Grażyna Szostek[*], *Agnieszka Grudzińska*[*]

SEMANTYCZNY PARSER JĘZYKA POLSKIEGO

Przekazywanie informacji za pomocą komputera sprawia, że ilość przekazywanej w ten sposób informacji rośnie lawinowo, przerastając coraz częściej zdolność odbioru przeciętnego człowieka. Rośnie więc z każdym dniem potrzeba stworzenia komputerowych narzędzi (programów), które mogłyby dokonywać wstępnej *klasyfikacji* lub nawet *selekcji* informacji kierowanej do konkretnego odbiorcy. Opracowanie takich narzędzi wymaga szeroko zakrojonych badań nad technikami automatycznego prztwarzania tekstu [9]. Istotnym składnikiem każdego przetwarzania musi być semantyczny parser języka polskiego.

1. Wprowadzenie

Język naturalny jest podstawowym medium komunikacji międzyludzkiej. Nic więc dziwnego, że od kilkudziesięciu lat trwają prace nad poznaniem mechanizmu mówienia i rozumienia ludzkiej wypowiedzi. Wśród wielu sformułowanych dotąd koncepcji wyróżnia się *conceptual dependency*, tj. formalny język opisu semantyki wypowiedzi, zintegrowany z algorytmem rozumienia tekstu.

2. Zarys teorii związków pojęciowych (Conceptual Dependency)

Podstawą reprezentacji różnorodnych informacji jest reprezentacja *zdarzenia elementarnego*, przyjęta i opracowana przez R. Schanka. Jest to prosta struktura, o niezmiennej konstrukcji. Niezależnie od formy wypowiedzi jej postać zawsze oparta jest na tym samym schemacie. Struktura ta wiąże pojęcia ogólne, takie jak *czynność, sprawca czynności, obiekt podlegający czynności, źródło i cel czynności*, a nazywana jest formułą CD (CD – *Conceptual Dependency*).

[*] Instytut Matematyki, Wyższa Szkoła Pedagogiczna, Rzeszów

Każde zdarzenie opisuje się formułą CD w następujący sposób:

AKTOR

AKCJA wykonana przez aktora

OBIEKT podlegający akcji

KIERUNEK, w którym akcja jest zorientowana OD – DO.

Ta postać formuły jest obligatoryjna, nawet gdy w zdaniu nie możemy odnaleźć wszystkich jej elementów. Czytając zdanie i dokonując jego rozbioru gramatycznego, uzupełniamy poszczególne elementy formuły, nadając im konkretne wartości. Elementy, których nie znajdziemy w tekście, pozostaną bez nadanych wartości [11].

Rozpatrzmy dwa zdania:

Janek dał Marysi książkę.

Marysia wzięła od Janka książkę.

Początkowo wszystkie pola w formułach są puste, wartości nieznane. Rozpatrując zdanie pierwsze, nietrudno odszukać aktora – Janka, obiekt – książkę, akcję – dawać oraz kierunek „do Marysi". Jeśli chodzi o drugie zdanie: aktor to Marysia, obiekt – też książka, akcja – wziąć, kierunek „od Janka". Nie mamy jednak pełnych informacji o kierunkach, w pierwszym przypadku nie znamy wartości pola „od", w drugim pola „do". Na razie mamy:

aktor		Janek	aktor		Marysia
akcja		dawać	akcja		wziąć
obiekt		książka	obiekt		książka
kierunek	DO	Marysia	kierunek	DO	*nieznane*
	OD	*nieznane*		OD	Janek

Rozważając te zdania, możemy uzupełnić też dwie nieznane wartości, ponieważ znaczenie słowa „dawać" precyzuje osobę dawcy, czyli „kierunek od", jako sprawcę dawania, a znaczenie czasownika „brać" wskazuje na aktora, który jest biorcą, czyli pole „kierunek do" będzie miało tę właśnie wartość.Teraz mamy już uzupełnione formuły:

aktor		Janek	aktor		Marysia
akcja		dawać	akcja		wziąć
obiekt		książka	obiekt		książka
kierunek	DO	Marysia	kierunek	DO	Marysia
	OD	Janek		OD	Janek

Formuły te różnią się wartością dwóch pól – aktor i akcja. Można zauważyć, że czasowniki „dawać" i „wziąć" są w pewien sposób podobne, a ich znaczenie podstawowe jest takie samo. Efektem obu jest przeniesienie prawa własności przedmiotu (albo prawa posiadania w danej chwili) z jednej osoby na inną. W tym przypadku Janek przekazał prawo do posiadania książki na Marysię. A więc akcję tę można nazwać wspólnym dla obu czasowników wyrażeniem np. TRANSFER POSSESSION.

aktor		Janek		aktor		Marysia
akcja		TRANSFER POSSESION		akcja		TRANSFER POSSESION
obiekt		książka		obiekt		książka
kierunek	DO	Marysia		kierunek	DO	Marysia
	OD	Janek			OD	Janek

Mając już tak określony akt elementarny TRANSFER POSSESION, możemy poinformować maszynę, że kiedy pojawi się taka akcja, a pola aktor i „kierunek do" mają takie same wartości, to znaczy, że chodzi o słowo „wziąć", a kiedy pola aktor i „kierunek od" – o słowo „dawać". Odtąd, zawsze kiedy będzie taka sytuacja, uruchamiane będą odpowiednie reguły wnioskowania. Oprócz zysku czasu i zmniejszenia ilości operacji mamy jeszcze jedną cenną właściwość tego sposobu zapisu. Chodzi o to, że te same lub podobne zdarzenia są opisywane prawie takimi samymi formułami.

