzia Fuzzy Logic Designer oraz parametrów algorytmu FCM w środowisku MATLAB w celu wygenerowania sterownika . . . . .	85
5.1	Przykładowy test po jego rozwiązaniu przez studenta w kursie „Wstęp do uczenia maszynowego” z włączonym modułem adaptacji treści . . . . .	100
5.2	Przykładowe ćwiczenie w kursie „Wstęp do uczenia maszynowego” . . . . .	101
5.3	Algorytm adaptacji treści w kursie z rozszerzoną strategią dydaktyczną . . . . .	103
5.4	Model zbioru wartości obu zmiennych wejściowych definiowanych w sterowniku obliczającym poziom kompetencji w zakresie podstaw uczenia maszynowego . . . .	106
5.5	Model zbioru wartości obu zmiennych wejściowych definiowanych w sterowniku obliczającym <i>question exchange requirement</i> . . . . .	109
5.6	Model zbioru wartości zmiennej wyjściowej definiowanej w sterowniku obliczającym <i>question exchange requirement</i> . . . . .	109



## Spis wyciągów

3.1	Kod przykładowego adaptera w kursie z adaptacyjną zawartością . . . . .	56
3.2	Eksperscki sterownik Mamdaniego zapisany w języku FCL . . . . .	69
3.3	Reguła widoczności testu z pytaniami dla osób o niższym niż średniowysoki poziomie zjawiska disengagement . . . . .	71
3.4	Kod źródłowy adaptera pozwalającego na komunikację pomiędzy kursem a sterownikiem rozmytym . . . . .	72
4.1	Sterownik Mamdaniego wygenerowany w środowisku MATLAB na zbiorze danych anotowanych przez studentów-nauczycieli dla klasyfikacji binarnej, gdzie rekordy nie powtarzają się a anotacje studentów-nauczycieli zostały uśrednione	86
A.1	Eksperscki sterownik Mamdaniego przekonwertowany do języka JavaScript przy pomocy konwertera <i>fcl2js</i> . . . . .	117
B.1	Kod języka Python wykorzystujący bibliotekę pyFume do generowania sterownika Takagi-Sugeno . . . . .	125
C.1	Kod języka Python prezentujący wygenerowany sterownik Takagi-Sugeno . . . .	127



## Bibliografia

- [1] Abdallah Shami Abdallah Moubayed Mohammadnoor Injadat i Hanan Lutfiyya. „Student Engagement Level in an e-Learning Environment: Clustering Using K-means”. W: *American Journal of Distance Education* 34.2 (2020), s. 137–156. DOI: 10.1080/08923647.2020.1696140. eprint: <https://doi.org/10.1080/08923647.2020.1696140>. URL: <https://doi.org/10.1080/08923647.2020.1696140>.
- [2] Ajith Abraham. „Rule-Based Expert Systems”. W: *Handbook of Measuring System Design*. John Wiley & Sons, Ltd, 2005. Rozd. 130. ISBN: 9780471497394. DOI: <https://doi.org/10.1002/0471497398.mm422>. eprint: <https://onlinelibrary.wiley.com/doi/pdf/10.1002/0471497398.mm422>. URL: <https://onlinelibrary.wiley.com/doi/abs/10.1002/0471497398.mm422>.
- [3] AICC. <https://www.aicc.org/>. Dostęp: 2024-02-17.
- [4] Hala Al Kuwatly, Maximilian Wich i Georg Groh. „Identifying and Measuring Annotator Bias Based on Annotators’ Demographic Characteristics”. W: *Proceedings of the Fourth Workshop on Online Abuse and Harms*. Red. Seyi Akiwowo, Bertie Vidgen, Vinodkumar Prabhakaran i Zeerak Waseem. Online: Association for Computational Linguistics, list. 2020, s. 184–190. DOI: 10.18653/v1/2020.a1w-1.21. URL: <https://aclanthology.org/2020.a1w-1.21>.
- [5] N.A. Ali, Fathy Eassa i Essam Hamed. „Personalized Learning Style for Adaptive E-Learning System”. W: *International Journal of Advanced Trends in Computer Science and Engineering* 8 (sty. 2019), s. 223–230. DOI: 10.30534/ijatcse/2019/4181.12019.
- [6] Dalia Kamal Fathi Alnagar. „Using artificial neural network to predicted student satisfaction in e-learning”. W: *Am J Appl Math Stat* 8.3 (2020), s. 90–5.
- [7] Elham Alsadoon. „The impact of an adaptive e-course on students’ achievements based on the students’ prior knowledge”. W: *Education and Information Technologies* 25.5 (wrz. 2020), s. 3541–3551. ISSN: 1573-7608. DOI: 10.1007/s10639-020-10125-3. URL: <https://doi.org/10.1007/s10639-020-10125-3>.
- [8] Ashton Anderson, Daniel Huttenlocher, Jon Kleinberg i Jure Leskovec. „Engaging with massive online courses”. W: *Proceedings of the 23rd International Conference on World Wide Web*. WWW ’14. Seoul, Korea: Association for Computing Machinery, 2014, s. 687–698. ISBN: 9781450327442. DOI: 10.1145/2566486.2568042. URL: <https://doi.org/10.1145/2566486.2568042>.

- [9] Ivon Arroyo, Kimberly Ferguson, Jeff Johns, Toby Dragon, Hasmik Meheranian, Don Fisher, Andrew Barto, Sridhar Mahadevan i Beverly P. Woolf. „Repairing Disengagement With Non-Invasive Interventions”. W: *Proceedings of the 2007 Conference on Artificial Intelligence in Education: Building Technology Rich Learning Contexts That Work*. NLD: IOS Press, 2007, s. 195–202. ISBN: 9781586037642.
- [10] Branka Arsovic i Nenad Stefanovic. „E-learning based on the adaptive learning model: case study in Serbia”. W: *Sādhanā* 45.1 (paź. 2020), s. 266. ISSN: 0973-7677. DOI: 10.1007/s12046-020-01499-8. URL: <https://doi.org/10.1007/s12046-020-01499-8>.
- [11] Alison Ashby, John TE Richardson i Alan Woodley. „National student feedback surveys in distance education: an investigation at the UK Open University”. W: *Open Learning: The Journal of Open, Distance and e-Learning* 26.1 (2011), s. 5–25.
- [12] Essaid El Bachari, El Hassan Abdelwahed i Mohamed El Adnani. „Design Of An Adaptive E-Learning Model Based On Learner’s Personality”. W: 2011. URL: <https://api.semanticscholar.org/CorpusID:10285041>.
- [13] Panos Balatsoukas, Anne Morris i Ann O’Brien. „Learning Objects Update: Review and Critical Approach to Content Aggregation.” W: *Educational Technology & Society* 11 (kw. 2008), s. 119–130.
- [14] Søren Balle, Anne Petersen i Anne-Mette Nortvig. „A Literature Review of the Factors Influencing E-Learning and Blended Learning in Relation to Learning Outcome, Student Satisfaction and Engagement”. W: *Electronic Journal of e-Learning* 16 (mar. 2018).
- [15] Renate van der Bent, Johan Jeuring i Bastiaan Heeren. „The Diagnosing Behaviour of Intelligent Tutoring Systems”. W: *Transforming Learning with Meaningful Technologies*. Red. Maren Scheffel, Julien Broisin, Viktoria Pammer-Schindler, Andri Ioannou i Jan Schneider. Cham: Springer International Publishing, 2019, s. 112–126. ISBN: 978-3-030-29736-7.
- [16] Nina Bergdahl. „Engagement and disengagement in online learning”. W: *Computers & Education* 188 (2022), s. 104561. ISSN: 0360-1315. DOI: <https://doi.org/10.1016/j.compedu.2022.104561>. URL: <https://www.sciencedirect.com/science/article/pii/S0360131522001324>.
- [17] Nina Bergdahl, Jalal Nouri i Uno Fors. „Disengagement, engagement and digital skills in technology-enhanced learning”. W: *Education and Information Technologies* 25.2 (mar. 2020), s. 957–983. ISSN: 1573-7608. DOI: 10.1007/s10639-019-09998-w. URL: <https://doi.org/10.1007/s10639-019-09998-w>.
- [18] James C. Bezdek, Robert Ehrlich i William Full. „FCM: The fuzzy c-means clustering algorithm”. W: *Computers & Geosciences* 10.2 (1984), s. 191–203. ISSN: 0098-3004. DOI: [https://doi.org/10.1016/0098-3004\(84\)90020-7](https://doi.org/10.1016/0098-3004(84)90020-7). URL: <https://www.sciencedirect.com/science/article/pii/0098300484900207>.
- [19] *Biblioteka its-student-progress*. <https://git.wmi.amu.edu.pl/jacekmar/its-student-progress>. Dostęp: 2024-01-26.
- [20] Ekaba Bisong. „Logistic Regression”. W: *Building Machine Learning and Deep Learning Models on Google Cloud Platform: A Comprehensive Guide for Beginners*. Berkeley, CA: Apress, 2019, s. 243–250. ISBN: 978-1-4842-4470-8. DOI: 10.1007/978-



