Pozyskiwanie specyfikacji systemów z opisów tekstowych

Zbigniew Huzar
Wydziałowy Zakład Informatyki, Politechniki Wrocławskiej
z.huzar@ci.pwr.wroc.pl

Marek Łabuzek
Wydziałowy Zakład Informatyki, Politechniki Wrocławskiej
labuzek @sun10.ci.pwr.wroc.pl

Streszczenie

W artykule przedstawiona została metoda generacji diagramów przypadków użycia i diagramów klas języka UML, które modelują rzeczywistość przedstawianą w opisie tekstowym. Jako etap pośredni tworzony jest model znaczenia opisu. Wykazana jest istotność użycia modelu znaczenia w zadaniu automatycznego pozyskania specyfikacji oraz zaproponowana postać tego modelu – system powiązanych ram.

1. Wstęp

Projektowanie systemów oprogramowania rozpoczyna się opisem a następnie analizą dziedziny, w której system funkcjonować. Pierwsze opisy mają charakter tekstowy. W podejściu obiektowym, np. [3], [4], na podstawie analizy takich tekstów próbuje się zidentyfikować podstawowe byty oraz zachodzące pomiędzy nimi związki. Istnieje szereg zaleceń i wskazówek pozwalających na wydobycie tych informacji i przedstawienie ich w postaci przynajmniej częściowo sformalizowanej, lepiej nadającej się do dalszego wykorzystania w procesie wytwarzania oprogramowania. Wydaje się, że proces wydobywania takiej informacji można – przynajmniej w pewnym zakresie – prowadzić automatycznie na podstawie analizy tekstu zapisanego w języku naturalnym. Wynikiem takiej analizy mogą być

diagramy przypadków użycia i diagramy klas, które są pierwszymi artefaktami zapisywanymi w języku UML [2] w procesie wytwarzania oprogramowania [4].

Artykuł przedstawia zarys algorytmicznej metody konstruowania diagramów klas i diagramów przypadków użycia na podstawie opisów tekstowych. Proponowany schemat postępowania składa się z dwóch etapów. W pierwszym etapie, na podstawie opisu tekstowego, generuje się reprezentację znaczenia tekstu w postaci zbioru powiązanych ram. W drugim etapie, na podstawie reprezentacji znaczenia, generuje się diagramy klas i przypadków użycia. W pierwszym etapie korzysta się z parsera, analizującego składnię kolejnych zdań tekstu. Sposób realizacji tego parsera jest dowolny, z tym że budowa drzew rozbioru składniowego powinna być zgodna z przyjętą gramatyką. Korzysta się także z bazy semantyki leksykalnej, którą można uważać za słownik języka z dodatkowymi informacjami o znaczeniu poszczególnych słów.

2. Modelowanie znaczenia

2.1. Wprowadzenie

Istnieją dwa główne podejścia do definicji znaczenia. Jedno z nich definiuje znaczenie jako warunki prawdziwości zdania, czyli odpowiedź na pytanie: *Jaki musi być świat, aby zdanie było prawdziwe?*. Drugie podejście zakłada, że znaczenie wypowiedzi jest ściśle związane z jej odbiorcą i jest mentalnym obrazem (fragmentu) świata wytwarzanym przez niego podczas odbioru danej wypowiedzi.

Proponowana metoda modelowania znaczenia wykorzystuje drugie podejście. Przemawia za tym odpowiedniość takiego podejścia do zadania generacji specyfikacji z tekstu opisu rzeczywistości, które jest w istocie procesem pozyskania wiedzy z tekstu, czyli zbudowania pewnej bazy wiedzy, będącą reprezentacją modelu fragmentu rzeczywistości opisywanego w tekście. Podejście „bazowiedzowe" do znaczenia pozwala w naturalny sposób zaimplementować system realizujący zadanie automatycznej generacji specyfikacji, a ponadto pozwala na gromadzenie doświadczeń. System taki przetwarzając tekst, tworzy jego model znaczenia, który jest obrazem opisywanej w tekście rzeczywistości, a następnie na podstawie tego obrazu generuje specyfikację w odpowiednim formacie.