Zjawisko „pojęciowego" podobieństwa czasowników zostało oczywiście zauważone i wykorzystane w teorii CD. W zapisie zdań w postaci formuł zastosowano uproszczenie dotyczące nazywania czynności. U podstaw wprowadzenia tej zmiany leży spostrzeżenie, że słysząc zdanie, nie rozpatrujemy go dosłownie, ale bierzemy pod uwagę treści i informacje, które ze sobą niesie. Treści te nie są związane z konkretnym czasownikiem; gdyby został w zdaniu użyty inny, równoważny znaczeniowo, czasownik, informacja, którą otrzymalibyśmy, byłaby dokładnie ta sama. Korzystając z tego faktu Schank wprowadził tzw. *akcje elementarne*, które w formułach pełnią rolę czasowników. Znaczenie aktu elementarnego jest ogólniejsze niż pojedynczego czasownika, który zastępuje, jest to znaczenie podstawowe słowa. Jeśli dwa różne zdania opisujące to samo zdarzenie zapiszemy w postaci formuł CD, obie będą miały tę samą akcję elementarną. Akty grupują czasowniki, które mają zbliżone znaczenie, jeśli chodzi o efekt czynności, pomimo że ich konkretne znaczenia są bardzo różne. Przykładem mogą być słowa „wziąć" i „dać", których efektem jest ta sama sytuacja: jedna osoba coś zyskała, druga straciła. Czynność, którą określiliśmy jako TRANSFER POSSESSION, w teorii Schanka nazwana została ATRANS. Znaczenie ATRANS nie ogranicza się tylko do podarunków, opisuje też pożyczenie, a nawet kupno przedmiotu; występuje tu podwójny ATRANS: kupowanej rzeczy i pieniędzy.

Jeżeli do formuł wprowadzimy w miejsce czasowników akcję ATRANS, pozostanie tylko jedna różnica – osoba aktora, a formuły będą teraz wyglądać:

aktor		Janek		aktor		Marysia
akcja		ATRANS		akcja		ATRANS
obiekt		książka		obiekt		książka
kierunek	DO	Marysia		kierunek	DO	Marysia
	OD	Janek			OD	Janek

W porównaniu z ludzkimi opisami, które różnią się szczegółami lub nastawieniem osoby relacjonującej, użycie akcji elementarnych daje pewną generalizację reprezentacji rzeczywistości, relacja jest bezosobowa, obiektywna, rzeczowa i zawiera tylko fakty.

2.1. Akcje elementarne

Akcji elementarnych jest bardzo niewiele, w porównaniu z liczbą czasowników jest to wręcz znikoma ilość. Trudno uwierzyć, że ten niewielki zbiór może zastąpić setki czasowników używanych przez ludzi. Trzeba też wziąć pod uwagę fakt, że niektóre czasowniki w ogóle nie muszą mieć odpowiednika wśród akcji, ponieważ nie wnoszą do opisu rzeczywistości żadnych istotnych informacji, np. słowo „zdziwić" informuje o zmianie uczuć, ale nie można skonkretyzować akcji, której rezultatem jest zmiana stanu rzeczywistości.

Dotychczas wybrano i opracowano jedenaście akcji elementarnych, które efektywnie mogą zastąpić wszystkie czasowniki. Są to:

ATRANS	PTRANS	PROPEL	MTRANS
MBUILD	ATTEND	SPEAK	GRASP
MOVE	INGEST	EXPEL	

Przedstawimy je pokrótce, aby idea teorii stała się jaśniejsza, a także dlatego, że działanie parsera w dużej mierze opiera się na akcjach. [8, 11].

MTRANS

Jest to jedna z ważniejszych akcji elementarnych. Oznacza transfer informacji pomiędzy jedną a drugą osobą lub różnymi częściami umysłu. Odpowiada takim czasownikom, jak „czytać", „powiedzieć", „usłyszeć", „nauczyć", „obiecać" i wielu innym, których znaczenie to przekazywanie wiadomości. Porównajmy dwa zdania, obydwa zapisujemy formułami CD, używając MTRANS, ale zobaczmy, jakie różnice występują na pozostałych polach.

Robert zobaczył Kasię.

Robert czyta gazetę.

Zapisując te zdania w postaci formuł, posłużymy się dodatkowo akcją ATTEND, która zostanie omówiona trochę później (oznacza użycie narządu zmysłów).

Robert zobaczył Kasię:

aktor		Robert
akcja		MTRANS
obiekt		obraz Kasi
kierunek	DO	część mózgu odpowiedzialna za widzenie
kierunek	OD	oczy Roberta

instrument:

aktor		Robert
akcja		ATTEND
obiekt		oczy
kierunek	DO	Kasia
kierunek	OD	*nieznane*

Robert czyta gazetę:

aktor		Robert
akcja		MTRANS
obiekt		informacje z gazety
kierunek	DO	umysł Roberta
	OD	gazeta

instrument:

aktor		Robert
akcja		ATTEND
obiekt		oczy
kierunek	DO	Kasia
kierunek	OD	*nieznane*

Można zauważyć, że interpretacje formułowe obu zdań znacznie różniących się treścią są bardzo do siebie podobne. Okazuje się, że „patrzeć" i „czytać" są odmianami MTRANS-u, tylko w jednym wypadku obiektem jest obraz przedmiotu, w drugim informacja w nim zawarta.

