РУСЕНСКИ УНИВЕРСИТЕТ ‘АНГЕЛ КЪНЧЕВ‘ РУСЕ

Факултет природни науки и образование

Катедра Математика

АВТОРЕФЕРАТ

Илияна Петрова Раева

За присъждане на образователна и научна степен ’доктор’ по научна специалност Математическо моделиране и приложение на математиката

на тема:

Математически модел на лингвистичен анализатор на естествено-езиков текст

Научен ръководител: проф. д-р Радослав Павлов
Институт по Математика и Информатика
Българска Академия на Науките

гр. Русе, 2013 г.

Обща характеристика на дисертацията

Актуалност

Разработването на математически модел на естествен език е актуална задача на съвременната теория на програмирането, информатиката и лингвистиката. Естественият език (ЕЕ) е сложна система от думи и изречения, които могат да се съставят от думите с помощта на определени правила, наречени граматика на езика. Крайната цел на анализа е да се определи смисъла на изречението. Независимо от съвременните научни постижения в това направление и развитието на информационните технологии, все още не са решени напълно проблемите за разбиране на ЕЕ и проблемите, свързани със семантиката на езика.

Компютърната обработка на естествено-езиков текст е тясно свързана с необходимостта от изграждане на модел за генериране на вътрешно представяне на семантиката на всяко изречение, което предстои да бъде обработвано.

От казаното по-горе може да се направи извода, че изследването на ЕЕ като цяло е твърде сложна задача. Имитирането на човешкото мислене от компютъра изисква процес на разработване на сложни математически модели и алгоритми, работа на големи екипи и скъпи проекти.

Основните резултати, свързани с представената тематика, биха могли да намерят приложение в областта на автоматизирания превод, диалогови системи, експертни системи, интелигентни системи за обучение, автоматизирана обработка на текстове и други.

Цел и задачи на дисертацията

Основната цел на дисертацията е да се разработи математичен модел на лингвистичен анализатор, приложим за решаване на определени типове приложни задачи за анализ на фрагменти от естествено - езиков текст.

Формулираната основна цел е постигната чрез изпълнението на следните конкретни задачи:

- Да се разработи алгоритъм за дедуктивен извод от естестено-езиков текст;
- Да се разработи алгоритъм за синтактичен анализ на текст с представяне на синтактично дърво на изреченията;
- Да се разработи алгоритъм за семантичен анализ на изречение на базата на зададени стратегии за семантична интерпретация и метод за построяване на логична форма на изречение;
- Като резултат от разработените алгоритми да се направи програма за дедуктивен извод и логически анализ на текст на синтактично и

семантично ниво;

- Да се тестват отделните модули на програмата с подходящо избрани примери с цел доказване на нейната ефектиност.

Структура на дисертационния труд

Разработеният в настоящата работа математически модел на лингвистичен анализатор на ЕЕ извършва автоматизиран анализ на фрагменти на естествено-езиков текст, преминаващ през различните нива на изследване: логика, синтаксис, семантика. Дисертационният труд се състои от увод, четири глави, основни изводи и резултати;

Първа глава на дисетрацията има обзорен характер:
- Дава представа за същността на моделирането на ЕЕ;
- Разгледани са основните съвременни математически модели: формални семантики, теоретико - множествени, игрови, формално-логически;
- Дефинирани са понятията "лингвистична променлива"и "размита функция на лингвистична променлива";
- Представен е метод на изчисляване на лингвистични вероятности на неопределени ситуации.

Във **втора глава** е реализирана програма за дедуктивен извод от естествено езиков текст. Програмата извършва превод на даден текст(задача) във вид на система от дизюнкти. Чрез интерпретация в семантична резолюция дава отговор на верността на дадено твърдение. Програмата използва правилата за извод от математическата логика.

За успешното реализиране на програмата:
- разработена е система от алгоритми за дедуктивен извод от текст, написан на български език;
- разгледан е принципът на резолюцията на Ербран за логически твърдения и неговите модификации;
- представен е теоретичният модел на системата като съвкупност от алгоритми и правила за извод;
- разработени са два тестови примера;
- направено е описание на системата: концептуален модел, структура, реализация.

Трета глава представя модел на система за автоматично задаване на коректен синтактичен анализ на изречения и за определени фрази, представяни в логична форма, решава някои семантични неопределености.

В **четвърта глава** са показани някои основни принципи на моделиране на семантика на ЕЕ:
- разработен е модул от алгоритми за решаване на задачи за превод на

синтактично и семантично ниво;
-представени са съвременни алгоритми за решаване на семантична неед-нозначност чрез използване на атрибутивно стойностни матрици;
- модулът съдържа система от интерпретационни правила за анализ на определени задачи с ограничения;
- описан е математическият модел и логическата структура на система за задаване на семантика на думи, изрази и изречения;
- на основата на коректна синтактична структура, зададена чрез безкон-текстна граматика е направена семантична интерпретация на неопреде-лени твърдения и фрази представени в логична форма.

Основните резултати и приноси са обобщени в края на дисертацион-ния труд.

Моделът, представен в настоящата работа се базира на дългогодишен труд и работа в областта на компютърната лингвистика, при която се е налагало решаването на множество приложни задачи за анализ на текст. Направен е опит за обобщение и разработване на сравнително пълна система от алгоритми.

Разработените програми, фрагменти от програми и алгоритми са на-мерили вече своята реализация в практически и приложни задачи при реализиране на научни проекти, задачи от областта на математическата лингвистика. Като тестов пример е представено решение на логическа задача, давана на олимпиада по математическа лингвистика.

Получени резултати

Постигнатите резултати в настоящата работа покриват поставените предварително цели и задачи.

Постигната е основната цел, а именно разработването на математи-чески модел на лингвистичен анализатор за решаване на лингвистични задачи.

Анализи и обзори

1) Направен е подробен обзор и анализ на основните области на изслед-ване, свързани с представената тематика. Дефинирана е целта, актуал-ността на проблема и поставените за изпъленение задачи в дисертационния труд.
2) Разгледани са основните видове съвременни математически модели : формални семантики, теоретико - множествени, игрови, формално логи-чески;
3) Дефинирани са основните понятия и теореми от:
- теорията на размитите множества и функции;
- пропозиционалната и предикатна логика;
- теория на множествата.

4) Представен е метод на изчисляване на лингвистични вероятности на неопределени ситуации.

Разработени алгоритми:
1) За дедуктивен извод от естествено-езиков текст на български език
2) За графично представяне на синтаксиса на обработваните изречения
3) За задаване на семантика на думи, фрази и изрази, представяне AVM матрици и частично решаване на семантична нееднозначност.

Теоретични модели
1) Разработен е теоретичен модел на система за дедуктивен извод от естествено - езиков текст.
2) Модел за синтактичен анализ
3) Модел за задаване на семантика

Практическа реализация на алгоритмите
Проектирана и разработена като приложна част от настоящата работа е софтуерна програма, която реализира разработените алгоритми и която се състои от три основни модула, някои от които са достатъчно общи за да бъдат ценни от практическа гледна точка.
1) Модул за дедуктивен извод от естествено - езиков текст. Програмата трансформира даден текст във вид на система от дизюнкти. Чреа интерпретация в семантична резолюция дава отговор за верността на дадено твърдение.
2) Модул за задаване на коректен синтактичен анализ.
3) Модул за решаване на семантична нееднозначност.

Тестови примери
С цел, доказване на ефективността на програмата са разработени два тестови примера.

Приноси

Дисертацията представлява изследване в интердисциплинарната област - математическа и компютърна лингвистика. В работата на дисертацията са използвани методи и средства от теорията на размитите множества и функции, областта на логиката, теорията на графите, методи за анализ на естествен език и др.

Научни приноси

-Разработен е теоретичен модел на програма, която използва правилата за извод в пропозиционалната логика и чрез метода на отричане на

целта доказва дали дадено твърдение (цел) е истина или не. Уникалното в програмата е, че работи на български език.

- Разработен е метод на изчисляване на размити лингвистични вероятности на неопределени ситуации, като допълнение на метода за изчисляване на вероятности от класическата теория на моделирането(Глава 1.4). Методът е в основата на формализацията на приближени разсъждения, описанието на размити алгоритми, в които често е удобно да се определят отношенията между променливите и да им се присвоят конкретни значения с помощта на блок схеми.
- В обща теоретична рамка са формализовани три алгоритмични процедури, които са изцяло оригинални(Глава 2.4).

Приложни приноси

* Реализирана е програма, която използва правилата за извод от пропозиционалната логика и чрез метода на отричане на целта доказва дали дадено твтрдение (цел) е истина или не. Софтуерното приложение се състои в три форми, всяка от които визуализира резултатите получени посредством отделните алгоритми.

- Разработени са три алгоритъма. (2.4 и 3.2)
- Разработена е система от правила за извод .(2.2)
- Програмата е тествана с три примера: (2.4)

* Разработена е програма за обработка на свързан текст и задаване на семантика. програмата съдържа 16 основни и два рекурентни модула(3.2)

Публикации на дисертацията

[1] Раева И, "Приложение на размитите функции на лингвистични променливи за семантичен анализ на нееднозначни твърдения, записаи на естествен език", СУБ - Русе, "Математика, информатика и физика" Том 1 Серия 5, стр. 69-74, 2001;

[2] Раева И., Р. Раев, Метод на изчисляване на лингвистични вероятности и прилагането им към лошо определени ситуации, Научни трудове РУ, т. 48, серия 6.1, стр. 45-49, 2009

[3] Раева И. , "Система за моделиране на нееднозначни семантики" ИЗВЕСТИЯ на СУБ - Русе, серия - Математика, информатика и физика, ISSN 1311- 9184, 2013;

[4] Raeva I., Fuzzy Functions of Linguistic Variables and Their Application for Semantic Analysis of Ambiguity Mathematical Statements Written in Natural Language, University of Rousse, 2000;

[5] Raeva I. , L. Petrkieva, System for Generating Natural Language

Semantics of a Simple Sentence, Proceedings of the 14-th International Conference SAER'2000;

[6] Raeva I., R. Raev, D. Stankovsli, "Computer System for Issue from Natural Language Text through Interpretation in Semantic resolution, iCENT, Ohrid Macedonia, 2007.

Перспективи

- Реализираната система за дедуктивен извод за естествено-езиков текст на български език, може да се използва като система за вземане на решения, експертна система, както и като система за обучение;
- Предложеният подход за автоматична обработка на изречения предоставя богати възможности за разработка и самостоятелност по време на проектирането на базите от данни и програмирането на C++ модулите. Системата има възможност за разширяване на обхвата на изреченията и увеличаване на тяхната сложност;

Съдържание на дисертацията

Увод

В увода е направен обзор на основните области на изследване, свързани с представената тематика. Поставени са конкретната цел и задачи на дисертацията и е направен кратък преглед на съдържанието.

