

mgr inż. WITOLD GŁOWACZ
AGH Akademia Górniczo-Hutnicza
Wydział Elektrotechniki, Automatyki, Informatyki i Elektroniki
Katedra Automatyki
al. A. Mickiewicza 30
30-059 Kraków

Implementacja automatycznego systemu dialogowego oparta o język Python i bazę danych MySQL

Przedstawiono implementację automatycznego systemu dialogowego opartego o język Python i bazę danych MySQL. Automatyczny system dialogowy pozwala nawiązać kontakt pomiędzy użytkownikiem i komputerem. System dialogowy analizuje zdania użytkownika i generuje swoje zdania w języku naturalnym. Język naturalny jest zdefiniowany za pomocą metod formalnych, w tym gramatyk i automatów. System może być użyty do sterowania silnikami elektrycznych w górnictwie i zwiększenia poziomu bezpieczeństwa.

1. WPROWADZENIE

Nowoczesne systemy bezpieczeństwa transportu mają własności systemów ekspertowych. W systemach ekspertowych wyróżnić można cztery elementy: bazę wiedzy, maszynę wnioskującą, interfejs użytkownika i bazę objaśnień. Baza wiedzy są to reguły opisujące relacje między faktami, opisujące one jak system ma się zachować w danym momencie działania. Maszyna wnioskująca dopasowuje fakty do przesłanek i uaktywnia reguły. Interfejs użytkownika pozwala użytkownikowi kontaktować się z maszyną wnioskującą i bazą wiedzy. Baza objaśnień zawiera znaczenia symboli używane przez użytkownika i system ekspertowy, umożliwiające porozumienie się. Systemy ekspertowe przechowują zdobytą wiedzę uzyskaną z treningu i doświadczenia. Mogą być implementowane do wzbogacenia wnioskowania bota. Bot jest softwarowym robotem, który prowadzi rozmowę z użytkownikiem w oparciu o swoją bazę wiedzy, do której włączono bazy wiedzy systemów ekspertowych [8]. Rozmowa użytkownika z botem może być prowadzona w różnych językach, w szczególności w naturalnym języku polskim. Bot analizuje zdania użytkownika i formułuje własne zdania używając reguł wybranego języka i bazy wiedzy.

2. JĘZYK PYTHON

Język Python został napisany przez Guido van Rossuma w 1990 roku [10]. W literaturze można znaleźć różnorodne porównania Pythona z Perlem, PHP oraz Javą. Python jest w równym stopniu podobny do każdego z wymienionych, jak i różny od nich. Podobieństwa i różnice łatwo określić, zapoznając się z jego kilkoma podstawowymi cechami:

- Python jest językiem w pełni obiektywnym. Odróżnia go to zarówno od Perla, jak i od PHP (pomimo faktu, że PHP z wersji na wersję coraz bardziej wspiera obiektywność – wciąż jednak czyni to słabo).
- Pomimo tego, że jest językiem obiektywnym, Python nie wymusza obiektywnego stylu programowania. Pozwala programiście tworzyć również programy strukturalne i funkcyjne.
- Zmienne nie mają typów, wartości natomiast tak. Ściślej rzecz ujmując, typy zmiennych w Pythonie są w pełni dynamiczne. Oznacza to podobną do PHP łatwość zmiany zastosowania zmiennych, choć jednocześnie takie rozwiązanie potrafi dawać inne efekty niż w Javie czy C++, gdzie przypisywane wartości muszą być zgodne z typem zmiennej i kompilator ściśle tego pilnuje.

- Python jest językiem interpretowanym – podobnie jak PHP czy Perl, a w przeciwieństwie do C/C++ czy Javy. To znacznie ułatwia tworzenie i modyfikowanie programów, jednak nieco negatywnie odbija się na wydajności. Sytuację nieco poprawia wbudowany mechanizm konwersji kodów źródłowych do zrozumiałej dla interpretera postaci binarnej, co przy kolejnych uruchomieniach skryptu może dawać spore wzrosty wydajności.
- Python, podobnie jak Java, jest dostarczany wraz z bogatym zestawem bibliotek (głównie dedykowanych operacjom sieciowym). Python szeroko korzysta z dziedziczenia, dlatego łatwo jest nie tylko implementować biblioteki w programach, ale także je dziedziczyć, rozszerzać i tworzyć nowe. Dzięki temu, podobnie jak w przypadku Javy, powstaje wiele ciekawych rozszerzeń tych bibliotek, tworzonych przez niezależnych programistów.
- Środowisko Pythona jest w pełni interaktywne. Można na bieżąco wprowadzać kolejne polecenia i oglądać wyniki. Bardzo ułatwia to tworzenie i debugowanie programów pisanych w tym języku.
- Python intensywnie się rozwija, z tendencją do wprowadzania maksymalnej zwięzłości kodu. Przybliża go to do Perla, co nie podoba się części webmasterów, którzy twierdzą, że dzieje się to ze szkodą dla przejrzystości kodu.