ATRANS

Ta akcja zgodnie z tym, co wcześniej zostało napisane, jest używana w tych przypadkach, w których należy wyrazić przekazanie prawa posiadania z jednej osoby na drugą. Dotyczy, oprócz sytuacji transferu posiadania, także sytuacji sprawowania kontroli nad obiektem fizycznym.

PTRANS

PTRANS jest akcją elementarną oznaczającą fizyczne przemieszczanie się obiektu. Używana jest jako odpowiednik „iść", „jechać". W tym przypadku pole aktor ma tę samą wartość co pole obiekt. Przyjrzyjmy się przykładowi:

Tomek pojechał do Warszawy:

aktor		Tomek
akcja		PTRANS
obiekt		Tomek
kierunek	DO	Warszawa
kierunek	OD	*nieznane*

PROPEL

Ta akcja elementarna odpowiada tym sytuacjom, w których wymaga się użycia siły, np. do wprawienia przedmiotu w ruch. W interpretacji zdania:

Kamień uderzył chłopca

użyjemy właśnie tej akcji, chociaż nie wiadomo, przez kogo została wykonana.

Wiemy jednak, że ktoś ją wprawił w ruch i że kieruje się w stronę chłopca. Zapiszemy to zdarzenie w sposób nieco bardziej skomplikowany niż poprzednie, ponieważ trzeba tu uwzględnić dwa obiekty kamień i dziecko.

aktor	*nieznane*		
akcja	PROPEL		
obiekt	kamień		
kierunek	chłopiec		
	spowodowało	*stan*	kontakt
		obiekt 1	kamień
		obiekt 2	chłopiec

INGEST

Akcja INGEST jest używana, gdy trzeba wyrazić czasowniki oznaczające wprowadzanie jakiejś substancji do wnętrza organizmu ludzkiego lub zwierzęcego. Chodzi o słowa typu „jeść", „pić" „oddychać", ale również „brać narkotyki". Na przykład:

Janek wypił szklankę mleka.

aktor		Janek
akcja		INGEST
obiekt		mleko
kierunek	DO	usta Janka
kierunek	OD	szklanka

instrument:

aktor		Janek
akcja		PTRANS
obiekt		szklanka zawierająca mleko
kierunek	DO	usta Roberta
kierunek	OD	stół

instrument:

aktor		Janek
akcja		MOVE
obiekt		ręka Janka
kierunek	DO	szklanka
kierunek	OD	*nieznane*

instrument:

aktor		Janek
akcja		GRASP
obiekt		szklanka zawierająca mleko
kierunek	DO	ręka Janka

Przy okazji możemy zaobserwować rozrost zapisu spowodowany ekspansją instrumentów. Taki zapis jest trochę sztuczny, ale po zastanowieniu przyznamy, że stwierdzając fakt wypicia szklanki napoju, wiemy o tych wszystkich poszczególnych czynnościach, są one jednak na tyle typowe i ogólnie znane, że nie warto ich wspominać. Dopiero w sytuacji, gdy tok czynności odbiega od normy (np. gdy pije osoba, która nie może posłużyć się rękoma), zaznaczamy to w wypowiedzi.

MBUILD

MBUILD oznacza czynność umysłu, tworzenie nowej wiedzy na podstawie już posiadanych wiadomości. MBUILD na wejściu dostaje informacje zapamiętane i przechowywane w mózgu lub pobierane bezpośrednio z otoczenia (obiekt DO), produkuje nową informację (obiekt Z).

Janek odgadł, gdzie są pieniądze:

aktor		Janek
akcja		MBUILD
obiekt DO		LOC (pieniądze) to ?
obielt Z		C (pieniądze) to X
kierunek	DO	umysł
	OD	pamięć

Pozostałe akcje elementarne nie mają tak istotnego znaczenia w tworzeniu interpretacji tekstu, pełnią funkcję w pewien sposób pomocniczą, opisując np. instrumenty w formułach, gdzie nadrzędną rolę pełni któraś z akcji przedstawionych powyżej.

ATTEND – zaangażowanie któregoś z organów zmysłów, np. oczu,
SPEAK – produkowanie dźwięków języka naturalnego,
MOVE – ruch części ciała,
GRASP – zaciśnięcie ręki na obiekcie, uchwycenie przedmiotu,
EXPEL – usunięcie substancji z ciała ludzkiego lub zwierzęcego.

Akcje elementarne reprezentują tylko część wiedzy – wiedzę elementarną o zdarzeniach. Do przechowywania wiedzy o sekwencjach akcji służą jednostki zwane *skryptami lub scenariuszami.* Wiedza o ludzkich intencjach jest zapisana w jednostkach nazywanych *tematami* (themes). Drugim elementem opisu ludzkich intencji są *plany* (plans). Oprócz tego Schank wyróżnił jednostki przechowujące wiedzę o sposobie zrealizowania planu, które nazwał *plan boxes* [1, 2, 3].

3. Parser semantyczny[1)]

Struktura jednostek wiedzy oraz kształt algorytmów procesu rozumienia są z mocy teorii uniwersalne. Natomiast konkretne jednostki wiedzy i konkretne reguły użyte w algoryt-

[1)] Parser skonstruowano dla rozumienia krótkiej notatki prasowej.

mach są tylko przykładowe. Skonstruowano je bowiem dla rozumienia następującej notatki prasowej:

„Jakież było zdziwienie ekipy remontowej, która przyjechała, by dokonać planowanej od roku naprawy mostu na drodze prowadzącej do zbankrutowanego PGR-u w Suwalskiem. Most zniknął. Policji udało się znaleźć jedynie przerdzewiałe fragmenty konstrukcji wbite w dno rzeki".

