

Uniwersytet Warszawski
Wydział Matematyki, Informatyki i Mechaniki

**Adam Markiewicz, Tomasz Grabowski, Albert Rozmus,
Krzysztof Rutkowski, Wiktor Zuba**

Nr-y albumów: 334774, 305145, 248353, 319379, 320501

Sugerowanie wyboru ścieżki kształcenia zintegrowane z USOS

**Praca licencjacka
na kierunku INFORMATYKA**

Praca wykonana pod kierunkiem
dra Roberta Dąbrowskiego
Pion Zastępcy Kanclerza ds. Informatycznych

Czerwiec 2015

Oświadczenie kierującego pracą

Potwierdzam, że niniejsza praca została przygotowana pod moim kierunkiem i kwalifikuje się do przedstawienia jej w postępowaniu o nadanie tytułu zawodowego.

Data

Podpis kierującego pracą

Oświadczenie autora (autorów) pracy

Świadom odpowiedzialności prawnej oświadczam, że niniejsza praca dyplomowa została napisana przeze mnie samodzielnie i nie zawiera treści uzyskanych w sposób niezgodny z obowiązującymi przepisami.

Oświadczam również, że przedstawiona praca nie była wcześniej przedmiotem procedur związanych z uzyskaniem tytułu zawodowego w wyższej uczelni.

Oświadczam ponadto, że niniejsza wersja pracy jest identyczna z załączoną wersją elektroniczną.

Data

Podpis autora (autorów) pracy

Streszczenie

Praca poświęcona jest systemowi Hermes, który ma na celu wspomaganie procesu rejestracji na studiach. Jest to zrealizowane poprzez rekomendowanie studentowi najlepiej dopasowanych przedmiotów - zgodnych z jego zainteresowaniami, ścieżką studiów, lub też po prostu najłatwiejszych dla jego konkretnego przypadku. Dzieje się to w oparciu o algorytmy uczenia maszynowego korzystające z danych wszystkich innych studentów wgranych do bazy danych systemu. System pozwala również na przewidywanie ocen z przedmiotów oraz rekomendacje seminariów. W pracy znajduje się ponadto opis funkcjonalności, architektury, implementacji, zastosowanych algorytmów oraz organizacji pracy nad projektem.

Słowa kluczowe

Rekomendacje
Predykcje
Wnioskowanie
Analiza danych

Dziedzina pracy (kody wg programu Socrates-Erasmus)

11.0 Matematyka, Informatyka:
11.3 Informatyka

Klasyfikacja tematyczna

Information systems
Information systems applications
Decision support systems
Data analytics

Tytuł pracy w języku angielskim

Recomendation of educational path integrated with USOS

Spis treści

1. Wstęp	5
1.1. Opis Projektu	5
1.2. od Autorów	6
2. Funkcjonalności	7
2.1. Funkcjonalności dla Studentów	7
2.1.1. Użytkownik anonimowy	7
2.1.1.1. Predykcja ocen	9
2.1.1.2. Rekomendacja seminariów	11
2.1.1.3. Rekomendacja przedmiotów	13
2.1.1.4. Rekomendacja przedmiotów do seminarium	15
2.1.2. Użytkownik zalogowany	17
2.2. Funkcjonalności dla Administracji	17
3. Architektura	19
3.1. Schemat Architektury	19
3.2. Schemat współdziałania komponentów architektury i przepływu danych	20
4. Technologia	23
5. Zastosowane Algorytmy	25
5.0.1. Algorytmy użyte przy proponowaniu przedmiotów	26
5.0.2. Algorytmy używane przy predykcji seminariów	27
5.0.3. Algorytmy używane przy predykcji ocen	27
6. Organizacja pracy oraz podział obowiązków	29
6.1. Praca nad systemem	29
6.2. Organizacja Pracy	30
6.3. Podział Obowiązków	31
7. Podsumowanie	33
Bibliografia	35

Rozdział 1

Wstęp

1.1. Opis Projektu

Głównym celem projektu Hermes było stworzenie serwisu internetowego wspierającego studentów w procesie zoptymalizowania wyboru przedmiotów. Serwis ma za zadanie umożliwić studentom lepsze planowanie ścieżki studiów i kariery zawodowej poprzez proponowanie przedmiotów, które mogą pasować do ich upodobań i predyspozycji.

Dodatkowym spełnionym wymaganiem projektu jest oferowanie usług przewidywania dla konkretnych studentów ocen z przedmiotów, których jeszcze nie ukończyli lub bądź nie podjęli.

Serwis oferuje również wsparcie dla uniwersytetu w postaci przewidywania ilości studentów którzy zapiszą się na konkretny przedmiot.

W celu obliczania predykcji system wykorzystywać będzie uczenie maszynowe na statystykach wszystkich studentów Uniwersytetu, a także ewentualnie wybrane przedmioty i uzyskane oceny przez studenta proszącego o propozycję.

Projekt został zrealizowany na zlecenie działu sieci komputerowych Uniwersytetu Warszawskiego.

1.2. od Autorów

Wybraliśmy ten projekt, ponieważ sami jesteśmy studentami i jesteśmy świadomi trudności związanych z wyborem przedmiotów. Oferta programowa Uniwersytetu Warszawskiego jest ogromna. W chwili pisania tej pracy po wyszukaniu w systemie usos zwróconych jest ponad 20000 przedmiotów! Dodatkowo opisy często są niejasne, szcątkowe, bądź brakuje sylabusu. Czasem tematyka przedmiotu obieralnego jest na tyle odmienna od dotychczasowego materiału poznanego przez studenta, że może on nie być świadomym, czy dany materiał odpowiada jego predyspozycjom i preferencjom.