2.2. Założenia modelu znaczenia

W celu modelowania obrazu świata opisanego w przetwarzanym tekście, należy określić, z jakich konstrukcji ten obraz może się składać, a więc także, jakie aspekty świata chce się uwzględniać. Zakłada się, że świat w pewnym konkretnym momencie składa się ze zbioru bytów fizycznych i abstrakcyjnych oraz zbioru zachodzących między nimi zależności (na przykład strukturalnych lub przestrzennych), które łącznie będą dalej nazywane konfiguracją bytów. Zakłada się również, że opisy rzeczywistości przedstawiają różne właściwości świata, na podstawie których można, m. in. tworzyć wyobrażenie o różnych możliwych jego konfiguracjach, czy też o możliwych przejściach między konfiguracjami. Własności te wyrażone są przez następujące rodzaje znaczeń.

Abstrakcyjne byty dziedzinowe

Abstrakcyjne byty dziedzinowe odnoszą się do klas bytów fizycznych i abstrakcyjnych występujących w danej dziedzinie. Każdy byt należący do abstrakcyjnego bytu dziedzinowego, dalej krócej – bytu abstrakcyjnego, nazywa się instancją bytu. Abstrakcyjne byty dziedzinowe mogą mieć różne własności, które mogą odnosić się do wartości atrybutów instancji (np. *duży*, jako wartość atrybutu *wielkość*, *dwuletni* – wartość atrybutu *wiek*) albo przeszłych lub teraźniejszych konfiguracji bytów (np. *oczekujący*, *sprzedany*). Byty abstrakcyjne mogą być proste lub złożone.

Relacje między abstrakcyjnymi bytami dziedzinowymi

Wyróżnia się trzy rodzaje relacji: specjalizację, powiązania i akcje. Abstrakcyjne byty dziedzinowe mogą być bardziej i mniej ogólne, co oznacza, że odnoszą się do większego lub mniejszego zbioru bytów. Relacja specjalizacji zachodzi między dwoma bytami abstrakcyjnymi, jeżeli jeden jest ogólniejszy niż drugi. Można też powiedzieć o specjalizacji lub o generalizacji danego abstrakcyjnego bytu dziedzinowego.

Powiązania odnoszą się do statycznych zależności występujących między instancjami bytów. Szczególnym rodzajem powiązania jest relacja agregacji zachodząca między bytem złożonym a bytami wchodzącymi w jego skład.

Akcje odnoszą się do zmian w konfiguracji bytów. Potraktowanie akcji jako relacji jest uproszczone – relacja uwzględnia byt abstrakcyjny inicjujący akcję, inne byty abstrakcyjne, które biorą udział w akcji, ale pomija czas trwania, przebieg oraz skutki akcji. Takie uproszczone rozumienie akcji występuje często w opisach rzeczywistości.

Relacja jest określona przez jej nazwę, rodzaj (specjalizacja, powiązanie czy akcja) oraz co najmniej dwóch uczestników relacji będących bytami abstrakcyjnymi. Dla akcji inicjujący byt abstrakcyjny jest pierwszym uczestnikiem relacji. Do każdego uczestnika relacji przypisane są dwie liczby naturalne określające minimalną i maksymalną liczbę instancji danego bytu abstrakcyjnego, które mogą

wchodzić w relację przy ustalonych instancjach pozostałych uczestników relacji. Oprócz tego do pierwszego uczestnika może być przypisana informacja, że jest on bytem złożonym, co oznacza również, że pozostali uczestnicy relacji są składowymi tego bytu.

Relacje czasowe między akcjami

Relacje czasowe między akcjami reprezentują cztery rodzaje zależności: *po akcji B zawsze zachodzi akcja A*, *podczas akcji B zawsze zachodzi akcja A* oraz *po akcji B może zajść akcja A* oraz *podczas akcji B może zajść akcja A*. Relacje *po* pozwalają na modelowanie następstwa akcji, które w opisach rzeczywistości występuje jawnie lub w postaci ciągu zdań opisujących kolejno wykonywany akcje. Relacje *podczas* pozwalają na modelowanie zależności między jednocześnie zachodzącymi akcjami, na przykład *Wybierając urządzenia, klient może przywołać wszystkie dane każdego z nich*.

