Міністерство освіти і науки України

Харківський національний університет радіоелектроніки

Факультет Комп'ютерних наук
(повна назва)

Кафедра Програмної інженерії
(повна назва)

АТЕСТАЦІЙНА РОБОТА
Пояснювальна записка

другий (магістерський)
(рівень вищої освіти)

Дослідження методів синтаксичного аналізу для системи автоматичної обробки текстової інформації
(тема)

Виконав: студент 2 курсу, групи ПЗСм-18-1
(курс, назва групи)

Долуда В.П.
(прізвище, ініціали)

спеціальності 121-Інженерія програмного забезпечення
(код і повна назва спеціальності)

Освітньо-професійної програми
(тип програми)

Програмне забезпечення систем
(повна назва освітньої програми)

Керівник: к.т.н. доцент Валенда Н.А.
(посада, прізвище, ініціали)

Допускається до захисту

Зав. кафедри, проф. _____ Дудар З.В.
(підпис)

2019 р.

Харківський національний університет радіоелектроніки

Факультет Комп'ютерних наук

Кафедра Програмної інженерії

Рівень вищої освіти другий (магістерський)

Спеціальність 121-Інженерія програмного забезпечення

Тип програми освітньо-професійна програма

Освітня програма Програмне забезпечення систем

ЗАТВЕРДЖУЮ:

Зав. кафедри _____
(підпис)

«____» _____ 2019 р.

ЗАВДАННЯ

НА АТЕСТАЦІЙНУ РОБОТУ

студентові Долуді Валерії Павлівні
(прізвище, ім'я, по батькові)

1. Тема роботи: *Дослідження методів синтаксичного аналізу для системи автоматичної обробки текстової інформації*
затверджена наказом університету від «___» _____ 2019 р. № ___
2. Термін подання студентом роботи до екзаменаційної комісії _____ 2019 р.
3. Вихідні дані до роботи: *методи синтаксичного аналізу, граматика для пошуку словосполучень, пояснювальна записка. Використовувати ОС Windows.*
4. Перелік питань, що потрібно опрацювати в роботі: *мета роботи, аналіз проблемної галузі і постановка задачі, огляд етапів обробки текстів природної мови, огляд існуючих алгоритмів для синтаксичного аналізу речення, огляд реалізованих існуючих парсеров для синтаксичного аналізу, огляд словників валентності, створення граматики для пошуку словосполучень, огляд синтаксичного парсера у складі семантичного аналізу.*

5. Консультанти розділів роботи

Найменування розділу	Консультант (посада, прізвище, ім'я, по батькові)	Позначка консультанта про виконання розділу	
		Підпис	Дата
Метод синтаксичного аналізу	доц. Валенда Н.А.		

КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів роботи	Терміни виконання етапів роботи	Примітка
1.	Аналіз предметної галузі	20 вересня 2019р.	
2.	Огляд існуючих методів	7 жовтня 2019р.	
3.	Дослідження синтаксичного аналізу для системи автоматичної обробки текстової інформації	18 листопада 2019р.	
4.	Підготовка пояснювальної записки	27 листопада 2019р.	
5.	Підготовка презентації та доповіді	5 грудня 2019р.	
6.	Попередній захист	10 грудня 2019р.	
7.	Нормоконтроль, рецензування	13 грудня 2019р.	
8.	Занесення диплома в електронний архів	16 грудня 2019р.	
9.	Допуск до захисту у зав. Кафедри	18 грудня 2019р.	

Дата видачі завдання «___» _____2019 р.

Студент _____ Долуда В.П.
(підпис)

Керівник роботи _____ к.т.н. доц. Ваденда Н.А.
(підпис)

РЕФЕРАТ / ABSTRACT

Атестаційна робота містить: 78 с., 10 рис., 1 табл., 9 формул, 4 додатки, 20 джерела.

КОНТЕКСТНО-ВІЛЬНА ГРАМАТИКА, КОРПУС ТЕКСТУ, ЛІНГВІСТИЧНИЙ ПРОЦЕСОР, ПРИРОДНА МОВА, СИНТАКСИЧНИЙ АНАЛІЗ, СЛОВНИК ВАЛЕНТНОСТІ, СЛОВОСПОЛУЧЕННЯ, ТЕКСТ.

Об’єкт дослідження – методи синтаксичного аналізу текстової інформації.

Метою роботи є знаходження проблем при синтаксичному аналізі та пошук шляхів їх вдосконалити.

В результаті роботи розглянуто предметну галузь, розглянуто парсери для синтаксичного аналізу, запропоновано використовувати комбінацію двох підходів та створено граматику для словосполучень.

CONTEXT-FREE GRAMMAR, CORPUS OF TEXT, LINGUISTIC PROCESSOR, NATURAL LANGUAGE, SYNTAX ANALYSIS, VERBAL COMMUNICATION, VOCABULARY OF VALENCY, VOCABULARY, TEXT.

The object of research is methods of syntax analysis of rext information.

The aim of this study is to find problems related to syntax analysis and to search for ways to improve it.

In the study subject area and parsers for syntax analysis were discussed, usage of combination of two methods was suggested, and grammar for phrases was created.

ЗМІСТ

Вступ 7
1 Аналіз проблемної галузі 9
1.1 Етапи лінгвістичного процесора 10
1.2 Існуючи алгоритми 13
1.3 Текстові корпуси 18
1.4 Постановка задачі 20
2 Аналіз методів для дослідження 22
2.1 Огляд існуючих методів реалізації синтаксичного аналізу 22
2.1.1 ABBYY Syntactic and Semantic Parser 22
2.1.2 Парсер граматики зв'язків (LinkParser) 24
2.1.3 Синтаксичний парсер лінгвістичного процесора ЕТАП-3 25
2.1.4 SyntAutom 27
2.1.5 SemSin 28
2.2 Огляд словників валентності 29
2.2.1 Толково-комбінаторний словник 29
2.2.2 Семантичний словник Тузова 30
2.2.3 RussNet 32
2.2.4 Framebank 33
3 Метод синтаксичного аналізу словосполучень 35
3.1 Граматика словосполучень для російської мови 35
3.2 Синтаксичний парсер 45
3.3 Синтаксичний аналіз у складі семантичного аналізу 47
3.4 Програмна реалізація 49
3.5 Можливості використання отриманих результатів 50
Висновки 52
Перелік посилань 54
Додаток А Програмний код для морфологічних правил 56

Додаток Б Слайди презентації 58
Додаток В Відгук керівника роботи 74
Додаток Г Рецензії 76

ВСТУП

Актуальність автоматичної обробки тестової інформації пояснюється значним збільшенням кількості інформації та ускладненням її обробки. Зараз автоматична обробка текстів може застосовуватися для вирішення безлічі завдань, з багатьма з яких кожна людина стикається щодня. До більш простих належать перевірка орфографії та автовиправлення, фільтрація спаму, автоматичний переклад невеликих фрагментів тексту. Існують і більш складні завдання: знаходження релевантних відповідей на питання, повноцінний машинний переклад великих текстів, системи антиплагіату, аналіз тональності тексту, визначення тематики тесту, складання анотації документа, автоматичне реферування та спрощення тексту, конструювання рекомендаційних систем, що працюють з великими масивами неструктурованих даних.