- 1-4842-4470-8\_20. URL: [https://doi.org/10.1007/978-1-4842-4470-8\\_20](https://doi.org/10.1007/978-1-4842-4470-8_20).
- [21] Blackboard. <https://www.blackboard.com/>. Dostęp: 2024-05-02.
- [22] Blackboard - Criteria for Releasing Content. [https://help.blackboard.com/Learn/Instructor/Original/Course\\_Content/Release\\_Content/Criteria\\_for\\_Releasing\\_Content](https://help.blackboard.com/Learn/Instructor/Original/Course_Content/Release_Content/Criteria_for_Releasing_Content). Dostęp: 2024-05-02.
- [23] Lori Breslow, David E Pritchard, Jennifer DeBoer, Glenda S Stump, Andrew D Ho i Daniel T Seaton. „Studying learning in the worldwide classroom research into edX’s first MOOC.” W: *Research & Practice in Assessment* 8 (2013), s. 13–25.
- [24] J. S. Brown. „Pedagogical, natural language, and knowledge engineering techniques in SOPHIE-I, II and III”. W: Academic Press, 1982.
- [25] Ramón Zatarain Cabada, María Lucía Barrón Estrada i Carlos Alberto Reyes García. „EDUCA: A web 2.0 authoring tool for developing adaptive and intelligent tutoring systems using a Kohonen network”. W: *Expert Systems with Applications* 38.8 (2011), s. 9522–9529. ISSN: 0957-4174. DOI: <https://doi.org/10.1016/j.eswa.2011.01.145>. URL: <https://www.sciencedirect.com/science/article/pii/S0957417411001783>.
- [26] J.J. Castro-Schez, C. Glez-Morcillo, J. Albusac i D. Vallejo. „An intelligent tutoring system for supporting active learning: A case study on predictive parsing learning”. W: *Information Sciences* 544 (2021), s. 446–468. ISSN: 0020-0255. DOI: <https://doi.org/10.1016/j.ins.2020.08.079>. URL: <https://www.sciencedirect.com/science/article/pii/S0020025520308331>.
- [27] Pinar Cavas. „Factors affecting the motivation of Turkish primary students for science learning”. W: *Science education international* 22 (2011), s. 31–42. URL: <https://api.semanticscholar.org/CorpusID:145425254>.
- [28] Roy Y Chan, Krishna Bista i Ryan M Allen. „Is Online and Distance Learning the Future in Global Higher Education?: The Faculty Perspectives during COVID-19”. W: *Online teaching and learning in higher education during COVID-19*. Routledge, 2021, s. 3–12.
- [29] Guanliang Chen, Rafael Ferreira, David Lang i Dragan Gasevic. „Predictors of Student Satisfaction: A Large-Scale Study of Human-Human Online Tutorial Dialogues.” W: *International Educational Data Mining Society* (2019).
- [30] Lucy Chipchase, Megan Davidson, Felicity Blackstock, Ros Bye, Peter Colthier, Nerida Krupp, Wendy Dickson, Deborah Turner i Mark Williams. *Conceptualising and measuring student disengagement in Higher Education: A synthesis of the literature*. URL: <https://www.scienu.ca/journal/index.php/ijhe/article/view/11010>.
- [31] Stephen L. Chiu. „Fuzzy Model Identification Based on Cluster Estimation”. W: *Journal of Intelligent and Fuzzy Systems* 2 (1994). 3, s. 267–278. DOI: 10.3233/IFS-1994-2306. URL: <https://doi.org/10.3233/IFS-1994-2306>.
- [32] William J. Clancey. „From Guidon to Neomycin and Heracles in Twenty Short Lessons: ORN Final Report 1979-1985”. W: *AI Mag.* 7 (1986), s. 40–60. URL: <https://api.semanticscholar.org/CorpusID:822338>.

- [33] Mihaela Cocea i Stephan Weibelzahl. „Disengagement Detection in Online Learning: Validation Studies and Perspectives”. W: *IEEE Transactions on Learning Technologies* 4.2 (2011), s. 114–124. DOI: 10.1109/TLT.2010.14.
- [34] Mihaela Cocea i Stephan Weibelzahl. „Log file analysis for disengagement detection in e-Learning environments”. W: *User Modeling and User-Adapted Interaction* 19.4 (paź. 2009), s. 341–385. ISSN: 1573-1391. DOI: 10.1007/s11257-009-9065-5. URL: <https://doi.org/10.1007/s11257-009-9065-5>.
- [35] Nello Cristianini i Elisa Ricci. „Support Vector Machines”. W: *Encyclopedia of Algorithms*. Red. Ming-Yang Kao. Boston, MA: Springer US, 2008, s. 928–932. ISBN: 978-0-387-30162-4. DOI: 10.1007/978-0-387-30162-4\_415. URL: [https://doi.org/10.1007/978-0-387-30162-4\\_415](https://doi.org/10.1007/978-0-387-30162-4_415).
- [36] Keeley A. Crockett, Annabel Latham, David Mclean i James D. O’Shea. „A fuzzy model for predicting learning styles using behavioral cues in an conversational intelligent tutoring system”. W: *2013 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)* (2013), s. 1–8. URL: <https://api.semanticscholar.org/CorpusID:5158737>.
- [37] Antoinette Davis. „Measuring student satisfaction in online mathematics courses - RESEARCH”. W: *Kentucky Journal of Excellence in College Teaching and Learning* 14 (2017), s. 2.
- [38] Angel De Vicente i Helen Pain. „Motivation self-report in ITS”. W: (lip. 2000).
- [39] L. Dias, L. Faria, C. Martins, V. Marques i E. Pratas. „Adaptive Learning Management System To Support An Intelligent Tutoring Module”. W: *EDULEARN14 Proceedings*. 6th International Conference on Education and New Learning Technologies. Barcelona, Spain: IATED, lip. 2014, s. 598–607. ISBN: 978-84-617-0557-3.
- [40] Marcia D. Dixson. „Measuring Student Engagement in the Online Course: The Online Student Engagement Scale (OSE).” W: *Journal of asynchronous learning networks* 19 (2015). URL: <https://api.semanticscholar.org/CorpusID:143391417>.
- [41] *Dokumentacja języka FCL*. [https://jfuzzylogic.sourceforge.net/html/pdf/iec\\_1131\\_7\\_cd1.pdf](https://jfuzzylogic.sourceforge.net/html/pdf/iec_1131_7_cd1.pdf). Dostęp: 2024-01-27.
- [42] Smadar Donitsa-Schmidt, Rony Ramot i Beverley Topaz. „Shaping the future of distance learning in teacher education: MOOCS during COVID-19”. W: *Perspectives in Education* 40.1 (2022), s. 250–267. DOI: 10.18820/2519593X/pie.v40.i1.15. eprint: <https://journals.co.za/doi/pdf/10.18820/2519593X/pie.v40.i1.15>. URL: <https://journals.co.za/doi/abs/10.18820/2519593X/pie.v40.i1.15>.
- [43] Fabiano A. Dorça, Vitor C. Carvalho, Miller M. Mendes, Rafael D. Araújo, Hiran N. Ferreira i Renan G. Cattelan. „An Approach for Automatic and Dynamic Analysis of Learning Objects Repositories through Ontologies and Data Mining Techniques for Supporting Personalized Recommendation of Content in Adaptive and Intelligent Educational Systems”. W: *2017 IEEE 17th International Conference on Advanced Learning Technologies (ICALT)*. 2017, s. 514–516. DOI: 10.1109/ICALT.2017.121.
- [44] *Eduexe - eLearning authoring tool*. <https://eduexe.com/home>. Dostęp: 2024-01-25.
- [45] *Edumatic ITS*. <https://its.wmi.amu.edu.pl>. Dostęp: 2024-01-25.