Algorytm działania parsera opiera się na założeniu, które poczynił Schank [13]. Chodzi mianowicie o spostrzeżenie, że do zrozumienia tekstu, czyli do wydobycia intencji mówiącego, nie trzeba rozumieć wszystkich elementów wypowiedzi. Wystarczy znać sens i uwzględniać znaczenie tylko niektórych słów, konkretnie tych, których wartość konceptualna może być pomocna w wykryciu intencji bohatera opowiadania. Ze względu na to słowa zostały podzielone na:

- słowa ważne, czyli te, których wartość konceptualna służy do budowy konceptualnej interpretacji tekstu,
- słowa pomocnicze, czyli takie, których wartość konceptualna uprawomocnia lub zwiększa precyzję jednej z konceptualnych wartości słowa ważnego,
- słowa puste, tj. te, których wartość konceptualna nie ma znaczenia przy budowie konceptualnej struktury konkretnego tekstu.

Wszystkie słowa są zapisane w specjalnym słowniku zbudowanym specjalnie dla potrzeb parsera, o czym pisaliśmy już wcześniej. Słownik jest budowany dla określonej grupy zagadnień, dla których możemy z góry przewidzieć zakres słownictwa, ponieważ każde ze słów musi być opracowane dla potrzeb parsera. Wyrazy są ułożone w słowniku w kolejności alfabetycznej, co ułatwia wyszukiwanie. W każdej linii na pierwszym miejscu jest forma podstawowa, po niej wszystkie formy gramatyczne, a dopiero po nich kategoria konceptualna tego słowa. Trzeba zaznaczyć, że od rodzaju kategorii konceptualnej słowa zależy jej konkretyzacja, np. akcje elementarne nie są uzgadniane przez żadną szczegółową kategorię, ponieważ są to tzw. kategorie samoreprezentujące. Oznacza to, że kategoria tego typu reprezentuje samą siebie i nie trzeba jej nadawać żadnej konkretnej wartości. Natomiast takie kategorie, które odpowiadają polom „aktor", „obiekt" „kierunek", są konkretyzowane kategoriami szczegółowymi, np. imieniem osoby (aktor) lub nazwą przedmiotu (obiekt).

Parser pracuje cyklicznie, przegląda i analizuje tekst od prawej do lewej, jak człowiek. Na każde przeczytane słowo przypada jeden cykl, który obejmuje:

- identyfikację słowa,
- generowanie oczekiwań,
- weryfikacja oczekiwań.

Identyfikacja słowa polega na wydzieleniu ciągu liter ograniczonego spacjami, a następnie przeszukaniu słownika w celu sprawdzenia, czy ten wyraz w nim się znajduje. Jeżeli parser nie znajdzie analizowanego słowa w słowniku, dochodzi do wniosku, że ma do czynienia ze słowem pustym i przystępuje do analizy następnego elementu tekstu. Jeśli zostanie odnalezione, parser pobiera jego kategorię konceptualną. Kiedy słowo zostanie odszukane, parser musi rozstrzygnąć, czy słowo to jest ważne, czy pomocnicze. Słowo pomocnicze zostaje wpisane do podręcznej pamięci pomocniczej i program przechodzi do kolejnego wy-

razu, w przypadku słowa ważnego parser wydobywa jego wartość konceptualną, którą przekazuje do następnych bloków działań (tzn. 2 i 3). Blok identyfikacji słowa wznawia działanie, kiedy pozostałe bloki skończą swoją pracę.

Blok *generowania oczekiwań* otrzymuje od poprzedniego bloku wartość konceptualną słowa a następnie sprawdza, czy w którejś z formuł stanowiących konceptualny zapis scenariuszy ta kategoria się nie pojawiła. Gdy taka formuła zostanie znaleziona, utworzona zostaje lista oczekiwań, w której umieszczane są wszystkie kategorie występujące w formule. Później program może powrócić do bloku identyfikacji i analizy następnego elementu tekstu. Taki sposób działania można porównać do zachowania człowieka, który usłyszawszy początek wypowiedzi, czeka na dalszy ciąg.

Weryfikacja oczekiwań to blok czynności pośredniczących pomiędzy identyfikacją słowa a generowaniem oczekiwań. Weryfikując oczekiwania, program sprawdza, czy wartość konceptualna danego słowa zawiera kategorię, która znajduje się na liście oczekiwań. Jeżeli tak, blok weryfikacji zezwala na przekazanie tej wartości konceptualnej blokowi generowania oczekiwań. Kiedy zidentyfikowanego słowa nie ma na liście oczekiwań parser musi sprawdzić, czy to słowo nie wiąże się z inną formułą CD albo nie jest ważne ze względu na strukturę obiektu [8].

Tak wygląda ogólny schemat działania parsera w systemie rozumiejącym tekst opartym na Schankowskiej teorii języka. Jednak w konkretnych rozwiązaniach algorytm ten musi być modyfikowany i uzupełniany o elementy właściwe dla danego problemu, podobnie jak i inne moduły tego systemu, ponieważ określone zastosowania i sposoby podejścia do problemu rozumienia ludzkiej mowy wymuszają położenie nacisku na różne elementy teorii.

