**Chmura Obliczeniowa i Technologie Big Data**

*Moduł „PyPork” realizujący mechanizmy logiki rozmytej dla platformy Apache Pig*

|  |  |
| --- | --- |
| Kierownik | dr hab. inż. Dariusz Mrozek, Prof. PŚ |
| Autorzy | Jan Skowronek  Zuzanna Lempa  Paweł Kudzia  Wojciech Wątroba |
| Kierunek | Informatyka SSM |
| Specjalizacja | IDSI |
| Data oddania | DD.MM.YYYY |
| Technologia | Apache Pig |

Spis treści

[1. Wstęp 3](#_Toc68288369)

[2.1. Cel projektu 3](#_Toc68288370)

[2.2. Założenia projektu 3](#_Toc68288371)

[2. Analiza tematu 4](#_Toc68288372)

[2.1. Platforma Hadoop 4](#_Toc68288373)

[2.2. Platforma Apache Pig 5](#_Toc68288374)

[2.3. Logika rozmyta 6](#_Toc68288375)

[3. Specyfikacja zewnętrzna 8](#_Toc68288376)

[3.1. Przygotowanie środowiska 8](#_Toc68288377)

[3.2. Przykład użycia biblioteki „PyPork” 10](#_Toc68288378)

[4. Specyfikacja wewnętrzna 11](#_Toc68288379)

[5. Podsumowanie 12](#_Toc68288380)

# 1. Wstęp

Big data to termin, który odnosi się do dużych i różnorodnych zbiorów danych. Przetwarzanie tego typu danych jest zadaniem wyjątkowo trudnym i wymagającym, ale z drugiej strony analiza ogromnych zbiorów okazuje się być wartościowa, ponieważ pozwala zdobyć nowy rodzaj wiedzy. Z tego też powodu na przestrzeni lat powstały narzędzia i rozbudowane platformy programistyczne umożliwiające m.in. na efektywne gromadzenie, przechowywanie i przetwarzanie dużych ilości informacji. Warto zaznaczyć, że dane mogą pochodzić z wielu różnych źródeł, co niesie za sobą konieczność użycia specjalistycznego oprogramowania, za pomocą którego można usprawnić proces przetwarzania. Korzyści z używania narzędzi Big Data są niezaprzeczalne, dlatego bardzo ważne jest, aby istniała możliwość rozszerzania o nowe funkcjonalności i tak już mocno rozbudowanych platform.

W ramach kursu „Chmura Obliczeniowa i Technologie Big Data” należy zrealizować własny moduł rozszerzający dla jednej z przykładowych platform programistycznych stosowanych w dziedzinie Big Data.

## 2.1. Cel projektu

Celem projektu jest rozszerzenie przykładowej platformy programistycznej o moduł, który umożliwi realizację rozmytego przetwarzania danych. Moduł należy opracować dla platformy Apache Pig będącej częścią ekosystemu Hadoop.

## 2.2. Założenia projektu

Nowy moduł dla Apache Pig powinien dostarczać podstawowe funkcjonalności rozmytego przetwarzania danych.

Główne założenia, które powinny zostać spełnione to:

1. Rozmyte filtrowanie w klauzuli WHERE,
2. Projekcja z rozmytą transformacją,
3. Tworzenie zmiennych lingwistycznych,
4. Grupowanie rozmyte względem zmiennej lingwistycznej,
5. Rozmyte złączenia poprzez:
   1. Zmienną lingwistyczną,
   2. Rozmywanie liczb.

Bibliotekę typu „fuzzy” należy zaimplementować wykorzystując do tego UDF, czyli funkcje rozszerzające, które definiuje użytkownik chcący wzbogacić istniejące rozwiązania.

Funkcje mogą zostać zaimplementowane w wybranych językach programowania, wspieranych przez aplikację Apache Pig. Jednym z nich jest język Python i to właśnie on posłużył do wykonania projektu.

# 2. Analiza tematu

W niniejszym rozdziale przedstawiono opis najważniejszych zagadnień teoretycznych związanych z projektem „PyPork”. Czytelnikowi przybliżone zostaną podstawowe informacje na temat dwóch bardzo popularnych platform Big Data, którymi są Hadoop i Apache Pig. Dodatkowo zaprezentowane zostaną kluczowe pojęcia z logiki rozmytej, których znajomość i dokładne zrozumienie okazują się być całkiem przydatne w wielu zastosowaniach, a po drugie są niezbędne do prawidłowej realizacji opisywanego projektu.