- [46] Gökhan Engin, Burak Aksoyer, Melike Avdagic, Damla Bozanlı, Umutcan Hanay, Deniz Maden i Gurdal Ertek. „Rule-based Expert Systems for Supporting University Students”. W: *Procedia Computer Science* 31 (2014). 2nd International Conference on Information Technology and Quantitative Management, ITQM 2014, s. 22–31. ISSN: 1877-0509. DOI: <https://doi.org/10.1016/j.procs.2014.05.241>. URL: <https://www.sciencedirect.com/science/article/pii/S1877050914004189>.
- [47] Soukaina Ennouamani i Zouhir Mahani. „An overview of adaptive e-learning systems”. W: *2017 Eighth International Conference on Intelligent Computing and Information Systems (ICICIS)*. 2017, s. 342–347. DOI: 10.1109/INTELCIS.2017.8260060.
- [48] Meng Joo Er i Yi Zhou. „Automatic generation of fuzzy inference systems via unsupervised learning”. W: *Neural Networks* 21.10 (2008). ICONIP 2007, s. 1556–1566. ISSN: 0893-6080. DOI: <https://doi.org/10.1016/j.neunet.2008.06.007>. URL: <https://www.sciencedirect.com/science/article/pii/S0893608008001214>.
- [49] Wei Fan i Kun Zhang. „Bagging”. W: *Encyclopedia of Database Systems*. Red. LING LIU i M. TAMER ÖZSU. Boston, MA: Springer US, 2009, s. 206–210. ISBN: 978-0-387-39940-9. DOI: 10.1007/978-0-387-39940-9\_567. URL: [https://doi.org/10.1007/978-0-387-39940-9\\_567](https://doi.org/10.1007/978-0-387-39940-9_567).
- [50] Jacqueline Feild, Nicholas Lewkow, Sean Burns i Karen Gebhardt. „A generalized classifier to identify online learning tool disengagement at scale”. W: *Proceedings of the 8th International Conference on Learning Analytics and Knowledge*. LAK '18. Sydney, New South Wales, Australia: Association for Computing Machinery, 2018, s. 61–70. ISBN: 9781450364003. DOI: 10.1145/3170358.3170370. URL: <https://doi.org/10.1145/3170358.3170370>.
- [51] Richard Felder i L.K. Silverman. „Learning and teaching styles in engineering education. *Engr*”. W: *Education* 78 (sty. 2002), s. 674–681.
- [52] Giuseppe Fenza, Francesco J. Orciuoli i Demetrios G. Sampson. „Building Adaptive Tutoring Model Using Artificial Neural Networks and Reinforcement Learning”. W: *2017 IEEE 17th International Conference on Advanced Learning Technologies (ICALT)* (2017), s. 460–462. URL: <https://api.semanticscholar.org/CorpusID:3418139>.
- [53] *FIS file format*. <https://functionbay.com/documentation/onlinehelp/default.htm#!Documents/introductiontothefisfileformat.htm>. Dostęp: 2024-01-30.
- [54] Neil D Fleming. „VARK visual, aural/auditory, read/write, kinesthetic”. W: *New Zealand: Bonwell Green Mountain Falls* (2006).
- [55] Yoav Freund i Robert E Schapire. „A Decision-Theoretic Generalization of On-Line Learning and an Application to Boosting”. W: *Journal of Computer and System Sciences* 55.1 (1997), s. 119–139. ISSN: 0022-0000. DOI: <https://doi.org/10.1006/jcss.1997.1504>. URL: <https://www.sciencedirect.com/science/article/pii/S002200009791504X>.
- [56] Jerome H Friedman. „Greedy function approximation: a gradient boosting machine”. W: *Annals of statistics* (2001), s. 1189–1232.

- [57] Caro Fuchs, Simone Spolaor, Marco S. Nobile i Uzay Kaymak. „pyFUME: a Python Package for Fuzzy Model Estimation”. W: *2020 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*. 2020. DOI: 10.1109/FUZZ48607.2020.9177565.
- [58] Johannes Fürnkranz. „Decision Tree”. W: *Encyclopedia of Machine Learning*. Red. Claude Sammut i Geoffrey I. Webb. Boston, MA: Springer US, 2010, s. 263–267. ISBN: 978-0-387-30164-8. DOI: 10.1007/978-0-387-30164-8\_204. URL: [https://doi.org/10.1007/978-0-387-30164-8\\_204](https://doi.org/10.1007/978-0-387-30164-8_204).
- [59] Grzegorz Gapiński. *Budowa adaptacyjnych kursów e-learningowych w podejściu hybrydowym*. Praca magisterska, 2024.
- [60] *Generator raportów*. <https://git.wmi.amu.edu.pl/jacekmar/its-raport-generator>. Dostęp: 2024-05-31.
- [61] Arash Geramian i Ajith Abraham. „Customer classification: A Mamdani fuzzy inference system standpoint for modifying the failure mode and effect analysis based three dimensional approach”. W: *Expert Systems with Applications* 186 (2021), s. 115753. ISSN: 0957-4174. DOI: <https://doi.org/10.1016/j.eswa.2021.115753>. URL: <https://www.sciencedirect.com/science/article/pii/S0957417421011295>.
- [62] Yue Gong. „Student Modeling in Intelligent Tutoring Systems”. Prac. dokt. Worcester Polytechnic Institute, 2014.
- [63] Joselyn Goopio i Catherine Cheung. „The MOOC dropout phenomenon and retention strategies”. W: *Journal of Teaching in Travel & Tourism* 21.2 (2021), s. 177–197. DOI: 10.1080/15313220.2020.1809050. eprint: <https://doi.org/10.1080/15313220.2020.1809050>. URL: <https://doi.org/10.1080/15313220.2020.1809050>.
- [64] Lalit Mohan Goyal, Mamta Mittal i Jasleen Kaur Sethi. „Fuzzy model generation using Subtractive and Fuzzy C-Means clustering”. W: *CSI Transactions on ICT* 4.2 (grud. 2016), s. 129–133. ISSN: 2277-9086. DOI: 10.1007/s40012-016-0090-3. URL: <https://doi.org/10.1007/s40012-016-0090-3>.
- [65] Wolfgang Greller, Mohammad Issack Santally, Ravindra Boojhawon, Yousra Rajabalee i Roopesh Kevin Sungkur. *Using Learning Analytics to Investigate Student Performance in Blended Learning Courses*. 2017. URL: <https://www.zfhe.at/index.php/zfhe/article/view/1004>.
- [66] Foteini Grivokostopoulou, Isidoros Perikos i Ioannis Hatzilygeroudis. „An Educational System for Learning Search Algorithms and Automatically Assessing Student Performance”. W: *International Journal of Artificial Intelligence in Education* 27.1 (mar. 2017), s. 207–240. ISSN: 1560-4306. DOI: 10.1007/s40593-016-0116-x. URL: <https://doi.org/10.1007/s40593-016-0116-x>.
- [67] Crina Grosan i Ajith Abraham. „Rule-Based Expert Systems”. W: *Intelligent Systems: A Modern Approach*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011, s. 149–185. ISBN: 978-3-642-21004-4. DOI: 10.1007/978-3-642-21004-4\_7. URL: [https://doi.org/10.1007/978-3-642-21004-4\\_7](https://doi.org/10.1007/978-3-642-21004-4_7).
- [68] Frederic Guay, Robert Vallerand i Céline Blanchard. „On the Assessment of Situational Intrinsic and Extrinsic Motivation: The Situational Motivation Scale (SIMS)”. W: *Motivation and Emotion* 24 (wrz. 2000), s. 175–213. DOI: 10.1023/A:1005614228250.