Глава I

Същност на моделирането на естествен език. Основни понятия и теореми. Формални модели

Математическият модел описва реално съществуващ обект/феномен с апарата на математиката, а самият процес на описание се нарича математическо моделиране. Въпреки неоспоримите предимства, математическият модел има и своите недостатъци, като например факта, че той представя само една или няколко страни от обекта и в този смисъл той не е пълен. От друга страна, създаването на опростено представяне на сложни системи позволява те да бъдат анализирани и изследвани с количествени методи и характеристики. Целта е да се построи лесно разбираем и удобен за потребителите модел, който да позволява ефективен и прецизен анализ.

1.1 Съвременни математически модели

Логическата структура, задаваща синтаксиса на формален език е следната:

• Дефинират се две множества

$\underleftrightarrow{M}$ - множеството, състоящо се от краен или безкраен брой елементи, наречени атоми (или терми);

$\underleftrightarrow{V}$ - крайно множество от логически връзки и операции.

• От множеството от формули T= $\underleftrightarrow{M} \bigcup \underleftrightarrow{V}$ се отделя подмножество $\underleftrightarrow{P}$ от правилно построени формули (ППФ). Дефинира се множество $\underleftrightarrow{H}$ от правила за конструиране на ППФ, т.е. определя се ефективна процедура, която позволява да се изясни дали даден израз е ППФ.

• От множеството ППФ се определя подмножество A, чиито елементи са наречени аксиоми. Понятието аксиома, също трябва да бъде ефективно.

• Съществува крайно множество R=$\{R_1, R_2, ..., R_n\}$ от релации, наречени правила за извод. Отново е необходимо да съществува ефективна процедура, която да определя може ли всеки член на произволна редица от ППФ да бъде изведен от една или няколко предшестващи ППФ на тази редица посредством няколко от фиксираните правила.

Чрез така дефинираните множества, дадена формална система (FS) се представя като наредената четворка от множества:

$$\lhd \text{ T, H, A, R } \rhd$$

където T= $M \bigcup V$

1.2 Основни видове семантики

Да се дефинира семантика на ФЕ означава да се дефинира семантика на всяка ППФ във ФЕ, съответстваща на дадена граматика. Това се осъществява чрез интерпретатация на всяка ППФ от дадения език, при която на всяка променлива се задава стойност от дадена област на интерпретация.

В работата са разгледани следните модели:
- Теоретико-множествен;
- Функционално семантичен;
- Игрови;
- Логически.

<u>Понятие за лингвистична променлива</u>

Лингвистичните променливи се използват като средство за приближено описание на явленията, които са твърде сложни или некоректно определени и не се поддават на точно описание.

. *Лингвистичната променлива се дефинира чрез характеристично множество:*

$$P=<X,\ U,\ R(X,\ u)>$$

където
X - име на променлива;
U - универсално множество (крайно или безкрайно) за X;
u $\in U$ *(u -общо название на елементите на U)*
R(X, u) $\subset U$ *- ограничение за значенията на X.*

. *Нека* $X_1, X_2, ..., X_n$ *са универсални множества съответно* $U_1, U_2, ..., U_n$. *Наредената n-торка от променливи X=<* $X_1, X_2, ..., X_n$*> се нарича n-арна съставна променлива . Декартовото произведение U=* $U_1 x U_2 x..x U_n$ *- универсално множество на X при ограничение R(*$X_1, X_2, ..., X_n$*) в U.*

Това отношение може да се определи чрез характеристичната функция:
$\mu_R : U_1 x U_2 x..x U_n \to (0, 1)$, където

$$\mu_R(u_1, u_2, \ldots, u_n) = \begin{cases} 1 & \text{ако}(u_1, u_2, \ldots, u_n) \in R(X_1, X_2, ..., X_n), \\ 0 & \text{ако}(u_1, u_2, \ldots, u_n) \notin R(X_1, X_2, ..., X_n). \end{cases} \quad (1)$$

Размити множества и функции на лингвистични променливи

. *Нека U е универсално множество. Размито множество M над U се задава чрез функцията на принадлежност*
$\mu_A : U \to (0, 1)$, *която съпоставя на всеки елемент* $u \in U$ *число* μ_x*(u) от интервала [0, 1], характеризиращо степента на принадлежност на елемента u на множеството M.*

Въпреки, че затвореният интервал [0, 1] е най - често използваното множество за дефиниране на функция на принадлежност на размито множество, в някои случаи се използват и други множества с частична или пълна наредба. Не се изисква елементите на тези множества да бъдат непременно числа. Те могат да се интерпретират като елементи, представящи различна сила на степен на принадлежност. Тази обобщена функция се записва във вида: $\mu_A : X \to L$, където L е произволно множество, което е поне частично наредено. Поради това, че най-често

L е мрежа, то множествата, дефинирани чрез тази обобщена функция са наречени L - размити множества(LRS). LRS са важни за някои приложения.
Въпреки, че множеството [0, 1] е пълно наредено, то множеството $[0,1]^n$ за някои $n \geq 2$ са само частично наредени.

1.3 Метод на изчисляване на лингвистични вероятности и прилагането им към лошо определени ситуации

Предложеният в настоящата глава подход за изчисляване на размити вероятности може да се разглежда като допълнение и развитие на класическата теория от гледна точка на лингвистичния модел на описание на тази теория.[30]

1.4 Лингвистични вероятности

Нека е зададена променливата X над крайно универсално множество:

$$U = u_1 + u_2 + ... + u_n$$

Ще предполагаме, че ограничението, обусловено от X съвпада с U. С други думи, всяка точка от U може да бъде избрана за значение на променливата X. На всеки елемент u_i, i=1, 2, ..., n ще съпоставим лингвистична вероятност P_i - булева лингвистична променлива. Променливата p_i, $0 \leq p_i \leq 1$ е базова променлива за P_i. За определеност ще предположим че Универсалното множество V, съответстващо на P_i, представлява само по себе си или интервала [0, 1] или крайно множество.
V = 0+0.1+0.2+...+0.9+1
Важно свойство на лингвистичните вероятности $P_1, P_2, ..., P_n$ е, че те са β взаимодействащи в смисъла на [13]. Взаимодействието между P_i е следствие на ограничението ($+\otimes$- аритметична сума)

$$p_1 + p_2 + ... + p_n = 1 \tag{2}$$

където p_i са базови променливи (т.е. числови вероятности), свързани с P_i. По-конкретно нека R($p_1 + p_2 + ... + p_n = 1$) означава размито n-арно отношение в $[0,1]x...x[0,1]$, представляващо (1.4). Нека освен това RP_i е

ограничение на променливата P_i. Тогава ограничението, определено от n -арната размита променлива $P_1, P_2, ..., P_n$, може да се запише във вида: R($P_1, P_2, ..., P_n$) = R(P_1)x R(P_2)x...x R(P_n) $\bigcap R(p_1+p_2+...+p_n = 1)$ Оттук следва, че без ограничение (1.3) размитите променливи $P_1, P_2, ..., P_n$ биха били невзаимодействащи.

1.5 Изчисления с лингвистични променливи

В много случаи приложенията на теорията на вероятностите за изчисление на средни стойности (математическо очакване и дисперсия), често се използва линейната комбинация от вида: ($+\otimes$ - аритметична сума) [39]:

$$z = a_1p_1 + a_2p_2 + ... + a_np_n \tag{3}$$

където a_i са реални числа, а p_i са стойности на вероятности от интервала [0, 1]. При така зададените a_i и p_i изчисляването на комбинацията z не представлява трудност. Това изчисляване обаче изглежда нетривиално, когато разглежданите вероятности са лингвистични по своята природа т.е. когато:
z= $a_1P_1 + a_2P_2 + ... + a_nP_n$, където P_i са такива лингвистични значения на вероятностите, като "правдоподобно "неправдоподобно "много правдоподобно "близо до"и др. Съответно z не е реално число, както в (1.5), а размито подмножество на реалната ос W$\otimes$(-$\propto, \propto$). В предположението, че размитите множества $P_1, P_2, ..., P_n$ са не взаимодействащи, ограничението придобива вида (1.4).

Нека $\mu(p_1, p_2, ..., p_n)$ е функция на принадлежност на ограничението R($P_1, P_2, ..., P_n$). Тогава като приложим принципа на обобщението и (1.5), z може да се изрази във вид на размитото множество ($+, \otimes$ - аритметична сума).

$$\int_W \frac{\mu(p_1, p_2, ..., p_n)}{a_1p_1 + a_2p_2 + ... + a_np_n} \tag{4}$$

което може да се запише по следния начин:

$$\int_W \frac{\mu(p_1)x\mu(p_2)x...x\mu p_n)}{a_1p_1 + a_2p_2 + ... + a_np_n} \tag{5}$$

При това в последния интеграл p_i удовлетворяват ограничението:
$p_1 + p_2 + ... + p_n = 1$, По такъв начин може да се представи линейната комбинация на стойностите на лингвистичните вероятности на размити множества на реалната ос . Изразът за Z може да се запише по друг

по-удобен за ичисление начин. Тъй като $\mu(z)$ е функцията на принадлежност на множеството Z и z$\in$ W. Тогава от (1.7) следва, че

$$\mu(z) = \underbrace{V}_{p_1,p_2,\ldots,p_n} \mu(p_1)x\mu(p_2)x...x\mu p_n) \tag{6}$$

При ограничения
Z= a_1p_1+a_2p_2+...+a_np_n и p_1+p_2+...+$p_n = 1$. В този случай изчисляването на Z се свежда до нелинейното програмиране с линейни ограничения. По-точно тази задача може да се формулира по следния начин: Да се максимализира Z при следните ограничения: (+, $\otimes$ - аритметична сума):

$$\mu_1(z_1) \geq z$$
$$\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots$$
$$\mu_n(z_n) \geq z \tag{7}$$

$$\text{Z}= a_1p_1 + a_2p_2 + ... + a_np_n$$
$$\text{и } p_1 + p_2 + ... + p_n = 1$$

За илюстриране на направените по-горе разсъждения може да се разгледа следният пример: Нека са дадени събитията:
P_1 = "правдоподобно"
P_2 ="неправдоподобно"
където:

$$\text{„правдоподобно“} = \int_0^1 \frac{\mu_{\text{правдоподобно}}(p)}{p} \tag{8}$$

"неправдоподобно- $\neg$ "правдоподобно"
Тогава

$$\mu_{\text{правдоподобно}}(p) = \mu_{\text{неправдоподобно}}(1-p), 0 \leq p \leq 1 \tag{9}$$

Да предположим, че искаме да изчислим математическото очакване:
(+, $\otimes$ аритметична сума) във вида
Z = a_1правдоподобно + a_2 неправдоподобно
Откъдето получаваме:
$\mu(z) = \underbrace{V}_{p_1,p_2,\ldots,p_n} \mu_{\text{правдоподобно}}(p_1)$ x $\mu_{\text{неправдоподобно}}(p_2)$
$p_1 + p_2 = 1$
Следователно:

$\mu_{\text{правдоподобно}}(p_1) = \mu_{\text{неправдоподобно}}(p_2)$ и следователно (1.11) се свежда до:
$\mu(z) = \mu_{\text{правдоподобно}}(p_1)$
Z= $a_1p_1 + a_2(1 - p_2)$
Или в по-явна форма:

$$\mu(z) = \mu_{\text{правдоподобно}}(\frac{z - a_2}{a_1 - a_2}) \tag{10}$$

От този резултат следва, че определено знание за вероятността n_1 води до съответната размитост на математическото очакване Z= $a_1p_1 + a_2(1 - p_2)$. Ако предположим, че универсалното множество от стойности на вероятности има V=0+0.1+0.2+...+0.9+1, то изразът за z се получава непосредствено като се използва принципът на обобщението във вида (1.4).
Да предположим:

$$P_1 = 0.3 = 0.8/0.2 + 1/0.3 + 0.6/0.4 \tag{11}$$

$$P_2 = 0.7 = 0.8/0.6 + 1/0.7 + 0.6/0.8 \tag{12}$$

и

$$(+, \otimes -) : Z = a_1p_1 \otimes a_2(p_2) \tag{13}$$

където символът $\otimes$ се използва за избягване на неяснотата със значение - обединение. Като заместим (1.16) и (1.17) в (1.18) получаваме:

$$Z = a_1(0.8/0.2 + 1/0.3 + 0.6/0.4) \otimes a_2(0.8/0.6 + 1/0.7 + 0.6/0.8)$$

$$= ((0.8/0.2)a_1 + (1/0.3)a_1 + (0.6/0.4)a_1) \otimes ((0.8/0.6)a_2 + (1/0.7)a_2 + (0.6/0.8)a_2) \tag{14}$$

След разкриване на скобите в дясната част на (1.19) трябва да се има предвид и ограничението $p_1 + p_2 = 1$, което означава, че член от вида $\mu_1/p_1a_1 \otimes \mu_2/p_2a_2$ се свежда до:

$$\mu_1/p_1a_1 \otimes \mu_2/p_2a_2 = \begin{cases} \mu_1 \wedge \mu_2(p_1a_1 \otimes p_2a_2) & \text{ако } p_1 + p_2 = 1 \\ 0 & \text{в противен случай} \end{cases}$$

По такъв начин получаваме:

$$Z = 1(0.3a_1 \otimes 0.7a_2) + 0.6(0.2a_1 \otimes 0.8a_2) + 0.6(0.4a_1 \otimes 0.6a_2)$$

т.е.изразът за Z като размито подмножество на реалната ос W=(- $\propto, \propto$).

Посоченият метод е в основата на:
-метод за формализация на приближени разсъждения . За разлика от традиционната логика, главен инструмент е правилото "modus ponens a

т. нар. "композиционно правило на извода".
- описание на размити алгоритми, в които често е удобно да се определят отношенията между променливите и да им се присвоят конкретни значения с помощта на блок схеми.
- вземане на решения при неопределени ситуации (приближено описание на реална ситуация).
Развитието на този подход е крачка напред в идеята за разбирането на поведението на хуманитарните системи.

Глава II

Система за дедуктивен извод на естествено-езиков текст

В дедуктивните модели за представяне и обработка на знания, проблемът който предстои да бъде решен се записва във вид на формална система с цел да се потвърди или опровергае на основата на аксиоми и правила за извод от формалната логика. Автоматизираното доказване на теореми датира от работите на Ербран, който е приложил механична процедура за дедуктивен извод.

2.1 Принцип на резолюцията за логически твърдения. Теорема на Ербран

Всяка формула на предикатното смятане, представена в префиксна нормална форма от вида:

$$\Pi = \forall x_1 \forall x_2 \ldots \forall x_r [D_1 \& D_2 \& \ldots \& D_k] \tag{15}$$

където $D_1, D_2, \ldots, D_k$ са дизюнкти, а $x_1, x_2, \ldots, x_r$ - различни променливи, влизащи в тези дизюнкти, се дефинира чрез нейното характеристично множество:

$S = \{D_1, D_2, \ldots, D_k\}$

В този смисъл на множеството от дизюнкти:

$S = \{\neg P(x, f(x)), P(x, y) \vee \neg R(x, g(y)), Q(x, \vee P(x, a))\}$

се съпоставя следната префиксна форма

$\forall x \forall y (\neg P(x, f(x)) \& (P(x, y) \vee \neg R(x, g(y))) \& Q(x \vee P(x, a)))$,

която се дефинира с характеристичното множество $S = \{D_1, D_2, D_3\}$.

Когато се говори, че множеството S е неизпълнимо(противоречиво), се има предвид неизпълнимостта на формулата (2.1).

. *Множеството от дизюнкти S е неизпълнимо тогава и само тогава, когато приема значение "ЛЪЖА" при всички интерпретации във всички области на интерпретация. Поради невъзможността да се разгледат всички интерпретации във всички области е необходимо да се намери специална област на интершретация, в която ако се установи неизпълнимостта на множество от дизюнкти, може да се направи извод за неизпълнимостта на това множество във всички други области на интерпретация[58].*

Такава област съществува и се нарича Универсум на Ербран. Универсума на Ербран се дефинира по следния начин:

. *(Процедура на извода на Ербран)*
Нека са дадени:
- $S = \{D_1, D_2, \ldots, D_k\}$ - множество от дизюнкти;
- H_0 е множество от константи, появяващи се в множеството от дизюнкти S. Ако в S няма константи, то в H_0 се включва някаква константа, например $H_0 = \{a\}$;
- $F = \{f^1(t_1, t_2, \ldots, t_n), f^2(t_1, t_2, \ldots, t_n), \ldots, f^n(t_1, t_2, \ldots, t_n)\}$, където f^i са всички n-местни функционални символи, срещащи се в множеството H, а $t_1, t_2, \ldots, t_n$ са - елементи на множеството H_i;
За $i = 0, 1, 2, \ldots \qquad H_{i+1} = H_i \cup \{f^n(t_1, t_2, \ldots, t_n)\}$
Тогава $H = H_\infty$ се нарича универсум на Ербран за S, а H_i - негово ниво i.

Нека

$$S = \{\mathbf{P}(x, \mathbf{a}, g(y)) \bigvee Q(x), \neg P(f(x), a, y) \bigvee Q(a)\} \tag{16}$$

Тогава:
$H_0 = \{a\}$; $H_1 = \{a, f(a), g(a)\}$;
$H_2 = \{a, f(a), g(a), f(f(a)), f(g(a)), g(f(a)), g(g(a))\}$;
...
$H_\infty = \{a, f(a), g(a), \ldots\}$
За множеството $S = \{\mathbf{Q}(\mathbf{x}) \bigvee \neg R(y), \neg O(x) \bigvee P(y)\}$ се получава $H = H_0 = H_1 = \ldots = H_\infty = \{a\}$

Интерпретацията на универсума на Ербран се нарича $H-$ интерпретация.

. *Интерпретацията I_H е H-интерпретация на множеството дизюнкти S, ако са изпълнени следните съответствия:*

- *На всеки предикатен символ P_j^n съответства n-местно отношение в H;*

- *На всеки функционален символ f_j^n съответства n-местна операция в H(т.е. функция отразяваща H^n в H);*

- *На всяка предметна константа от S съответства константа H (т.е. всички константи се изобразяват сами себе си).*

Нека $= \{A_1, A_2, \ldots, A_k, \ldots\}$ е база на Ербран за S. Тогава H-интерпретация I_H може да се представи чрез множеството: $I_H = \{m_1, m_2, \ldots, m_k, \ldots\}$, в която $m_i A_i$ или $\neg A$ за $i = 1, 2, \ldots$. При това: ако m_i отговаря на A_i, то A_i получава значение 'ИСТИНА', а в противен случай приема значение 'ЛЪЖА'.

Нека

$$S = \{\neg P(f(x)) \vee O(x), R(x)\}. \tag{17}$$

Тогава $H = \{a, a(f(a)), \ldots\}$
$A = \{P(a), Q(a), R(a)P(f(a)), Q(f(a)), R(f(a)), \ldots\}$
Получаваме следните H-интерпретации:
$\mathbf{I_H^1} = \{\mathbf{P(a), Q(a), R(a), P(f(a)), Q(f(a)), R(f(a)), \ldots}\}$
$\mathbf{I_H^2} = \{\neg\mathbf{P(a), \neg Q(a), \neg R(a), \neg P(f(a)), \neg Q(f(a)), \neg R(f(a)), \ldots}\}$
$\mathbf{I_H^3} = \{\mathbf{P(a), \neg Q(a), \neg R(a), P(f(a)), \neg Q(f(a)), \neg R(f(a)), \ldots}\}$

При I_H^1 двата дизюнкта са изпълними, при I_H^2 първият дизюнкт е изпълним, вторият - не, при I_H^3 всички дизюнкти са неизпълними, те приемат значение 'ЛЪЖА'.

В случай, че интерпретацията не е зададена върху универсума на Ербран, а върху произволна област D, то може да се установят следните съответствия между интерпретациите.

. *Нека е зададена интерпретация I, върху определена област D. Казва се, че H-интерпретацията I_H съответства на интерпретация I, ако е изпълнено следното:*
Нека $h_1, h_2, \ldots, h_n$ са елементи на H и нека всяко h се изобразява в някой от елементите d_i на областта D. Тогава, ако всяко $P(d_1, d_2, \ldots, d_n)$ приема значение 'ИСТИНА' или 'ЛЪЖА' при интерпретация I, то $P(h_1, h_2, \ldots, h_n)$ също приема стойност 'ИСТИНА' или 'ЛЪЖА' при I_H.

На тази основа може да дефинираме следната теорема:

. *Множеството дизюнкти S е неизпълнимо, тогава и само тогава, когато S приема стойност 'ЛЪЖА' при всичките си H-интерпретации.*

От Теорема 2.1 следва, че за да се установи неизпълнимостта на множеството от дизюнкти, е достатъчно да се разгледат само H-интерпретации.

. *Множеството от дизюнкти S е неизпълнимо, тогава и само тогава, когато съществува крайно множество от неизпълними фундаментални дизюнкти в S. От тук следва, че за да се установи не изпълнимостта на множество от дизюнкти е необходимо да се съставят множества $S_1, S_2, \ldots, S_n, \ldots$, съдържащи фундаментални дизюнкти и последователно да се проверява тяхната истинност.*

Теоремата на Ербран гарантира, че ако множеството S е неизпълнимо, то дадената процедура ще открие такова n, че S_n да приеме стойност 'ЛЪЖА'. Да допуснем, че $S = \{P(x) \vee \neg Q(x, g(x)), \neg P(f(a)), Q(y, z)\}$. Тогава фундаментални примери за дизюнкти са елементите на следните множества:
$S_1 = (\mathbf{P(f(a))} \vee \neg \mathbf{Q(f(a), g(f(a)))}, \neg \mathbf{P(f(a), g(f(a)))}$.