## 2.1. Platforma Hadoop

Apache Hadoop jest otwartą platformą programistyczną napisaną w języku Java, która przeznaczona jest do rozproszonego składowania i przetwarzania wielkich zbiorów danych. Z reguły tego typu przetwarzanie odbywa się przy pomocy klastrów komputerowych, czyli większej liczby współpracujących ze sobą komputerów. Zwiększenie liczby dostępnych węzłów obliczeniowych powoduje wzrost szybkości analizowania i przetwarzania danych, co jak się okazuje ma znaczenie w przypadku, gdy tych danych jest bardzo dużo i dodatkowo pochodzą z różnych źródeł. Istotną kwestią jest również to, iż platforma Hadoop wchodzi w skład większej liczby projektów rozwijanych przez fundację Apache.

Platforma Hadoop składa się z następujących modułów:

* Hadoop Common – typowe narzędzia, które używane są przez pozostałe moduły,
* Hadoop Distributed File System (HDFS) – rozproszony system plików, zapewnia dostęp do danych wykorzystywanych przez aplikację,
* Hadoop YARN – pozwala na planowanie zadań i zarządzanie zasobami klastra,
* Hadoop MapReduce – system do równoległego przetwarzania dużych ilości danych, implementacja paradygmatu MapReduce,
* Hadoop Ozone – magazyn obiektowy.

Warto zaznaczyć, iż samo określenie Hadoop utożsamiane jest z pewnym ekosystemem w skład którego wchodzi wiele dodatkowych aplikacji. Można do nich zaliczyć poniższe m.in.:

* Apache Pig,
* Apache Spark,
* Apache Storm,
* Apache Hive,
* Apache HBase,
* Apache Phoenix,
* Apache ZooKeeper,
* Apache Flume,
* Apache Sqoop,
* Apache Oozie,
* Cloudera Impala.

Jak wspomniano wcześniej do wykonania modułu typu „fuzzy” wykorzystany został Apache Pig, który jak się okazuje w stosunkowo prosty sposób pozwala zwiększyć i tak już dużą liczbę dostępnych funkcjonalności platformy Hadoop.

## 2.2. Platforma Apache Pig

Apache Pig to platforma do analizy dużych zbiorów danych, która udostępnia język wysokiego poziomu przeznaczony do tworzenia programów operujących na danych, które mogą pochodzić z różnych źródeł. Jest to bardzo ważną cechą, ponieważ dzięki Apache Pig można jednocześnie przetwarzać dane, które pochodzą z relacyjnych bazy danych, plików tekstowych, czy plików CSV. Często bywa tak, że dane nie mają ściśle zdefiniowanej struktury, co zdarza się bardzo często. Szeroki obszar zainteresowań Big Data dotyczy zarówno informacji w pełni ustrukturyzowanych, jak i nieustrukturyzowanych. Pierwszy rodzaj danych pochodzi przede wszystkim z systemów zarządzania relacyjnymi bazami danych, zaś dane nieustrukturyzowane to wszelkiego rodzaju wiadomości e-mail, wpisy ze stron internetowych (np. artykuły z blogów), aktywności w mediach społecznościowych. Przy okazji warto wspomnieć o istnieniu częściowo ustrukturyzowanych danych, do których zaliczają się wspomniane wcześniej pliki tekstowe oraz np. dzienniki systemowe.

Istotną zaletą programów napisanych w oparciu o platformę Apache Pig jest to, że wykonywane zadanie można w prosty sposób zrównoleglić, ponieważ całość odbywa się dzięki zastosowaniu systemu Hadoop. Podstawowym elementem infrastruktury Apache Pig jest kompilator, którego zadanie sprowadza się do tworzenie sekwencji programów Map-Reduce. Platforma Apache Pig dostarcza proceduralny język programowania Pig Latin, do którego zalet można zaliczyć m.in.:

* Łatwość tworzenia kodu, ponieważ Pig Latin został opracowany w taki sposób, aby mógł być stosowany przez osoby, które na co dzień nie pracują z językiem Java lub posiadają niewielkie doświadczenie programistyczne,
* Duże możliwości optymalizacji, dzięki czemu zadania mogą być automatycznie optymalizowane przez moduły niższego poziomu, co pozwala użytkownikowi w pełni skupić się na określeniu oczekiwanego rozwiązanie niż pisaniu wydajnego kodu,
* Możliwość tworzenia własnych rozwiązań i modułów poprzez zdefiniowanie funkcji specjalnego przeznaczenia.