- [69] Neil Guppy, Dominique Verpoorten, David Boud, Lin Lin, Joanna Tai i Silvia Bartolic. „The post-COVID-19 future of digital learning in higher education: Views from educators, students, and other professionals in six countries”. W: *British Journal of Educational Technology* 53.6 (2022), s. 1750–1765. DOI: <https://doi.org/10.1111/bjet.13212>. eprint: <https://bera-journals.onlinelibrary.wiley.com/doi/pdf/10.1111/bjet.13212>. URL: <https://bera-journals.onlinelibrary.wiley.com/doi/abs/10.1111/bjet.13212>.
- [70] Safiyeh Rajae Harandi. „Effects of e-learning on Students’ Motivation”. W: *Procedia - Social and Behavioral Sciences* 181 (2015). Proceedings Of The 3rd International Conference On Leadership, Technology And Innovation Management, s. 423–430. ISSN: 1877-0428. DOI: <https://doi.org/10.1016/j.sbspro.2015.04.905>. URL: <https://www.sciencedirect.com/science/article/pii/S1877042815031985>.
- [71] Khe Foon Hew i Wing Sum Cheung. „Students’ and instructors’ use of massive open online courses (MOOCs): Motivations and challenges”. W: *Educational research review* 12 (2014), s. 45–58.
- [72] Khe Foon Hew, Xiang Hu, Chen Qiao i Ying Tang. „What predicts student satisfaction with MOOCs: A gradient boosting trees supervised machine learning and sentiment analysis approach”. W: *Computers & Education* 145 (2020), s. 103724. ISSN: 0360-1315. DOI: <https://doi.org/10.1016/j.compedu.2019.103724>. URL: <https://www.sciencedirect.com/science/article/pii/S0360131519302775>.
- [73] Tin Kam Ho. „Random decision forests”. W: *Proceedings of 3rd international conference on document analysis and recognition*. T. 1. IEEE. 1995, s. 278–282.
- [74] D. Hooshyar, Rodina Ahmad, Moslem Yousefi, Farrah Yusop i Shi-Jinn Horng. „A flowchart-based intelligent tutoring system for improving problem-solving skills of novice programmers: Flowchart-based Intelligent Tutoring System”. W: *Journal of Computer Assisted Learning* 31 (kw. 2015). DOI: [10.1111/jcal.12099](https://doi.org/10.1111/jcal.12099).
- [75] Shih-Chun Hsu, J.-Y.-J. Hsu i I-Jen Chiang. „Automatic generation of fuzzy control rule by machine learning methods”. W: *Proceedings of 1995 IEEE International Conference on Robotics and Automation*. T. 1. 1995, 287–292 vol.1. DOI: [10.1109/ROBOT.1995.525299](https://doi.org/10.1109/ROBOT.1995.525299).
- [76] Dirk Ifenthaler. „Learning Management System”. W: *Encyclopedia of the Sciences of Learning*. Red. Norbert M. Seel. Boston, MA: Springer US, 2012, s. 1925–1927. ISBN: 978-1-4419-1428-6. DOI: [10.1007/978-1-4419-1428-6\\_187](https://doi.org/10.1007/978-1-4419-1428-6_187). URL: [https://doi.org/10.1007/978-1-4419-1428-6\\_187](https://doi.org/10.1007/978-1-4419-1428-6_187).
- [77] *jFuzzyLogic*. <https://jfuzzylogic.sourceforge.net/html/index.html>. Dostęp: 2024-01-27.
- [78] Ankur Joshi, Saket Kale, Satish Chandel i D Kumar Pal. „Likert scale: Explored and explained”. W: *British journal of applied science & technology* 7.4 (2015), s. 396–403.
- [79] Carl Jung i John Beebe. *Psychological types*. Routledge, 2016.
- [80] Janusz Kacprzyk. *Wieloetapowe Sterowanie Rozmyte*. Wydawnictwa Naukowo-Techniczne, 2001.
- [81] Iraklis Katsaris i Nikolas Vidakis. „Adaptive e-learning systems through learning styles: A review of the literature”. W: *Advances in Mobile Learning Educational*

- Research* 1.2 (paź. 2021), s. 124–145. DOI: 10.25082/AMLER.2021.02.007. URL: <https://www.syncsci.com/journal/AMLER/article/view/AMLER.2021.02.007>.
- [82] Michelle M. Kazmer. „The process of disengaging from online learning community revealed through examination of threaded discussions”. W: *International Journal of Web Based Communities* 8.4 (2012). PMID: 49564, s. 521–536. DOI: 10.1504/IJWBC.2012.049564. URL: <https://www.inderscienceonline.com/doi/abs/10.1504/IJWBC.2012.049564>.
- [83] John M. Keller. „Development and use of the ARCS model of instructional design”. W: *Journal of instructional development* 10.3 (wrz. 1987), s. 2–10. ISSN: 0162-2641. DOI: 10.1007/BF02905780. URL: <https://doi.org/10.1007/BF02905780>.
- [84] E. E. Kerre. „A First View on the Alternatives of Fuzzy Set Theory”. W: *Computational Intelligence in Theory and Practice*. Red. Bernd Reusch i Karl-Heinz Temme. Heidelberg: Physica-Verlag HD, 2001, s. 55–71. ISBN: 978-3-7908-1831-4.
- [85] Akrom Tegar Khomeiny, Tegar Restu Kusuma, Anik Nur Handayani, Aji Prasetya Wibawa i Agus Hery Supadmi Irianti. „Grading System Recommendations for Students using Fuzzy Mamdani Logic”. W: *2020 4th International Conference on Vocational Education and Training (ICOVET)*. 2020, s. 1–6. DOI: 10.1109/ICOVET50258.2020.9230299.
- [86] Hassan Khosravi, Kendra Cooper i Kirsty Kitto. „RiPLE: Recommendation in Peer-Learning Environments Based on Knowledge Gaps and Interests”. W: *Journal of Educational Data Mining* 9.1 (maj 2017), s. 42–67. DOI: 10.5281/zenodo.3554627. URL: <https://jedm.educationaldatamining.org/index.php/JEDM/article/view/239>.
- [87] Hassan Khosravi, Shazia Sadiq i Dragan Gasevic. „Development and adoption of an adaptive learning system: Reflections and lessons learned”. W: *Proceedings of the 51st ACM technical symposium on computer science education*. 2020, s. 58–64.
- [88] Dennis F. Kibler i Pat Langley. „The Experimental Study of Machine Learning”. W: 1991. URL: <https://api.semanticscholar.org/CorpusID:18797039>.
- [89] Aleksandra Klašnja-Milićević, Mirjana Ivanović, Boban Vesin i Zoran Budimac. „Enhancing e-learning systems with personalized recommendation based on collaborative tagging techniques”. W: *Applied Intelligence* 48.6 (2018), s. 1519–1535. ISSN: 0924-669X. DOI: 10.1007/s10489-017-1051-8. URL: <https://doi.org/10.1007/s10489-017-1051-8>.
- [90] Zdenko Kovacic i Stjepan Bogdan. *Fuzzy controller design: theory and applications*. CRC press, 2018.
- [91] Kerri-Lee Krause. „Understanding and promoting student engagement in university learning communities”. W: *Paper presented as keynote address: Engaged, Inert or Otherwise Occupied* (2005), s. 21–22.
- [92] Y Kurniasari, B Suseta, N Hendiyani i A M Abadi. „Classification of Open Unemployment Rate in Indonesia with Mamdani Fuzzy Inference System”. W: *Journal of Physics: Conference Series* 1581.1 (lip. 2020), s. 012010. DOI: 10.1088/1742-6596/1581/1/012010. URL: <https://dx.doi.org/10.1088/1742-6596/1581/1/012010>.