Relacja między akcjami jest określona przez rodzaj (*po* albo *podczas*) i informację o tym, czy zachodzi zawsze, czy tylko może zajść, oraz przez dwa argumenty będące akcjami.

2.3. Model znaczenia jako system ram

Jako reprezentacja modelu znaczenia wybrana została baza wiedzy w postaci zbioru powiązanych ze sobą ram według idei Minsky'ego [7]. Wśród innych sposobów reprezentacji wiedzy (język predykatów lub sieci semantyczne), ramy wyróżniają się tym, że pozwalają w prosty sposób modelować system pojęć powiązanych relacjami.

Ważnym argumentem są też wspomniane w [5] problemy identyfikacji i reprezentacji abstrakcyjnych bytów dziedzinowych. Jednym z nich jest występowanie w tekstach fraz rzeczownikowych w różnych wariantach składniowych. Na przykład, frazy *urządzenia zapisane w rejestrze* oraz *zapisane w rejestrze urządzenia* opisują ten sam byt abstrakcyjny, ale różnią się kolejnością wyrazów. Podobnie frazy *urządzenia mające gwarancję* oraz *urządzenia, które mają gwarancję* opisują ten sam byt abstrakcyjny, ale różnią się użytą konstrukcją składniową. Dlatego nie można identyfikować i reprezentować bytów abstrakcyjnych używając tekstu frazy rzeczownikowej lub ich rozbioru składniowego. Rozwiązaniem jest użycie modelu znaczenia uwzględniającego złożoną strukturę fraz rzeczownikowych. Jest on także potrzebny do prawidłowego uwzględniania zjawiska anafory przez użycie skróconej nazwy bytu abstrakcyjnego tak, jak w poniższej parze zdań:

Sklep sprzedaje urządzenia elektryczne.
Każde urządzenie ma określoną nazwę i model. (1)

Z modelowaniem znaczenia uwzględniającym strukturę frazy rzeczownikowej wiąże się problem odróżnienia relacji, których źródłem są konstrukcje wewnątrz frazy rzeczownikowej, od pozostałych relacji. Na przykład, zdanie (2) opisuje klasę *urządzenie elektryczne mające numer seryjny* w kontekście, w którym następnym zdaniem jest (3).

Urządzenia elektryczne mające numer seryjny wpisane są do rejestru. (2)

Takimi urządzeniami są komputer, drukarka i monitor. (3)

Relacja *ma* między *urządzenie elektryczne mające numer seryjny* oraz *numer seryjny* ma swoje źródło w samej frazie rzeczownikowej i stanowi część definicji bytu abstrakcyjnego *urządzenie elektryczne mające numer seryjny*. Natomiast relacja *jest wpisane* między *urządzenie elektryczne mające numer seryjny* oraz *rejestr* nie ma takiego charakteru – jest pewną informacją o tym bycie abstrakcyjnym. Standardowa reprezentacja (4) znaczenia zdania (2) wykorzystująca język predykatów nie odzwierciedla tego faktu, gdyż odpowiada ona raczej zdaniu (5), w którym jest opisywany inny byt abstrakcyjny – *urządzenie elektryczne*.

$$\forall x \, (\, (urządzenie_elektryczne(x) \wedge \exists y \, (\, numer_seryjny(y) \wedge ma(x, y) \,)) \\ \Rightarrow \exists z \; wpisane_do(\, x, z) \,) \quad (4)$$

Jeżeli urządzenie elektryczne ma numer seryjny, wpisane jest do rejestru. (5)

Model znaczenia wykorzystujący ramy pozwala w prosty sposób dołączyć do uczestników relacji informację, czy stanowi ona część ich definicji.