S_1 е изпълнимо тогава и само тогава, когато:

$$\begin{aligned}
&\mathbf{P(f(a))} \vee \neg \mathbf{Q(f(a)), g(f(a))} \& \neg \mathbf{P(f(a))} \& \mathbf{Q(f(a)), g(f(a))} = \\
&= \mathbf{P(f(a))} \& \neg \mathbf{P(f(a))} \& \mathbf{Q(f(a)), g(f(a)))} \vee \neg \mathbf{Q(f(a), g(f(a)))} \& \\
&\& \neg \mathbf{P(f(a))} \& \mathbf{Q(f(a)), g(f(a))} = \square \qquad \textbf{nil(празен дизюнкт)}
\end{aligned}$$

Основната идея на принципа на резолюцията се заключава в проверката дали множеството дизюнкти S, съдържа в себе си празния дизюнкт $\square$. Ако това е така, то S е неизпълнимо. Ако S не съдържа $\square$, то се предприемат стъпки за извеждане на нови дизюнкти, до тогава, докато се изведе $\square$. По същество принципът на резолюцията се явява разширение на еднолигералното правило на Дейвис и Патнем.

В настоящия модел ще разглеждаме принципа на резолюцията като система от правила за извод, с помощта на които се извеждат нови дизюнкти от дадено множество S и намиране на фундаментална система от дизюнкти като подмножество на множеството S за доказане на неизпълнимостта на дадена формула. В резултат на получаване на празен дизюнкт, системата дава отговор за верността на дадено твърдение.

Негативната страна на процедурата на Ербран се състои в експоненциалното нарастване на множеството S при нарастване на i. Като решение на този проблем в настоящата работа е предложено използването на по- ефективна модификация на принципа на резолюцията, а именно - семантична резолюция.

2.2 Теоретичен модел на системата. Алгоритми и правила за извод.

Нека да разгледаме формалната аксиоматична система $ФС_1$ за пропозиционалното смятане. [8]
1) Изходните терми(атоми) са букви от латинската азбука със или без индекси;
2) Логическите връзки на системата са $\neg, \vee, \wedge, \rightarrow$. Те се интерпретират съответно като отрицание, конюнкция, дизюнкция и импликация. Използват се и скоби. Други изходни символи за системата няма.
3) Правила за образуване на ППФ: а) Всички атоми са ППФ;
б) Ако А и В са ППФ, то $\neg A, A \vee B, A \wedge B, A \rightarrow B$ също са ППФ;
в) Други правила за построяване на ППФ няма.

При интерпретация на ППФ, се интерпретират всички атоми от входното множество, а след това се интерпретират и тези, които са получени в резултат от изпълнението на логическите операции. Окончателното множество, получено като резултат от интерпретация на операциите, се счита за резултат от интерпретационните правила на всички ППФ.
Системата от аксиоми представлява съвкупност от 11 аксиоми, които са общозначими ППФ.

Правила за извод
Правило 1: Всички аксиоми са изводими;
Правило 2: Правило на заместването (ПЗ). Нека А е ППФ, съдържаща атом Х. Тогава, ако А е теорема в пропозиционалното смятане, то като се замени атома Х навсякъде, където той се среща в А, на позициите, където той се появява в произволна ППФ В, се получава също теорема.
Правило 3: Правило modus ponens(правило на дедуктивния извод)
Ако А и $A \rightarrow B$ са теореми, то В също е теорема.
Правило 4: Други правила за извод няма
Формално логически докаателства на твърдения
• Нека С е дадена изводима ППФ във ФС. Ще докажем, че $\tau X \rightarrow P$
Доказателство:
1) $\tau Y \rightarrow (X \rightarrow Y)[a1]$;
2) $\tau P \rightarrow (X \rightarrow P)[]$;
3) $\tau X \rightarrow P$ [modus ponens]

• Ще докажем, че $\tau X \rightarrow X$
Доказателство:
1) $\tau(X \rightarrow (Y \rightarrow Z)) \rightarrow ((X \rightarrow Y) \rightarrow (X \rightarrow Z))[a2]$;
2) $\tau(X \rightarrow (Y \rightarrow X)) \rightarrow ((X \rightarrow Y) \rightarrow (X \rightarrow X))$ [ПЗ] ;
3) $\tau(X \rightarrow Y) \rightarrow (X \rightarrow X$ [modus ponens]

4) $\tau(X \to P) \to (X \to X$ [ПЗ]
5) $\tau X \to X$ [пример 1 и modus ponens]
Всяка ППФ В е изводима в множеството А= $\{A_1, A_2, .., A_n\}$, ако :
1) Всяка A_i е изводима от $A_1, A_2, .., A_n$;
2) Всяка изводима ППФ от ФС, е изводима и от $A_1, A_2, .., A_n$;
3) От изводимостта на А и $A \to B$ от $A_1, A_2, .., A_n$ следва изводимостта на В от $A_1, A_2, .., A_n$;

Ако множеството А= $\{A_1, A_2, .., A_n\}$ се състои от аксиоми или други изводи и ППФ, то множеството изводими от тях ППФ съвпада с множеството на всички изводими ППФ в дадена ФС.

Понятието "изводимост"се доказва чрез теоремата за дедукцията на Ербран: [25]

Теорема : Ако $A_1, A_2, .., A_n$ + В то $\tau A_1 \to (A_2 \to (...(A_n \to B)...))$
Вижда се, че ако ППФ В е изводима от $A_1, A_2, .., A_n$, то ППФ
$\tau A_1 \to (A_2 \to (...(A_n \to B)...))$ е теорема.
Доказателството на тази теорема следва от следните правила за ФС
Правило 5 (правило на силогизма)
Ако $\tau A \to B$ и $B \to C$, то $\tau A \to C$
Доказтелство:
1) $A \to B, B \to C, A \to C$; [mp]
2) $(\tau A \to B) \to ((B \to C) \to (A \to C)$; [ТД]
3) $\tau(B \to C) \to (A \to C)$; [mp]
4) $\tau A \to C$; [mp]
Правило 6 (Разновидност на правилото на силогизма)
Ако А $\tau B, C\tau D$ и В, $D\tau E$, то $A, C\tau E$
1) $\tau C \to D$ [теорема на дедукцията];
2) $B\tau D \to E$ [теорема на дедукцията];
3) $B\tau C \to D$ [силогизъм от 1 и 2];
4) $\tau B \to (C \to E$ [теорема на дедукцията];
5) $\tau A \to B$ [теорема на дедукцията];
6) $\tau A \to (C \to E)$ [силогизъм от 4 и 5];
Правило 7: (правило на размяна на местата)
$\tau A \to (B \to C)$ то и $\tau B \to (A \to C)$;
1) $A \to (B \to C), B, A \to \tau C$; [mp]
2) $A \to (B \to C), \tau B \to (A \to C$; [теорема за дедукцията]
По аналогичен начин се доказват следните две правила:
Правило 8 (правило на обединението)
Ако $\tau A \to (B \to C)$, то $\tau \wedge B \to C$;
Правило 9 (правило на разединението)
Ако $\tau \wedge B \to C$ то $\tau A \to (B \to C)$
Системата от аксиоми притежава свойствата пълнота, непротиворечи-

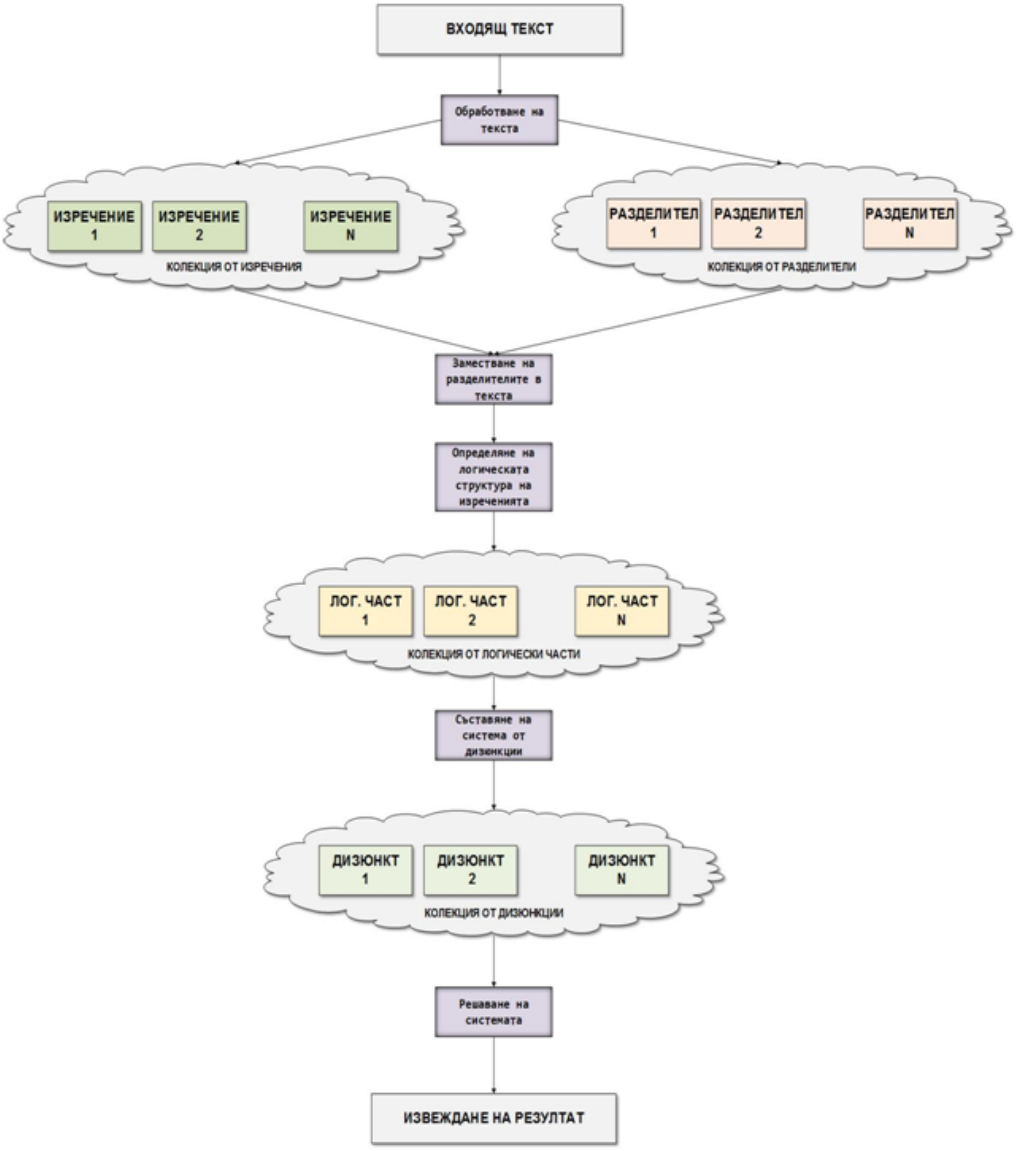
вост и независимост [20] Фактът, че всяка теорема във формалната система на пропозиционалното смятане е общозначима ППФ произтича от това, че всички аксиоми, тавтологии и правила за извод, приложими към тавтологии, винаги се привеждат към тавтологии. Обратното твърдение е теоремата за пълнота.

Теоретичният модел на системата включва: пет основни правила за извод. три алгоритъма и два разработени тестови примера.