Funkcje rozszerzające (UDF) pozwalają wzbogacić standardowy zestaw funkcjonalności języka Pig Latin. W tym celu Apache Pig wspiera aż sześć języków programowania:

* Java,
* Jython,
* Python,
* JavaScript,
* Ruby,
* Groovy.

W chwili obecnej największe wsparcie dostępne jest tylko i wyłącznie dla języka Java. Dzięki własnym funkcjom użytkownicy mogą rozszerzyć język Pig Latin i dzięki temu dostosować wszystkie niezbędne etapy potoku przetwarzania danych, do których można zaliczyć ładowanie i przechowywanie danych, transformację oraz agregację.

## 2.3. Logika rozmyta

Logika rozmyta stanowi rozszerzenie tradycyjnej logiki binarnej i pozwala na opisywanie zjawisk fizycznych w bardziej naturalny sposób dla człowieka. Zazwyczaj w życiu codziennym opis różnych zjawisk nie opiera się na ścisłym definiowaniu parametrów. Oczywiście dokładna wiedza na temat np. wzrostu pacjenta, czy aktualnej temperatury jest ważna, jednak w życiu codziennym bardzo często używa rezygnuje się z tej precyzji na rzecz określeń zdecydowanie mniej precyzyjnych jak np. "ten mężczyzna jest wysoki" lub "dzisiaj jest bardzo ciepło".

Logika rozmyta okazała się przydatna w różnych zastosowaniach inżynierskich, przede wszystkim w takich, gdzie klasyczna logika, która dokonuje klasyfikacji na podstawie dwóch wartości logicznych (prawda i fałsz) nie potrafi skutecznie poradzić sobie z wieloma sprzecznościami i niejednoznacznymi stwierdzeniami. Przykłady użycia to np. zastosowanie elementów teorii zbiorów rozmytych w elektronicznych systemach sterowania, zadaniach związanych z eksploracją danych (data mining) czy w budowie systemów ekspertowych, które na podstawie pewnej bazy wiedzy, modeli oraz reguł wnioskowania mogą rozwiązywać różne problemy.

Zbiór rozmyty jest uogólnieniem pojęcia zbioru ostrego, który polega na dopuszczenie, aby funkcja przynależności zbioru mogła przyjmować wartości z zakresu . W sposób bardziej formalny można napisać, że zbiór rozmyty to zbiór par, w pewnej numerycznej przestrzeni rozważań

dla każdego

gdzie jest funkcją przynależności zbioru rozmytego . Każdemu elementowi zbioru przypisywany jest stopień przynależności do zbioru , przy czym istotne jest, aby wartości funkcji przynależności mieściły się w dopuszczalnym przedziale .

Kolejnym równie istotnym zagadnieniem będącym częścią logiki rozmytej jest pojęcie zmiennej lingwistycznej. Można napisać, że zmienna lingwistyczna to po prostu zmienna, której wartościami są pewne słowa lub zdania wyrażone w języku naturalnym. Przykładowe zmienne lingwistyczne zostały zaprezentowane w tabeli 2.3.1.

Tabela 2.3.1. Przykłady zmiennych lingwistycznych.

|  |  |
| --- | --- |
| **Kategoria** | **Wartość** |
| Prędkość | mała, średnia, duża |
| Wzrost | niski, średni, wysoki |
| Widoczność | bardzo zła, zła, średnia, dobra, bardzo dobra |
| Ciśnienie | niskie, wysokie |

Spełnienie przesłanki w logice rozmytej może być reprezentowane przez wspomniane wcześniej pojęcie funkcji przynależności . Podczas projektowania systemów, w których zastosowane mają zostać elementy logiki rozmytej, bardzo ważną kwestią jest dobór rodzaju funkcji przynależności, która jednoznacznie będzie określać poszczególne zbiory rozmyte zmiennej lingwistycznej.