Dla celów reprezentacji modelu znaczenia proponuje się, aby każda rama reprezentowała byt abstrakcyjny, relację między bytami abstrakcyjnymi lub relację między akcjami. Rama ma trzypoziomową budowę: składa się z ciągu rekordów, być może o różnej strukturze, każdy rekord zbudowany jest z kilku pól, natomiast pole ma nazwę i wartość. Nazwa pola jest napisem, natomiast wartościami pól mogą być napisy lub referencje do ram, które są unikalnymi identyfikatorami ram. Realizacja systemu ram jest stosunkowo łatwa. Po opracowaniu dokładnej budowy ram, pozostaje praca czysto programistyczna. Przykładem istniejącego systemu wykorzystującego ramy jest *GUS* [1], który pozwala na zamawianie usług lotniczych przez prowadzenie dialogu w języku naturalnym z systemem.

3. Tworzenie modelu znaczenia

Model znaczenia jest tworzony dla kolejnych zdań tekstu opisu rzeczywistości. Przetwarzając zdanie, najpierw analizuje się jego składnię, która

w wyniku daje drzewo jego rozbioru składniowego. Opisuje ono strukturę fraz w zdaniu. Następnie, na podstawie takiego drzewa tworzony jest model znaczenia danego zdania.

Moduł analizy składni zwany parserem, może zostać zrealizowany w dowolny sposób spełniający dwa następujące wymagania. Po pierwsze – powinien uwzględniać konstrukcje składniowe występujące w przetwarzanych tekstach. Repertuar tych konstrukcji należy opracować na podstawie gramatyki opisowej języka (np. [9] dla języka polskiego) oraz korpusu przykładowych tekstów. Dla celu przetwarzania opisów rzeczywistości należy uwzględnić następujące konstrukcje:

1. określenia rzeczownika
   - przymiotniki (*duży*, *końcowy*, *elektryczny*)
   - imiesłowy przymiotnikowe (*sprzedany*, *oczekujący*, *zapisany w rejestrze*)
   - fraza rzeczownikowa w dopełniaczu (*numer seryjny komputera*)
   - zdanie przydawkowe (*urządzenia*, *które mają gwarancję*)

2. szeregi fraz rzeczownikowych połączonych przecinkami i spójnikami szeregowymi (*nazwa, model i cena*)

3. określenia czasownika i jego derywatów (imiesłowów i odsłowników)
   - frazy rzeczownikowe (*klient wybiera urządzenia*)
   - frazy przyimkowe (*zapisywanie w rejestrze*)
   - fraza przymiotnikowa (*gwarancja jest przypisana do konkretnego urządzenia*, *każda transakcja zostaje zapisana w rejestrze*)
   - fraza czasownikowa – dla czasownika „móc” (*urządzenie może mieć gwarancję*)

4. frazy okolicznikowe
   - imiesłowy przysłówkowe współczesne (*wybierając urządzenia*)
   - wyrażenia przyimkowe z przyimkiem „po” lub „podczas” oraz odsłownikiem (*po zapisaniu transakcji*, *podczas wybierania urządzeń*)

5. zdania pojedyncze w trybie oznajmującym, czasie teraźniejszym, trzeciej osobie (*Urządzenia mają numer, nazwę i cenę*.).

Po drugie – drzewa tworzone przez parser powinny mieć budowę zgodną z ustaloną formalną gramatyką języka. Dla języka angielskiego przykładem takiej gramatyki jest *XTAG* [11]. Dla języka polskiego nie istnieje podobne opracowanie. Formalnym ujęciem polskiej gramatyki, które może stanowić podstawę takiego opracowania jest [10]. Sam parser można napisać w Prologu [8]. W chwili pisania artykułu istnieje działający prototyp takiego parsera wykorzystujący formalną gramatykę fragmentu języka polskiego obejmującą wyżej wymienione konstrukcje.

Podczas tworzenia modelu potrzebna jest także baza semantyki leksykalnej. Przechowuje ona informacje o semantyce różnych wyrazów. Dla rzeczowników przechowywana jest informacja, czy opisują osobę, natomiast dla czasowników – informacja czy czasownik opisuje akcję czy powiązanie. Dla tych ostatnich dodatkowo pamiętane jest też czy czasownik opisuje relację specjalizacji, agregacji czy inną relację. Informacje tego typu można znaleźć dla języka angielskiego na przykład w bazie *WordNet* [6].