Метою даної роботи було вибрано пошук шляхів для вдосконалення слабких сторін систем автоматичної обробки тексту.

Об’єктом дослідження є процес обробки текстової інформації.

Предметом дослідження було вибрано методи синтаксичного аналізу для систем автоматичної обробки текстової інформації.

Синтаксичний аналіз – це один з етапів обробки тексту машиною, в рамках якого велося дослідження. Попередній етап – рівень морфології – на сьогодні більш опрацьовано, та лінгвістичний процесор може досить легко розібрати поданий текст на рівні морфології на достатньому рівні якості[4]. Проте наступний рівень – рівень синтаксису – викликає проблеми, головним чином через вільний порядок слів в реченні, бо практично неможливо перебрати всі можливі конфігурації слів в реченні, та багатозначність, яка притаманна конструкціям мови на усіх рівнях аналізу. Тому немає одного універсального рішення, яке б працювало в любому випадку, та існують різні підходи.

Для досягнення поставленої мети були досліджені теоретичні матеріали про загальну схему створення лінгвістичного процесору, яка включає підходи для

створення кожного рівня окремо та їх зв’язок один з одним. Також було розглянуто сучасні системи для синтаксичного аналізу текстової інформації, способи їх роботи, проблеми під час реалізації , обмеження під час роботи. Для створення власної LR-граматики були вивчені теоретичні аспекти її побудови.

Під час дослідження було виявлено те, що побудова правил граматики для структури цілого речення російської мови є проблематичною [12]. Оскільки граматика російської мови дозволяє вільний порядок слів, кількість правил стає дуже великою. Але навіть не дивлячись на це, створення правил веде до втрати частини можливостей мови і виділення лише найпримітивніших конструкцій. Тому було запропоноване рішення виділяти за допомогою правил граматики лише частини речення – словосполучення. Словосполучення мають чітку структуру та не дозволяють вільного порядку слів. У запропонованому підході пропонується використовувати словники валентності замість створення правил для всього речення. Такий підхід удосконалить синтаксичний аналіз шляхом спрощення його проведення та зменшення неоднозначностей.

Наукова новизна роботи полягає у комбінованому підході до синтаксичного аналізу, який включає синтаксичний аналіз для груп слів з чітким порядком слів, та елементи семантичного аналізу для усього речення в цілому, що дозволяє істотно зменшити кількість варіантів розбору та спростити процес.

Запропонований підхід доцільно буде використовувати в лінгвістичному процесорі замість пошуку повного дерева речення на етапі синтаксичного аналізу. Такий підхід легко інтегрується в лінгвістичний процесор та надає необхідні данні для роботи наступного аналізатору – семантичного.

Крім того отриманні данні можна використовувати для чорнової розмітки корпусів тексту. Корпуси тексту на даний момент широко використовуються для навчання систем штучного інтелекту. Оскільки для якісного навчання потрібна велика кількість корпусів текстів, а розмітка зазвичай робиться людиною, то переклад частини роботи на мишину є актуальним.

1 АНАЛІЗ ПРЕДМЕТНОЇ ГАЛУЗІ

У якості природної мови в даній роботі буде розглядатися російська мова. Це обумовлено широким розповсюдженням російської мови. У XX столітті російська мова увійшла в число глобальних мов. Журнал «Language Monthly» опублікував інформацію, за якою приблизно 300 млн осіб у світі володіють російською мовою, що робить її п’ятою за поширеністю. Також не маловажна поширеність мови в Інтернеті, оскільки це основне місце для використання автоматичної обробки тексту. За даними дослідження W3Techs російська мова друга по використанню в Інтернеті. Також можна добавити, що електронна база даних реєстру перекладів «Index Translationum» свідчить, що є велика кількість перекладів з російської та на російську мову у світі, для чого також необхідно використовувати автоматичну обробку тексту. Серед мов, на які найчастіше перекладають, російська на сьомому місці, а з яких найчастіше перекладають – четвертому.

У російській мові, як у будь-якої іншої, є свої особливості, які необхідно враховувати при автоматичній обробці тексту. Перше, що варто відзначити, це вільний порядок слів, який унеможливлює скласти загальну схему речення. Мови можна класифікувати за базовим порядком складових. Виділяють три основні складові в реченні: підмет (S), присудок (V) і пряме доповнення (O). Виділяють шість базових типів SOV, SVO, VSO, VOS, OVS, OSV в залежності від найбільш частого порядку частин в реченні [1]. Російська мова належить до групи SVO, до яких відноситься 75% від загального числа мов, де першим буде суб’єкт, який виконує дію, після буде дія, а останнім буде об’єкт, над яким дія виконується. Але на відміну від мов із чітким порядком слів, які граматично правильно можуть мати тільки одну структуру побудови речення, в російській мові будуть правильними всі шість.

Ще одною з найбільших складнощів при обробці текстів російською мовою є неоднозначність його одиниць, що виражається в явищах полісемії, синонімії омонімії.

Полісемія – наявність в однієї одиниці мови двох та більше пов'язаних між собою значень. Виділяють полісемію слів, де одне і теж слово використовують для позначення різних предметів чи явищ («земля» – суша, грунт, конкретна планета) та граматичну полісемію, де слово (зазвичай дієслово) можна вжити в декількох значеннях. Наприклад, у реченні «В окно постучали» дієслово вжито в невизначено-особистому значенні, а у реченні «Мы постучали по дереву.» дієслово вживається у власне-особистому значенні.