3.1. Schemat działania

Parser zbudowany przez nas dla potrzeb naszego systemu opiera się na podstawach stworzonych przez Schanka i jego współpracowników, ale ze względu na cel programu, a także zastosowany język programowania konieczne stały się modyfikacje i rozwinięcie podstawowego schematu działania.

Zasadniczy schemat działania pozostaje niezmieniony w stosunku do algorytmu opisanego w poprzednim rozdziale, tzn. są trzy główne etapy parsingu:

1) identyfikacja słowa,
2) generowanie oczekiwań,
3) weryfikacja oczekiwań.

Słownik dla wybitnie fleksyjnego języka polskiego musi być jednak specyficzny właśnie dla tego języka. Nie chodzi tu o to, że ma być zbudowany z polskich słów, ponieważ to jest oczywiste, ale o fakt, że należy uwzględnić wszystkie formy gramatyczne każdego wyrazu. Wiąże się to z tym, że parser po odczytaniu słowa musi odszukać je w słowniku, a rzadko się zdarza, że wyrazy występują w tekście w formie podstawowej. Słownik w parserze języka polskiego pełni więc obok swojej podstawowej roli także zadanie kojarzenia form fleksyjnych pojawiających się w tekście z formą podstawową słowa. Aby słownik miał przejrzystszą budowę, był mniejszy i łatwiej przebiegało szukanie słów, zastosowałyśmy następujący zapis: na pierwszym miejscu znajduje się forma podstawowa wyrazu, za nią wszyst-

kie możliwe formy gramatyczne (a ściślej końcówki), a dopiero po nich kategoria konceptualna słowa. Należy wspomnieć, że jedne wyrazy mają krótką szczegółową kategorię (np. „iść" odpowiada kategorii PTRANS), natomiast inne wywołują całe struktury (np. „zbudować" może mieć jako odpowiednik „FROM nic TO DOM"). Takie typy kategorii wiążą się z tym, że ten słownik nie jest po prostu zbiorem słów w jednym języku z ich odpowiednikami w drugim języku (jak np. słownik polsko-angielski), ponieważ kategorie są wyrazem treści konceptualnych, które niesie słowo. Przy tym podejściu nie jest ważne, jaką częścią mowy jest słowo, ale informacje o świecie lub zmianie rzeczywistości, które ono reprezentuje. Krótko mówiąc, każda jednostka słownika wyraża pośrednią zależność pomiędzy wyrażeniem językowym a wiedzą konceptualną, ujętą przez kategorię.

Zgodnie z Schankowską teorią parsera nasz parser czyta tekst słowo po słowie i analizuje każde z nich. Po odczytaniu wyrazu przeszukuje słownik, aby sprawdzić, czy słowo w nim się znajduje. Gdy go nie ma w słowniku, wyraz traktowany jest jako pusty, a więc nieprzydatny dla tworzenia konceptualnej interpretacji tekstu, i pomijany w dalszej analizie. Kiedy parser znajdzie słowo w słowniku, sprawdza, jakiego typu ono jest. Gdy okaże się słowem pomocniczym, wpisywane jest do podręcznej pamięci pomocniczej. W przypadku natrafienia na słowo ważne, niosące ze sobą konkretne wartości konceptualne podejmowany jest szereg akcji. Po odczytaniu jego kategorii konceptualnej sprawdzana jest lista kategorii, którym została już nadana podczas analizy poprzednich słów tekstu konkretna wartość, roboczo nazywana zbiorem zmiennych uzgodnionych. Kiedy okazuje się, że ta kategoria już ma nadaną wartość konceptualną, nie trzeba podejmować żadnych dalszych działań, ponieważ oznacza to, że taka kategoria pojawiła się już wcześniej przy analizie jakiegoś innego słowa. W przeciwnym przypadku (tzn. gdy kategorii nie ma wśród zmiennych uzgodnionych) sprawdzana jest lista oczekiwań (jeśli analizowane właśnie słowo jest pierwszym elementem tekstu, lista jeszcze nie istnieje), a następnie dodaje się tę kategorię wraz z przyporządkowaną jej wartością konceptualną do zbioru kategorii, którym nadana została wartość konceptualna (czyli zbioru zmiennych uzgodnionych). Kiedy kategoria zostaje odnaleziona na liście oczekiwań, będzie z niej usunięta, ponieważ analizowana właśnie kategoria spełniła to oczekiwanie.

Z kolei parser przechodzi do następnego etapu, do generowania oczekiwań. Pierwszym krokiem podejmowanym w tym bloku jest przeglądanie formuł CD tworzących skrypty. Spośród nich wybierane są te formuły, w których występuje rozpatrywana właśnie kategoria konceptualna. Po znalezieniu takiej formuły jest ona przez parser uaktywniana (ściągana do pomocniczego pliku, do którego dostęp ma tylko parser), ale wcześniej sprawdza się, czy formuła ta nie została już wcześniej uaktywniona przez inną kategorię, oczywiście w tym wypadku nie trzeba jej ponownie aktywizować. Następnym krokiem jest właściwe generowanie oczekiwań, polegające na wyłuskaniu z formuły wszystkich kategorii konceptualnych w niej występujących. Oczekiwania te wpisywane są do listy oczekiwań, ale wcześniej, na etapie wyszukiwania ich z formuł, przeszukiwany jest zbiór kategorii z już uzgodnionymi wartościami oraz lista oczekiwań, aby w żadnej z tych struktur kategorie nie były dublowane. Zbiór zmiennych uzgodnionych jest sprawdzany w celu uniknięcia sytuacji, w której wygenerowane zostałoby oczekiwanie na wartość konceptualną kategorii, która już w wyniku analizy któregoś z wcześniejszych elementów tekstu zyskała konkretną

wartość. Podobnie w przypadku listy oczekiwań: nie ma sensu dopisywać do niej oczekiwania, które już zostało wygenerowane przy analizie innej formuły.