Tworzenie modelu znaczenia na podstawie drzewa rozbioru składniowego zdania polega na jego przeglądaniu w poszukiwaniu wymienionych wyżej konstrukcji i tworzeniu dla każdej z nich różnych elementów modelu znaczenia: nowych ram oraz rekordów i pól w już istniejących ramach.

Dla fraz rzeczownikowych, także tych, które są częścią innej frazy rzeczownikowej, tworzone są byty abstrakcyjne. Na przykład *urządzenie elektryczne mające numer seryjny* opisuje dwa byty abstrakcyjne: odpowiadający całej frazie oraz frazie *numer seryjny*. Wyjątkiem są tu frazy typu *dane urządzenia*, *informacje o kliencie*, dla których nie jest generowany byt abstrakcyjny dla rzeczownika *dane* i *informacje*, a jedynie dla frazy rzeczownikowej podrzędnej. Na przykład *dane urządzenia* są utożsamiane z bytem abstrakcyjnym *urządzenie*, a *informacje o kliencie* – z bytem *klient*.

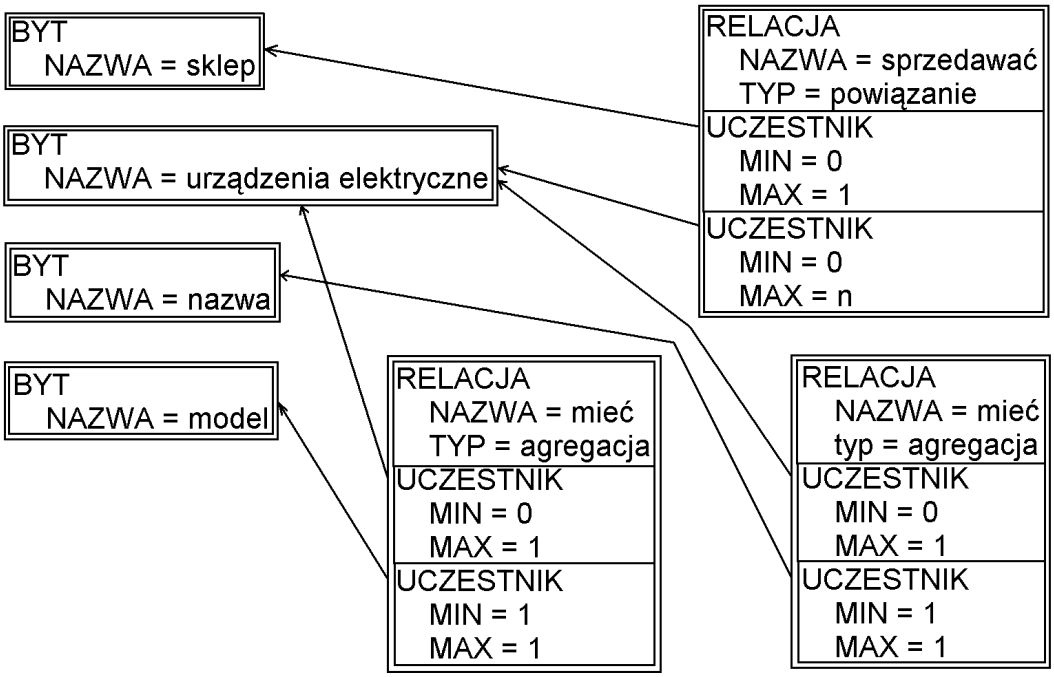
Dla przymiotników i imiesłowów przymiotnikowych bez dodatkowych określeń są tworzone właściwości odpowiednich bytów abstrakcyjnych. Dla imiesłowu *oczekujące* w frazie *zamówienie oczekujące* zostanie utworzona właściwość bytu *zamówienie*.