Do najpopularniejszych funkcji przynależności można zaliczyć funkcję trójkątną oraz funkcję trapezową, aczkolwiek wyróżnia się jeszcze inne funkcje jak np. funkcję klasy , czy funkcję klasy .

Trójkątna funkcja przynależności jest opisywana przy pomocy trzech parametrów: .

Trapezowa funkcja przynależności jest opisywana za pomocą czterech parametrów: .

Lewa zewnętrzna funkcja przynależności klasy jest opisywana poprzez użycie dwóch parametrów: .

Prawa zewnętrzna funkcja przynależności klasy jest opisywana z wykorzystaniem dwóch parametrów: .

Dodatkowo istnieje również Gaussowska funkcja przynależności, która definiowana jest przy pomocy dwóch parametrów: .

gdzie: jest środkiem, a to wariancja.

Autorzy modułu „PyPork” zdecydowali się zaimplementować ww. funkcje przynależności, jednak należy mieć na uwadze, że funkcji przynależności jest zdecydowanie więcej.

Rozwiązanie (w postaci kodu) przedstawiono w rozdziale 4 niniejszego raportu.

# 3. Specyfikacja zewnętrzna

Rozdział 3. poświęcony jest sprawom technicznym dotyczącym zastosowanych narzędzi. To właśnie tutaj opisano jak należy przygotować lokalne środowisko testowe w celu wykonania własnego modułu poszerzającego możliwości Apache Pig.

## 3.1. Przygotowanie środowiska

W celu pracy z platformą Apache Pig należy przygotować odpowiednie środowisko, które zazwyczaj przyjmuje postać jakiegoś systemu operacyjnego z rodziny Linux. Z reguły wiele narzędzi programistycznych jest tworzonych właśnie pod ten system. Nie inaczej jest w przypadku Hadoop i Apache Pig, w związku z czym pierwszym krokiem do rozpoczęcia pracy było skonfigurowanie odpowiedniej maszyny wirtualnej. Autorzy projektu zdecydowali, że w tym celu wykorzystają jedną z popularnych dystrybucji Linuksa, którą niewątpliwie jest Ubuntu. Instalacja systemu jest prosta i sprowadza się do postępowania według wskazówek wyświetlanych na ekranie. W przypadku, gdy instalowana jest wersja desktop systemu Ubuntu, warto wybrać opcję „minimal”, dzięki której zainstalowane zostaną tylko niezbędne pakiety i aplikacje.

Po prawidłowym zainstalowaniu systemu na maszynie wirtualnej można przejść do kolejnego kroku, który dotyczy instalacji Hadoop i Apache Pig. Aby zminimalizować mogące się pojawić problemy i przy okazji uprościć ręczną konfigurację, utworzony został skrypt powłoki Bash, który realizuje następujące zadania:

* Instalacja środowiska uruchomieniowego Java,
* Pobranie i rozpakowanie paczek Hadoop i Apache Pig,
* Przeniesienie plików Hadoop i Apache Pig do odpowiednich katalogów,
* Utworzenie zmiennych środowiskowych,
* Sprawdzenie poprawności wykonanej instalacji za pomocą polecenia wypisującego informacje o zainstalowanej wersji danego oprogramowania.

Na listingu 3.1.1. przedstawiony został przykładowy skrypt instalacyjny, który należy uruchomić z uprawnieniami administratora systemu.

#!/bin/bash

sudo apt install openjdk-8-jdk

sudo apt install openjdk-8-jre

java -version

# main directory where Hadoop and Pig will be installed

MAIN\_DIRECTORY=/usr/local

cd $MAIN\_DIRECTORY

# download hadoop

sudo wget http://apache.claz.org/hadoop/common/hadoop-2.10.1/hadoop-2.10.1.tar.gz