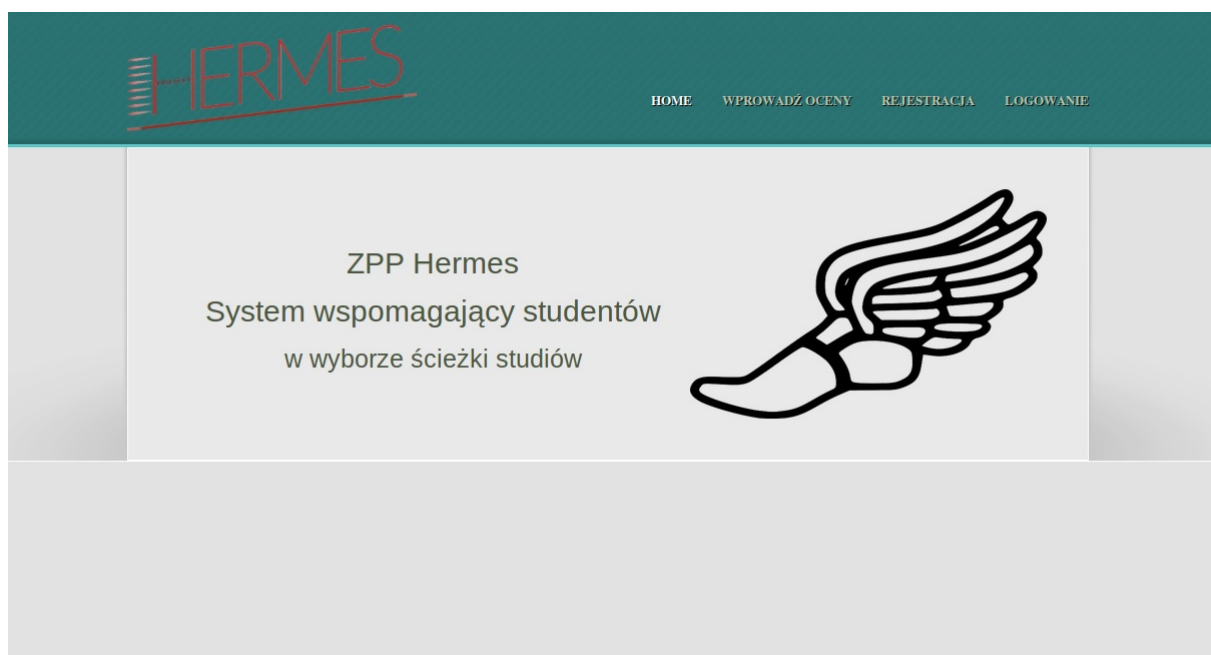
Pragniemy zaadresować te problemy za pomocą systemu Hermes. Pisząc go, przyświecała nam idea stworzenia drogowskazu dla studentów, którzy nie są pewni, w jakim kierunku powinni się specjalizować. Wierzimy, iż system dzięki udanym sugestiom zwiększy liczbę studentów, którzy będą naprawdę zainteresowani tematyką przedmiotu, a zmniejszy liczbę studentów zapisanych na przedmioty zupełnie niedopasowane do ich predyspozycji, co często kończy się słabą oceną bądź nawet brakiem zaliczenia. Dzięki temu skorzystają na tym studenci, którzy z większym prawdopodobieństwem zapiszą się na pasujące do ich osiągnięć i predyspozycji przedmioty, z których zdobędą możliwie wiele interesującej ich wiedzy. Zmniejszy się również dla nich ryzyko zapisania się na przedmiot, który w ogóle nie pasuje do ich zainteresowań. Zyska również Uniwersytet, którego studenci będą bardziej zadowoleni z oferty programowej, wzrośnie jakość wiedzy absolwentów oraz lepiej zostanie wykorzystany potencjał studentów. Pomoże to uczelni być lepiej postrzeganą, bardziej pożądaną przez przyszłych studentów a także zwiększyć jej prestiż.

Tomasz Grabowski
Adam Markiewicz
Albert Rozmus
Krzysztof Rutkowski
Wiktor Zuba

Rozdział 2

Funkcjonalności

Przy wejściu na stronę systemu użytkownik otrzymuje stronę startową która jest jednocześnie interfejsem.



Ekran startowy strony

Poniższe podrozdziały opisują funkcjonalności zależne od wybranej opcji.

2.1. Funkcjonalności dla Studentów

2.1.1. Użytkownik anonimowy

W tym podrozdziale znajduje się opis funkcjonalności dostępnych dla użytkownika anonimowego - w domyśle studenta.

W przypadku wyboru zakładki "wprowadź oceny" student uzyskuje możliwość wprowadzenia danych dotyczących statystyki jego zaliczeń przedmiotów w celu uzyskania następujących predykcji: predykcja oceny z przedmiotu, rekomendacja seminarium oraz rekomendacja przedmiotów na podstawie osiągnięć bądź rekomendacja przedmiotów dla wybranego seminarium. Interfejs ten umożliwia dodawanie dowolnej liczby przedmiotów za pomocą przycisku "plus". Dodatkowo przy wpisywaniu nazwy przedmiotu pojawia się podpowiedź sugerująca istniejące nazwy przedmiotów z bazy pasujące do wpisanego wzorca.

The screenshot shows the HERMES application interface. At the top is a dark teal header with the HERMES logo on the left and navigation links (HOME, WPROWADŹ OCENY, REJESTRACJA, LOGOWANIE) on the right. The main content area is light gray. On the left, there is a sidebar with several sections:

- Wybierz algorytm rekomendacji przedmiotów:**
 - ☐ strategia wykorzystująca algorytm regulowy
 - ☐ lista priorytetowa najłatwiejszych przedmiotów
 - ☐ dobierz w sposób losowy
 - ☐ strategia najbliższych sąsiadów
- ☐ Wybierz seminarium do rekomendacji przedmiotów
- A dropdown menu showing "Systemy rozproszone".
- Wybierz algorytm rekomendacji seminarium:**
 - ☐ strategia wykorzystująca algorytm lasów losowych
 - ☐ strategia najbliższych sąsiadów
- ☐ Wybierz przedmiot do predykcji oceny:
- A dropdown menu showing "Zaawansowane systemy operacyjne".

- On the right, there are two identical input forms:
- Przedmiot:** A text input field.
- Ocena:** A row of seven buttons labeled B, 2, 3, 3+, 4, 4+, 5, 5!.
- Below these forms is a circular button with a plus sign (+).
- At the bottom right is a dark gray button labeled "WYŚLIJ".

interfejs po wyborze opcji "wprowadź oceny"

2.1.1.1. Predykcja ocen

Użytkownik wybiera predykcję oceny z przedmiotu za pomocą opcji ”wybierz przedmiot do predykcji oceny”

The screenshot shows the HERMES application interface. At the top, there is a teal header with the HERMES logo and navigation links: HOME, WPROWADŹ OCENY, REJESTRACJA, and LOGOWANIE. The main content area is divided into two columns. The left column contains a form for selecting recommendation algorithms and subjects. It includes a section for 'Wybierz algorytm rekomendacji przedmiotów' with four radio button options: 'strategia wykorzystująca algorytm regulowy', 'lista priorytetowa najłatwiejszych przedmiotów', 'dobierz w sposób losowy', and 'strategia najbliższych sąsiadów'. Below this is a checkbox for 'Wybierz seminarium do rekomendacji przedmiotów' and a dropdown menu currently showing 'Systemy rozproszone'. Another section, 'Wybierz algorytm rekomendacji seminarium', has two radio button options: 'strategia wykorzystująca algorytm lasów losowych' and 'strategia najbliższych sąsiadów'. At the bottom of this column is a checked checkbox for 'Wybierz przedmiot do predykcji oceny:' and a dropdown menu showing 'Zaawansowane systemy operacyjne'. The right column contains two identical input sections. Each has a 'Przedmiot:' text input field and an 'Ocena:' row of seven green buttons labeled B, 2, 3, 3+, 4, 4+, 5, and 5!. Below the second 'Ocena:' row is a plus icon in a circle and a 'WYŚLIJ' button.

wybór opcji predykcji oceny z przedmiotu

Po wybraniu tej opcji użytkownik może z rozwijanej listy wybrać przedmiot, z którego chce predykcję.

The screenshot shows the HERMES application interface. On the left, there is a dropdown menu titled "Wybierz algorytm rekomendacji przedmiotów:" with a list of subjects. The subject "Matematyka obliczeniowa 2" is highlighted. On the right, there is a list of subjects with their predicted grades. The subjects and their grades are:

Przedmiot	Ocena
Analiza matematyczna dla informatyków 1	B 2 3 3+ 4 4+ 5 5!
Analiza matematyczna dla informatyków 2	B 2 3 3+ 4 4+ 5 5!
Podstawy matematyki	B 2 3 3+ 4 4+ 5 5!
Wstęp do programowania	B 2 3 3+ 4 4+ 5 5!

wybór przedmiotu do predykcji

Po jego wybraniu i użyciu opcji "wyślij" użytkownik otrzymuje wynik w postaci następującej strony:

The screenshot shows the HERMES application interface displaying the predicted grade for the selected subject. The predicted grade is 3.0.