Синонімія – це тотожність або близькість значень різних за звучанням мовних одиниць одного мовного рівня, такого як слів, словосполучень, морфем, синтаксичних конструкцій [2]. Наприклад, синонімія слів: «головоломка» – «загадка» – «софизм» – «чайнворд» – «задача» – «кроссворд».

Омонімія – збіг за формою двох різних за змістом мовних одиниць. Саме наявність загальної семантичної ознаки відрізняє полісемію від омонімії. Наприклад, слово «три» може бути числівником та може бути одною з форм наказового способу дієслова «тереть». Оскільки ці слова семантично не пов’язані, то це є явищем омонімії.

Крім указаних вище складнощів при моделюванні природної мови треба відмітити зміни, що постійно відбуваються. Зміни в мові можна відчути вже через один або два десятка років. Найбільші зміни відбуваються в словниковому складі. Але навіть синтаксис, морфологія і фонетика зазнають змін. Як наслідок, немає можливості розробити формальну модель конкретної природної мови один раз і побудувати на її основі лінгвістичний процесор. Потрібне постійне поповнення знань про мову на всіх його рівнях і корекція існуючих моделей.

1.1 Етапи лінгвістичного процесора

Вищезазначена складність природної мови веде до складностей в її обробці. Для того, щоб спростити цей процес, виконується розбиття окремі етапи, відповідні

рівням мови. Більшість сучасних лінгвістичних процесорів відносяться до модульного типу, в якому за кожен етап аналізу тексту відповідає окремий модуль процесора. Є кілька способів розбиття тексту на рівні, проте загальновизнано виділяти рівень речень (синтаксичний рівень), рівень слів (морфологічний рівень) та рівень фонем (фонологический рівень). Питання про кількість рівнів і їх склад в лінгвістиці досі залишається відкритим. Серед рівнів часто виділяють підрівні. При вирішенні деяких особливих задач в процесорі можуть бути присутніми не всі рівні. В загальному випадку лінгвістичний процесор виконує чотири етапи.

Перший етап це графематіческій або лексичний аналіз, який ще можна назвати сегментацією. Вхідними даними для лексичного аналізу є не оброблений текст на природній мові. На цьому етапі програма поділяє текст на речення, а в речення на слова та токени. Крім слів у реченні можуть буди роздільники, прізвища, імена, ініціали, електронні адреса, дати, числа, заголовки, примітки. Також може виконуватися виділення стійких оборотів, які не мають словозмінних варіантів. Отримана інформація на наступному етапі передається морфологічному і синтаксичному аналізаторам.

Наступним етапом виконується морфологічний аналіз. На цьому рівні процесор працює окремо з кожним словом. Для нього визначається частина речі та знаходяться всі його характеристики. Такими характеристиками для різних частин речі можуть бути рід, число, відмінок, особа, схиляння, час. Також на цьому етапі визначається лема слова, тобто початкова форма слова. Наприклад, для іменника це називний відмінок, однина. Актуальною на цьому етапі є проблема однозначного розпізнавання, тобто позбавлення омонімії леми в контексті. Розрізняють два типи слів-омонімів – повні і неповні, слова однієї і тієї ж частини мови, у яких збігаються не всі та всі форми відповідно. На етапі морфологічного аналізу приділяють увагу до неповних омонімів, які важливо вирішити зараз, в той час як повні омоніму на цьому етапі можуть на впливати на аналіз. Ще одну складність представляють омоформи і омографи. Омоформи – слова, що збігаються в одній, рідше – в декількох граматичних формах. Омографи – слова, що мають різне прочитання, відповідно і різне значення, але мають однакове написання. На

етапі морфологічного аналізу може бути виділений лексичний підрівень. Лексема – це семантичний інваріант всіх словоформ, тобто сукупність всіх граматичних форм для одного слова. Віднесення конкретного слова з речення до його леми у словнику не можна вважати повноцінним морфологічним аналізом, але можна вважати його частиною. Морфологічний рівень включає також підрівень морфем. Морфема – це мінімальна значуща частина слова, така як корінь, префікс, суфікс, закінчення, постфікси.

На третьому кроці виконується синтаксичний аналіз. Використовуючи данні, отриманні про кожне слово в реченні із морфологічного аналізу, виявляються синтаксичні зв’язки слів між собою та будується граматична структура речення. Для представлення результатів даного етапу часто використовують таку структуру, як дерево. Це зручний спосіб показати відразу і структуру речення і залежності між словами, який також має зручну для людини графічну інтерпретацію. При правильному результаті роботи синтаксичного аналізатора дерево має один корінь. Однак через омонімії в тих випадках, коли граматичні форми різних слів збігаються або через збіг деяких словоформ в лексемі, можливі отримання різних дерев розбору. З них вибирається та, яка має найменшу кількість висячих вершин. Такий розбір речення вважається найбільш достовірним. В рамках синтаксичного рівня може бути виділений підрівень словосполучень – синтаксично пов'язаних груп слів. На цьому підрівні не знаходять зв’язки між всіма словами в реченні, а шукають більш прості зв’язки в групах з декількох слів, та не пов’язують ці групи між собою. Такий аналіз є меншим, ніж переклад тексту, тому такий підхід можна назвати підрівнем синтаксичного аналізу. Також можна виділити надрівень для синтаксичного аналізу – це пошук складного синтаксичного цілого або сверхфразова єдність. Складне синтаксичне ціле – це сукупність речень (приблизно може бути абзацом), об’єднаних змістом. Пошук таких частин відбувається з допомогою лексико-граматичних засобів, таких як лексичні повтори і анафоричні посилання. Це вказівки на попередні слова тексту, які в даному місці замінені на займенники, щоб уникнути повтору [2].

Останнім етапом э семантичний аналіз. Результати роботи аналізатора можуть відрізнятися в залежності від початкової задачі обробки тексту. У загальному випадку на даному етапі алгоритму необхідно визначити яке інформаційне навантаження несе текст. Наприклад, результатом роботи даного етапу може бути структура, що складається з текстових фактів. Такий результат буде зручним для задач реферування та спрощення тексту. Також результатом проведення семантичного аналізу є виявлення смислового змісту тексту і його оцінка. Це необхідно для рішення задач визначення теми тексту та його емоційного забарвлення. Для подання такого результату використовується зв'язний семантичний граф (семантична мережа), в вузлах якого знаходяться слова або словосполучення, а на дугах, які їх з’єднують, знаходиться семантична вага. Як частину семантичного аналізу можна виділити локальний семантичний аналіз – семантичний аналіз в межах одного речення [2].