3. BAZA DANYCH MySQL

MySQL jest najpopularniejszą na świecie bazą danych dostępną na licencji open source [11]. Dzięki swojej wydajności, szybkości działania, stabilności i rozbudowanym mechanizmom zabezpieczeń jest wykorzystywana niemal do wszystkich zadań, do jakich może być potrzebna baza danych. Stanowi zaplecze dla wielu aplikacji korporacyjnych, witryn WWW i sklepów internetowych. Bogactwo funkcji i możliwości to ogromna zaleta środowiska MySQL, jednakże zapamiętanie składni, parametrów i sposobów stosowania wszystkich poleceń i rozszerzeń jest praktycznie niemożliwe. Każdy, kto chce biegle posługiwać się tą bazą, powinien mieć pod ręką materiały opisujące stosowany w niej język SQL.

Cechy MySQL:

- Wydajny dzięki C i C++.
- API dla wielu języków programowania: C, C++, Eiffel, Java, Perl, PHP, Python, Ruby, Tcl.
- Pełna wielowątkowość, korzystająca z wątków kernela. Oznacza to, że MySQL będzie pracował na maszynie wieloprocessorowej.

- Opcjonalna obsługa transakcji.
- B-drzewa z kompresowanymi indeksami, które znacząco wpływają na czas wyszukiwania i pobierania danych (wierszy) z bazy.
- Istnieje możliwość "osadzenia" (ang. *embed*) serwera MySQL w aplikacji.
- Duża liczba typów danych w kolumnach. Liczby, ciągi znakowe, obiekty binarne (BLOB), data & czas, typy wyliczeniowe, zestawy. W MySQL daną kolumnę można dostosować do pewnej wielkości danych, które będą w niej przechowywane (np. TINYINT, a nie INT), tym samym uzyskuje się większą wydajność i mniejsze zużycie pamięci (również dyskowej). Istnieje możliwość definiowania niektórych typów danych jako narodowych (różne standardy kodowania).
- Obsługa klauzul agregujących i grupujących SQL.
- Złączenia zewnętrzne (LEFT & RIGHT).
- Komenda **SHOW** pozwalająca przeglądać informacje na temat baz, tabel i indeksów. Komenda **EXPLAIN** opisująca pracę optymalizatora zapytań.
- Bardzo prosty (z punktu widzenia administratora) system zabezpieczeń. Wszystkie hasła są szyfrowane.
- Połączenia z serwerem przez: TCP/IP, ODBC, JDBC.
- Lokalizacja (w sensie językowym) serwera. Komunikaty m.in. po polsku.

4. DEFINICJA AUTOMATYCZNEGO SYSTEMU DIALOGOWEGO

Przetwarzanie języka naturalnego (ang. *Natural Language Processing*, NLP) jest działem sztucznej inteligencji zajmującym się automatyzacją analizy, tłumaczenia i generowania informacji w języku naturalnym. W jego skład wchodzi:

- reprezentacja wiedzy zawartej w tekstach [2], [7],
- teoria gramatyk i automatów [1], [3], [4],
- teoria języków formalnych [3],
- teoria zbiorów rozmytych [5], [6], [9].

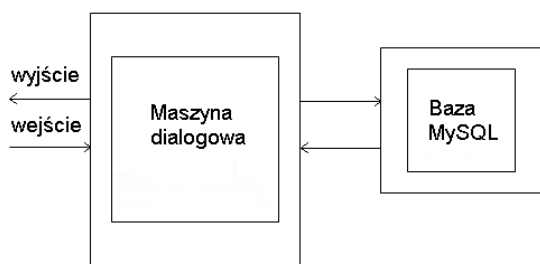
Przetwarzanie języka naturalnego można podzielić na:

- przetwarzanie informacji w postaci tekstowej lub symbolicznej,
- rozpoznawanie i generowanie mowy.