Dla pozostałych rodzajów określeń rzeczownika, określeń czasownika i jego derywatów, imiesłowów przysłówkowych oraz zdań tworzona jest relacja. Dla każdej konstrukcji ustalona jest kolejność uczestników relacji utworzonych przy przetwarzaniu danej konstrukcji, na przykład dla zdania pierwszym uczestnikiem staje się podmiot zdania. Dla każdej relacji ustalany jest jej rodzaj (specjalizacja, powiązanie czy akcja). Pierwszym wyznacznikiem jest rodzaj związku składniowego, dla której relacji została utworzona. Akcje są tworzone tylko dla zdania (*Klient wybiera urządzenia*), imiesłowu przysłówkowego współczesnego (*wybierając urządzenia*) oraz odsłownika (*wybranie urządzenia*). W tych przypadkach jest sprawdzane czy w bazie semantyki leksykalnej, istnieje informacja o rodzaju czasownika. Jeżeli jest to czasownik akcji, tworzona relacja jest akcją, jeżeli czasownik statyczny – relacja jest powiązaniem lub specjalizacją. Jeżeli w bazie nie ma informacji o danym czasowniku sprawdzany jest pierwszy uczestnik relacji: jeżeli odpowiada on osobie lub całemu systemowi, to relacja będzie akcją, w przeciwnym razie – powiązaniem lub specjalizacją. Jeżeli relacja nie jest akcją, to dodatkowo sprawdzany jest w bazie semantyki leksykalnej rodzaj relacji, jaką opisuje czasownik. Jeżeli opisuje specjalizację, to dla pierwszego uczestnika relacji jest zapamiętywana informacja o jego generalizacji, będącej drugim uczestni-

kiem relacji, a sama relacja jest usuwana. Jeżeli natomiast opisuje agregację, to przy pierwszym uczestniku zapamiętywany jest ten fakt.

Dla zdań zawierających frazy okolicznikowe utworzona zostanie relacja między akcjami. Na przykład dla zdania *Wybierając urządzenia, klient może przywołać wszystkie dane każdego z nich*. zostanie utworzona relacja między *wybieranie urządzenia* a *przywołanie danych urządzenia*.

Na przykład dla zdań (1) utworzone zostaną następujące ramy:



4. Generacja diagramów

Generacja diagramów przypadków użycia i klas odbywa się w dwóch krokach. W pierwszym przetwarzane są kolejno wszystkie ramy relacji między bytami abstrakcyjnymi. W drugim przetwarzane są ramy relacji między akcjami. Dzięki wykorzystaniu modelu znaczenia w przetwarzaniu tekstu oraz dostowaniu jego zawartości i formy do generowanych diagramamów, algorytm generacji jest prosty.

Przetwarzanie relacji w pierwszym kroku generacji zależy od jej rodzaju. Dla akcji generowany jest aktor, którego nazwą jest nazwa bytu abstrakcyjnego inicjującego akcję, przypadek użycia o nazwie będącej rzeczownikowym opisem akcji oraz relacja między wygenerowanym aktorem i przypadkiem użycia. Na przykład dla zdania *Klient wybiera urządzenia* wygenerowane zostaną aktor *klient*, przypadek użycia *wybieranie urządzenia* oraz relacja między nimi. Dla

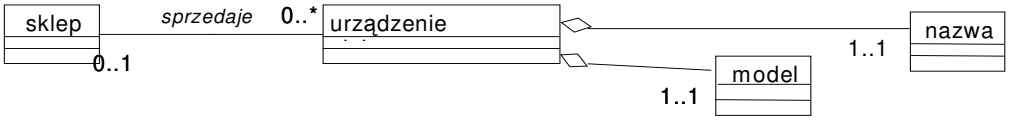
powiązania, najpierw są przetwarzani wszyscy uczestnicy relacji. Jeżeli dla danego uczestnika nie została jeszcze wygenerowana klasa, jest ona generowana. Jej nazwą jest nazwa uczestnika i, jeśli istnieje generalizacja uczestnika, to klasa jest specjalizacją klasy wygenerowanej dla generalizacji tego uczestnika. Następnie jest generowana relacja między klasami wygenerowanymi dla uczestników relacji. Dla każdego uczestnika generowana jest informacja o krotności przy klasie wygenerowanej dla niego oraz ewentualne oznaczenie agregacji.