Окрім семантичного аналізу, можна ще виділити етап прагматики чи дискурса. Це етап аналізу тексту в цілому. Основне завдання цього виду аналізу текстів полягає в зв’язуванні побудованих на етапі семантичного або синтаксичного аналізу графів простих речень в єдину когнітивну карту за особливими правилами. Когнітивна карта – це граф, у вузлах якого знаходяться об'єкти або суб'єкти, між якими встановлені взаємини у вигляді послідовних дій – дієслівних груп [3]. Функціональна сторона етапу полягає у встановленні залежностей між об'єктами – іменними та іншими групами.

1.2 Існуючи алгоритми

Результат як морфологічного аналізу, так і наступних етапів в загальному випадку неоднозначний. Отже немає загальновизнаного підходу до проведення морфологічного, синтаксичного і семантичного аналізів, який би давав повністю вірний результат. Існує кілька підходів до проведення кожного етапу.

Модулі морфологічного аналізу словоформ розрізняються в основному за такими параметрами:

– результат роботи. Лема з різним набором морфологічних характеристик (рід, число, відмінок, вид, особа і т.п.) заданої словоформи;

– метод аналізу. Можливі методи з опорою на словник словоформ мови, на словник основ, або ж без опори на словник;

– наявність або відсутність можливості обробки словоформи лексеми, не включеної в словник [4].

Для реалізації синтаксичного етапу запропоновано велику кількість різних ідей і методів, що відрізняються способом опису синтаксису мови, способом використання цієї інформації при аналізі або синтезі пропозицій, а також способом представлення синтаксичної структури речення. Можна виділити три основні підходи: генеративний підхід, заснований на ідеях граматик, створених Н. Хомським; підхід, який основується на лінгвістичній моделі «Смисл ⇔ Текст» І. Мельчука, а також підхід, в рамках якого робляться ті чи інші спроби подолати обмеження перших двох підходів [5].

В рамках генеративного підходу синтаксичний аналіз проводиться, як правило, на основі формальної контекстно-вільної граматики, яка описує фразову структуру речення, або ж на основі деякого розширення контекстно-вільної граматики. Ці граматики виходять з послідовного лінійного членування речення на фрази (різні словосполучення) і тому можуть відображати одночасно як його синтаксичну, так і лінійну структури. Отримана в результаті ієрархічна синтаксична структура речення природної мови описується деревом складових, в листях якого знаходяться слова речення, кожне піддерево відповідає одній синтаксичній конструкції, отриманої з речення, тобто фразі. Дуги висловлюють відносини вкладених конструкцій. Дерева складових більше підходять для опису мов з жорстким порядком слів, подання з їх допомогою розірваних конструкцій вимагає розширення використовуваного граматичного формалізму [4]. Зате в рамках цього підходу більш природно описуються конструкції з непідрядними

відносинами. Як буде виглядати дерево складових для речення «Я слышу стук в дверь» представлено на рисунку 1.1.

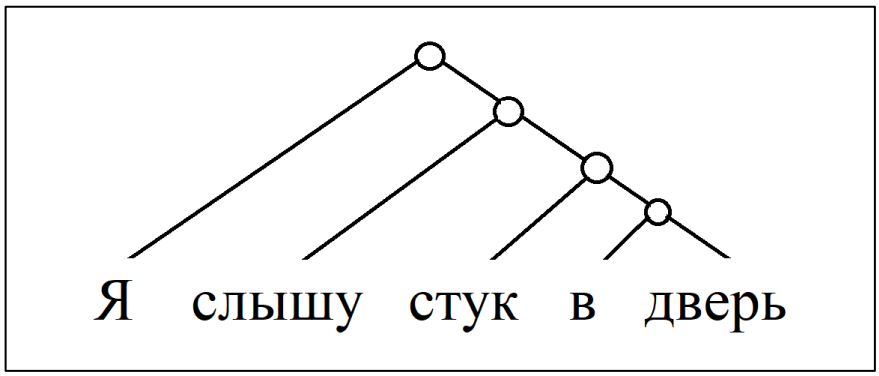

Рисунок 1.1 – Дерево складових

В рамках другого підходу для подання синтаксичної структури пропозиції використовується більш наочний спосіб – дерева залежностей. Слова речення розташовані не тільки в листях, а во всіх вузлах дерева. В корені дерева знаходиться слово-предикат, зазвичай дієслово-присудок, а кожна дуга дерева, що зв'язує пару вузлів, інтерпретується як синтаксичний підрядний зв'язок між ними, причому напрямок зв'язку відповідає напрямку даної дуги. Оскільки при цьому синтаксичні зв'язки між словами і порядок слів у реченні відокремлені, то на основі дерев залежностей можуть бути описані розірвані конструкції [6], які досить часто виникають в мовах з вільним порядком слів. Як буде виглядати дерево залежностей для речення «Я слышу стук в дверь» представлено на рисунку 1.2.

Загальна складність для обох підходів – уявлення однорідних членів речення.

Синтаксичні моделі в усіх описаних вище підходах намагаються врахувати обмеження, що накладаються на з'єднання мовних одиниць у мовленні, при цьому так чи інакше використовується поняття валентності [4]. Валентність – це здатність слова або інший одиниці мови приєднувати інші одиниці певним синтаксичним способом; актант – це слово або синтаксична конструкція, що заповнює цю валентність. Наприклад, російська дієслово «передать» має три основні валентності, які можна виразити наступними питальними словами: хто? кому? що?

В рамках генеративного підходу валентності слів (перш за все, дієслів) описуються переважно у вигляді спеціальних фреймів [4], а в рамках підходу, заснованого на деревах залежностей – як моделі управління [6].