- [93] *Kurs e-learningowy "Zastosowania zbiorów rozmytych"*. [https://marcinsz.students.wmi.amu.edu.pl/zastosowania\\_zbiorow\\_rozmytych/](https://marcinsz.students.wmi.amu.edu.pl/zastosowania_zbiorow_rozmytych/). Dostęp: 2024-01-25.
- [94] Pat Langley. „Machine learning as an experimental science”. W: *Machine Learning* 3 (2004), s. 5–8. URL: <https://api.semanticscholar.org/CorpusID:7364455>.
- [95] Rosa Lanzilotti i Teresa Roselli. „An Experimental Evaluation of Logiocando, an Intelligent Tutoring Hypermedia System”. W: *Int. J. Artif. Intell. Educ.* 17 (2007), s. 41–56. URL: <https://api.semanticscholar.org/CorpusID:18420865>.
- [96] Eitel J. M. Lauría, Joshua D. Baron, Mallika Devireddy, Venniraiselvi Sundararaju i Sandeep M. Jayaprakash. „Mining academic data to improve college student retention: an open source perspective”. W: *Proceedings of the 2nd International Conference on Learning Analytics and Knowledge*. LAK '12. Vancouver, British Columbia, Canada: Association for Computing Machinery, 2012, s. 139–142. ISBN: 9781450311113. DOI: 10.1145/2330601.2330637. URL: <https://doi.org/10.1145/2330601.2330637>.
- [97] Mohamed Lazrag i Mustapha Machkour. „A Multi-agent Architecture for an Intelligent Tutoring System”. W: *Journal of Advanced Research in Dynamical and Control Systems* 12 (maj 2020), s. 2020. DOI: 10.5373/JARDCS/V12SP4/20201572.
- [98] Jeongju Lee, Hae-Deok Song i Ah Jeong Hong. „Exploring Factors, and Indicators for Measuring Students' Sustainable Engagement in e-Learning”. W: *Sustainability* 11.4 (2019). ISSN: 2071-1050. DOI: 10.3390/su11040985. URL: <https://www.mdpi.com/2071-1050/11/4/985>.
- [99] Youngju Lee i Jaeho Choi. „A review of online course dropout research: Implications for practice and future research”. W: *Educational Technology Research and Development* 59 (2011), s. 593–618.
- [100] Jiacheng Liao, Yan Liang i Jiahui Pan. „Deep facial spatiotemporal network for engagement prediction in online learning”. W: *Applied Intelligence* 51.10 (paź. 2021), s. 6609–6621. ISSN: 1573-7497. DOI: 10.1007/s10489-020-02139-8. URL: <https://doi.org/10.1007/s10489-020-02139-8>.
- [101] Graf von Malotky, Nikolaj Troels i Alke Martens. „Analyzing the Usage of the Classical ITS Software Architecture and Refining It”. W: *Intelligent Tutoring Systems*. Red. Andre Coy, Yugo Hayashi i Maiga Chang. Cham: Springer International Publishing, 2019, s. 40–46. ISBN: 978-3-030-22244-4.
- [102] E.H. Mamdani. „Application of fuzzy algorithms for control of simple dynamic plant”. English. W: *Proceedings of the Institution of Electrical Engineers* 121 (12 grud. 1974), 1585–1588(3). ISSN: 0020-3270. URL: <https://digital-library.theiet.org/content/journals/10.1049/piee.1974.0328>.
- [103] *marcin-szczepanski/jFuzzyLogic*. <https://github.com/marcin-szczepanski/jFuzzyLogic>. Dostęp: 2024-01-27.
- [104] Jacek Marciniak. „Building E-learning Content Repositories to Support Content Reusability”. W: *International Journal of Emerging Technologies in Learning (ijET)*. 2014. URL: <https://api.semanticscholar.org/CorpusID:21377926>.
- [105] Jacek Marciniak. „Building Intelligent Tutoring Systems Immersed in Repositories of E-learning Content”. W: *Procedia Computer Science* 35 (2014). Knowledge-Based and Intelligent Information & Engineering Systems 18th Annual Conference, KES-2014 Gdynia, Poland, September 2014 Proceedings, s. 541–550. ISSN: 1877-0509.

- DOI: <https://doi.org/10.1016/j.procs.2014.08.135>. URL: <https://www.sciencedirect.com/science/article/pii/S1877050914011004>.
- [106] Jacek Marciniak. *Inteligentne systemy e-learningowe wykorzystujące ontologie typu wordnet*. pl. Poznań: Wydawnictwo Naukowe UAM, 2015.
- [107] Jacek Marciniak i Marcin Szczepański. „Individualized learning in a course with a tight schedule”. W: *Procedia Computer Science* 176 (2020). Knowledge-Based and Intelligent Information & Engineering Systems: Proceedings of the 24th International Conference KES2020, s. 2059–2068. ISSN: 1877-0509. DOI: <https://doi.org/10.1016/j.procs.2020.09.242>. URL: <https://www.sciencedirect.com/science/article/pii/S1877050920321451>.
- [108] Jacek Marciniak, Marcin Szczepański, Krzysztof Dyczkowski, Karolina Mazurowska, Radosław Stanczewski, Joanna Grzybek i Dorota Marciniak. „The Use of a Fuzzy Rule-Based System in Adaptive e-Learning Content Based on Intercultural Competence”. W: *2023 IEEE International Conference on Fuzzy Systems (FUZZ)*. 2023, s. 1–6. DOI: [10.1109/FUZZ52849.2023.10309813](https://doi.org/10.1109/FUZZ52849.2023.10309813).
- [109] Jacek Marciniak, Andrzej Wójtowicz, Barbara Kołodziejczak, Marcin Szczepański i Anna Stachowiak. „Impact of Course Scheduling on Student Performance in Remote Learning”. W: *ITiCSE '22*. Dublin, Ireland: Association for Computing Machinery, 2022, s. 400–406. ISBN: 9781450392013. DOI: [10.1145/3502718.3524788](https://doi.org/10.1145/3502718.3524788). URL: <https://doi.org/10.1145/3502718.3524788>.
- [110] Jacek Marciniak, Andrzej Wójtowicz, Barbara Kołodziejczak, Marcin Szczepański, Dorota Marciniak i Anna Stachowiak. „Granular or Long: Influence of the Content Structure on Student Interaction with Learning Materials”. W: *2023 IEEE Frontiers in Education Conference (FIE)*. 2023, s. 1–8. DOI: [10.1109/FIE58773.2023.10343030](https://doi.org/10.1109/FIE58773.2023.10343030).
- [111] *MATLAB*. <https://www.mathworks.com/products/matlab.html>. Dostęp: 2024-01-30.
- [112] *MATLAB - Fuzzy Logic Designer*. <https://www.mathworks.com/help/fuzzy/fuzzylogicdesigner-app.html>. Dostęp: 2024-01-30.
- [113] Jíerí Mazurek, Cristina Pérez Rico, Carlos Fernández, Jean-Pierre Magnot i Tristan Magnot. „The 5-item likert scale and percentage scale correspondence with implications for the use of models with (fuzzy) linguistic variables”. eng. W: *Revista de Métodos Cuantitativos para la Economía y la Empresa* 31 (2021), s. 3–16. ISSN: 1886-516X. DOI: [10.46661/revmetodoscuanteconempresa.4010](https://doi.org/10.46661/revmetodoscuanteconempresa.4010). URL: <https://hdl.handle.net/10419/286220>.
- [114] Scott McQuiggan, Bradford Mott i James Lester. „Modeling self-efficacy in intelligent tutoring systems: An inductive approach”. W: *User Model. User-Adapt. Interact.* 18 (lut. 2008), s. 81–123. DOI: [10.1007/s11257-007-9040-y](https://doi.org/10.1007/s11257-007-9040-y).
- [115] *Moodle.org*. <https://moodle.org/>. Dostęp: 2024-01-25.
- [116] *Moodle.org Documentation - Tracking Progress*. [https://docs.moodle.org/404/en/Tracking\\_progress](https://docs.moodle.org/404/en/Tracking_progress). Dostęp: 2024-05-04.
- [117] Abdallah Moubayed, Mohammadnoor Injadat, Abdallah Shami i Hanan Lutfiyya. „Student engagement level in an e-learning environment: Clustering using k-means”. W: *American Journal of Distance Education* 34.2 (2020), s. 137–156.



