Rozpoznawanie mowy dla potrzeb VR

Adam Wojciechowski

Rozpoznawanie mowy - definicja

Rozpoznawanie mowy

to zdolność programu do rozpoznania (ale nie rozumienia) słów lub zdań wygłaszanych przez użytkownika.

Historia

1958 - Cyfrowy system konwersacyjny Davida, Mathewsa i McDonalda

1962 - Pierwszy komercyjny generator mowy - model 7772 firmy IBM.

1984 - Pierwszy system do rozpoznawania mowy na dużej maszynie. Analiza każdego wyrazu trwa wiele minut. Urządzenie rozpoznawało około 5000 pojedynczych słów angielskich.

1986 - Prototyp systemu Tangora 4: dzięki specjalizowanym mikroprocesorom po raz pierwszy przetwarzanie mowy może odbywać się na stacji roboczej i w czasie rzeczywistym. System zawiera już mechanizm kontroli kontekstowej

1990 - Dragon Systems przedstawia pierwszą amerykańską wersję systemu Dragon Dictate.

1992 - Technologia Tangora jako model klient-serwer: niezbędny jest system IBM RS/6000 z systemem operacyjnym AIX. Rejestracja głosu odbywa się na pecetach pracujących pod kontrolą OS/2.

1993 - Personal Dictation firmy IBM jest pierwszym typowo pecetowym systemem przetwarzającym głos. Jego cenę ustalono na 1000 dolarów. Philips Dictation Systems przedstawia pierwszą wersję pakietu do ciągłego rozpoznawania mowy.

Historia c.d.

1996 - IBM OS/2 Warp 4 - pierwszy komercyjny system operacyjny z wbudowanymi funkcjami rozpoznawania mowy i nawigacji głosem.

1997 - Speech Magic Philipsa - rozwiązanie klasy klient-serwer. Spółka Lernout & Hauspie prezentuje pierwszy anglojęzyczny pakiet do rozpoznawania mowy.

1998 - IBM, Dragon, Lernout & Hauspie oraz Philips opracowują komercyjne wersje swoich produktów.

2001 - Rozpoznawanie mowy dla mas: premiera korzystających z mechanizmów analizy głosu MS Office XP i Corel WordPerfect Office 2002.