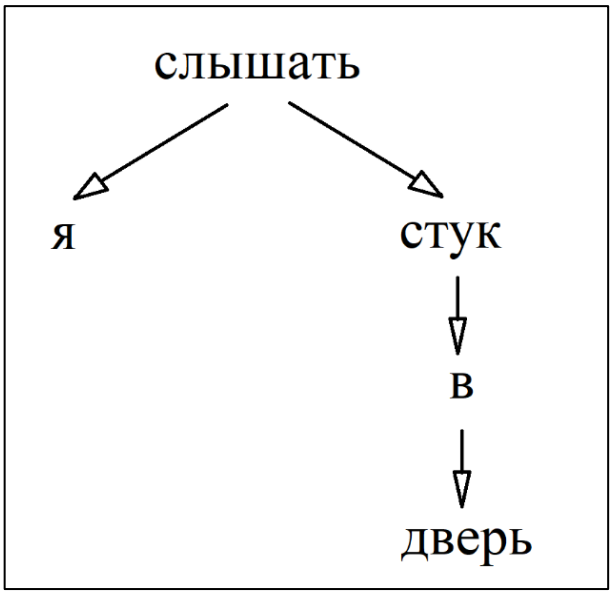

Рисунок 1.2 – Дерево залежностей

Модулі синтаксичного аналізу в обох розглянутих підходах спираються на граматики природної мови. Загальна кількість правил граматики може бути від декількох десятків до декількох сотень, в залежності від використовуваного словника: чим більше інформації представлено в словнику, тим коротше може бути граматика і навпаки. Так, в моделі «Смисл – Текст» [5] упор робиться на словник, а не на граматику; в словнику, що використовується зберігається інформація, що відноситься до різних рівнів мови, зокрема, про моделі управління слів і нестандартну сполучуваність слів.

Етап семантичного аналізу тексту найменш опрацьований. Для локального семантичного аналізу були запропоновані так звані відмінкові граматики і семантичні відмінки (валентності) [6], на базі яких семантика пропозиції описується через зв'язку головного слова (зазвичай дієслова) з його семантичними актантами, тобто через семантичні відмінки. Це спосіб схожий на описаний вище спосіб для проведення синтаксичного аналізу. Наприклад, дієслово «передать»

описується семантичними відмінками того, хто дає, адресата і об'єкта передачі. Використовуючи термінологію штучного інтелекту, сукупність семантичних відмінків часто називають семантичним фреймом, що описує відповідну ситуацію. Для подання семантики всього тексту зазвичай використовуються два формальних підходи:·

– формули обчислення предикатів, що виражають властивості, стан, процеси, дії і відносини;

– семантичні мережі – розмічені графи, в яких вершини відповідають поняттям, а дуги – відносинам між ними.

Мало досліджений рівень прагматики і дискурсу. В основному розроблені методи аналізу локальної зв'язності тексту, в першу чергу, вирішення анафоричних посилань [7]. Анафора – використання виразних властивостей мови, які можуть бути коректно інтерпретовані тільки в контексті, як правило, з урахуванням попереднього фрагмента тексту.

В даний час існує два основних підходи для створення лінгвістичних процесорів: перший заснований на правилах (rulebased), а в основі другого лежить машинне навчання. Історично першим є підхід на правилах, який полягає в описі необхідної лінгвістичної інформації у вигляді формальних правил. Правила створюються лінгвістами або фахівцями з проблемної області оброблюваних текстів. Описані вище способи проведення окремих етапів аналізу є представниками даного підходу. Незважаючи на те, що це був перший спосіб обробляти мову, в його рамках продовжують розробляти і удосконалювати алгоритми, та він залишається як і раніше актуальним.

Більш новим способом є машинне навчання. Джерелом лінгвістичної інформації для такого підходу виступають не правила, а відібрані тексти проблемної області. Серед методів, що застосовуються в рамках підходу, виділяють методи навчання з учителем (supervised), методи навчання без вчителя (unsupervised), методи часткового навчання з учителем (bootstrapping) [7]. Найчастіше застосовується навчання з учителем, при якому відбувається побудова машиною математичної і програмної моделі – машинного класифікатора, який вміє

розпізнавати різні класи одиниць тексту (слів, словосполучень та інших конструкцій) або самих текстів. Побудова класифікатора відбувається на спеціально розміченій навчальній вибірці, в якій одиницям мови, які потрібно навчитися розпізнавати, приписані мітки, які кодують важливі ознаки. Навчанням є, по суті, виявлення загальних закономірностей, властивих текстам на природній мові, на основі даних навчальної вибірки.

Обидва розглянутих підходи мають свої переваги і недоліки. Створення правил вимагає досить кваліфікованої працівника, як правило, лінгвіста. Дуже часто навіть лінгвіст не може передбачити заздалегідь всі окремі випадки, які треба відобразити в правилах. У той же час правила зазвичай є декларативними і легко зрозумілі, тому їх просто підтримувати: модифікувати і розширювати, тим самим доповняючи функціонування лінгвістичного процесора. Машинне навчання не вимагає ручної праці по складанню правил і скорочує час розробки систем, проте необхідні знання для вибору відповідних методів навчання. Крім того, класифікатори, які створює машина непрозорі для розуміння людиною, тому що не мають явної лінгвістичної інтерпретації.

1.3 Текстові корпуси

Раніше частіше застосовувався підхід на правилах, оскільки було мало розмічених текстових корпусів. З появою різних розмічених даних все частіше вдаються до машинного навчання, як швидкого способу отримання потрібного класифікатора. Отже, текстові корпуси на даний час є актуальним і затребуваним матеріалом.

Корпус текстів – це масив текстів, зібраний за певним принципом, який відповідає поставленій задачі (по жанру, авторської приналежності і т.п.), а також володіє такою лінгвістичної розміткою, яка потрібна для даного випадку,

наприклад, морфологічною, акцентною, синтаксичною, дискурсивною або ін. Для всіх корпусів можна виділити наступні властивості. Корпус повинен бути:

– в електронному вигляді;

– репрезентативний – повинен добре відображати об’єкт, який моделює;

– розмічений;

– прагматично орієнтований – повинен бути створений під певну задачу.

Розмітка полягає в приписуванні текстів і їх компонентів спеціальних тегів: лінгвістичних і зовнішніх. Виділяють наступні лінгвістичні типи розмітки: морфологічна, семантична, синтаксична, дискурсна і т. д. До деяких корпусах застосовуються подальші структурні рівні аналізу. Зокрема, деякі невеликі корпуси можуть бути повністю синтаксично розмічені. Такі корпуси зазвичай називають глибоко анотованими або синтаксичними, а сама синтаксична структура при цьому є деревом залежностей.