Proponowana ocena z Matematyka obliczeniowa 2

3.0

The footer of the page contains the following information:

NOWOŚCI	KONTAKT	NIC CIEKAWEGO
Hermes rusza!	EMAIL: info-hermes@google.com TELEFON: 000-000-000	

- ZPP

Przykładowy rezultat zapytania o predykcję oceny

2.1.1.2. Rekomendacja seminariów

Użytkownik chcąc otrzymać rekomendowane seminaRIA musi wybrać opcję otrzymania rekomendacji seminarium wg jednego z dwóch algorytmów: strategii lasów losowych bądź najbliższych sąsiadów.

The screenshot shows the HERMES application interface. At the top, there is a navigation bar with the HERMES logo and links for HOME, WPROWADZ OCENY, REJESTRACJA, and LOGOWANIE. The main content area is divided into two sections. On the left, there is a form for selecting recommendation algorithms. It has two main sections: 'Wybierz algorytm rekomendacji przedmiotów' and 'Wybierz algorytm rekomendacji seminarium'. The first section has four radio buttons: 'strategia wykorzystująca algorytm regulowy', 'lista priorytetowa najłatwiejszych przedmiotów', 'dobierz w sposób losowy', and 'strategia najbliższych sąsiadów'. The second section has two radio buttons: 'strategia wykorzystująca algorytm lasów losowych' and 'strategia najbliższych sąsiadów'. Below these, there are two dropdown menus: 'Wybierz seminarium do rekomendacji przedmiotów' (currently showing 'Systemy rozproszone') and 'Wybierz przedmiot do predykcji oceny:' (currently showing 'Zaawansowane systemy operacyjne'). On the right, there is a list of recommended seminars. Each entry consists of a 'Przedmiot' (Subject) and an 'Ocena' (Rating). The subjects are 'Analiza matematyczna dla informatyków 1', 'Analiza matematyczna dla informatyków 2', 'Podstawy matematyki', and 'Wstęp do programowania'. The ratings are shown as a row of buttons: B, 2, 3, 3+, 4, 4+, 5, 5!. In the first two entries, the '3+' button is highlighted. In the third entry, the '4' button is highlighted. In the fourth entry, the '3+' button is highlighted.

Wybierz algorytm rekomendacji przedmiotów:

- ☐ strategia wykorzystująca algorytm regulowy
- ☐ lista priorytetowa najłatwiejszych przedmiotów
- ☐ dobierz w sposób losowy
- ☐ strategia najbliższych sąsiadów

☐ Wybierz seminarium do rekomendacji przedmiotów

Systemy rozproszone

Wybierz algorytm rekomendacji seminarium:

- ☒ strategia wykorzystująca algorytm lasów losowych
- ☒ strategia najbliższych sąsiadów

☐ Wybierz przedmiot do predykcji oceny:

Zaawansowane systemy operacyjne

Przedmiot: Analiza matematyczna dla informatyków 1

Ocena: B 2 3 3+ 4 4+ 5 5!

Przedmiot: Analiza matematyczna dla informatyków 2

Ocena: B 2 3 3+ 4 4+ 5 5!

Przedmiot: Podstawy matematyki

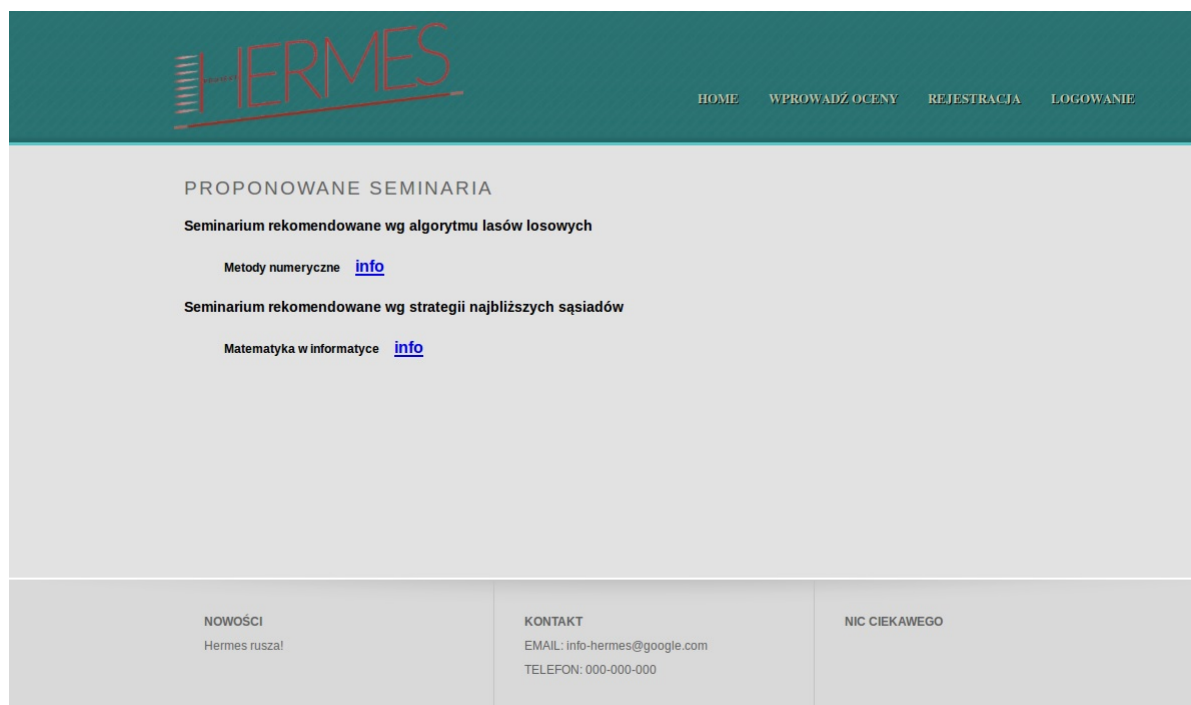
Ocena: B 2 3 3+ 4 4+ 5 5!

Przedmiot: Wstęp do programowania

Ocena: B 2 3 3+ 4 4+ 5 5!