Język naturalny został wykształcony przez ludzi na drodze długotrwałej ewolucji. Pełni różnorodne funkcje. Stanowi skuteczny środek przekazywania myśli, uczuć, opinii i wiedzy. W jego następstwie później powstało pismo, które umożliwiło ich trwałe zachowywanie.

5. REALIZACJA AUTOMATYCZNEGO SYSTEMU DIALOGOWEGO

Automatyczne systemy dialogowe oparte są na botach, które są programami informatycznymi zawierającymi: bazę wiedzy i maszynę dialogową. Strukturę bota pokazano na rys. 1.



Rys. 1. Struktura bota

Bot przyjmuje do analizy zdanie użytkownika. Bada ich składnię i znaczenie. Formułuje odpowiedzi i pytania, które wysyła do użytkownika. Nowe zdania są zapisane w słowniku bota w celu poszerzenia jego bazy wiedzy.

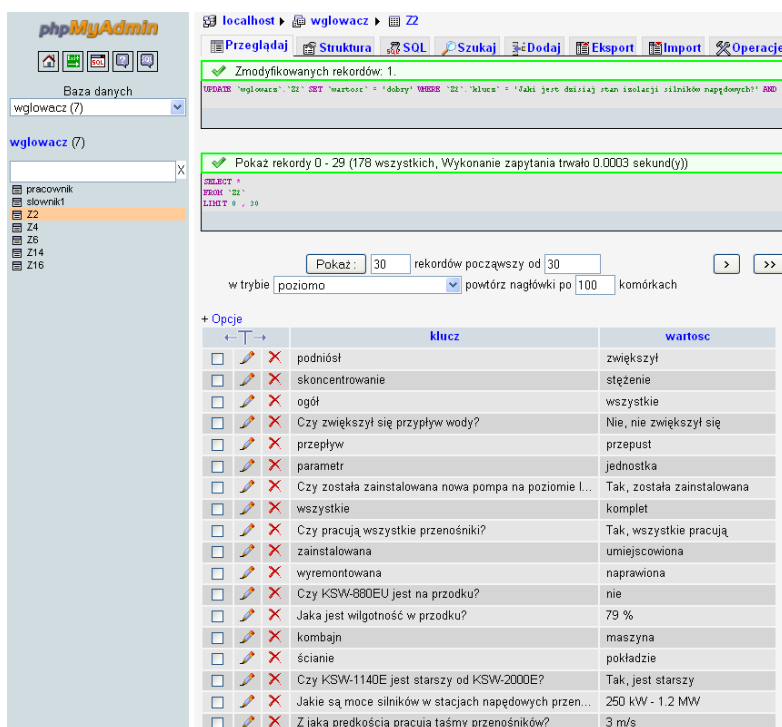
Poniżej przedstawiono fragment kodu źródłowego bota:

```
def bot (txt,licznik1,licznik2):
    if txt == "":
        return txt
    txa=txt
    import MySQLdb
    db=MySQLdb.connect(host='mysql.agh.edu.pl',use
r='...',passwd='...',db='...')
    cursor = db.cursor()
    cursor.execute ("SELECT * FROM Z2")
    result=cursor.fetchall()
    for row in result:
        a[row[0]]=row[1]
    cursor.execute ("SELECT * FROM Z4")
    result=cursor.fetchall()
    licznik1=result[0][1]
    licznik2=result[1][1]
    .....
    cursor.execute("INSERT INTO Z6(klucz,wartosc)
VALUES('%s','%s')"%(txa,txt))
    licznik2=licznik2+1
    .....
    cursor.execute("UPDATE Z4 SET war
tosc=%d'WHERE nazwa='licznik1'"%
(licznik1,))
    cursor.execute("UPDATE Z4 SET war
tosc=%d'WHERE nazwa='licznik2'"%
```

```
(licznik2,))
cursor.close()
db.close()
return txt
.....
name = sys.argv [0]
hfile = name.replace (".cgi", ".history")
title = name.replace (".cgi", "").capitalize ()
licznik1=0
licznik2=0
readline.set_history_length (8)
try:
    readline.read_history_file (hfile)
except IOError:
    pass
print "Content-type: text/html\r\n\r\n"
print "<html>"
print "<head><title>{0}</title></head>".format
(title)
print "<body>"
print "<h3>{0}</h3>".format (title)
form = cgi.FieldStorage ()
message = form.getvalue ("message", "")
readline.add_history ("User: " + message)
readline.write_history_file (hfile)
timer = datetime.now ()
message = bot (message,licznik1,licznik2)
print str (datetime.now () - timer)
readline.add_history ("Bot: " + message)
readline.write_history_file (hfile)
history = "".join (open (hfile, 'r').readlines ())
history = cgi.escape (history).replace ("\n",
"<br />\n")
print "<hr><p>{0}</p><hr>".format (history)
print "<form method='post' action='{0}'>".format
(name)
print "<p>User: <input type='text' name='message'
size='45'/></p>"
print "</form>"
print "Bot:<p
style='width:300px;height:20px;border:solid blue
1px'>{0}</p>" .format (cgi.escape (message))
print "</body>"
print "</html>"
```