Програмата използва правилата за извод в пропозиционалната логика и чрез метода на отричане на целта (resolution refutation) доказва дали дадено твърдение (цел) е истина или не.
Предназначена е за получаване на дедуктивен извод от естествено-езиков текст, въведен на български език, чрез интерпретация в семантично определен клас и за решаване на лингвистични задачи от областта на логиката, изискващи отговор на въпроса дали дадено твърдение е истина или лъжа. За изходни се използват поредица от твърдения, представени във вид на система от дизюнкти, чрез символи на предикатното смятане.

2.3 Концептуален модел

Представеният концептуален модел включва три модула
[M_1] Модул за обработка на изреченията
Въвежда се множество от логически твърдения, подредени в свързан текст, който изисква отговор за отхвърляне или потвърждаване на дадено твърдение . След въвеждането на текст от потребителя, програмата разбива този текст на отделни изречения (Алгоритъм 2.4.1) . Основният елемент в логиката на алгоритъма е разделителя. Разделителите отделят твърденията, които се съдържат в дадено изречение и определят техните ‘знаци‘ - положителен, ако твърдението е истина, или отрицателен, ако е лъжа. След като се разделят изреченията трябва да се определи логическата структура на всяко изречение. На база на логическата структура на всяка отделна част от изречението се поставя съответният знак. Системата от дизюнкции се създава като отделните части на всяко изречение се подредят в система от уравнения, всяко от които съдържа логическите части и техните знаци. За да бъде гъвкав алгоритъма е важно програмата да не работи със строго определени разделители, а да се даде възможност тези разделители да се унифицират.



Фиг. 9 Концептуален модел на системата

Един добър подход да се постигне това е да се моделира универсално поведение на всеки разделител в зависимост от логическата структура, която той определя. След което към него да се определят знаците на твърденията, които го съставят. По този начин думите, които съставят разделителя, могат просто да се заместят с съответните знаци и тогава за програмата няма да има значение под каква форма е зададен текста (език, структура словоред и др.).
Логическата структура на разделителя се определя във вида:

$\mathbf{A} \rightarrow \mathbf{B} \Leftrightarrow \neg\mathbf{A} \vee \neg\mathbf{B}$

Думите, които се заместват в изречението се задават предварително, като колекция от тип низ (дума) и могат да бъдат на всякакъв език. Заместващите ги символи могат да имат следния вид: '|-' $\rightarrow$ отрицателно твърдение ($\neg\mathbf{A}$)

'|+' $\rightarrow$ положително твърдение ($\mathbf{A}$)

' ' $\rightarrow$ празен символ (дума, която не влияе на логическия смисъл на изречението).

[M_2] Модул за трансформиране на системата от логически твърдения в система от дизюнкти

В първоначалното представяне на изреченията като система от логически формули и изрази, се използват логическите връзки отрицание, конюнкция, дизюнкция, импликация, равнозначност. За да се осъществи дедуктивен извод от системата, тя трябва да се представи като система от дизюнкти. които включват само операциите отрицание, дизюнкции и конюнкции. Програмата извършва представянето като използва правилата за извод [Алгоритъм 2.4.2 и т. 2.2.5]

[M_3] Модул за извършане на дедуктивен извод

Дедуктивният извод се осъществява по правилата за изключване на дизюнкти [Алгоритъм 2.4.3) . Изводът за верността на дадено твърдение се дефинира в съотетстие с теорията за принципа за резолюцията и неговата модификация в семантична резолюция.

Реализираната система за дедуктивен извод от естествено-езиков текст, в семантична резолюция би могла да бъде използвана като система за вземане на решения или експертна система. Системата решава ефективно определен клас от логически задачи от областта на математическата и конпютърна лингвистика. Тазработеният тестов пример на задача давана на олимпиада по лингвистика показва че тя би могла успешно да се използва и като система за обучение.

Глава III

Автоматична обработка на естествено-езиков текст

В настоящата глава е представен модел на система за лингвистичен анализ на синтаксиса и семантиката на естествено - езиков текст. Програмата обработва изречения написани на български език [1] в свързан текст. На всяко изречение задава съответно дървовидно представяне и

на базата на коректен синтактичен анализ с определени ограничения, задава семантика на неопределени и семантично нееднозначни изрази и фрази от определен вид. Задаването на синтаксиса на изречение се осъществява чрез дефиниране на граматиката на езика [15]. Един сравнително прост пример на граматика на даден формален език е пораждащата граматика. В настоящата работа се използва безконтекстна фразово-структурна граматика(ФСГ).

В класическата дефиниция на ФСГ правилата са от вида:
ХАУ $\rightarrow$ ХZУ за А $\rightarrow$ Z или А$\rightarrow$ Y _ Y
където А е нетерминален символ, а Х, У, Z са редици от терминални и нетерминални символи, от които поне втория е ненулев.

Да разгледаме следния пример на ФСГ с правила:

$$
\begin{array}{ll}
i & S \rightarrow NP, VP \\
ii & NP \rightarrow ArtN \\
iii & VP \rightarrow V, NP \\
iv & N \rightarrow \text{момче, топка и т. н .} \\
v & V \rightarrow \text{ рита, донесе и т. н .} \\
vi & \rightarrow \text{suff(-то), suff(-та), suff(-тя) и т. н .}
\end{array}
\tag{18}
$$

В правилата v и vi, символите N(съкратено от noun) и V(съкратено от verb) могат да имат повече от едно съответствие, което означава, че те допускат алтернативи.

За осъществяването на коректен синтактичен анализ, са въвеждени следните синтактични категории:

NP1 = define singular noun phrase /определена същ. фраза, ед.ч. (19)
NP2 = indefine singular noun phrase /неопределена същ. фраза, ед.ч.
NP3 = define plular noun phrase/определена същ. фраза, мн.ч.
NP4 = indefine plular noun phrase/неопределена същ. фраза, мн.ч. .

На базата на тези категории може да се разглеждат четири различни локални дървета за глаголните фрази в (3.7), които ще изискват или четири отделни правила, или единични дизюнктивни правила.

$$
\begin{aligned}
VP &\rightarrow V, NP_1 \\
VP &\rightarrow V, NP_2 \\
VP &\rightarrow V, NP_3 \\
VP &\rightarrow V, NP_4
\end{aligned}
\tag{20}
$$

$$
VP = V, \left\{ \begin{array}{c} NP_1 \\ NP_2. \\ NP_3 \\ NP_4 \end{array} \right\} \tag{21}
$$

При сложните категории ситуацията е твърде различна. Дадени са сложните категории на четири типа съществителни фрази.

$$
\begin{bmatrix} nominal+ \\ phrasal+ \\ definite+ \\ numbersing \end{bmatrix} \qquad \begin{bmatrix} nominal+ \\ phrasal+ \\ definite- \\ numbersing \end{bmatrix} \tag{22}
$$

$$
\begin{bmatrix} nominal+ \\ phrasal+ \\ definite+ \\ numberplur \end{bmatrix} \qquad \begin{bmatrix} nominal+ \\ phrasal+ \\ definite- \\ numberplur \end{bmatrix} \tag{23}
$$

Тези четири категории се съпоставят на глаголните фрази (3.7).

$$
VP = V, \begin{bmatrix} nominal+ \\ phrasal+ \end{bmatrix} \tag{24}
$$

‘VP‘ и ‘V‘ могат да бъдат, както на позицията на обекта на посочените четири типа съществителни фрази, така и на позицията на пропозиционални обекти или на позицията на субекти в недефинирани клаузи. При втора ситуация разликата между простите и сложните категории е в това, че при сложните категории, един конституент би могъл да варира като друг. За разлика от модерната ФСГ, където се изисква категориите в съответните локални дървета да имат едни и същи характеристики, то тук е възможно да се съпостави единичен израз.

За импликацията: PP $\rightarrow P$, NP

$$
\begin{bmatrix} number_\alpha \\ person_\beta \\ gender_\gamma \end{bmatrix} \qquad \begin{bmatrix} number_\alpha \\ person_\beta \\ gender_\gamma \\ numbersing \end{bmatrix} \tag{25}
$$

Естествено езикова - семантика

Да се зададе естествено - езикова семантика означава да се дефинира формален език с проста семантика и на този език да се съпостави естествен език.

Формалният език трябва да притежава следните свойства [13]:

* да бъде еднозначен;

* да има прости интерпретационни правила и правила за извод;

* да притежава логическа структура, определена чрез структурата на изречението.

Структура на глагола

Класифицирането на глаголите се осъществява чрез семантичните падежи, които те изискват. Основните семантични падежи, които са използвани са:

$AGENT$ субект в активно изречение или като субект "чрез"в пасивно изречение

$INSTRUMENT$ като субект в активно изречение без агент или предлог - с

$EXPERIENCER$ като субект в активно изречение без агент

$BENEFICIRARY$ като IOBJ с преходни глаголи или предлога - за

$AT - LOC$ предлози в, на, след и др.

$AT - POSS$ притежателни NP или SUBJECT на изречение без AGENT

$TOLOC$ предлози към, във

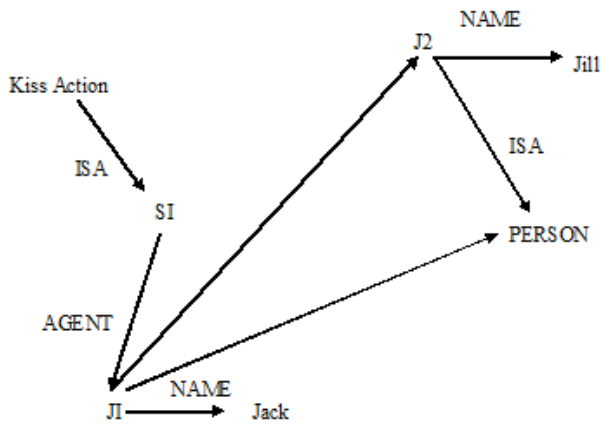
$TO - POSS$ предлози към IOBJ с някои глаголи

$FEROM - LOC$ предлог ”от”, ”от там” и др.

$FEROM_POSS$ предлог ”от”

Логична форма на изречение

Най- често логичната форма на изречение се изгражда на основата на семантичната мрежа [16], представяща структурата на даденото изречение.



Фиг 14. Семантичната мрежа представяща структурата на изречение

На фигурата е представен семантичната мрежа на английското изречение Jack kissed Jill.
На тази мрежа съответства следната логична форма: (PASS s_1 KISS ACTION (AGENT s_1 (NAME j_1 PERSON "Jack"))
(THEMEs_1 (NAME j_2 PERSON "Jill")))
Всяка логична форма включва следните компоненти:
- Оператор, определящ типа на структурата (в случая PASS означава, че представянето е на събитие, което е станало в миналото);
- Име на обекта, който е описан (в случая падежът на KISS ACTION е наречен s_1);
- Модификатори на обектите, което може да бъде списък от структури на ЛФ. В дадения пример модификаторите са двата падежа AGENT и THEME
Други оператори, които се използват за изграждане на структурата на изреченията са PRESS (сегашно време), FUT(бъдеще време), INF(инфинитив).