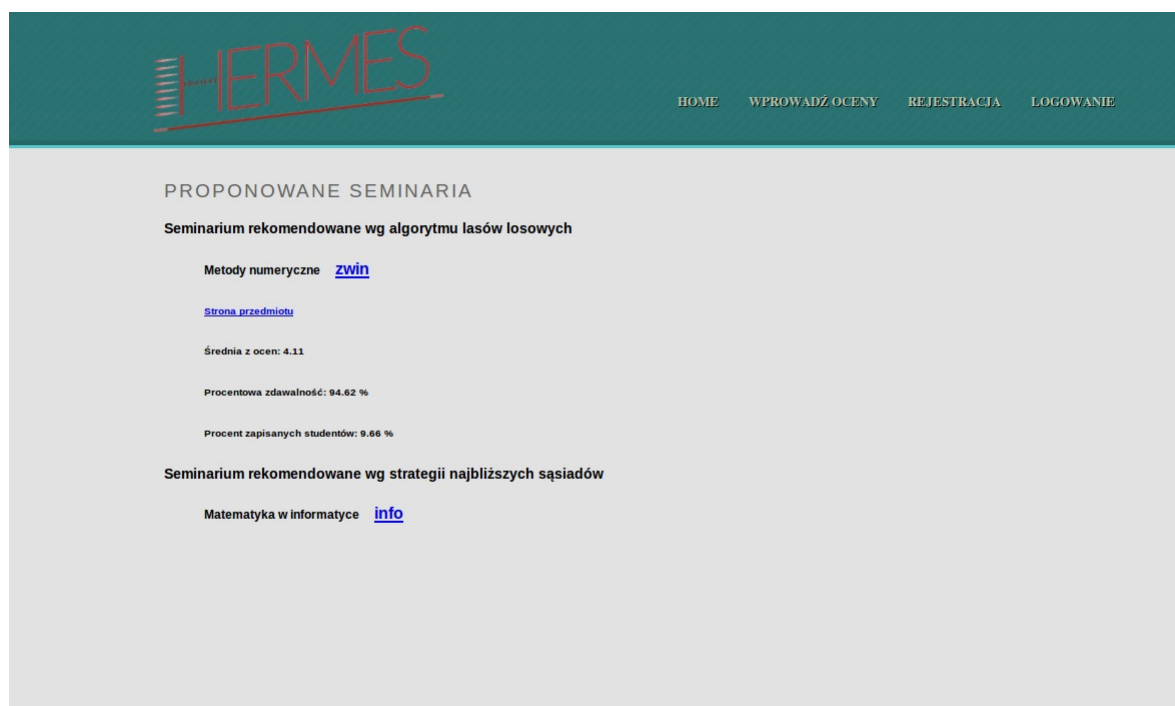
wybór opcji rekomendacji seminarium

Na podstawie danych wprowadzonych przez studenta oraz wybranych przez niego algorytmów rekomendacji seminarium student otrzymuje wyniki predykcji



Przykładowy rezultat prośby o zarekomendowanie seminarium

Dla każdego seminarium użytkownik może podejrzeć informacje dotyczące zwróconego przedmiotu, takie jak jego strona w systemie usos czy też podstawowe statystyki jak zdawalność i średnia ocena



Przykładowy rezultat prośby o obliczenie rankingu seminariów

2.1.1.3. Rekomendacja przedmiotów

Po wybraniu opcji rekomendacji przedmiotów student otrzymuje wybór kilku opcji. Może wybrać opcje "znajdź najłatwiejsze", szczęśliwy traf - "sposób losowy", "ranking wg algorytmu najbliższych sąsiadów" oraz "znajdź wg algorytmu regułowego".

The screenshot shows the HERMES system interface. At the top, there is a header with the HERMES logo and navigation links: HOME, WPROWADŹ OCENY, REJESTRACJA, and LOGOWANIE. The main content area is divided into two columns. The left column contains a form for selecting recommendation algorithms and seminars. The right column displays a list of recommended subjects with their corresponding grades.

Wybierz algorytm rekomendacji przedmiotów:

- ☐ strategia wykorzystująca algorytm regułowy
- ☒ lista priorytetowa najłatwiejszych przedmiotów
- ☒ dobierz w sposób losowy
- ☐ strategia najbliższych sąsiadów

☐ Wybierz seminarium do rekomendacji przedmiotów

Systemy rozproszone

Wybierz algorytm rekomendacji seminarium:

- ☐ strategia wykorzystująca algorytm lasów losowych
- ☐ strategia najbliższych sąsiadów

☐ Wybierz przedmiot do predykcji oceny:

Zaawansowane systemy operacyjne

Przedmiot: Analiza matematyczna dla informatyków 1

Ocena: B 2 3 3+ 4 4+ 5 5!

Przedmiot: Analiza matematyczna dla informatyków 2

Ocena: B 2 3 3+ 4 4+ 5 5!

Przedmiot: Podstawy matematyki

Ocena: B 2 3 3+ 4 4+ 5 5!

Przedmiot: Wstęp do programowania

Ocena: B 2 3 3+ 4 4+ 5 5!

Interfejs użytkownika dla opcji rekomendacji przedmiotów - wybór algorytmów

Jako wynik każdej z tych rekomendacji student otrzymuje ranking 5 przedmiotów które wg danego algorytmu mają największą szansę pasować do zainteresowań studenta.

The screenshot shows the HERMES system interface. At the top, there is a teal header with the HERMES logo on the left and navigation links (HOME, WPROWADŹ OCENY, REJESTRACJA, LOGOWANIE) on the right. Below the header, the page is divided into two main sections. The first section is titled "NAJŁATWIEJSZE PRZEDMIOTY" and contains a list of five recommended subjects, each with an "info" link. The second section is titled "SZCZĘŚLIWY TRAF" and contains a list of four recommended subjects, each with an "info" link.

NAJŁATWIEJSZE PRZEDMIOTY

Lista proponowanych przedmiotów:

1. Weryfikacja wspomagana komputerowo [info](#)
2. Sztuczna Inteligencja i systemy doradcze [info](#)
3. Optymalizacja 1 [info](#)
4. Kryptografia [info](#)
5. Wstęp do biologii obliczeniowej [info](#)

SZCZĘŚLIWY TRAF

Lista proponowanych przedmiotów:

1. Algorytmika [info](#)
2. Sztuczna Inteligencja i systemy doradcze [info](#)
3. Data mining [info](#)
4. Optymalizacja 1 [info](#)

Przykładowy rezultat dla rekomendacji przedmiotów wg zainteresowań studenta

Każdy przedmiot po rozwinięciu opisu zawiera proste statystyki, tak jak dla rekomendacji seminariów : ogólną zdawalność, średnią z ocen, ogólny procent studentów uczęszczających na ten przedmiot czy link do strony przedmiotu w systemie USOS.

This screenshot shows the detailed view of a recommended subject in the HERMES system. The layout is similar to the previous one, but the second item in the "NAJŁATWIEJSZE PRZEDMIOTY" list, "Sztuczna Inteligencja i systemy doradcze", is expanded. It now shows a "zwin" link, a "Strona przedmiotu" link, and three statistics: "Średnia z ocen: 3.85", "Procentowa zdawalność: 88.08 %", and "Procent zapisanych studentów: 29.35 %". The other subjects in the list remain unchanged.

NAJŁATWIEJSZE PRZEDMIOTY

Lista proponowanych przedmiotów:

1. Weryfikacja wspomagana komputerowo [info](#)
2. Sztuczna Inteligencja i systemy doradcze [zwin](#)
[Strona przedmiotu](#)
Średnia z ocen: 3.85
Procentowa zdawalność: 88.08 %
Procent zapisanych studentów: 29.35 %
3. Optymalizacja 1 [info](#)
4. Kryptografia [info](#)
5. Wstęp do biologii obliczeniowej [info](#)

SZCZĘŚLIWY TRAF

Lista proponowanych przedmiotów:

Opis przedmiotu po rozwinięciu (tu Sztuczna Inteligencja)

2.1.1.4. Rekomendacja przedmiotów do seminarium

Ostatnią funkcjonalnością dla studenta jest możliwość otrzymania rekomendacji do semianarium. Student, chcąc otrzymać taką predykcję, musi zaznaczyć opcję "Wybierz seminarium do rekomendacji przedmiotów"