sudo tar xzf hadoop-2.10.1.tar.gz

sudo mkdir hadoop

sudo mv hadoop-2.10.1/\* hadoop

sudo rm -r hadoop-2.10.1

# download pig

sudo wget http://apache.claz.org/pig/pig-0.17.0/pig-0.17.0.tar.gz

sudo tar xzf pig-0.17.0.tar.gz

sudo mkdir pig

sudo mv pig-0.17.0/\* pig

sudo rm -r pig-0.17.0

# home directories for Hadoop and Pig

MY\_HADOOP\_HOME=$MAIN\_DIRECTORY/hadoop

MY\_PIG\_HOME=$MAIN\_DIRECTORY/pig

# create environmental variables

echo export HADOOP\_HOME=$MY\_HADOOP\_HOME >> ~/.bashrc

echo export HADOOP\_INSTALL=$MY\_HADOOP\_HOME >> ~/.bashrc

echo export HADOOP\_MAPRED\_HOME=$MY\_HADOOP\_HOME >> ~/.bashrc

echo export HADOOP\_COMMON\_HOME=$MY\_HADOOP\_HOME >> ~/.bashrc

echo export HADOOP\_HDFS\_HOME=$MY\_HADOOP\_HOME >> ~/.bashrc

echo export HADOOP\_COMMON\_LIB\_NATIVE\_DIR=$MY\_HADOOP\_HOME/lib/native >> ~/.bashrc

echo export HADOOP\_OPTS="-Djava.library.path=$MY\_HADOOP\_HOME/lib/nativ" >> ~/.bashrc

echo export YARN\_HOME=$MY\_HADOOP\_HOME >> ~/.bashrc

# if you installed java 7 change name here

echo export JAVA\_HOME=/usr/lib/jvm/java-1.8.0-openjdk-amd64 >> ~/.bashrc

# add Hadoop and Pig to PATH

HADOOP\_PATH=$MY\_HADOOP\_HOME/sbin:$MY\_HADOOP\_HOME/bin

PIG\_PATH=$MY\_PIG\_HOME/bin

echo export PATH=$PATH:$HADOOP\_PATH:$PIG\_PATH >> ~/.bashrc

source ~/.bashrc

hadoop version

pig -version

echo "Script completed."

Listing 3.1.1. Skrypt instalacyjny.

Następnie można zweryfikować poprawności instalacji Apache Pig. Najprostszy sposób polega na uruchomieniu aplikacji w trybie lokalnym korzystając z komendy przedstawionej na listingu 3.1.2.

pig -x local

Listing 3.1.2. Uruchomienie Apache Pig w trybie lokalnym.

Jeśli wszystko zostało prawidłowo skonfigurowane, aplikacja powinna uruchomić się w trybie interaktywnym, w którym możliwe jest wpisywanie poleceń języka Pig Latin. Aby zapoznać się z podstawowymi funkcjami można np. wczytać plik CSV i wyświetlić jego zawartość.

## 3.2. Przykład użycia biblioteki „PyPork”

W celu sprawdzenia poprawności opisywanej biblioteki można skorzystać z dostarczonego pliku skryptowego w języku Pig Latin, który znajduje się w katalogu pig\_examples. W przykładzie pokazane zostało użycie operacji rozmytych na danych pochodzących z pliku CSV.

Należy pamiętać, aby na początku własnych skryptów zarejestrować główny plik biblioteki, który należy umieścić w łatwo dostępnych miejscu jak np. główny katalog użytkownika. Rejestracja skryptu odbywa się za pomocą operatora REGISTER w podobny sposób jak zostało to zaprezentowane na listingu 3.2.1.

Listing 3.2.1. Uruchomienie Apache Pig w trybie lokalnym.

REGISTER '/home/piggy/PyPork/main\_fuzzy.py' using jython as fuzzy

W języku Pig Latin do wczytywania danych wykorzystuje się operator LOAD. Przykład jego użycia można zobaczyć na listingu 3.2.2.

Listing 3.2.2. Wczytanie danych z pliku (LOAD).

users = LOAD 'user\_data' USING PigStorage(',') AS (firstname: chararray, lastname:chararray,wzrost:int, wiek:int);

Aby móc wyświetlić informacje w konsoli systemowej, w tym celu należy użyć operator DUMP. Przykładowe wywołanie zostało przedstawione na listingu 3.2.3.

Listing 3.2.3. Wyświetlenie informacji w konsoli systemowej (DUMP).

DUMP fuzzy\_users;

# 4. Specyfikacja wewnętrzna

W niniejszym rozdziale opisana została architektura biblioteki „PyPork”.

Cały projekt składa się z kilku plików skryptowych języka Python, w ramach których zaimplementowane zostały poszczególne funkcje odpowiedzialne za realizację określonych zadań.