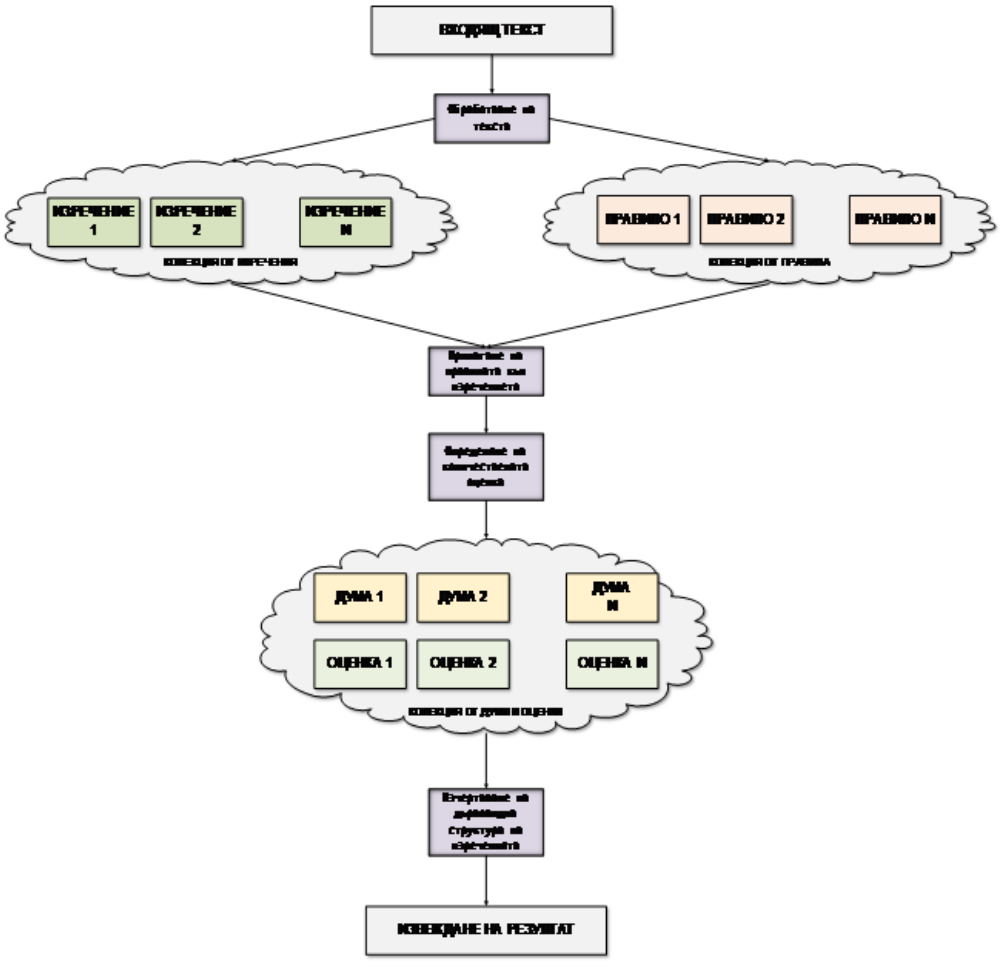
Представени са моделите на изграждане на логична форма на: съществителна фраза, прости изречения и вмъкнати изречения.

3.1 Реализация и съвместимост на програмната система

.NET Framework е среда за разработка и изпълнение на приложения. Тя предоставя програмен модел, библиотеки от типове и единна инфраструктура за разработка на приложения и поддържа различни езици за програмиране. Приложенията, базирани на .NET Framework, се компилират до междинен код (на езика IL) и се изпълняват от средата за

изпълнение на .NET Framework - CLR. Компилираният .NET код може да работи без да се прекомпилира върху различни платформи, за които има имплементация за .NET Framework (Windows, Linux, FreeBSD).

Концептуален модел



Фиг. 15 Концептуален модел на системата за обработка на изречение

Описание и структура на програмните модули

В приложението са включени 16 основни модула и два ресурсни модула.

Структурата на потребителския интерфейс се базира на:

- Модел на система за лингвистичен анализ на естествено - езиков текст на български език да е с възможно най-опростена структура

(не е натоварен с анимации или картинки), за да може максимално оптимизирано да се поднася информацията на потребителя;

- Цветовете да са специално подбрани, за да не дразнят очите и при евентуално по-дълго разглеждане на web приложението да не водят до умора;

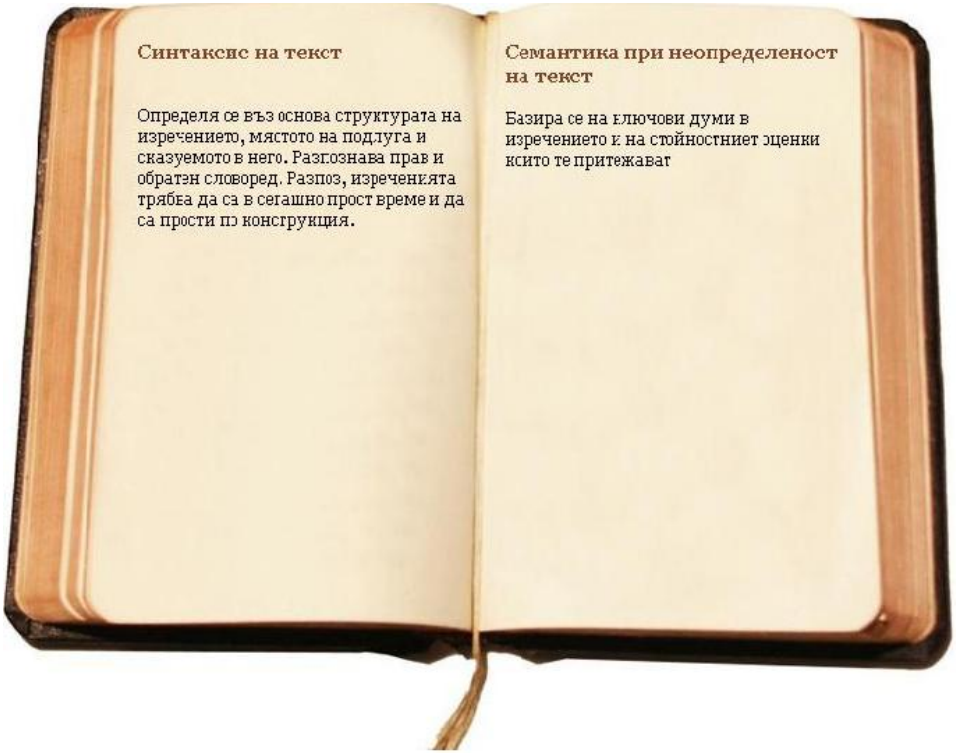
- Лесно и бързо да може да се достига до желания резултат

Програмната среда е създадена, за да улеснява както потребителя, така и администратора. В нея се включват подсказващи менюта и приложения с цел по-лесното достигане до крайния резултат, който си е поставил потребителят. В процеса на разяснение на сайта ще се видят предупредителни и информационни съобщения, които допълнително улесняват работата с програмната среда.

Функцията is_podlog() приема един параметър и това е текст. Тя анализира текста и определя подлога в изречението. За целта използва библиотека от правила и факти, определящи въз основа синтаксиса на изречението, неговата структура, разположението на останалите думи. Тя работи само за прости изречения в сегашно време. Анализът на думата представлява сложна проверка следяща представката, корена и наставката на думата. Ако тези частици съвпадат с тези от базата от данни, то програмата връща като резултат цифра, отговаряща на кода на определената дума.

3.2 Визуализация и диалогови форми на системата

Основната част на системата е нейната начална страница. Тя представлява разгърната книга и се състои от две части. Първата представлява и описва модулите за определяне синтаксис на текст, а втората за определяне на еднозначността от породена нееднозначност на смисъла в дадено изречение. Имената, озаглавяващи двата модула са и бутони водещи ги до конкретните приложения. Целесъобразно дизайнът е изчистен от излишни линкове и бутони с цел улеснение на потребителите.



Фиг. 17 Главна потребителска форма

Моделът на система за лингвистичен анализ на естествено - езиков текст на български език представлява сериозен проект, който изисква много време и усилия за постигане на добри резултати. Самото задание предоставя богати възможности за разработка и самостоятелност по време на проектирането на базите от данни и програмирането на C# модулите. При разработката на проекта възниква необходимост от:

- работа с интернет източници;
- добри знанията по C# програмиране;
- вграждане на функции в C# модулите;
- добри познания на WPF средата на Майкрософт;
- прегледно проектиране на потребителския интерфейс.

До момента Уеб базираната система има възможности за анализ на текст и представянето му в дървовидна структура.Тя може да определя семантиката на изречение, въз основа на това да слага количествена оценка и определя еднозначна семантика на изречение породена от думи,

които са нееднозначни. Възможностите за бъдещи разширения, които биха могли да се направят са следните:

- Разширяване обхвата на видовете изречения, включващи сложните форми;.
- Възможност за анализиране на изречения и в други времена;
- Семантиката на изречението при нееднозначност да се базира върху размити множества и семантични гнезда.

Глава IV

Моделиране на нееднозначни семантики

Премахването на нееднозначността на смисъла на думите (Word Sense Disambiguation - WSD) е дългогодишен проблем в компютърната лингвистика. Една подобна система трябва да може да приема като вход недефиниран текст и на всяка дума да подбира най-подходящото значение с приемлива точност и ефективност. Най-разпространеният подход е премахването на нееднозначността чрез контекста на думата, като за целта се използва информация за всяко значение на думата. [47].

4.1 Описание на математическия модел

Логическа структура

Синтаксисът на езика, който обработва системата е описан от следната безконтекстна граматика:

$L =< T, H, A, R >$, където
$T = \{Noun, Name, Pronominal, Adjective, Adverb, Preposition, Verb, Number, Article, \lambda\}$;
$H = \{S, VP, PP, NP, NP2, NP3, Art1, Adj1, Adv1, \lambda\}$;
$A = (S)$
$R = \{$

$s \rightarrow N, VP$
$NP \rightarrow Art1, Num1, Adj1, Noun$
$NP \rightarrow Name$
$NP \rightarrow Pronominal$
$NP \rightarrow Verb, Adv1, NP2$
$NP2 \rightarrow NP3, PP$
$NP2 \rightarrow NP3$
$NP3 \rightarrow NP2$
$NP3 \rightarrow PP$
$PP \rightarrow PrepositionNP$
$Art1 \rightarrow Art$
$Art \rightarrow \lambda$
$Num1 \rightarrow Number$
$Num1 \rightarrow \lambda$
$Adj1 \rightarrow Adjective$
$Adj1 \rightarrow \lambda$
$Adv1 \rightarrow Adverb$
$Adv1 \rightarrow \lambda$

Проверката на синтаксиса на тази безконтекстна граматика е осъществен чрез следната матрица на предшествията(Dumb - празно място).