Розмічені корпуси створюються зазвичай експертами-лінгвістами. Оскільки розмітка текстів досить трудомістка і довга робота, яка потребує фахівців, для прискорення створення корпусів користуються краудсорсингом, при якому розмітка виконується волонтерами, а потім редактується. Прикладом такого сервісу є OpenCorpora [8]. Їх мета – створити морфологічно, синтаксично і семантично розмічений корпус текстів російською мовою, в повному обсязі доступний для дослідників і редагований користувачами. Така форма краудсорсингу стала можливою завдяки розподіленню завдання розмітки на невеликі завдання, з більшістю з яких може впоратися людина без спеціальної лінгвістичної підготовки. На своєму сайті вони пропонують бажаючим взяти участь в розмітці іменованих сутностей, в знятті морфологічної неоднозначності, додаванні текстів в корпус та інше.

Також варто відзначити найбільш відомимі та обширні корпуси для російської мови «Национальный корпус русского языка» (НКРЯ) [9] та «Генеральный Интернет-Корпус Русского Языка» (ГИКРЯ) [10].

Інший спосіб створення корпусу – напівавтоматична розмітка, коли спочатку працює вже готовий модуль аналізу тексту, а його результати підправляються

людиною-експертом. Ще один шлях – пошук природної розмітки тексту. Наприклад, для машинного навчання в задачах оцінки тональності текстових відгуків можуть бути використані тексти інтернет відгуків з уже проставленими оцінками [4].

1.4 Постановка задачі

Метою роботи є дослідження проблем синтаксичного аналізу та розробка методу синтаксичного аналізу для мов з нечітким порядком слів у реченні.

Дослідження різних методів синтаксичного аналізу, реалізованих у ряді парсерів, визначають однією з головних проблем складність побудови повного єдиного дерева речення під час синтаксичного аналізу. Причиною являються багато неоднозначностей на всіх рівнях аналізу. Для вирішення цієї проблеми пропонується застосовувати синтаксичний аналіз для груп слів, які мають чіткий порядок. Метод синтаксичного аналізу пропонується реалізувати як комбінацію синтаксичного та семантичного аналізу. Було прийнято рішення зменшити цю задачу до задачі розділення речення на словосполучення, а завершити синтаксичний аналіз використовуючи словник валентності.

Для реалізації методу синтаксичного аналізу необхідно виконати наступні основні етапи:

– формалізувати синтаксис словосполучень – побудувати модель, яка в формальному вигляді описує синтаксис даних мовних одиниць;

– запропонувати механізм співставлення граматичних ознак головних та залежних слів;

– запропонувати метод синтаксичного аналізу швидкість якого близька до лінійного;

– запропонувати метод семантико-синтаксичного аналізу речень на основі словника валентності.

Етап формалізації синтаксису словосполучень включає в себе наступні дії:

– виділення основних видів словосполучень і їх головних слів;

– побудова структурних схем словосполучень в залежності від головних слів, опис правил узгодження головних та залежних слів словосполучення;

– вибір моделі для опису синтаксису словосполучень;

– опис синтаксису словосполучень за допомогою обраної моделі.

Для виділення словосполучень буде використовувати контекстно-вільна граматика, яка буде шукати послідовності частин мови в реченні, які можуть бути словосполученнями. Це можна зробити просто, оскільки словосполучення мають жорсткий порядок слів. На граматику будуть накладатися морфологічні правила, які існують в російській мові для узгодження слів в словосполученнях. Ці правила допоможуть перевірити чи дійсно знайдена послідовність слів э словосполученням та виділити головне і залежні слова. Для реалізації отриманої граматики буде використовуватися технологія LR-аналізу.

Також буде розроблено веб-додаток, у якому користувач матиме можливість ввести свій текст та отримати синтаксичний розбір речення у вигляді знайдених словосполучень та віднесення їх до групи об’єкту, дії чи суб’єкту.

2 АНАЛІЗ МЕТОДІВ ДЛЯ ДОСЛІДЖЕННЯ

2.1 Огляд існуючих методів реалізації синтаксичного аналізу

У пунктах 2.1.1-2.1.5 нижче будуть розглянуті існуючи парсери для синтаксичного аналізу тексту, їх основні підходи, переваги та недоліки. Для огляду бралися кращі з тих, що брали участь в змаганні синтаксичних аналізаторів текстів російською мови в 2011-2012 року. Розглянуті системи використовують різні методи синтаксичного розбору, показуючи в той же самий час кращі результати, з чого можна зробити висновок, що на поточний момент немає єдиного найефективнішого рішення.

2.1.1 ABBYY Syntactic and Semantic Parser

ABBYY Syntactic and Semantic Parser для аналізу тексту використовує два основних інструмента: словник синтаксичних парадигм слів та дерево універсальних семантичних значень і відносин між ними. Словник задає правила вживання лексеми залежно від її класу. В ньому містяться дані про морфологічну парадигму та про велику кількість синтаксичних рівнів. Кожен синтаксичний рівень складається з набору синтаксичних конфігурацій, які уявляюсь собою граматичний вираз, який може зіставляється з нульом або більше інших заповнених граматичних слотів. Для кожного з таких виразів задається множина семантичних слотів, які розглядаються як семантичні інтерпретації [12].

Лексичний підхід, на якому заснована система, для будови дерева універсальних семантичних значень використовує граматику керованих вершинами фразових категорій – Head-driven Phrase Structure Grammar (HPSG). Цей метод використовує лексикон з ієрархічною організацією, де кожна лексична одиниця мови описується ієрархічною структурою властивостей, що містить

граматичну та семантичну інформацію та механізм уніфікації. Уніфікація це метод, що дозволяє двом сумісним елементам структури властивостей з'єднувати інформацію, яку вони містять, в один більший елемент. Це дозволяється робити в тому випадку, якщо вони не містять в своїх структурах конфліктуючих типів або різних атомарних значень одних і тих же властивостей.

У HPSG вводиться два універсальних синтаксичних принципу, а саме:

– принцип вершини HFP (Head Feature Principle) полягає в том, що для будь- якої фразової категорії, де визначена вершина, значення властивості HEAD материнського вузла і значення властивості HEAD дочірнього вузла повинні бути уніфіковані;

– принцип моделі управління (The Valence Principle), що означає, що значення властивостей SPR (специфікатор) і COMPS (комплементи) материнського вузла ідентичні значенням аналогічних властивостей вершинного дочірнього вузла.