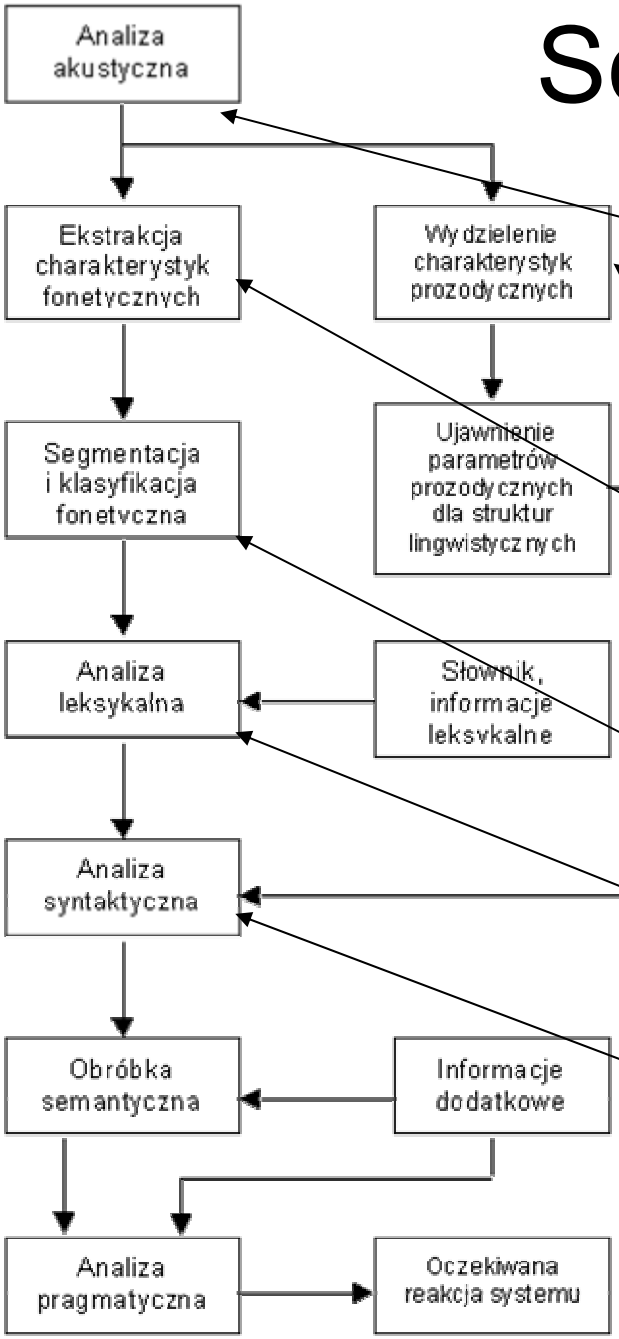
Firmy związane z rozpoznawaniem mowy

- IBM i Lotus program ViaVoice do pakietu biurowego SmartSuite
- Microsoft środowisko SpeechAPI do MS Office
- L&H (Lernout & Hauspie) wykupiona przez Microsoft zbankrutowała – rozpoczęła pracę nad rozpoznawaniem mowy polskiej
- Neurosoft z Wrocławia i Drive z Sopotu pracują nad takim oprogramowaniem
- Politechnika Częstochowska i Śląska

Zalety i wady stosowanych w systemach rozpoznawania podstawowych jednostek językowych

Typ jednostki	Zalety	Wady
Alofon	- stosunkowo wyraźnie różnią się akustycznie, - informacja o granicach między słowami, -małe wymagania dotyczące reguł na niskich poziomach	- trudności w budowie środków technicznych do wydzielania, - zbyt duża liczba alofonów, - większość ma parametry zależne od otoczenia
Fonem	- niewielka liczba klas, - bezpośrednio występują w słownikach (transkrypcja fonetyczna)	- są fonetycznie trudno rozpoznawalne, potrzebne są dodatkowe reguły na niższych i wyższych poziomach rozpoznawania
Diafon	- uwzględniona informacja o przejściach międzyfonemowych, - możliwość uzyskania reguł koartykulacyjnych	- duża liczba diafonów, - trudności ze stosowaniem dużej liczby reguł fonologicznych
Sylaba	- łatwość rozdziału, - duża liczba reguł koartykulacyjnych, - reguły fonologiczne zawierają warunki odnośnie granic sylab	- trudności dokładnego określenia granicy, - duża liczba sylab
Słowo	- zmniejsza się liczba poziomów rozpoznawania	- złożony zbiór wzorców klas w przypadku dużych słowników, - trudne do opisu w słownikach reguły fonologiczne

Schemat blokowy procesu rozpoznawania mowy



Pomiar natężenia sygnału mowy w funkcji czasu – karta graficzna

Intonacja, akcent, rytm wpływają na znaczenie

Podział kontinuum na okna czasowe, a następnie przy pomocy FFT transformacja do dziedziny częstotliwości

Ustalenie ciągów fonetycznych

Ciągi fonemów porównywalne są ze słownikiem

Określana zostaje struktura zdań

Fonemy i ich cechy

Fonemy to jedne z najmniejszych części języka mówionego. Około 55 wystarcza aby wypowiedzieć zdanie w dowolnym ziemskim narzeczu. Poszczególny język wykorzystuje maksymalnie 30 naraz. Język polski zawiera 37 fonemów.