	Noun	Name	Pron.	Adj.	Prep.	Verb	Adv.	Numb.	Dumb	Art
Noun	0	0	0	0	1	1	0	0	1	0
Name	0	0	0	0	1	1	0	0	1	0
Pron.	0	0	0	0	1	1	0	0	1	0
Adj.	1	0	0	0	0	0	0	0	0	0
Prep.	1	1	1	1	0	0	0	1	0	1
Verb	1	1	1	1	1	0	1	1	0	1
Adv.	1	1	1	1	1	0	0	1	0	1
Numb.	1	0	0	1	0	0	0	0	0	0
Dumb	0	0	0	0	0	0	0	0	1	0
Art	1	0	0	1	0	0	0	1	0	0

Таблица 1

Въпреки, че системата обработва изречения, в които нееднозначните думи са съществителни, прилагателни имена или глаголи, тя може да се счита за подходящ модел за изследване на ефективността на алгоритъма, тъй като посочените части на речта са по-голямата част от частите на речта в естествения език.

На ключовите дейности нееднозначна "дума + критерий"се задава стойност на функцията на принадлежност към дадено семантично гнездо чрез коефициент на съвместимост, т.е. процентната стойност на статистическата вероятност се опрделя от отношението на броя на използванията на дадена дума с точно определено значение към общия брой на случаите, в които те се срещат заедно с кое да е тяхно значение.

Алгоритъм за представяне на частите на изречението

Системата е самообучаваща се в обогатяването на речника и в зависимостите между думите, така че не зависи съществено от типа на обработвания текст. Методът използва система от 10 речника, в които се съхранява инфор-

мация за типа на съответната дума и за някои от основните и характеристични параметри. (не всички от тях се използват в системата).

Като A е текстов тип и числото след това показва колко байта са заделени за него, N е целочислен тип, а S е тип short. В самите речници Spirit (одушевеност) е означен с t и f (true, false), Number - s и p(singular, plural), Count (изброимо) - t и f (true, false) Type - o и c (ordinalm cardinal) - (редни и бройни), Person (лице) - 1, 2, 3. Coefficient - процентната стойност на статистическата вероятност.
Общият коефициент на варианта на изречението от дадения текст се изчислява като произведение на коефициентите на съместимост на двойките дума - критерий и многозначна дума. Процентната стойност на вероятността на дадения вариант спрямо останалите варианти на същото изречение в същия текст се изчислява по следния начин: (x, y, z и w са стойностите на вероятностите на съотетните варианти на изречения в проценти, a, b, c и d са общите коефициенти на съответните варианти)

При 2 варианта	При 3 варианта	При 4 варианта
$x + y = 100$	$x + y + z = 100$	$x + y + z + w = 100$
	$\frac{x}{y} = \frac{a}{b}$	$\frac{x}{y} = \frac{a}{b}$
	$\frac{y}{x} = \frac{b}{c}$	$\frac{y}{z} = \frac{b}{c}$
	$\Rightarrow$	$\frac{z}{w} = \frac{c}{d}$
$x = \frac{100a}{a+b}$	$x = \frac{100a}{a+b+c}$	$x = \frac{100a}{a+b+c+d}$
$y = \frac{100b}{a+b}$	$y = \frac{100b}{a+b+c}$	$y = \frac{100b}{a+b+c+d}$
	$z = \frac{100c}{a+b+c}$	$z = \frac{100c}{a+b+c+d}$
		$w = \frac{100d}{a+b+c+d}$

Таблица 2

Ограниченията, които са наложени върху типа изречения, обработвани от системата са следните:

Системата приема на вход текст, който може да се въведе ръчно при стартиране на програмата или да се използва стандартен текстов файл, въведен предварително. Програмата позволява да се запише въведения текст като текстов файл с разширение *txt . При по-нататъшна обработка ако има непознати думи или комбинации от думи, системата изисква въвеждане на допълнителни данни за съответната дума.

На изход програмата извежда атрибутивно-стойностна матрица (Atributive Value Matrix, AVN) във вид на дърво на всеки вариант на изречение, логична

форма на всеки вариант, коефициентите на съвместимост на думите критерии и многозначните думи, както и общата вероятност на варианта на изречението като е предвидена възможност за определяне на допустимия минимум на вероятност на изречението, под който изречението се отхвърля. Ако коефициента на съвместимост е 1, то може да се определи дали в логичната форма да се появява като множител и дали да се показват двойките думи с този коефициент.
Заключение
Приложенията биха могли да бъдат в системи за корекции и обработка на документи(договори), системи за обучение, експертни системи и др.

На всяко обработвано изречение и на всяка съществителна фраза е съпоставена логическа форма. По този начин, на базата на синтаксиса се решават определен вид семантични нееднозначности, породени от думите в изреченията.

4.2 Семантична интерпретация на някои неопределени описания, представени в логична форма

Безконтекстната граматика (СТО) безспорно е един от най-силните методи, използван за описания на естествено - езикови структури. [22],[58]. Както бе посочено, правилата на граматиката най-общо се записват по следния начин:

$$< symbol >=< symbol >_1 < symbol >_2 \ldots < symbol >_n$$

Съществува друг подход, съдържащ идеята за оригинална фразова структура, основана на CFG[27]. В това представяне, оригиналните класове не са записани чрез символите: ‘S‘, ‘NP‘, ‘VP‘, а с фразите, които описват прецизно очевидните свойства на елементите от класовете. Различните унификационни граматически формализми използват атрибутивно стойностни матрици(или feature structures)

Дефиниция 4.1. *Матрицата* $\begin{bmatrix} t_1 & v_1 \\ \ldots & \ldots \\ t_n & v_n \end{bmatrix}$, *където* $t_1, t_2, \ldots, t_n$ *са имена на атрибути, такива като* $CAT, NUM, PERS, \ldots$ *и* $v_1, v_2, \ldots, v_n$ *са съответни стойности на атрибутите, е наречена* ***атрибутивно стойностна матрицата (AVM)****.*

Съществуват три основни изисквания за тези описания:

- описанията трябва да бъдат частични, което дава възможност за добавяне на нови детайли във времето;
- описанията, използвани в схемите, трябва да бъдат съвместими;

- трябва да съществува възможност за комбиниране на по-малките структури, за изграждане на по-големи и сложни описания.

За всяка отделна дума, фраза и изречение, съществува AVM. На различните AVMs се съпоставят различни семантики на всяка дума.

Например е известно, че английската дума "stock"ама значение (снабдявам), като съществително (стока), и като прилагателно (складиран). Нещо повече изборът на едно от тези значения зависи единствено от контекста. Матриците на представяне на значенията на думата "stock"са следните(1):

$$\begin{bmatrix} cat & noun \\ num & sing \\ pers & 3 \\ head & stock \end{bmatrix}, \begin{bmatrix} cat & verb \\ num & sing \\ tense & pres \\ head & stock \end{bmatrix}, \begin{bmatrix} cat & noun \\ num & sing \\ head & stock \end{bmatrix}$$

Логична форма и AVM думи, фрази и изречения

По надолу са дадени фрази, изречения и описания, които са представени в логична форма(LF) на основата на AVM. На всяка LF е съпоставена n -арна релация, дефинирана върху размито множество с размита функция на принадлежност. Чрез дефинирането на размита функция на n-променливи за всяка фраза или изречение е дефинирана лексикално функционална семантика(ЛФС). ЛФС ограничава значението на разглежданите изрази. Елементите на множествата, върху които са дефинирани релациите и функциите са думи от естествен език, наречени - лингвистични променливи.

- Логична форма на прилагателно име

  Наличието на прилагателно в съществителната фраза, често води до неопределеност на израза и от там до нееднозначност на цялото изречение. Фразата "The interesting book"в някои ситуации е неопределена. Ние не бихме могли точно да определим думата "интересен". Можем да си зададем въпроса: "Колко интересна е книгата?"Нещо повече различните хора биха били на различни мнения по отношение на факта, че книгата е интересна.

AVM на фазата е представена:

$$\begin{bmatrix} cat & NP \\ det & def \\ num & sing \\ pers & 3 \\ mod & \begin{bmatrix} cat & ADJ \\ property & interesting \end{bmatrix} \\ head & book \end{bmatrix}$$

На основата на това представяне можем да запишем следната LF

Def/sing book (x_1) (property interesting(x_2)

Нека $D_1 = \{b_1, b_2, \ldots, b_n\}$ е множеството от книги D_2 следното множество: $D_2 = \{interesting, moreinteresting, lessinteresting, \ldots\}$. Нека множествата имат функции на принадлежност съответно $\mu_1(x_1)$ и $\mu_2(x_2)$. Съществува размите бинарна релация x, R, x_2 дефинирана върху размито множество $D_1 x D_2 (R_1 \subseteq D_1 x D_2)$ се функция на принадлежност $\mu_k(x_1 x_2)$, такава че за всеки x_1, x_2 $((x_1 x_2) \in D_1 x D_2)$ се съпоставя число от инервала [0, 1]. Значението на функцията $\mu_p(x_1 x_2)$, можем да определим като субективна мярка за степента на изпълнението на релацията. Нека да разгледаме множеството от функции: $F = \{f_{PRED}, f_{NUM}, f_{ART}, f_S, f_{NP}, f_{VERB}, \ldots\}$, чиито елементи съответстват на имената на атрибутите в AVM т.е, на нетерминалните символи на върховете на дървото на разбора на дадената дума, фраза или изречение. Стойността на функциите са елементи на множеството от стойности на атрибутите. Записваме

$P = \{f_{PREP}\}, \{f_{PREP}, f_{NUM}\}, \{f_{NUM}, f_{ART}\}, \ldots, \{f_{PREP}, f_{NUM}, \ldots, f_{VERB}\}$

Нека да дефинираме следните размити функции на две променливи: F_1 : $D_1 x D_2 \rightarrow$ всяка една от които съпоставя на всяка наредена двойка $< X, y >$ елемент от множеството P, където $X \in D_1, y \in D_2$.

Нещо повече, лексикално -функционалната семантика(LFS) на дадена съществителна фраза ще се представя чрез следната интерпретация:

$F_1(x_1, x_2) = \{f_{art}, f_{num}\}$

$F_1(x_1 = \text{book}, x_2 = \text{interesting} =< \text{def, sing} >)$

Можем да запишем LFS на фразата "Less interesting books която е следната:

$F_1(x_1, x_2) = \{f_{art} = \text{indef}, f_{NUM} = p_1\}$

По този начин ние можем да дефинираме различни функции и да генерираме AP фрази от вида на споменатия по-горе тап.

- Логична форма на съществителна фраза

  Ние можем да разгледаме следните четири размити множества:

M = {big, bigger, biggest, no that big, ...}
N = {not nice, nicer, nicerest, ..., nice ...}
R = {blue, dark blue, light blue, ...}
B = {ball(1), ball(2), ...} - set of balls, V.

Съответно с функции на принадлежност: $\mu(x), v(x), p(x), \pi(x)$.

Нека K2 да бъде релацията, зададена върху декартовото произведение MxNxRxB ($R_2 MxNxRxB$). Ще дефинираме функцията P с област K_2 и кообласт.