Наведений підхід на основі граматичних правил має наступні недоліки:

– строгий лінійний порядок складових в правій частині правила не дозволяє використовувати такі та подібні правила в мовах з відносно вільним порядком синтаксичних складових, до яких відноситься і російська. Це означає, що в лінійному ланцюжку речення іменна група повинна стояти перед прийменниковою;

– правила не здатні аналізувати структури, які слабо піддаються проектування і формалізації, в той час, як вони граматично допустимі в багатьох мовах;

– абсолютна залежність синтаксичних правил від правильності і повноти окремо взятого словникового лексикону;

– трудомісткість розробки лексикону для російської мови;

– відсутність поділу аналізу на рівні і словники (морфологічний, синтаксичний і семантичний) позбавляє архітектуру лексикону прозорості.

Незважаючи на зазначені недоліки підходу, необхідно визнати великий експериментальний потенціал побудованої на HPSG моделі для дослідників в

області штучного інтелекту. Метод аналізу тексту, який використовується ABBYY Syntactic and Semantic Parser, дозволяє виконувати повний аналіз пропозицій з високою точністю. Однак собівартість створення таких ресурсів і специфіка комерційної діяльності, в рамках якої вони були створені, дозволяє припустити, що у вільному доступі ці ресурси не з’являться, і вказує на проблематичність відтворення подібних ресурсів за доступний для огляду час будь-яким науковим колективом.

2.1.2 Парсер граматики зв’язків (LinkParser)

LinkParser з самого початку створювався як апарат для автоматичної системи аналізу речень, що дозволило авторам відійти від академічних уявлень, прийнятих в лінгвістичній традиції. Базова відміна LinkParser полягає в тому, що використовувана модель аналізу є контекстно-вільною граматикою [13]. Кожна одиниця словника граматики описується формулою, що складається з з’єднувачів (connector). З’єднувач складається з імені типу зв’язку (наприклад, S – суб’єкт, O – об’єкт), і суфікса, що визначає вектор напрямку з’єднання. В залежності від суфікса з'єднувач може бути право-спрямованим і ліво-спрямованим. Ліво-спрямований і право-спрямований з’єднувач одного типу утворюють зв’язок (link). Для зв’язку з’єднувачей у формули використовується чотири зв’язки: оператор кон’юнкції (&), оператор диз’юнкції (or), позначення факультативности ({}) і необмеженість повторення (@).

В аналізаторі LinkParser використовується близько ста різних з’єднувачей, які розрізняються переважно нижньому індексом. Число базових дескрипторів при цьому порівняно невелике. Слова узагальнюються в класи з селективним і субкатегоріальним ознаками. В ході аналізу словам в системі приписуються значення їх базових класів – селективних ознак.

У LinkParser вводяться наступні загальні структурні обмеження:

– зв’язки між словами в реченні не повинні перетинатися;

– всі слова в лінійній послідовності повинні бути з’єднані між собою;

– в реченні повинен виконуватися порядок реалізації з’єднань, заданий у формулі несиметричною кон'юнкція для односпрямованих з’єднувачей;

– для однієї пари слів не може бути проведено більше одного зв’язку.

На жаль, LinkParser використовується для роботи в англійською мовою. Використання його для російської мови ускладнено по ряду наступних причин:

– ідея використання ліво- і право-спрямованих з’єднувачей, втрачає свою силу для мови з відносно вільним напрямком зв’язків (особливо для дієслівних груп);

– якщо припустити, що кожний можливий напрямок зв’язку можна маркувати окремим з’єднувачем, то в цьому випадку різко зросте як число базових з’єднувачів, так і число диз’юнктів словоформ, що негативно позначається на швидкості роботи процесора;

– надмірність і значно зростаючий обсяг словника, які виникають в силу морфологічної розвиненості флективної мови, коли кожна морфологічна форма описується окремої формулою, що призведе до ускладнення складання з’єднувачів і до збільшення їх загального числа в граматиці.

Незважаючи на те, що LinkParser в чистому вигляді не можна застосовувати для аналізу речень на російській мові, він по праву вважається одним з найбільш елегантних і детально опрацьованих рішень завдання синтаксичного аналізу англійської мови.

2.1.3 Синтаксичний парсер лінгвістичного процесора ЕТАП-3

Синтаксичний парсер лінгвістичного процесора ЕТАП-3 визначає синтаксичну структуру фрази у вигляді дерева залежностей, яке будується за допомогою спеціальних правил (синтагм) [14]. Цих правил для кожної мови

налічується кілька сотень. Всі вони бінарні. Цей факт означає, що синтагма пов'язує синтаксичним відношенням не слова фрази, а деяку пару омонімів цих слів, якщо вони представлені на початку синтаксичного аналізу декількома (морфологічними або лексичними, або обома) омонимами. Таким чином, омоніми слів фрази можуть зв'язуватися синтаксичними відносинами незалежно один від одного.

В результаті роботи синтагм на першому етапі синтаксичного аналізу виникає граф синтаксичних гіпотез. На подальших етапах синтаксичного аналізатора сторонні зв'язку різними засобами фільтруються. Іншими словами, в основі алгоритму синтаксичного аналізу системи ЕТАП-3 лежить «фільтровий метод».

Однією з проблем, що виникає при роботі парсеру є сторонні інтерпретації, коли парсер допускає кілька варіантів розбору речення. Іншою проблемою є надмірність. Якщо лінгвіст, що обслуговує систему, зустрічає в тексті синтаксичну конструкцію, яка не врахована в синтагмах, то йому достатньо підправити одну з відповідних синтагм або створити нову, щоб виникло відсутнє синтаксичне відношення. Однак часто буває, що деяка мовна конфігурація, будучи занурена в інші контексти, утворює іншу синтаксичну конструкцію і повинна аналізуватися вже інакше. Передбачити всі ці контексти при написанні синтагм неможливо в принципі. Звідси випливає, що синтагми неминуче будуть породжувати в ряді випадків зайві, невірні синтаксичні гіпотези.

Система ЕТАП-3 використовує такі лінгвістичні ресурси.

Перший ресурс це корпус текстів. Завдяки тому, що система ЕТАП-3 знаходиться в експериментальній експлуатації вже досить давно, були синтаксично розмічені десятки тисяч фраз з різного роду текстів (зараз в корпусі текстів близько 37 000 фраз). Всі синтаксичні структури цих фраз спочатку «начорно» будувалися системою ЕТАП-3, а потім вручну редагувалися спеціалістами лінгвістами.