Kiedy pojawia się kategoria typu „from...to...", pomijana jest pierwsza grupa akcji, mianowicie sprawdzanie zmiennych uzgodnionych i listy oczekiwań, ponieważ nie jest to kategoria, która ma szczegółową wartość konceptualną. W takim przypadku parser przechodzi od razu do etapu wyszukiwania formuł i generowania oczekiwań, które odbywają się według opisanego powyżej schematu.

Odczyt słowa, sprawdzanie zbioru kategorii z uzgodnionymi wartościami konceptualnymi, przeszukiwanie listy oczekiwań oraz aktywizacja formuł CD i generowanie oczekiwań powtarzane są cyklicznie do momentu, w którym parser osiągnie koniec tekstu. Na podstawie przeczytanych słów ma wtedy:

- listę oczekiwań, które nie zostały spełnione,
- zbiór kategorii z ich uzgodnionymi wartościami konceptualnymi,
- zbiór słów pomocniczych, które nie zostały wykorzystane przy analizie, ale mogą być przydatne w bardziej zaawansowanym systemie,
- zbiór formuł CD, które parser zaktywizował na podstawie kategorii konceptualnych.

Zaktywizowane formuły przechowywane są w czystej postaci, tzn. takiej, w jakiej są zapisane w scenariuszach. Celem pracy parsera nie jest jednak wyszukanie formuł związanych z analizowanym tekstem, ale przedstawienie zdań języka naturalnego w interpretacji formułowej. Zadanie postawione przed parserem polega na wyprodukowaniu możliwie najbardziej uzgodnionych formuł CD, które zostały wybrane ze skryptów na podstawie przeczytanych słów. Uzgadnianie odbywa się poprzez wprowadzanie do formuł wartości konceptualnych kategorii, które w tych wybranych formułach występują. Należy zaznaczyć, że akcje elementarne nie są uzgadniane z żadną konkretną wartością, ponieważ są traktowane jako zmienne, które mają wartość PRAWDA lub FAŁSZ. PRAWDA wtedy, gdy parser znajdzie w tekście słowo, którego kategorią konceptualną jest dana akcja elementarna, FAŁSZ w przeciwnym wypadku.

Parser po dotarciu do końca tekstu przechodzi do etapu uzgodnienia kategorii występujących w zaktywizowanych formułach. Dla każdej z kategorii, tworzących zaktywizowane formuły przeszukuje zbiór kategorii, które mają nadane konkretne wartości konceptualne, i wprowadza te wartości do formuł obok odpowiednich kategorii. Kategorie, których nie może uzgodnić z żadną wartością konceptualną, pozostają *nieznane*, co zaznaczane jest przez dwa puste nawiasy „()" (taka forma zapisu związana jest z postacią formuły CD). Kiedy kategoria oznaczająca akcję elementarną nie może zostać uzgodniona, cała formuła jest usuwana ze zbioru formuł zaktywizowanych, ponieważ zaistnienie akcji jest podstawowym warunkiem tego, aby cała formuła miała wartość konceptualną. Według tego założenia formuła, w której wszystkie kategorie są uzgodnione, ale akcja elementarna ma wartość FAŁSZ, jest nieużyteczna dla procesu rozumienia tekstu i w ogóle nie powinna być brana pod uwagę. Zbiór formuł z uzgodnionymi kategoriami konceptualnymi (niemożliwe do uzgodnienia wartości są zapisane w postaci „()") jest końcowym produktem działań parsera, przekazywanym do dalszych modułów systemu.

3.2. Algorytm działania parsera

W tej części artykułu chcemy przedstawić algorytm oparty na schemacie działania parsera przestawionego powyżej. To pozwoli w bardziej dokładny i szczegółowy sposób zobaczyć i prześledzić wszystkie kroki parsera.

1) Start
2) Sprawdzamy, czy jest osiągnięty koniec historii:
   a) tak: wypisujemy listę formuł zaktywizowanych;
      koniec.
   b) nie: przechodzimy do kroku 3.
3) Odczytujemy słowo z historii;
4) Sprawdzamy, czy dane słowo występuje w słowniku:
   a) tak: przechodzimy do kroku 5.
   b) nie: słowu nadajemy kategorię „pomocnicze"
      przechodzimy do kroku 5.
5) Sprawdzamy, czy kategorie konceptualnie przyporządkowane danemu słowu w słowniku są wyczerpane:
   a) tak: przechodzimy do kroku 2.
   b) nie: przechodzimy do kroku 6.
6) Sprawdzamy, czy kategoria ma wartość „pomocnicze„:
   a) tak: zapisujemy słowo do pamięci pomocniczej;
      przechodzimy do kroku 5.
   b) nie: przechodzimy do kroku 7.
7) Sprawdzamy, czy kategoria ma wartość „akcja elementarna" lub „zmienna":
   a) tak: przechodzimy do kroku 9.
   b) nie: przechodzimy do kroku 8.
8) Sprawdzamy, czy kategoria ma wartość „from..to „:
   a) tak: przechodzimy do kroku 12.
   b) nie: przechodzimy do kroku 5.
9) Sprawdzamy, czy kategorii została nadana wartość:
   a) tak: przechodzimy do kroku 5.
   b) nie: przechodzimy do kroku 10.
10) Sprawdzamy, czy w liście oczekiwań występuje dane oczekiwanie:
   a) tak: usuwamy oczekiwanie z listy;
      przechodzimy do kroku 11.
   b) nie: przechodzimy do kroku 11.
11) Kategorii nadajemy wartość;
12) Sprawdzamy, czy wszystkie formuły nieaktywne zostały przeglądnięte:
   a) tak: przechodzimy do kroku 5.
   b) nie: pobieramy kolejną formułę:
      przechodzimy do kroku 13.
13) Sprawdzamy, czy formuła zawiera daną kategorię:
   a) tak: przechodzimy do kroku 14.
   b) nie: przechodzimy do kroku 12.