Тогава с

$F(x_1 = big, x_2 = nice, x_3 = blue, x_4 = ball) = \{f_{art} = def, f_{pers} = 3, f_{num} = sing\}$

е представена LFS на фразата: "The big ice blue ball чиято AVM е дадена на (3):

$$\begin{bmatrix} cat & NP \\ det & def \\ pers & 3 \\ mod_1 & \begin{bmatrix} cat & ADJ \\ size & nice \end{bmatrix} \\ mod_2 & \begin{bmatrix} cat & ADJ \\ quolify & nice \end{bmatrix} \\ mod_3 & \begin{bmatrix} cat & ADJ \\ colour & blue \end{bmatrix} \\ head & ball \end{bmatrix}$$

LF на фразата е следната: Def/sning ball(x.)(colour blue (x_2)(quality nice(x_3)(size big(x_4))))

- Логична форма на изречение

Нека да разгледаме изречението ‘Nelly has a small bag‘ . AVM на това изречение е следната:
Нека да запишем LF
S (subj (NP name Nelly
main-v has
tense present
obj (NPDET a
ADJ small
HEAD bag)).
Можем да дефинираме:
$F(g(x,), g_2(y_1, y_2)) = \{f_s, f_{np}, f_{py}\}$, където g(x,) Н $g_2(y,, y_2)$ са следните функции:
$G_1(x,) = \{f_{noun}, f_{pres}\}; g_2(x_1, x_2) = \{f_{art}, f_{pers}\}$ и
$G_2(y_1 = small, y_2 = bag) =< indef, 3 >$
$G_1(x = Nelly) =< name, 3 >$
Са съответно FLS на фразите ‘Nelly‘ и ‘a small bag‘.

- Логична форма на съществително име

Нека да вземем AVM на описанието за мъж:

$$\begin{bmatrix} hight & tall \\ age & young \\ character & stubborn \end{bmatrix}$$

Ние бихме могли да смятаме, че описанието се състои от фразата:
Висок мъж $\Rightarrow def/singman(x_1)(highttall(x_2))$.
Млад мъж $\Rightarrow def/singman(x_1)(areyoung(x_2))$.
Да дефинираме следните функции

$f_1 : NxA_1 \rightarrow M, f_3 : NxA_3 \rightarrow M, f_2 : NxA_2 \rightarrow M,$
където N е множеството от мъже,
$A_1 = \{tall, verytall, notverytall, \ldots\}, A_2 = \{young, younger, youngest, \ldots\},$
$A_4 = \{stubborn, morestubborn, \ldots\}$ $M = \{f_{art}, f_{pres}\}$. Това може да се обобщи по следният начин:
$F(f_1, f_2, f_3) \Rightarrow MxMxM$. FLS на описанието е следната:
$F(f_1(x_1 = man, x_2 = tall), f_2(x_1 = man, x_2 = young), f_3(x_1 = man, x_2 = stubborn)) = \{f_{def}, f_{pers=3}\}$

Размитите функции, които ние дефинираме са функции на много променливи. Тъй като тези променливи могат да имат стойности, които са думи от естествен език, те могат да бъдат разглеждани като лингвистични променливи. Някои от лингвистичните променливи могат да приемат измерими стойности на променливите, такива като височина, тегло, възраст и т.н. Тези променливи се наричат базови лингвистични променливи. В някои от случаите тези значения могат да бъдат интерпретирани като неопределени ограничения на променливите. Нещо повече: функциите могат да се разделят на два класа: тези, които съдържат базови лингвистични променливи и тези, които не съдържат базови променливи. Ще обърнем по-специално внимание на функциите от първия вид. Базовите променливи се измерват с числа, което ни позволява да оперираме с други методи, използвайки принципа на генерализация на размити множества [38]. Тези функции са числови функции, което позволява да се представят графично. Разглеждането на съществителна фраза или изречение като функция на n-променливи е подходящо за получаване на финалното представяне, което може да бъде направено по-нататък.. Този начин на представяне може да се използва за:
- генериране на нови КР, съответстващи на различни стойности на променливите на функциите;
-ограничаване на значението на неопределена фраза, чрез използване на базови променливи за нейното представяне.

Литература

[1] Андрейчин Л, Н. Костов, Еньо Николов, "Български език за институтите и начални учители| Народна просвета, София 1974;

[2] Бърнев П., П. Станчев, "Размити множества Народна Просвета София 1987;

[3] Василев Ц., Р. Русев, "Algorithm and D", Научни трудове РУ 2008, т.47, серия 5.2;

[4] Галушкин А., И. Дегтярев., "Основи на кибернетиката" , Москва 1984;

[5] Денев Й, Павлов Р "Дискретна математика" , София 1984;

[6] Карпов Ю, "Теория автоматов Москва, 2003;

[7] Левин Р, Д. Дрант, "Практическо въведение на технологията на изкуствения интелект и експертните системи в илюстрациите на Бейсик', 1987';

[8] Мендельсон Э.,"Введение в математическую логику", Москва, 1976;

[9] Митков Р ,"Увод в компютърната лингвистика" сп. Математическа и компютърна лингвистика"изд. АКЛ, Шумен, 1997г.

[10] Наков П., "Статистически машинен превод" , Презентация в Калифорнийски Университет, Бъркли .2007;

[11] Налимов В., "Вероятностная модель языка", Москва 1974;

[12] Павлов Р., "Математическа лингвистика", Народна просвета, 1982;

[13] Поспелов Д. А. ,"Логические методы анализа и синтеза схем М. Энергия", 1974;

[14] Саломаа А., "Жемчужины теории формальных языков", изд. Мир 1986;

[15] Стефанова П.,"Приложна Лингвистика и методика на ЧЕ обучение", Нов Български Университет , 2010г. ;

[16] Таненбаум А. , "Компьютерные сети"СПб, Питер, 2002;

[17] Тодорова М. ,"Езици за функционално и логическо програмиране", София 2010;

[18] Тузов В. А. Ъ"Математическая модель языка", Ленинград, 1984;

[19] Цвиркун А. Д., "Структура сложных систем" М. 1975;

[20] Шапорев С. Д., "Математическая логика - курс лекциий и практических занятий", Санк Ц Петербург, 2005;

[21] Allen J., "Natural Language Understanding", University of Rochester, 1987;

[22] Alsina C, E. Trillas and L. Valverd , " On some logical connectives for fuzzy set Theory" J. of Math. Analysis and Applications 93;

[23] Ashby W. R., "Measuring the internal informational Exchange in a system" Cybernetica 1 1964;

[24] Atkinson P., R. Vieira ,"Beginning Microsoft SQL Server 2012 Programming" , 2012;

[25] Avgers T. G. ,"Axiomatic derivation of the mutual information principle as a method of inductive inference" Kybernetes 12 1983;

[26] Bond F, "Translating the Untranslatable: A Solution to the Problem of Generating English Determiners" CSLI Series in Computational Linguistics. 2005;

[27] Borsley R., "Modern Phrase Structure Grammar", Blackwell publishers, 1996;

[28] Peter F. Brown, Vincent J. Della Pietra, Stephen A. Della Pietra, Robert. L. Mercer.,"The mathematics of statistical machine translation: parameter estimation" Computational Linguistics, 19(2), pp. 263-311, 1993;

[29] Budd Andy., "Simon Collison, Cameron Moll - CSS Mastery: Advanced Web Standards Solutions, friends of ED" 2006 ;

[30] Burdzy K., "The reproducibility property of fuzzy control systems" Fuzzy sets and Systems 9 1983;

[31] Cowell R, P. Stefien , "Probabilistic Networks and Expert Systems", Spigelhaltler, 2007;

[32] David S. Plat, "Introducing Microsoft .NET", t, 2001;

[33] Despeyroux J. T. Despeyroux, G. Kahn, "A simple applicative language Mini ML" Proceeding of the 1986 ACM conference Massachusetts USA;

[34] Feldman A., "M. Daymon WPF in Action with Visual Studio" 2008;

[35] Gatward Richard "A Software Engineering approach to Developing Computer", Proceedings SAER'99 p. p. 213 - 215;

[36] Grimes F, "Microsoft .NET for Programmers", 2002;

[37] Grishman R., "Computation Linguistics", Cambridge University Press, 1986;

[38] Harper M., "Aambiguous Noun Phrases in Logical Form", Computational Linguistics, Velum 18, number 4, 1993;

[39] Hartley R. ,"Transmission of information" The Bell Systems Technical J. v 7 1982;

[40] Hennessy M. "The Semantics of programming Languages. An Elementary Introduction using Structural Operational Semantivs", Wiley 1991;

[41] Hintikka J., "On semantic Information" In: Hinikka and Suppes, 1970;

[42] Howard M., David L., "Writing Secure Code", 2003;

[43] Klir G., T. A. Folder, "Fuzzy Sets, Uncertainty and Information", Prentice hall 1992;

[44] Knuth D. E., "Semantics of context - free Languages Correction - Math", Systems Theory 1971, vol. 5;

[45] Lakates L., "Proofs and Refutations", Cambridge University Press, 1976;

[46] Lecky-Thompson E., "Professional PHP5", 2007;

[47] Mamdani E. H. , J. Efstathion, "An Analysis of formal logic as inference mechanisms on expert systems" Intern. J. of Man Machine Studies 21 1984;

[48] Martin J. , "Elements of Formal Semantics", Academic Press Inc. Orlando;

[49] Nurmi]Nurmi H., "Approaches to collective decision making with Fuzzy preference relations", Fuzy sets and Systems v.6 1981;

[50] Nielson H., "Semantics with Applications", Chichester, New York, 1993;

[51] Orlovsky S., "Decision making with Fuzzy preference relations" Fuzzy sets and Systems v.1 1978;

[52] Ovchinnikov S., "Representations of transitive fuzzy relations", In: Skala Termini and Trillas1984 ;

[53] Polya G., "Mathematics and plausible reasoning", Artificial Intelligence, v. 19, 1982;

[54] Radev S., "Infinitely propositional normal model logic, studio logic sin Comp" Orlando, Florida, 1987;

[55] Rosner M. R. Johnson, "Computational Linguistics and Formal Semantics", Cambridge University Press, 1992;

[56] Skeet J, "C# in Depth", 2008;

[57] Tong R. "The evaluation of a fuzzy models derived from experimental data Fuzzy sets and Systems" v. 4 1980;

[58] Vagin V. N., "Deduction and generalization in the Decision Making System", Moscow, 1988;

[59] Vassileva 1999]Vassileva S. "A Model of Database for Management in Education", Proceedings SAERŸ99 p. 183;

[60] Woodcock J., Martin Loomes, "Software Engineering Mathematics", GB 1991;

[61] Kenji Yamada, Kevin Knight. A Syntax-based Statistical Translation Model. In Proceedings of ACL, pp. 523-530, 2001.

[62] Zadhe L. ,"Notion of Linguistic Variable and its application to taking a decision", Moscow, 1976;

[63] Zadhe L., "Fuzzy Sets and Information granularity" In: Gupta Ragade and Yager 1979;

[64] http://liternet.bg/web/book/dict/translate.htm

[65] http://fbm.uni-ruse.bg/d/uis/L-1.pdf

[66] http://www.fmi.uni-sofia.bg/Members/marian/