Katalog fuzzy przechowuje następujące pliki:

* linguistic.py – zawiera definicje funkcji, w których skład wchodzą m.in. funkcje operujące na wartościach lingwistycznych,
* membership\_functions.py – przechowuje definicje wybranych funkcji przynależności (np. trójkątna, trapezowa),
* operators.py – zawiera definicje funkcji, które odpowiadają rozmytym operatorom AND, OR i NOT.

W katalogu pig\_examples umieszczone zostały przykłady użycia opisywanego rozszerzenia PyPork. Można tutaj znaleźć:

* linguistic.pig – to właśnie ten plik zawiera kilka przykładów pokazujących działanie autorskich funkcji,
* user\_data – przechwouje przykładowe dane (format CSV),
* pattern.json – plik konfiguracyjny, który definiuje wartości rozmyte.

Ponadto główny katalog repozytorium zawiera kilka skryptów testujących, które sprawdzają poprawność działania poszczególnych funkcjonalności:

* linguistic\_test.py – zawiera testy sprawdzające obsługę wartości rozmytych,
* membership\_test.py – przechowuje przykładowe testy, które weryfikują poprawność zaimplementowanych funkcji przynależności,
* operatorsTest.py – sprawdza działania rozmytych operatorów AND, OR i NOT.

Poszczególne skrypty można oddzielnie uruchomić z poziomu konsoli systemowej wywołując standardowe polecenie pokazane na listingu 4.1.

Listing 4.1. Uruchamianie skryptów Python z poziomu konsoli systemowej.

python <nazwa pliku>.py

W celu ułatwienia pracy z biblioteką utworzono plik main\_fuzzy.py, w którym znalazły się wszystkie niezbędne funkcje. To właśnie ten plik jest niezbędny do pracy z platformą Apache Pig, co zresztą zostało zaprezentowane w poprzednim rozdziale.

# 5. Podsumowanie

Wykonanie projektu pt. „PyPork” pozwoliło autorom zapoznać się z platformą Apache Pig, która znajduje zastosowanie w analizie i przetwarzaniu dużych zbiorów danych. Platforma ta udostępnia proceduralny język Pig Latin, dzięki któremu możliwe jest pisanie zarówno prostych, jak i bardziej zaawansowanych skryptów umożliwiających przetwarzanie danych. Pig Latin oprócz udostępniania typowych mechanizmów charakterystycznych dla języków programowania tj. różne typy danych (tuple, bag, map), czy funkcje (np. LOAD, STORE), daje również możliwość definiowania własnych funkcji, co pozwala wprowadzić dodatkowe operacje w potoku przetwarzania. Opcja rozbudowy istniejących rozwiązań niewątpliwie jest jedną z kluczowych zalet platform Hadoop i Apache Pig, ponieważ dzięki temu standardowy zestaw narzędzi (który swoją drogą i tak jest obszerny) może być poszerzony. W tym przypadku Apache Pig wzbogacono o wybrane zagadnienia z logiki rozmytej, która przydaje się również w niezmiernie ciekawej dziedzinie informatyki jaką jest Big Data.

Autorzy projektu są zadowoleni z uzyskanych wyników i osiągniętego celu, choć nie da się ukryć, że aktualne rozwiązanie mogłoby zostać bardziej usprawnione. Jeśli dodatkowym celem byłoby udostępnienie biblioteki „PyPork” publicznie, wtedy koniecznym krokiem byłoby usprawnienie repozytorium w serwisie GitHub. Przydałaby się rozbudowana i jednocześnie czytelna dokumentacja biblioteki. Tego typu zmiana, choć stosunkowo prosta z pewnością podniosłaby ogólną jakość projektu. Być może popularyzacja przyczyniłaby się do powstania społeczności, której członkowie rozwijaliby istniejące rozwiązanie.

Podsumowując, realizacja projektu „PyPork” poruszającego dosyć niestandardową tematykę w ramach kursu „Chmura Obliczeniowa i Technologie Big Data”, zdecydowanie poszerzyła wiedzę dotyczącą technik pracy z językiem Python, zbiorów rozmytych i co najważniejsze specjalistycznych narzędzi służących do analizy, gromadzenia oraz przetwarzania dużych zbiorów danych. Nie da się ukryć, że wiedza o istnieniu takich narzędzi jak Hadoop, czy Apache Pig może być niezwykle przydatna podczas projektowania i realizacji komercyjnych systemów informatycznych, które sprowadzają się do przekazywania informacji.