Z reguły rozpoznawanie opieramy na fonemach, diafonach i sylabach

Podział systemów rozpoznawania mowy

Systemy	Forma sygnału	Słownik	Operatorzy	Informacje językowe	Otoczenie
Rozpoznawanie izolowanych słów	Pojedyncze słowa	10-300	Współpracujący z systemem	Ograniczone wykorzystanie dodatkowych informacji językowych	
Ograniczone rozpoznawanie mowy ciągłej	Mowa ciągła	30-500	Współpracujący z systemem	Ograniczone wykorzystanie dodatkowych informacji językowych	Cichy pokój
Ograniczone rozpoznawanie i rozumienie mowy ciągłej	Mowa ciągła	1000-2000	Nie współpracujący z systemem	Wykorzystanie informacji językowych	
Ograniczony dyktafon	Mowa ciągła	1000-10000	Współpracujący z systemem	Pełne wykorzystanie dodatkowych informacji językowych	Cichy pokój
Nieograniczone rozpoznawanie mowy ciągłej	Mowa ciągła	Nieograniczony	Nie współpracujący z systemem	Nie wykorzystuje się	Cichy pokój
Nieograniczone rozumienie mowy	Mowa ciągła	Nieograniczony	Nie współpracujący z systemem	Pełne wykorzystanie dodatkowych informacji językowych	

Główne zastosowania rozpoznawania mowy

Większość dotychczas realizowanych systemów rozpoznawania mowy była (i jest nadal) projektowana dla następujących zadań:

- rozpoznawanie izolowanych słów,
- rozpoznawanie ograniczonych ciągów izolowanych słów,
- rozpoznawania ciągów cyfr lub słów określonego formatu,
- wyszukiwanie i wydzielenie kluczowych słów w kontekście,
- ograniczone rozumienie mowy (komunikatów),
- rozpoznawanie mowy ciągłej.

Zalety sterowania głosem

- Sterowanie głosem może być realizowane znacznie szybciej niż za pośrednictwem np. klawiatury alfanumerycznej. Szybkość komunikowania się operatora sprzętu cyfrowego za pośrednictwem klawiatury nie przekracza około 0,5 słowa na sekundę. Szybkość pisania na maszynie przez profesjonalną maszynistkę wynosi około 1,5-2,5 słów/s. Jednocześnie spontaniczna mowa przebiega ze średnią szybkością od 2-3,6 słów/s.
- Przekazywanie informacji do maszyny za pomocą sygnału mowy umożliwia zwolnienie rąk operatora, które mogą równocześnie być wykorzystywane do manipulowania innymi układami sterowania lub wprowadzania danych.

Zalety sterowania głosem c.d.

- Przekazywanie informacji głosem może mieć miejsce w różnych nietypowych sytuacjach i położeniach operatora (np. pod wodą).
- Czas reakcji głosowej jest znacznie krótszy niż reakcji ruchowej, co pozwoliłoby na wykorzystanie sterowania głosem tam, gdzie wymagane jest szybkie sterowanie lub błyskawiczne wyłączenie układu na skutek zaistniałego niebezpieczeństwa lub zagrożenia.
- Układy sterowania głosem pozwoliłyby w znacznym stopniu złagodzić skutki kalectwa ludzi, objawiającego się ograniczeniem zdolności ruchowych rąk lub nóg, dzięki możliwości sterowania manipulatorami lub wózkami inwalidzkimi.
- Sterowanie głosem nie wymaga od operatora specjalnego przygotowania ani też treningu

Inicjalizacja aplikacji MS Speech API

Kontekst jest niezbędny do rozpoznawania mowy. Zazwyczaj cała aplikacja ma jeden kontekst, ale można stworzyć wiele kontekstów, które będą powiązane z różnymi częściami programu

(np.: menu, okna dialogowe, okna aplikacji)

```
//Globalna Definicja
CComPtr<ISpRecoContext> g_cpRecoCtxt;
// utworzenie kontekstu rozpoznawania mowy
hr = g_cpEngine->CreateRecoContext(&g_cpRecoCtxt );
if ( FAILED( hr ) ) //wyjście z aplikacji
```