- [118] Antonio Mucherino, Petraq J. Papajorgji i Panos M. Pardalos. „k-Nearest Neighbor Classification”. W: *Data Mining in Agriculture*. New York, NY: Springer New York, 2009, s. 83–106. ISBN: 978-0-387-88615-2. DOI: 10.1007/978-0-387-88615-2\_4. URL: [https://doi.org/10.1007/978-0-387-88615-2\\_4](https://doi.org/10.1007/978-0-387-88615-2_4).
- [119] M. Mudrák. „Analysis and implementation of adaptive course in Moodle”. W: *2017 15th International Conference on Emerging eLearning Technologies and Applications (ICETA)*. 2017, s. 1–6. DOI: 10.1109/ICETA.2017.8102513.
- [120] Mahbub Murshed, M Ali Akber Dewan, Fuhua Lin i Dunwei Wen. „Engagement detection in e-learning environments using convolutional neural networks”. W: *2019 IEEE Intl Conf on Dependable, Autonomic and Secure Computing, Intl Conf on Pervasive Intelligence and Computing, Intl Conf on Cloud and Big Data Computing, Intl Conf on Cyber Science and Technology Congress (DASC/PiCom/CBDCCom/CyberSciTech)*. IEEE. 2019, s. 80–86.
- [121] Mir Murtaza, Yamna Ahmed, Jawwad Ahmed Shamsi, Fahad Sherwani i Mariam Usman. „AI-Based Personalized E-Learning Systems: Issues, Challenges, and Solutions”. W: *IEEE Access* 10 (2022), s. 81323–81342. DOI: 10.1109/ACCESS.2022.3193938.
- [122] Isabel Briggs Myers i in. *Introduction to type®*. CPP, 1976.
- [123] Marina Nehme. W: *Legal Education Review* 20.1/2 (2010), s. 223–239. URL: <https://search.informit.org/doi/10.3316/informit.872489860313574>.
- [124] *OpenOlat - Infinite Learning*. <https://www.openolat.com/>. Dostęp: 2024-05-02.
- [125] *OpenOLAT 8.3 - User Manual*. [https://www.openolat.com/fileadmin/documents/openolat/OpenOLAT\\_8.3\\_User\\_Manual\\_EN.pdf](https://www.openolat.com/fileadmin/documents/openolat/OpenOLAT_8.3_User_Manual_EN.pdf). Dostęp: 2024-05-02.
- [126] Dotti Osterholt i Katherine Barratt. *Breaking the teaching and learning gridlock*. Grud. 2014. URL: <https://nebhe.org/journal/breaking-the-teaching-and-learning-gridlock/>.
- [127] Zdzisław Pawlak. „Rough sets”. W: *International Journal of Computer & Information Sciences* 11.5 (paź. 1982), s. 341–356. ISSN: 1573-7640. DOI: 10.1007/BF01001956. URL: <https://doi.org/10.1007/BF01001956>.
- [128] Tomasz Piłka, Bartłomiej Grzelak, Aleksandra Sadurska, Tomasz Górecki i Krzysztof Dyczkowski. „Predicting Injuries in Football Based on Data Collected from GPS-Based Wearable Sensors”. W: *Sensors* 23.3 (2023). ISSN: 1424-8220. DOI: 10.3390/s23031227. URL: <https://www.mdpi.com/1424-8220/23/3/1227>.
- [129] Manuel Piñeres, Darsana Josyula i Jovani Jiménez-Builes. „Multi-level pedagogical model for the personalization of pedagogical strategies in intelligent tutoring systems”. W: *Dyna (Medellin, Colombia)* 82 (grud. 2015), s. 185–193. DOI: 10.15446/dyna.v82n194.49279.
- [130] K. R. Premlatha i T. V. Geetha. „Learning content design and learner adaptation for adaptive e-learning environment: a survey”. W: *Artificial Intelligence Review* 44.4 (grud. 2015), s. 443–465. ISSN: 1573-7462. DOI: 10.1007/s10462-015-9432-z. URL: <https://doi.org/10.1007/s10462-015-9432-z>.
- [131] Joseph Psocka, Leonard Daniel Massey i Sharon A Mutter. *Intelligent tutoring systems: Lessons learned*. Psychology Press, 1988.
- [132] M del Puerto Paule Ruiz, M Jesús Fernández Díaz, Francisco Ortín Soler i Juan Ramón Pérez Pérez. „Adaptation in current e-learning systems”. W: *Computer*

- Standards & Interfaces* 30.1 (2008), s. 62–70. ISSN: 0920-5489. DOI: <https://doi.org/10.1016/j.csi.2007.07.006>. URL: <https://www.sciencedirect.com/science/article/pii/S0920548907000517>.
- [133] Aimad Qazdar, Chihab Cherkaoui, Brahim Er-Raha i D. Mammass. „AeLF: Mixing Adaptive Learning System with Learning Management System”. W: *International Journal of Computer Applications* 119 (czer. 2015), s. 1–8. DOI: 10.5120/21140-4171.
- [134] Feiyue Qiu, Guodao Zhang, Xin Sheng, Lei Jiang, Lijia Zhu, Qifeng Xiang, Bo Jiang i Ping-Kuo Chen. „Predicting students’ performance in e-learning using learning process and behaviour data”. W: *Scientific Reports* 12 (sty. 2022). DOI: 10.1038/s41598-021-03867-8.
- [135] Lei Qu, Ning Wang i W. Lewis Johnson. „Using Learner Focus of Attention to Detect Learner Motivation Factors”. W: *User Modeling 2005*. Red. Liliana Ardissono, Paul Brna i Antonija Mitrovic. Berlin, Heidelberg: Springer Berlin Heidelberg, 2005, s. 70–73. ISBN: 978-3-540-31878-1.
- [136] Nisha S Raj i Renumol V G. „A Rule-Based Approach for Adaptive Content Recommendation in a Personalized Learning Environment: An Experimental Analysis”. W: *2019 IEEE Tenth International Conference on Technology for Education (T4E)*. 2019, s. 138–141. DOI: 10.1109/T4E.2019.00033.
- [137] Paul Ramsden. „A performance indicator of teaching quality in higher education: The Course Experience Questionnaire”. W: *Studies in higher education* 16.2 (1991), s. 129–150.
- [138] Martina A Rau, Vincent Aleven, Nikol Rummel i Stacie Rohrbach. „Sense making alone doesn’t do it: Fluency matters too! ITS support for robust learning with multiple representations”. W: *Intelligent Tutoring Systems: 11th International Conference, ITS 2012, Chania, Crete, Greece, June 14-18, 2012. Proceedings 11*. Springer, 2012, s. 174–184.
- [139] Jingli Ren i Haiyan Wang. „Chapter 3 - Calculus and optimization”. W: *Mathematical Methods in Data Science*. Red. Jingli Ren i Haiyan Wang. Elsevier, 2023, s. 51–89. ISBN: 978-0-443-18679-0. DOI: <https://doi.org/10.1016/B978-0-44-318679-0.00009-0>. URL: <https://www.sciencedirect.com/science/article/pii/B9780443186790000090>.
- [140] Abhishek Revadekar, Shreya Oak, Aumkar Gadekar i Pramod Bide. „Gauging attention of students in an e-learning environment”. W: *2020 IEEE 4th Conference on Information & Communication Technology (CICT)*. 2020, s. 1–6. DOI: 10.1109/CICT51604.2020.9312048.
- [141] Bart Rienties, Tim Lewis, Ruth McFarlane, Quan Nguyen i Lisette Toetenel. „Analytics in online and offline language learning environments: the role of learning design to understand student online engagement”. W: *Computer Assisted Language Learning* 31.3 (2018), s. 273–293. DOI: 10.1080/09588221.2017.1401548. eprint: <https://doi.org/10.1080/09588221.2017.1401548>. URL: <https://doi.org/10.1080/09588221.2017.1401548>.
- [142] Georgios Rizos i Björn W. Schuller. „Average Jane, Where Art Thou? – Recent Avenues in Efficient Machine Learning Under Subjectivity Uncertainty”. W: *Information Processing and Management of Uncertainty in Knowledge-Based Systems*. Red. Marie-Jeanne Lesot, Susana Vieira, Marek Z. Reformat, João Paulo Carvalho,