14) Sprawdzamy czy dana formuła jest aktywna:
    a) tak: przechodzimy do kroku 12.
    b) nie: uaktywniamy formułę;
       przechodzimy do kroku 15.
15) Generujemy oczekiwania w oparciu o formułę.
16) Sprawdzamy, czy wszystkie oczekiwania zostały wygenerowane:
    a) tak: przechodzimy do kroku 12.
    b) nie: przechodzimy do kroku 17.
17) Sprawdzamy, czy oczekiwaniu (kategorii) została nadana wartość:
    a) tak: przechodzimy do kroku 16.
    b) nie: sprawdzamy stan oczekiwań;
       przechodzimy do kroku 18.
18) Sprawdzamy, czy w liście oczekiwań jest takie oczekiwanie:
    a) tak: przechodzimy do kroku 16.
    b) nie: dodajemy dane oczekiwanie;
       przechodzimy do kroku 16.

3.3. Przykładowe działanie parsera

Analizę notatki prasowej rozpoczynamy od odczytania pierwszego słowa: *jakież*. Odnajdujemy to słowo w słowniku i stwierdzamy, że jest to słowo puste, które nie podlega dalszej analizie.

Następnym słowem jest: *było*. Słowo *być* ma kategorię – słowo pomocnicze, czyli zapisujemy go do pamięci pomocniczej.

Kolejne słowo: *zdumienie* ma w słowniku dwie kategorie:

(FROM? – O BIEKT) (TO? + OBIEKT),

(FROM? + OBIEKT) (TO? – OBIEKT),

które oznaczają, że słowo jest ważne i podlega dalszej analizie. Analizę rozpoczynamy od pierwszej kategorii:

- przeglądamy po kolei wszystkie formuły nieuaktywnione,
- stwierdzamy, że nie ma formuł zawierających tę kategorię.

Czyli przystępujemy do analizy drugiej kategorii:

- przeglądamy formuły nieuaktywnione,
- zostaje znaleziona formuła:

  (?MTRANS (ACTOR? FACHOWCY) (OBJECT? FACHOWCY)

  (FROM? + OBIEKT) (TO? – OBIEKT))

Generujemy w oparciu o tę formułę oczekiwania:

- MTRANS – ta kategoria pojawiła się po raz pierwszy, więc dodajemy ją do listy oczekiwań,
- FACHOWCY – po sprawdzeniu zmiennych uzgodnionych stwierdzamy, że nie ma wśród nich tej kategorii, więc dodajemy ją do listy oczekiwań,
- OBIEKT – ponownie przeglądamy zmienne uzgodnione. Ta kategoria nie występuje, więc dodajemy ją do listy oczekiwań.

W ten sam sposób przeprowadzamy analizę kolejnych znalezionych zdań:

(?PTRANS (ACTOR? OBIEKT) (OBJECT? OBIEKT) (FROM? + OBIEKT)
(TO? – OBIEKT))

(?MTRANS (ACTOR? POLICJA) (OBJECT? POLICJA) (FROM? + OBIEKT)
(TO? – OBIEKT))

Przechodzimy do analizy słowa: *ekipy*. Po odnalezieniu tego słowa w słowniku stwierdzamy, że jest to słowo ważne. Kategorią słowa *ekipa* jest FACHOWCY, która jest nazwą zmiennej. Wartością zmiennej FACHOWCY jest właśnie słowo ekipa.

Analizę rozpoczynamy od przeglądnięcia zmiennych uzgodnionych, by sprawdzić, czy już taka zmienna nie została wcześniej uzgodniona. Okazuje się, że nie było jeszcze tej kategorii, więc przechodzimy do usunięcia kategorii z listy oczekiwań, jeśli w niej występuje. Następny krok polega na dodaniu kategorii do zmiennych uzgodnionych w postaci:

FACHOWCY = ekipa.

Teraz przeszukujemy formuły nieuaktywnione i znajdujemy następujące formuły:

(?PTRANS (ACTOR? FACHOWCY) (OBJECT? FACHOWCY) (FROM? X)
(TO ?OBIEKT))

(?MTRANS (ACTOR? FACHOWCY) (OBJECT? FACHOWCY)(FROM?
+ OBIEKT) (TO? OBIEKT))

(?MTRANS (ACTOR? FACHOWCY) (OBJECT? OBIEKT)(FROM? – OK)
(TO? + OK))

Analiza tych formuł jest taka sama jak poprzednio.

Analiza kolejnych słów jest zgodna z powyżej opisanymi schematami. Słowa, których kategorią jest akcja elementarna, analizujemy w ten sam sposób, jak słowa tekstowe o kategorii zmienna.