6. TESTOWANIE SYSTEMU

W oparciu o pomiary system ekspertowy dokonał oceny warunków bezpieczeństwa w kopalni i zapisał wyniki w swojej bazie wiedzy oraz bazie wiedzy bota. Wynikami oceny jest zainteresowany



Rys. 2. Baza wiedzy bota w MySQL

użytkownik, który prowadzi z botem następującą rozmowę:

Chatbot

User: Jakie jest stężenie metanu?
 Bot: 1,5 %
 User: Ile osób pracuje na ścianie?
 Bot: 16 osób
 User: Jaka jest temperatura w przodku?
 Bot: 27 stopni C
 User: Jesteś pewny?
 Bot: Tak

User:
 Bot:

Rys. 3. Fragment rozmowy użytkownika z botem prowadzonej za pomocą aplikacji napisanej w języku Python

7. WNIOSKI

Metody formalne pozwalają zdefiniować języki użytkownika i bota. Użycie języka naturalnego jest trudne ze względu na rozbudowaną strukturę. Z przeprowadzonych badań wynika, że zastosowanie bazy wiedzy MySQL w rozpoznaniu poszczególnych słów użytkownika i identyfikacji zdań użytkownika jest skuteczną techniką rozpoznawania informacji. Metody przetwarzania języka naturalnego mogą być wyko-

rzystane w projektowaniu automatycznych systemów dialogowych. Systemy dialogowe ze względu na dużą szybkość działania i posiadaną wiedzę zwiększają bezpieczeństwo pracy.

Literatura

1. Aho A.V., Ullman J.D.: The Theory of Parsing, Translation and Compiling, Vol. I: Parsing, Vol. II: Compiling, Englewood Cliffs, Prentice-Hall, 1973.
2. Ajdukiewicz K.: Język i poznanie. PWN, Warszawa 2006, s. 107-116.
3. Blikle A.: Automaty i gramatyki. Wstęp do lingwistyki matematycznej, Warszawa, PWN, 1971.
4. Głowacz W.: Automatyczny system dialogowy oparty na klasyfikatorze rozmytym i gramatyce precedensyjnej w bezpieczeństwie transportu. Mechanizacja i Automatyizacja Górnictwa 2010, nr 10, Katowice, s. 24-27.
5. Głowacz A., Głowacz W.: Dc machine diagnostics based on sound recognition with application of FFT and fuzzy logic, Przegląd Elektrotechniczny (Electrical Review), R. 84, Nr 12/2008, pp. 43-46.
6. Głowacz A., Głowacz W.: Diagnostyka maszyny prądu stałego oparta na rozpoznawaniu dźwięków z zastosowaniem LPC i logiki rozmytej. Przegląd Elektrotechniczny (Electrical Review), R. 85, Nr 6/2009, pp. 112-115.
7. Lubaszewski W.: Słowniki komputerowe i automatyczna ekstrakcja informacji z tekstu, Uczelniane Wydawnictwa Naukowo-Dydaktyczne AGH, 2009, s. 257-260.
8. Santangelo A., Augello A., Gentile A., Pilato G., Caglio S.: A Chat-bot based Multimodal Virtual Guide for Cultural Heritage Tours. Proc. of PSC, Las Vegas, 2006, pp. 114-120.
9. Zadeh L.A.: Fuzzy Sets, Information and Control, 8, 1965, pp. 338-353.
10. http://internetmaker.pl/artukul/3423,1,wprowadzenie_do_pythona_-_jezyk_oprogramowania_inny_niz_wszystkie.
11. http://pl.docs.pld-linux.org/uslugi_bazydanych__mysql.html.

Recenzent: dr inż. Antoni Zdrojewski