- Anna Wilbik, Bernadette Bouchon-Meunier i Ronald R. Yager. Cham: Springer International Publishing, 2020, s. 42–55. ISBN: 978-3-030-50146-4.
- [143] Hassan A. El-Sabagh. „Adaptive e-learning environment based on learning styles and its impact on development students' engagement”. W: *International Journal of Educational Technology in Higher Education* 18.1 (paź. 2021), s. 53. ISSN: 2365-9440. DOI: 10.1186/s41239-021-00289-4. URL: <https://doi.org/10.1186/s41239-021-00289-4>.
- [144] Naomie Salim i Norreen Haron. „The Construction of Fuzzy Set and Fuzzy Rule for Mixed Approach in Adaptive Hypermedia Learning System”. W: *Technologies for E-Learning and Digital Entertainment*. Red. Zhigeng Pan, Ruth Aylett, Holger Diener, Xiaogang Jin, Stefan Göbel i Li Li. Berlin, Heidelberg: Springer Berlin Heidelberg, 2006, s. 183–187. ISBN: 978-3-540-33424-8.
- [145] Pauline M Sampson, John Leonard, Julia W Ballenger i J Craig Coleman. „Student satisfaction of online courses for educational leadership”. W: *Online Journal of Distance Learning Administration* 13.3 (2010), s. 1–12.
- [146] S Sani, A B Bichi i Shehu Ayuba. „Artificial Intelligence Approaches in Student Modeling: Half Decade Review (2010-2015)”. W: *International Journal of Computer Science and Network* 5.5 (2016), s. 746–754.
- [147] Pedro Bispo Santos, Caroline Verena Bhowmik i Iryna Gurevych. „Avoiding bias in students' intrinsic motivation detection”. W: *International Conference on Intelligent Tutoring Systems*. Springer. 2020, s. 89–94.
- [148] Leo Willyanto Santoso. „Adaptive Educational Resources Framework for E-Learning Using Rule-Based System”. W: *Information and Communication Technology for Intelligent Systems*. Red. Tomonobu Senjyu, Parikshit N. Mahalle, Thinagaran Perumal i Amit Joshi. Singapore: Springer Singapore, 2021, s. 385–396. ISBN: 978-981-15-7078-0.
- [149] Ines Šarić-Grgić, Ani Grubišić, Ljiljana Šerić i Timothy Robinson. „Student Clustering Based on Learning Behavior Data in the Intelligent Tutoring System”. W: *International Journal of Distance Education Technologies* 18 (lut. 2020), s. 73–89. DOI: 10.4018/IJDET.2020040105.
- [150] Santiago Schez-Sobrinio, Cristian Gmez-Portes, David Vallejo, Carlos Glez-Morcillo i Miguel Á. Redondo. „An Intelligent Tutoring System to Facilitate the Learning of Programming through the Usage of Dynamic Graphic Visualizations”. W: *Applied Sciences* 10.4 (2020). ISSN: 2076-3417. DOI: 10.3390/app10041518. URL: <https://www.mdpi.com/2076-3417/10/4/1518>.
- [151] *scikit-learn: machine learning in Python*. <https://scikit-learn.org/stable/>. Dostęp: 2024-02-01.
- [152] Sfenrianto Sfenrianto, Yustinus B. Hartarto, Habibullah Akbar, Mukhneri Mukhtar, Efriadi Efriadi i Mochamad Wahyudi. „An Adaptive Learning System based on Knowledge Level for English Learning”. W: *International Journal of Emerging Technologies in Learning (ijET)* 13.12 (grud. 2018), pp. 191–200. DOI: 10.3991/ijet.v13i12.8004. URL: <https://online-journals.org/index.php/ijet/article/view/8004>.
- [153] *Sharable Content Object Reference Model (SCORM®)*. <https://adlnet.gov/past-projects/scorm/>. Dostęp: 2024-01-25.

- [154] Saman Shishehchi, Seyed Yashar Banihashem i Nor Azan Mat Zin. „A proposed semantic recommendation system for e-learning: A rule and ontology based e-learning recommendation system”. W: *2010 International Symposium on Information Technology*. T. 1. 2010, s. 1–5. DOI: 10.1109/ITSIM.2010.5561329.
- [155] Jeremiah Hasudungan Sihombing, Kusuma Ayu Laksitowening i Eko Darwiyanto. „Personalized E-Learning Content Based On Felder-Silverman Learning Style Model”. W: *2020 8th International Conference on Information and Communication Technology (ICoICT)*. 2020, s. 1–6. DOI: 10.1109/ICoICT49345.2020.9166452.
- [156] *Simpful*. <https://simpful.readthedocs.io/en/latest/>. Dostęp: 2024-01-31.
- [157] Ninni Singh, Vinit Kumar Gunjan, Amit Kumar Mishra, Ram Krishn Mishra i Nishad Nawaz. „SeisTutor: A Custom-Tailored Intelligent Tutoring System and Sustainable Education”. W: *Sustainability* 14.7 (2022). ISSN: 2071-1050. DOI: 10.3390/su14074167. URL: <https://www.mdpi.com/2071-1050/14/7/4167>.
- [158] D. Sleeman i J.S. Brown. *Intelligent Tutoring Systems*. Computers and people series. Academic Press, 1982. ISBN: 9780126486803.
- [159] Moh Solehuddin, Encep Sopandi, Dedi Saputra, Erwin Dhaniswara, Sri Yulianto, Zhang Wei i Shanshan Xu. „Development of Adaptive E-Learning Content to Increase Learning Effectiveness”. W: *Journal International Inspire Education Technology* 2 (lip. 2023), s. 87–98. DOI: 10.55849/jiiiet.v2i2.457.
- [160] Mousazadeh Somayeh, Maryam Dehghani, Farzaneh Mozaffari, Seideh Madineh Ghasemnegad, Hamideh Hakimi i Bagherian Samaneh. „The effectiveness of E-learning in learning: A review of the literature”. W: *International journal of medical research & health sciences* 5.2 (2016), s. 86–91.
- [161] Anna Stachowiak, Patryk Żywica, Krzysztof Dyczkowski i Andrzej Wójtowicz. „An Interval-Valued Fuzzy Classifier Based on an Uncertainty-Aware Similarity Measure”. W: *Intelligent Systems'2014*. Red. P. Angelov, K.T. Atanassov, L. Dookovska, M. Hadjiski, V. Jotsov, J. Kacprzyk, N. Kasabov, S. Sotirov, E. Szmidt i S. Zadrozny. Cham: Springer International Publishing, 2015, s. 741–751. ISBN: 978-3-319-11313-5.
- [162] P.V. Praveen Sundar. „Quasi Framework: A New Student Disengagement Detection In Online Learning”. W: *International Journal of Engineering Research & Technology (IJERT)* 1 (grud. 2012).
- [163] Marcin Szczepański i Jacek Marciniak. „Application of a fuzzy controller in adaptive e-learning content used to evaluate student activity”. W: *Procedia Computer Science* 225 (2023). 27th International Conference on Knowledge Based and Intelligent Information and Engineering Systems (KES 2023), s. 2526–2535. ISSN: 1877-0509. DOI: <https://doi.org/10.1016/j.procs.2023.10.244>. URL: <https://www.sciencedirect.com/science/article/pii/S1877050923014023>.
- [164] Gopalakrishnan T i P. Sengottuvelan. „A hybrid PSO with Naïve Bayes classifier for disengagement detection in online learning”. W: *Program* 50.2 (sty. 2016), s. 215–224. ISSN: 0033-0337. DOI: 10.1108/PROG-07-2015-0047. URL: <https://doi.org/10.1108/PROG-07-2015-0047>.