4. Podsumowanie

Zbudowany parser jest dowodem na to, że można zaimplementować Schankowską teorię związków pojęciowych tak, by zgodnie z jej założeniami reprezentować ludzką wiedzę o świecie i języku. Co więcej – język polski, chociaż składniowo dużo bardziej skomplikowany niż angielski, który dla teorii stanowił podstawę, także może być prawidłowo rozpoznawany i interpretowany za pomocą programu napisanego według algorytmu opartego na teorii CD. Równorzędność języków, w których teoria jest implementowana, jest jeszcze jednym dowodem na jej poprawność i uniwersalność, a więc potwierdza trafność wyboru właśnie tej teorii.

Przedstawiony parser działa tylko w określonym przedziale wiedzy, ograniczonym przez interpretowany tekst. Spowodowane to jest tym, że zasób informacji dla każdej dziedziny z życia ludzkiego jest ogromny, a zapisanie tych wiadomości w postaci skryptów i plan boxów to mrówcza praca. Dlatego zakres wiedzy, a przez to ilość słów, skryptów i plan boxów, został zawężony we wspólnym projekcie do informacji na tematy poruszane w krótkich notatkach prasowych. Jednak w celu sprawdzenia poprawności samego sposobu rozwiązania problemu i opracowanych algorytmów zasób wiedzy systemu został ograni-

czony do informacji niezbędnych dla zrozumienia przykładowej jednej notatki prasowej (o moście, który zniknął).

W związku z tymi ograniczeniami w wiedzy dostępnej dla całego systemu, wiedza parsera też została ograniczona. Dla parsera dostępne są skrypty zapisane w ciągach formuł, z których aktywizowane są tylko niektóre, powiązane znaczeniowo z tekstem. Zbiór formuł, które są interpretacją analizowanej notatki, został wzbogacony o kilka formuł zawierających kategorie nie związane ze słowami występującymi w tekście, aby symulować działanie parsera dla prawdziwego zbioru formuł, w którym nie wszystkie formuły są powiązane z tekstem notatki.

Słownik, którym posługuje się parser, również został ograniczony do zasobu słów z notatki, ponieważ zwiększenie ich ilości nie przyniosłoby żadnych dodatkowych informacji w przypadku tego tekstu. Rozszerzenie zasobu słów słownika wymaga nadania każdemu nowemu wyrazowi jego wartości konceptualnej, a także dopisania wszystkich jego form gramatycznych. W przyszłości, dla szerszych zastosowań, można byłoby powiązać parser ze słownikiem elektronicznym języka polskiego, który potrafiłby generować formy gramatyczne, jednak należałoby go wzbogacić o wartości konceptualne słów.

Cały system, po odpowiednim rozszerzeniu wiedzy każdego z modułów o informacje potrzebne do rozumienia tekstów z określonej dziedziny, mógłby być stosowany w praktyce do selekcji i formalnego zapisu wiedzy, np. w agencji prasowej, gdzie napływające depesze mogłyby być selekcjowane, a potem przechowywane w odpowiednich bazach danych. Uniwersalność systemu wiązałaby się jednak z przemyślanym opracowaniem słownika i story understandera, rozszerzeniem zbioru skryptów i plan boxów, a może również wymagałaby sięgnięcia po bardziej zaawansowane elementy teorii Schanka, takie jak akcje elementarne wyższych rzędów.

Literatura

[1] Czerniewski B.: *Mechanizm rozumienia tekstu jako inteligentny interfejs*. Praca magisterska, Katedra Informatyki AGH 1995

[2] Haj-Ali R., El-Zoghbi H.: *Inteligentny parser języka arabskiego*. Praca magisterska, Katedra Informatyki AGH 1996

[3] Kasprzyk A., Zbroja S.: *Rozumienie tekstu na podstawie wiedzy o stereotypowym przebiegu zdarzeń*. Praca magisterska, Katedra Informatyki AGH 1997

[4] Lubaszewski W.: *Czy nowe językoznawstwo*. JP LXIV, 1984

[5] Lubaszewski W.: *Robot – Bibliotekarz I: Rozumienie tekstu*. W: I Krajowa Konferencja Robotyki, Wrocław 1985

[6] Lubaszewski W.: *Archetype Driven Parser for Polish*. Praca niepublikowana

[7] Lubaszewski W.: *Jak komputer rozumie tekst polski*. JP LXIX, 1989

[8] Lubaszewski W.: *Rozumienie tekstu przez komputer*. Kraków, PAN 1990

[9] Raport Komisji Unii Europejskiej 1998 „LANGUAGE ENGINEERING. Progress and Prospects'98"

[10] Schank R.C.: *Conceptual Dependency. A Theory of Natural Language Understanading*. Cognitive Psychology 3, 1972

[11] Schank R.C.: *Conceptual Dependency Theory*. W: Schank R.C. (red.), Conceptual Information Processing, Amsterdam 1975, s. 22–82
[12] Schank R.C.: *Representing Meaning: An Artificial Intelligence Perspective*. W: Allen S. (red.) Text Processing. Proceedings of Nobel Symposium 51, Stockholm 1982, s. 25–63
[13] Schank R.C., Lebowitz M., Birnbaum L.A.: *Intergrated Partial Parsing*. Yale AI Project, Research Report # 143, 1978
[14] Schank R.C., Riesbeck C.K. (red.): *Inside Computer Understanding*. New Jersey, Hillsdale 1981

Recenzenci:
prof. dr hab. Mariusz Flasiński
prof. dr hab. Wiesław Lubaszewski