The screenshot displays the HERMES web application interface. At the top, there is a dark teal header with the HERMES logo on the left and navigation links (HOME, WPROWADŹ OCENY, REJESTRACJA, LOGOWANIE) on the right. The main content area is light gray and contains a recommendation tool. On the left, a box titled "Wybierz algorytm rekomendacji przedmiotów:" contains four radio button options: "strategia wykorzystująca algorytm regulowy", "lista priorytetowa najłatwiejszych przedmiotów", "dobierz w sposób losowy", and "strategia najbliższych sąsiadów". Below this, a checked checkbox "Wybierz seminarium do rekomendacji przedmiotów" is followed by a dropdown menu currently showing "Systemy rozproszone". Underneath, another section "Wybierz algorytm rekomendacji seminarium:" has two radio button options: "strategia wykorzystująca algorytm lasów losowych" and "strategia najbliższych sąsiadów". At the bottom of this box, an unchecked checkbox "Wybierz przedmiot do predykcji oceny:" is followed by a dropdown menu showing "Zaawansowane systemy operacyjne". To the right of these options, there are three rows of subject and grade selection. Each row has a "Przedmiot:" label and a text input field, followed by an "Ocena:" label and a row of seven buttons representing grades: B, 2, 3, 3+, 4, 4+, 5, and 5!. The first row has "Analiza matematyczna dla informatyków 1" and grade "4" selected. The second row has "Analiza matematyczna dla informatyków 2" and grade "3+" selected. The third row has "Podstawy matematyki" and grade "4" selected. A fourth row is partially visible with "Wstęp do programowania" and grade "3+" selected.

wybór opcji rekomendacji przedmiotów do seminarium

Po wybraniu tej opcji student z rozwijanej listy wybiera interesujące go seminarium spośród seminariów znajdujących się w bazie

The screenshot shows the HERMES web application interface. At the top, there is a navigation bar with the HERMES logo and links for HOME, WPROWADŹ OCENY, REJESTRACJA, and LOGOWANIE. The main content area is divided into two columns. The left column contains a section titled 'Wybierz algorytm rekomendacji przedmiotów:' with four radio button options: 'strategia wykorzystująca algorytm regulowy', 'lista priorytetowa najłatwiejszych przedmiotów', 'dobierz w sposób losowy', and 'strategia najbliższych sąsiadów'. Below these options is a checked checkbox labeled 'Wybierz seminarium do rekomendacji przedmiotów' and a dropdown menu. The dropdown menu is open, showing a list of seminars: 'Systemy rozproszone', 'Systemy rozproszone', 'Języki programowania', 'Zagadnienia programowania obiektowego', 'Wybrane aspekty inżynierii oprogramowania', 'Analiza, wizualizacja i optymalizacja oprogramowania', 'Innowacyjne zastosowania informatyki', 'Molekularna biologia obliczeniowa', 'Algorytmika', 'Metody numeryczne' (highlighted in orange), 'Matematyka w informatyce', and 'Zaawansowane systemy operacyjne'. The right column displays four rows of subject recommendations. Each row has a 'Przedmiot:' label followed by a text input field and an 'Ocena:' label followed by a row of seven buttons labeled B, 2, 3, 3+, 4, 4+, 5, and 5!. The subjects and their corresponding scores are: 1. 'Analiza matematyczna dla informatyków 1' with score 4; 2. 'Analiza matematyczna dla informatyków 2' with score 3+; 3. 'Podstawy matematyki' with score 4; 4. 'Wstęp do programowania' with score 3+.

wybór seminarium do rekomendacji przedmiotów

Po użyciu przycisku "wyślij" student otrzymuje ranking 5 najbardziej zgodnych do jego osiągnięć i wybranego seminarium przedmiotów

The screenshot shows the HERMES web application interface displaying the recommended subjects for the 'Metody numeryczne' seminar. The top navigation bar is the same as in the previous screenshot. The main content area has a title 'PRZEDMIOTY REKOMENDOWANE DLA SEMINARIUM METODY NUMERYCZNE' and a subtitle 'Lista proponowanych przedmiotów:'. Below this, there is a list of five recommended subjects, each followed by a blue 'info' link: 1. 'Weryfikacja wspomagana komputerowo info', 2. 'Wstęp do biologii obliczeniowej info', 3. 'Zaawansowane systemy operacyjne info', 4. 'Kompresja danych info', and 5. 'Przetwarzanie dużych danych info'.

rezultat predykcji przedmiotów związanych z seminarium

Analogicznie do poprzednich funkcjonalności możliwe jest podejrzenie informacji o przedmiotach.

2.1.2. Użytkownik zalogowany

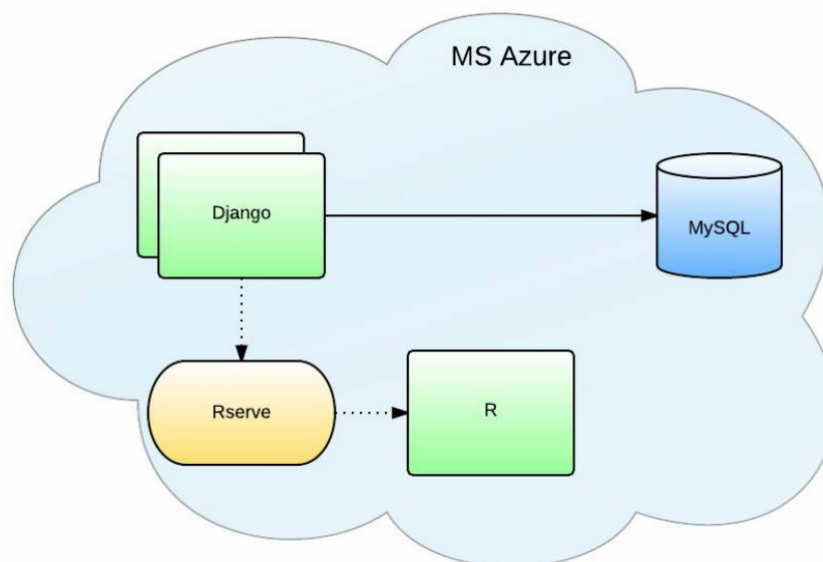
2.2. Funkcjonalności dla Administracji

Domyślnie miała być jeszcze funkcjonalność dla administracji, tzn predykcja popularności przedmiotów, niestety zabrakło nam czasu by ją zaimplementować.

Rozdział 3

Architektura

3.1. Schemat Architektury



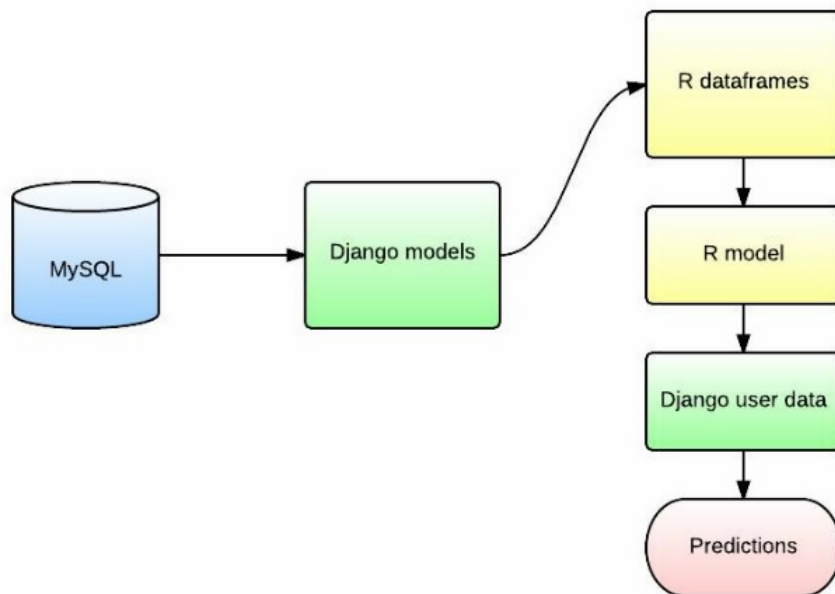
Architektura systemu Hermes

W architekturze i logice naszego systemu wyróżniamy następujące komponenty:

- **Chmura (MS Azure)** - Sercem systemu jest główny serwer wraz z innymi usługami znajdujący się w chmurze internetowej. Znajduje się na niej serwer bazy danych, serwer WWW oraz serwer usług analitycznych.
- **RDB (MySQL)** - Relacyjna baza danych zawierająca statystyczne dane dotyczące zdawalności przedmiotów przez studentów, ich zapełnienia, popularność itd. Stanowi bazę do tworzenia predykcji.
- **Sytem R (R)** - język i zestaw bibliotek uczenia maszynowego wykorzystywany w predykcjach.

- **Rserve (Rserve)** - serwer pośredniczący między stroną WWW a systemem R. Strona komunikuje się z R poprzez komunikaty wysyłane do Rserve.
- **Strona WWW (Django)** - interfejs za pomocą którego użytkownik może przysyłać prośby o wykonanie udostępnianych przez system rekomendacji.
- **Serwer WWW (Django)** - udostępnia użytkownikom stronę internetową, w naszym systemie pośredniczy między interfejsem użytkownika a bazą danych. Pośredniczy również w komunikacji z serwerem analitycznym.

3.2. Schemat współdziałania komponentów architektury i przepływu danych



Schemat przepływu danych w systemie

Schemat działania i komunikacji między poszczególnymi komponentami wygląda następująco:

1. Na samym początku przy uruchomieniu systemu django ładuje modele na podstawie bazy danych MySQL.
2. R zaczyna przetwarzać dane w bazie i na ich podstawie tworzy dataframes (tzn format danych zgodny z R) które są bazą obliczania predykcji oraz rekomendacji.
3. Po wykonaniu przetworzenia danych i obliczenia statystyk aktywuje się serwer WWW i system staje się dostępny dla użytkowników.
4. Użytkownik wchodzi na stronę i za pośrednictwem jej interfejsu może przesłać do serwera prośbę o rekomendację bądź predykcję.

5. Serwer WWW po odebraniu prośby o rekomendacje przesyła żądanie do Rserve z poleceniem obliczenia predykcji, wykorzystując się przy tym odebrane wcześniej dane użytkownika.
6. R dokonuje obliczeń na podstawie wyliczonego wcześniej modelu i zwraca do serwera WWW predykcję jako model R.
7. Serwer WWW odbiera rezultat zapytania, konwertuje go na format zgodny z formatem Django i wyświetla go użytkownikowi za pośrednictwem strony WWW.

Rozdział 4

Technologia

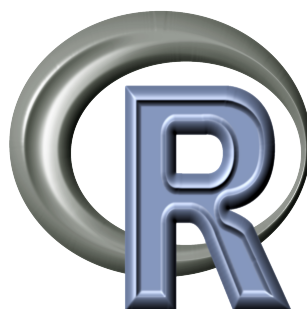
Technologie wykorzystywane w naszym systemie:



Microsoft Azure - Azure jest komercyjną platformą obsługiwaną przez Microsoft. Udostępnia ona usługi związane z chmurą internetową (tzw cloud-computing). W naszym systemie znajduje się na niej serwer WWW a także serwer bazy danych.



MySQL - Jeden z najpopularniejszych serwerów relacyjnych baz danych. Znajduje się w nim relacyjna baza danych zawierająca dane niezbędne do stworzenia modeli uczenia maszynowego.



R - język i zestaw bibliotek używany w celu analizy danych a także obliczania predykcji dla studentów bądź uniwersytetu.



- **Python oraz Django** - technologie, za pomocą których tworzymy webowy interfejs użytkownika.



- **USOS Api** - API udostępniane przez system USOS. Nasz system wykorzystuje go w celu zebrania danych zalogowanego użytkownika niezbędnych do rekomendacji.

Rozdział 5

Zastosowane Algorytmy

Przy tworzeniu algorytmów wykorzystaliśmy pakiet statystyczny R. Gwarantował on dużą kontrolę nad algorytmami, których używaliśmy. W naszej pracy pojawiły się wszystkie najważniejsze rodzaje algorytmów używanych w eksploracji danych:

- klasyfikacja
- poszukiwanie reguł asocjacyjnych
- grupowanie

Najwięcej pojawiło się zastosowań pierwszej z metod. Czasem używaliśmy też podejścia statystycznego, nie ucząc jakiegoś modelu, lecz podejmując decyzje na podstawie statystyk. W zastosowanych algorytmach często używaliśmy odpowiednio zdefiniowanej funkcji odległości między obiektami w naszym modelu. Obiektem był tu wektor n -elementowy ocen studenta pomnożonych o 2 oraz 11 w przypadku oceny 5! a także 0, gdy student nie uczestniczył w przedmiocie, dla dwóch takich wektorów definiujemy miarę odległości między studentami $v = (v_1, \dots, v_n)$ i $w = (w_1, \dots, w_n)$:

$$\text{dist}(v, w) = \sum_{k=1}^n \begin{cases} 10 & \text{gdy } (v_k = 0) \oplus (w_k = 0) \\ |w_k - v_k| & \text{wpp} \end{cases}$$

kod w R:

```
1 dist <- function(v,w) {  
2   sum <- 0  
3   for (i in 1:length(v)) {  
4     if (xor((v[i] == 0), w[i] == 0)) {  
5       sum <- sum + 10  
6     }  
7     else {  
8       sum <- sum + abs(v[i] - w[i])  
9     }  
10  }  
11  return(sum)  
12 }
```

dist.R

Podana funkcja mocniej wyraża fakt, że studenci brali różne przedmioty niż to że mieli różne oceny. Używaliśmy jej w algorytmach opartych na metodzie k najbliższych sąsiadów, której

używaliśmy w wielu predykcjach. W naszym systemie mieliśmy kilka rodzajów predykcji i rekomendacji:

- proponowanie przedmiotów
- predykcja seminarium
- predykcja ocen

5.0.1. Algorytmy użyte przy proponowaniu przedmiotów

Przy proponowaniu przedmiotów obieralnych użyliśmy 4 strategie:

- wyliczanie reguł asocjacyjnych
- strategia najbliższych sąsiadów
- przydział w oparciu o statystyki
- przydział losowy

Reguły asocjacyjne wyliczyliśmy za pomocą algorytmu apriori dostępnego w pakiecie 'arules'. Algorytm brał na wejściu koszyki przedmiotów, czyli zestawy przedmiotów obieralnych, które wzięli poszczególni studenci, zaś na wyjściu zwracał reguły postaci 'jeśli student wziął przedmioty p_1, \dots, p_k to prawdopodobnie weźmie też przedmiot p_l '. To 'prawdopodobnie' zależy od miar *support* i *confidence* podawanych na wejściu algorytmu. Miary *support* i *confidence* ustawiliśmy tak, by reguł nie było za dużo, ale też były sensowne.