W drugim kroku generacji diagramów na podstawie relacji między akcjami generowane są relacje między przypadkami użycia. Jeżeli relacja zachodzi zawsze, generowana jest relacja «uses» od pierwszego do drugiego argumentu. Jeżeli relacja nie zachodzi zawsze, generowana jest relacja «extends» od drugiego do pierwszego argumentu. Na przykład dla zdania (6) wygenerowana zostanie relacja «uses» od przypadku *przyjmowanie towaru* do *zapisywanie danych dostawcy*, a dla zdania (7) – relacja «extends» od przypadku *przywołanie urządzenia* do *wybieranie urządzenia*.

Podczas przyjmowania towaru magazynier zapisuje dane każdego dostawcy. (6)

Wybierając urządzenia, klient może przywołać wszystkie dane każdego z nich. (7)

Dla zdań (1) zostanie wygenerowany poniższy diagram.



5. Podsumowanie

Opisana metoda automatycznego generowania specyfikacji diagramu przypadków użycia i diagramu klas na podstawie analizy tekstowego opisu rzeczywistości daje się w pełni zautomatyzować. Parser analizujący składnię tekstu dla ustalonej gramatyki jest łatwo realizowalny w Prologu, natomiast tworzenie modelu znaczenia oraz generacja diagramów opisane są przez zestaw reguł, który może zostać zapisany w postaci dokładnego algorytmu.

Istotnym elementem metody jest wykorzystanie reprezentacji znaczenia. Pozwala ono uwzględnić ważne zjawiska językowe oraz uprościć analizę tekstu. Za sposób reprezentacji znaczenia został wybrany system powiązanych ram. Format ten umożliwia uwzględnienie specyficznych aspektów znaczenia opisów rzeczywistości.

Podstawowym kierunkiem rozwoju przedstawionej metody jest uwzględnienie większej liczby zjawisk językowych zarówno składniowych jak i semantycznych, na przykład związanych z anaforą. Ważnym byłaby też możliwość wykrywania występujących w tekście wieloznaczności oraz niespójności semantycznych. Stowarzyszonym kierunkiem rozwoju są także prace związane z językiem polskim: opracowanie formalnej gramatyki polskiej, parsera oraz bazy semantyki leksykalnej.

Bibliografia

[1] Bobrow D. G., Kaplan R., Kay M., Norman D., Thompson H. Winograd T. *GUS - A frame driven dialog sysytem*. Palo Alto, California Xerox Research Center, 1976

[2] Booch G., Jackobson I., Rumbaugh J. *The Unified Modeling Language Reference Manual*. Addison-Wesley, 1999

[3] Górski J. (red.) *Inżynieria oprogramowania w projekcie informatycznym*. Mikom, 1999

[4] Jacobson I., Booch G., Rambough J. *The Unified Software Development Process*. Addison-Wesly, 1999

[5] Łabuzek M. *Automatyczne generowanie specyfikacji na podstawie opisu rzeczywistości*. Materiały KKIO 2000, s. 85-92, 2000

[6] Miller G. A. i inni. *Introduction to WordNet: An On-line Lexical Database* w *Five Papers on WordNet*. Princeton University, 1993

[7] Minsky M. *A Framework for Representing Knowledge* in: *The Psychology of Computer Vision* (P. H. Winston, ed.). Mc Craw-Hill Computer Science Series, 1975

[8] Pereira F. C. N., Shieber S. M. *Prolog and Natural-Language Analysis*. Stanford University, 1987

[9] Saloni Z., Świdziński M. *Składnia współczesnego języka polskiego*. PWN, 1998

[10] Szpakowicz S. *Formalny opis składniowy zdań polskich*. Wydawnictwo Uniwersytetu Warszawskiego, 1983

[11] The XTAG Research Group, *A Lexicalized Tree Adjoinig Grammar for English*. www.cis.upenn.edu/~xtag, Univeristy of Pennsylvania, 1999