Inicjalizacja aplikacji MS Speech API

Poprzez załadowanie gramatyki środowisko wie co ma zrobić. Są dwa główne rodzaje gramatyk: do rozpoznawania ciągłego mowy (*dictation*) i do rozpoznawania w trybie komend (*command and control*). Zasady gramatyki mogą być stworzone w XML-u, skompilowane i dołączone do zasobów aplikacji.

```
// Załadowanie gramatyki
// zdefiniowana przez użytkownika ("SRGRAMMAR") typ zasobów.
hr = g_cpRecoCtxt->CreateGrammar(GRAMMARID1, g_cpCmdGrammar);

if ( FAILED( hr ) ) //opuść aplikację

hr =    g_cpCmdGrammar->LoadCmdFromResource(
        NULL,
        MAKEINTRESOURCEW(IDR_CMD_CFG),
        L"SRGRAMMAR",
        MAKELANGID( LANG_NEUTRAL, SUBLANG_NEUTRAL),
        TRUE);
if ( FAILED( hr ) ) //opuść aplikację
```

Przykładowa gramatyka w XML-u

```
<GRAMMAR LANGID="409">
   <DEFINE>
      <ID NAME="VID_Navigation_Move" VAL="254"/>
                                 <ID NAME="VID_Navigation_Stop" VAL="255"/>
                                 <ID NAME="VID_Navigation_Back" VAL="256"/>
                                 <ID NAME="VID_Navigation_Left" VAL="257"/>
                                 <ID NAME="VID_Navigation_Right" VAL="258"/>
                                 <ID NAME="VID_Navigation_Ahead" VAL="259"/>
   </DEFINE>
   <RULE ID="VID_Navigation_Move" TOPLEVEL="ACTIVE">
      <O>Please</O>
      <P>
         <L>  <P>Move Forward</P>  </L>
      </P>
   </RULE>
   <RULE ID="VID_Navigation_Stop" TOPLEVEL="ACTIVE">
      <O>Please</O>
      <P>
         <L>   <P>Stop</P>   </L>
      </P>
   </RULE>
   <RULE ID="VID_Navigation_Back" TOPLEVEL="ACTIVE">
     <O>Please</O>
     <P>
        <L>  <P>Move Back</P>  </L>
     </P>
   </RULE>
……….
……
</GRAMMAR>
```

Inicjalizacja aplikacji MS Speech API

Aktywowanie gramatyki

// Set rules to active, we are now listening for commands

hr = g_cpCmdGrammar->SetRuleState(NULL, NULL, SPRS_ACTIVE);

Po aktywacji rozpoznawanie dźwięków odbywa się w tle na zasadzie pojawiania się zdarzeń (*event*). SAPI wysyła zdarzenia w momentach charakterystycznych dla rozpoznawania mowy:

SPEI_SOUND_START - pojawienie się dźwięku

SPEI_SOUND_END – zakończenie dźwięku

SPEI_RECOGNITION – gdy rozpoznanie zakończyło się powodzeniem

Inicjalizacja aplikacji MS Speech API

Ustalenie, że tylko komunikat SPEI_RECOGNITION będzie przesyłany do aplikacji i żaden inny

hr = g_cpRecoCtxt->SetInterest(SPFEI(SPEI_RECOGNITION), SPFEI(SPEI_RECOGNITION));

Powiązanie komunikatów przesyłanych z SAPI z konkretnym oknem aplikacji; jeśli jest SPEI_RECOGNITION to jest WM_RECOEVENT

hr = g_cpRecoCtxt->SetNotifyWindowMessage(hWnd, WM_RECOEVENT, 0, 0);

Główna pętla symulacji WndProc obsługuje zdarzenie WM_RECOEVENT

```
case WM_RECOEVENT:
        ProcessRecoEvent( hWnd );
        break;
```