Strategia najbliższych sąsiadów została użyta w oparciu o miarę *dist* przedstawioną wyżej. Polegała ona na tym, że spośród pewnej liczby najbliższych studentowi, któremu chcemy coś zaproponować, sąsiadów, bierzemy przedmioty, które oni brali. W tym podejściu próbujemy zaklasyfikować studenta do podobnej grupy i przydzielić mu to co brali 'podobni' studenci. Kod w R:

```
1 recomNearestSub <- function(k, student) {
2   n <- dim(data)[1]
3   A <- matrix(0,n,2)
4   for (i in 1:n) {
5     A[i,1] <- dist(student, data[i,])
6     A[i,2] <- i
7   }
8   bestNb <- A[order(A[,1])[1:k],2]
9   studNotChosen <- which(student == 0)
10  recom <- c()
11  for (i in 1:length(bestNb)) {
12    nbSub <- which(data[bestNb[i],31:50] > 0)
13    recom <- unique(c(recom, nbSub[nbSub %in% studNotChosen]))
14  }
15  return(recom)
16 }
```

nearestSub.R

Przydział w oparciu o statystyki to propozycja przedmiotów, które prawdopodobnie byłoby studentowi najłatwiej zaliczyć. Statystykę 'łatwości' przedmiotu stanowiła średnia ważona supportu, średniej ocen i zdawalności przedmiotu.

Przydział losowy polegał na braniu z jenostajnym rozkładem prawdopodobieństwa przedmiotów, których student jeszcze nie wziął.

5.0.2. Algorytmy używane przy predykcji seminariów

Przy proponowaniu seminariów użyliśmy 2 strategii:

- klasyfikator najbliższych sąsiadów
- klasyfikator lasów losowych

Klasyfikator najbliższych sąsiadów, tak jak w przypadku proponowania przedmiotów, również korzystał ze zdefiniowanej funkcji odległości i ze zbioru decyzji dla najbliższych sąsiadów, czyli seminarium, które wybrali najbliżsi sąsiedzi, jako proponowane wybierał mode, czyli najczęściej występujące wśród sąsiadów seminarium.

Klasyfikator lasów losowych budowaliśmy w oparciu o pakiet 'randomForest'. Jest to klasyfikator, który buduje się w oparciu o klasyfikacje na pewnej liczbie drzew decyzyjnych, następnie agregując decyzje z tych drzew, najczęściej przez głosowanie. Dawał on gorsze wyniki niż klasyfikator najbliższych sąsiadów. Na danych testowych, skuteczność klasyfikatora knn wynosiła 0.45 podczas gdy skuteczność lasów losowych tylko 0.26. Jednak zaletą tego klasyfikatora jest to, że buduje on już wcześniej model i decyzja dla danych wejściowych studenta obliczana jest bardzo szybko, w przeciwieństwie do klasyfikatora knn. Kod w R:

```
1 library(randomForest)
2 classifierf<-randomForest(data[,1:50], data[,51])
3 predictRf <- function(student) {
4   return(round(predict(classifierf, student)))
5 }
```

rf.R

5.0.3. Algorytmy używane przy predykcji ocen

Przy predykcji ocen użyliśmy wyłącznie klasyfikatora najbliższych sąsiadów. W przypadku wyboru algorytmu uczenia nadzorowanego, musielibyśmy trenować nasz klasyfikator dla każdego przedmiotu, co przy dużej liczbie przedmiotów jest zadaniem dość karkołomnym. Dlatego zdecydowaliśmy się na algorytm uczenia nienadzorowanego, który dobrze się sprawdził w poprzednich predykcjach. Tutaj algorytm wybiera 100 najbliższych sąsiadów i wybiera tych, którzy wzięli dany przedmiot, następnie przelicza średnią ocen wybranych studentów z tego przedmiotu i zwraca ją jako wynik klasyfikacji.

Rozdział 6

Organizacja pracy oraz podział obowiązków

6.1. Praca nad systemem

Za namową zamawiającego na początku zdecydowaliśmy się zrealizować projekt za pomocą technologii koncernu Microsoft. Otrzymaliśmy od niego licencję Bizspark która dała nam licencję na swobodne wykorzystywanie produktów Microsoftu przez 2 lata. Dodatkowo Bizspark umożliwiał korzystanie z usługi Azure w zakresie abonamentu w wysokości 150 euro na miesiąc. Dlatego wybraliśmy chmurę Azure jako nasz serwer. Planowaliśmy dodatkowo zrealizować obliczanie predykcji za pomocą Sql Server Analysis Services a frontend za pomocą .NET.

Z projektem jednak wiązały się dość znaczące problemy. Pierwszym z nich była zmiana technologii użytej w celu obliczania predykcji, na którą zdecydowaliśmy się w lutym. Przed rozpoczęciem pracy nad projektem nikt z nas nie znał możliwości Sql Server Analysis Services (SSAS) ani nie tworzył strony internetowej w .NET. O ile z .NET nie było znaczących problemów, usługi analityczne Sql Servera stanowiły dla nas barierę nie do przejścia. Ze względu na dość kiepsko udomkumentowaną technologię SSAS, brak obiecanych szkoleń z tej technologii oraz brak zajęć na uniwersytecie jej poświęconym a także małej elastyczności algorytmów tej usługi podjęliśmy decyzję o rezygnacji z tego narzędzia. Zdecydowaliśmy się na wykorzystanie języka skryptowego R, z którym dobrze zaznajomione były 2 osoby w zespole. Zmieniliśmy także realizację frontendu z .NET na Django ponieważ główny powód wyboru .NET - pluginy dedykowane dla SSAS, przestał być ważny dla projektu. Z Django obeznane były wszystkie osoby w zespole, a dodatkowym atutem tego wyboru był jeden członek zespołu posiadający doświadczenie zawodowe w pisaniu aplikacji w Pythonie.

Kolejnym znaczącym problemem przy realizacji projektu była mocno utrudniona praca związana z analizą danych i dobieraniem optymalnych algorytmów. Zamawiający obiecał nam dostarczenie zaszumionych danych z systemu USOS. Czekaliśmy na nie, ponieważ trudno było samemu wymyślić i wygenerować nietrywialne korelacje, zbliżone do rzeczywistości.

Próbka takich danych mocno pomogłaby nam z doбором algorytmów predykcyjnych. Jednak zaszumienie okazało się w praktyce niemożliwe a "zwykłych" danych nie mogliśmy otrzymać z powodu ustawy o ochronie danych osobowych. O tych ograniczeniach praktycznie dowiedzieliśmy się dopiero w kwietniu, co mocno nam popsuło pracę nad algorytmami predykcyjnymi i analizą ich jakości która niestety nie została wykonana na danych produkcyjnych.

6.2. Organizacja Pracy

W naszej pracy korzystaliśmy z repozytorium git. Było ono umieszczone na stronie <http://www.github.com>. W ramach tego repozytorium mieliśmy następujące odnogi:

- **ZPP-frontend** - pierwsza wersja frontendu, zrealizowana w technologii C# .NET, a w kolejnej iteracji za pomocą pluginu do django w Visual Studio 2013
- **ZPP-Frontend2.0** - druga odnoga reprezentująca frontend realizowany w django na systemie linux. Jest to odnoga w której został stworzony końcowy efekt naszej pracy.
- **ZPP-SSDT** - nazwa tej odnogi bierze się od Sql Server Data Tools - na początku trzymaliśmy w tym repozytorium projekty Sql Server Analysis Services zajmujące się analizą danych. W kolejnej iteracji w tym repozytorium umieszczane były skrypty obliczające rekomendacje w języku R.
- **ZPP-slides** - w tej odnodze tworzona i przechowywana była prezentacja końcowa.
- **ZPP-pracalic** - odnoga w której powstawała praca licencjacka.