- [165] Tomohiro Takagi i Michio Sugeno. „Fuzzy Identification of Systems and Its Applications to Modeling and Control”. W: *IEEE Trans. Systems, Man, and Cybernet* 15 (sty. 1985), s. 116–132. DOI: 10.1016/B978-1-4832-1450-4.50045-6.
- [166] Mingjie Tan i Peiji Shao. „Prediction of Student Dropout in E-Learning Program Through the Use of Machine Learning Method”. W: *International Journal of Emerging Technologies in Learning (ijET)* 10.1 (lut. 2015), pp. 11–17. DOI: 10.3991/ijet.v10i1.4189. URL: <https://online-journals.org/index.php/ijet/article/view/4189>.
- [167] Paul A. Trout. „Disengaged students and the decline of academic standards”. W: *Academic Questions* 10 (1997), s. 46–56. URL: <https://api.semanticscholar.org/CorpusID:143828696>.
- [168] Huong May Truong. „Integrating learning styles and adaptive e-learning system: Current developments, problems and opportunities”. W: *Computers in Human Behavior* 55 (2016), s. 1185–1193. ISSN: 0747-5632. DOI: <https://doi.org/10.1016/j.chb.2015.02.014>. URL: <https://www.sciencedirect.com/science/article/pii/S0747563215001120>.
- [169] Hsiao-Lin Tuan, Chi-Chin Chin i Shyang-Horng Shieh. „The development of a questionnaire to measure students’ motivation towards science learning”. W: *International Journal of Science Education* 27 (kw. 2005), s. 639–654. DOI: 10.1080/0950069042000323737.
- [170] Paraskevi Tzouveli, Phivos Mylonas i Stefanos Kollias. „An intelligent e-learning system based on learner profiling and learning resources adaptation”. W: *Computers & Education* 51.1 (2008), s. 224–238. ISSN: 0360-1315. DOI: <https://doi.org/10.1016/j.compedu.2007.05.005>. URL: <https://www.sciencedirect.com/science/article/pii/S0360131507000504>.
- [171] Manisha Verma, Yuta Nakashima, Noriko Takemura i Hajime Nagahara. „Multi-label Disengagement and Behavior Prediction in Online Learning”. W: *Artificial Intelligence in Education*. Red. Maria Mercedes Rodrigo, Noburu Matsuda, Alexandra I. Cristea i Vania Dimitrova. Cham: Springer International Publishing, 2022, s. 633–639. ISBN: 978-3-031-11644-5.
- [172] Mari Aulikki Virtanen, Maria Kääriäinen, Eeva Liikanen i Elina Haavisto. „The comparison of students’ satisfaction between ubiquitous and web-based learning environments”. W: *Education and Information Technologies* 22 (2017), s. 2565–2581.
- [173] Yuan Wang i Ryan Baker. „Grit and intention: Why do learners complete MOOCs?” W: *International review of research in open and distributed learning* 19.3 (2018).
- [174] Zeerak Waseem. „Are you a racist or am I seeing things? Annotator influence on hate speech detection on twitter”. W: *Proceedings of the first workshop on NLP and computational social science*. 2016, s. 138–142.
- [175] Geoffrey I. Webb. „Naïve Bayes”. W: *Encyclopedia of Machine Learning*. Red. Claude Sammut i Geoffrey I. Webb. Boston, MA: Springer US, 2010, s. 713–714. ISBN: 978-0-387-30164-8. DOI: 10.1007/978-0-387-30164-8\_576. URL: [https://doi.org/10.1007/978-0-387-30164-8\\_576](https://doi.org/10.1007/978-0-387-30164-8_576).
- [176] *Weka 3: Machine Learning Software in Java*. <https://www.cs.waikato.ac.nz/ml/weka/index.html>. Dostęp: 2024-02-17.

- [177] Etienne Wenger. *Artificial Intelligence and Tutoring Systems: Computational and Cognitive Approaches to the Communication of Knowledge*. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1987. ISBN: 0934613265.
- [178] Maximilian Wich, Hala Al Kuwatly i Georg Groh. „Investigating Annotator Bias with a Graph-Based Approach”. W: *Proceedings of the Fourth Workshop on Online Abuse and Harms*. Red. Seyi Akiwowo, Bertie Vidgen, Vinodkumar Prabhakaran i Zeerak Waseem. Online: Association for Computational Linguistics, list. 2020, s. 191–199. DOI: 10.18653/v1/2020.alw-1.22. URL: <https://aclanthology.org/2020.alw-1.22>.
- [179] xAPI. <https://xapi.com/>. Dostęp: 2024-02-17.
- [180] L.A. Zadeh. „Fuzzy sets”. W: *Information and Control* 8.3 (1965), s. 338–353. ISSN: 0019-9958. DOI: [https://doi.org/10.1016/S0019-9958\(65\)90241-X](https://doi.org/10.1016/S0019-9958(65)90241-X). URL: <https://www.sciencedirect.com/science/article/pii/S001999586590241X>.
- [181] Nick Zepke i Linda Leach. „Improving student engagement: Ten proposals for action”. W: *Active Learning in Higher Education* 11.3 (2010), s. 167–177. DOI: 10.1177/1469787410379680. eprint: <https://doi.org/10.1177/1469787410379680>. URL: <https://doi.org/10.1177/1469787410379680>.
- [182] Amal Zouaq, Roger Nkambou i Claude Frasson. „Bridging the gap between ITS and elearning: Towards learning knowledge objects”. W: *Intelligent Tutoring Systems: 9th International Conference, ITS 2008, Montreal, Canada, June 23-27, 2008 Proceedings* 9. Springer. 2008, s. 448–458.
- [183] Patryk Żywica, Krzysztof Dyczkowski, Andrzej Wójtowicz, Anna Stachowiak, Sebastian Szubert i Rafał Moszyński. „Development of a fuzzy-driven system for ovarian tumor diagnosis”. W: *Biocybernetics and Biomedical Engineering* 36.4 (2016), s. 632–643. ISSN: 0208-5216. DOI: <https://doi.org/10.1016/j.bbe.2016.08.003>. URL: <https://www.sciencedirect.com/science/article/pii/S0208521616300997>.