Inicjalizacja aplikacji MS Speech API

Funkcja sprawdzająca jakie zdarzenie się pojawiło

```
void ProcessRecoEvent( HWND hWnd )
{
CSpEvent event;
while (event.GetFrom(g_cpRecoCtxt) == S_OK)
    {                                                   // sprawdzane jest jedynie
            switch (event.eEventId)                             //rozpoznanie mowy
            {
            case SPEI_RECOGNITION:
                    ExecuteCommand(event.RecoResult(), hWnd);
                    break;
            }
    }
}
```

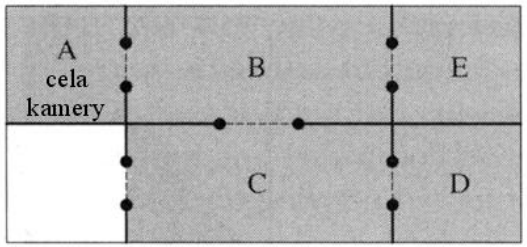
Inicjalizacja aplikacji MS Speech API

```
void ExecuteCommand(ISpPhrase *pPhrase, HWND hWnd)
{ SPPHRASE *pElements;
// wybranie frazy, której id figuruje w gramatyce
    if (SUCCEEDED(pPhrase->GetPhrase(&pElements)))
    {   switch ( pElements->Rule.ulId )
        {   case VID_Navigation:
            {   switch( pElements->pProperties->vValue.ulVal )
                {   case VID_Counter:
                        PostMessage( hWnd,WM_GOTOCOUNTER,NULL,NULL);
                        break;
                }
            }
            break;
        }
        ::CoTaskMemFree(pElements);
    }
}
```

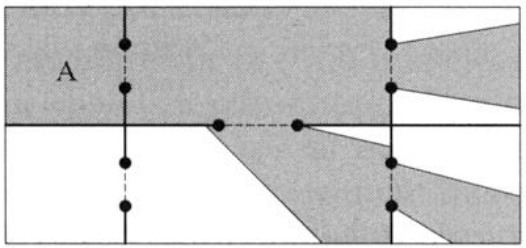
Podsumowanie

- Będzie popularne, gdy stanie się niezawodne
- Systemy rozpoznawania komend mają zastosowanie w interfejsach użytkownika
- (np.: telefony komórkowe, system operacyjny, aplikacje dla niepełnosprawnych)
- Warto również zwrócić uwagę na systemy generowania mowy (bardzo popularne)
- Przyszłość systemów rozpoznawania mowy znajduje się w obsłudze wszystkich otaczających nas urządzeń elektronicznych

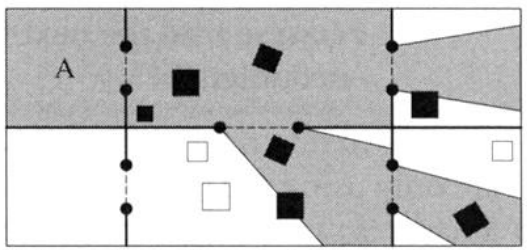
Portaling



(a) Cele widoczne z celi A - widoczność z celi do celi

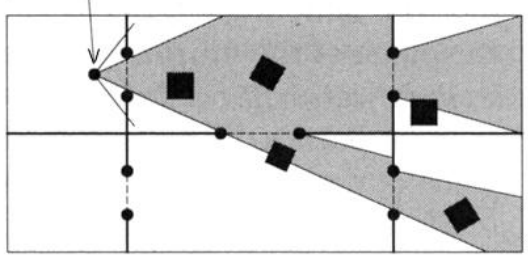


(b) Obszar potencjalnie widoczny dla obserwatora z celi A - widoczność obszaru celi



Obliczenia wstępne

(c) Potencjalnie widoczne obiekty z celi A - PVS



Obliczenia w czasie rzeczywistym

(d) Obiekty widoczne z kamery (punktu widzenia)