Przydział zadań do osób i planowanie pracy zostały zrealizowane za pomocą portalu Redmine.

Praca na początku projektu przebiegała dość chaotycznie, z długimi przerwami. Dużą blokadą były dla nas problem z efektywną pracą nad predykcjami z SSAS oraz brak danych bliższych rzeczywistości utrudniający nam wyjście poza teoretyczne rozważanie o algorytmach. Jednak pewne prace były w związku z tym systemem wykonywane, zbudowaliśmy pewną strukturę danych, wygenerowaliśmy dane testowe i próbowaliśmy generować predykcję. Z wszystkich tych rzeczy (nie licząc części danych testowych) zrezygnowaliśmy w dalszej fazie realizacji projektu.

Realną pracę wychodzącą poza studiowanie dokumentacji i metodę prób i błędów prowadzącą do nikąd zaczęliśmy dopiero po decyzji o zmianie technologii w lutym. Praca odbywała się wtedy w cotygodniowych iteracjach, a znacząco przyśpieszyła w kwietniu po ostatecznej decyzji o braku możliwości otrzymania jakichkolwiek danych przypominających rzeczywiste. Niestety ze względu na niezbyt długi pozostały wówczas czas nie udało nam się zrealizować wszystkich funkcjonalności, w szczególności integracji z USOS (przez brak czasu ale także utrudnioną współpracę z zamawiającym).

6.3. Podział Obowiązków

W projekcie Hermes podział pracy był następujący:

- **Tomasz Grabowski :**

- Pisanie licencjatu : jego redakcja, formatowanie, wstawienie zrzutów ekranu. Napisanie wszystkich rozdziałów poza rozdziałem 5 (autorstwa Krzysztofa Rutkowskiego) oraz tym podrozdziałem (każdy sam opisywał swój wkład).
- Pomoc przy tworzeniu frontendu, w szczególności: edycja szablonu wyświetlającego wyniki predykcji oraz rekomendacji, obliczanie i wyświetlanie statystyk każdego przemiotu (po użyciu w tym szablonie opcji info), realizacja tego elementu w Django (kod w języku Python).
- Generowanie danych testowych i przypadków testowych .
- Funkcjonalne testowanie systemu.
- We wstępnej wersji projektu (gdzie jeszcze korzystaliśmy z Sql Server Analysis Services) tworzenie przypadków oraz danych testowych a także projektowanie kostki OLAP i sposobu wykorzystania modułu Data Mining w celu obliczenia rekomendacji.

- **Adam Markiewicz:**

- Tworzenie frontendu - modele ORM, szablony, obsługa logowania, rejestracji i edycji ocen użytkowników, zintegrowanie bazy ocen użytkowników z resztą frontendu (kod w Pythonie i Javascriptcie), debugowanie
- Administrowanie serwerem WWW i bazą danych, wdrożenie systemu.
- Testy funkcjonalne systemu.
- We wstępnej wersji projektu korzystającej z SSAS - administracja serwerem WWW, hurtownią danych i kontrolerem domeny; projektowanie kostki OLAP i integracja jej z warstwą Data Mining w SSAS.

- **Albert Rozmus:**

- Praca przy frontendzie - strona wizualna i funkcjonalna
- Pomoc przy tworzeniu prezentacji
- Testy funkcjonalne systemu
- Wykorzystanie systemu Redmine do organizacji pracy

- **Krzysztof Rutkowski :**

- organizacja oraz zarządzanie pracą zespołu.
- znalezienie osoby która podjęła się stworzenia plakatu.
- tworzenie, implementacja i testowanie algorytmów uczenia maszynowego w R.
- integracja skryptów R z frontendem.

- pomoc przy tworzeniu frontendu: formularze do predykcji ocen i przedmiotów z seminarium, stworzenie modulu dla skryptów R, poprawianie jego błędów

- **Wiktor Zuba :**

- Obsługa bazy danych.
- Stworzenie modelu danych.
- Opracowanie rozkładu danych imitującego korelacje obecne w systemie USOS.
- Stworzenie skryptu generującego losowe dane z wybranego rozkładu.
- Konwersja danych do różnych formatów.
- Zainicjalizowanie pracy licencjackiej i stworzenie stron tytułowych.

Plakat został zrealizowany przez osobę wynajętą przez nas w tym celu.

Rozdział 7

Podsumowanie

Projekt napotkał na duże przeszkody w trakcie realizacji. Problemy z zamawiającym, z otrzymaniem sensownych danych testowych oraz z nauką Sql Server Analysis Services a także skutecznym wykorzystaniem jego algorytmów z dość znaczącymi limitacjami mocno wpłynęły na czas spędzony na efektywnej pracy nad projektem. Z ubolewaniem przyznajemy, iż realną pracę przynoszącą progres a nie błędzenie w ślepych zaułkach, książkach i dokumentacjach zaczęliśmy dopiero w marcu.

W celu realizacji projektu i usprawnienia pracy musieliśmy podjąć decyzję o zmianie technologii. Umożliwiło nam to mieć większy wpływ na tworzone algorytmy predykcyjne oraz sposób ich działania. Dużym plusem tej zmiany była możliwość wykorzystania solidnej znajomości teorii systemów decyzyjnych oraz jej zastosowania w R przez członków zespołu. Zmiana na Django także była skutkiem lepszej znajomości Pythona przez zespół. Uważamy, iż zmiany okazały się w końcowym rozrachunku pozytywne, ponieważ udało nam się zaimplementować system który zawiera większość planowanych na początku funkcjonalności. Dodatkowo, dzięki zastosowanym technologiom, w miarę łatwe jest rozwijanie systemu. W celu zaimplementowania nowych typów predykcji wystarczy napisać nowy skrypt w R i załączyć go do systemu oraz zintegrować z frontendem.

Problemy przy pracy nad projektem niestety uniemożliwiły spełnienie naszych wszystkich założeń. Zabrakło nam czasu na zaimplementowanie funkcjonalności dla pracowników predykujących popularność wybranych przedmiotów wśród studentów. Z przyczyn niezależnych od nas nie udało nam się również przetestować napisanych przez nas algorytmów na prawdziwych danych, przez co nie mieliśmy możliwości praktycznej weryfikacji predykcji.

Mamy jednak nadzieję, iż pomimo tych trudności system sprawdzi się na prawdziwych danych i okaże się realną, efektywną pomocą dla studentów.

Bibliografia

- [AZR] Microsoft *dokumentacja techniczna Azure*
<http://azure.microsoft.com/en-us/documentation/>
- [PTN] Python Software Foundation *dokumentacja techniczna języka Python*
<https://docs.python.org/2.7/>
- [DGO] Django Software Foundation *dokumentacja techniczna Django*
<https://docs.djangoproject.com/en/1.8/>
- [R] R Development Core Team *dokumentacja techniczna języka R*
<http://cran.r-project.org/manuals.html>
- [RSV] Simon Urbanek *dokumentacja techniczna serwera R - modułu RServe*
<http://www.rforge.net/Rserve/doc.html>