Для подолання надмірності і оптимізації процесу виділення правильної синтаксичної структури з графа гіпотетичних зв'язків застосовують ранжування синтаксичних гіпотез, породжуваних синтагмами, за допомогою впровадження в

систему навчального статистичного компонента. Таким чином, синтаксичний аналізатор ЕТАП-3 використовує емпіричну статистику, породжену лінгвістом експертом. Цим досягаються дві мети: розширюються рамки можливостей побудованої лінгвістом діючої моделі мови та точніше визначаються межі цих можливостей. Це призводить до того, що правильна структура виділяється першої або однією з перших.

2.1.4 SyntAutom

SyntAutom – система, заснована на правилах, побудованих вручну. Відмінна риса цієї системи в тому, що вона має тенденцію пов’язувати значущі слова, тоді як допоміжні слова переносяться на більш низькі рівні дерева залежностей.

Система використовує морфологічний словник, словник валентності дієслів, базу частотності морфологічні інтерпретацій слів, базу частотності бінарних відносин залежності між парами лексичних одиниць, (обчислюються за великим автоматично розміченому корпусу), емпіричні ваги, що додаються, коли автомат перетинає деякі стану автомата.

Таки підхід забезпечує наступні переваги:

– синтаксична і морфологічна неоднозначність вирішуються одночасно в рамках уніфікованого підходу;

– явний опис переходів автомата забезпечує гнучкий спосіб управління процесом парсинга тексту;

– стан автомата, що реалізує парсинг тексту в даній системі, найчастіше надає більше інформації, ніж в змозі забезпечити контекстно-вільна граматика;

– до системи можна легко додавати «локальні» функції, які викликаються тільки в специфічних умовах.

В той же самий час системі, як і іншим системам, заснованим на правилах, притаманні загальні труднощі:

– важко узгодити емпіричні ваги з вагами, які формуються статистичною моделлю;

– існують межі, за якими важко збільшити граматичне покриття, що обумовлено комбінаторним зростанням варіантів синтаксичного розбору і падінням точності синтаксичного аналізу.

2.1.5 SemSin

SemSin [15] – це семантико-синтаксичний аналізатор, в завдання якого входить зняття морфологічної омонімії, побудова синтаксичного дерева залежностей і часткове зняття лексичної неоднозначності.

Аналізатор використовує набір лінгвістичних ресурсів. Першим є словник і класифікатор В. А. Тузова. У ньому кожна лексема містить морфологічні характеристики, а також номер свого класу і моделі управління слів. У словнику містяться загальновживані слова, назви та імена власні. Також використовується база фразеологізмів, яка забезпечує розбір трьох типів словосполучень: незмінних, із змінним першим словом і повністю змінюваних. Наступною є база прийменників, що зберігає класи іменників, з якими вони взаємодіють, і назви зв’язків з головними словами прийменникових груп. Крім того використовується база продукційних правил, що містить близько 210 правил.

У процесі аналізу речення система сегментує його, встановлює головне слово сегмента, може об’єднувати сегменти, підпорядковувати їх. Вихідне речення розбивається по знакам пунктуації на окремі сегменти. Кожному сегменту при цьому присвоюється свій тип, виходячи з наявності чи відсутності підрядного союзу або дієслівної форми. Після завершення роботи сегментації проводиться побудова іменних і прийменникових груп всередині сегментів Таким чином в першій фазі синтаксичного аналізу визначається топологічна структура речення (виділення дієслівних груп і сегментів), у другій фазі відбувається виділення

фразових категорій в межах, визначених межами сегментів. Отже, в першій фазі аналіз пропозиції проводиться «зверху вниз», у другій – «знизу вверх», але на фрагментах менше довжини пропозиції. Така стратегія дозволяє значно знизити обсяг необхідних для її реалізації обчислень.

2.2 Огляд словників валентності

У цьому розділі стисло описані основні чотирьох найбільш повних словників валентності для російської мови. В опис включені способи подання інформації про дієслова та його актантів.

2.2.1 Толково-комбінаторний словник

Толково-комбінаторний словник [5] був створений в рамках теорії «Сенс-Текст» і опублікований І.А. Мельчук та А.К. Жовківський в 1984 р. Метою авторів словника було вичерпно відобразити всі семантичні і синтаксичні можливості слів. Також словник містить у собі функції тлумачного, синонімічного, фразеологічного, словотвірного і інших типів словників.

Словник створювався вручну колективом авторів, і за 10 років роботи над ним було підготовлено трохи більше 250 словникових статей. У словник входять одиниці різних частин мови та фразеологізмів. В основному, описуються три типи відносин між словами: семантичні, лексичні (лексичні функції), синтаксичні (моделі управління). Толково-комбінаторний словник є словником активного типу, тобто спрямованим на породження тексту за змістом. Більш того, це чисто теоретичний словник, спрямований на вузьке коло читачів, а саме, на лінгвістів.

Модель управління – спосіб опису синтаксичної і семантичної валентності дієслова в теорії «Сенс-Текст» [5]. Дієслово представляється у вигляді предиката, що приймає кілька змінних (X, Y, Z і т.д.), дається визначення. Сама модель управління виглядає як таблиця, стовпці якої співвідносяться з семантичними актантами лексеми. У другому рядку представлені способи поверхневої реалізації синтаксичного актанта. Одна лексема може володіти декількома варіантами моделі управління, в тому числі модифікаціями. Приклад моделі управляння у вигляді таблиці наведено на рисунку 2.1.

1 = X [кто радуется или восхищается]	2 = Y [чему радуется или чем восхи-щается]
1. $S_{им}$	1. $S_{тв}$ 2. *тем*, C_{conj} ПРЕДЛ желательно

Рис. 2.1 – Приклад моделі управління для дієслова «радоваться»

Також в словнику вказуються обмеження до моделі управління, тобто умови спільної зустрічальності актантов і закономірності їх сполуки. Наводяться приклади вживання і лексичний функції, що відображають парадигматичні і синтагматический зв’язок лексеми.

2.2.2 Семантичний словник Тузова

Семантичний словник В. А. Тузова містить в собі опис слів російської мови на формальній семантичній мові. Словник будувався на базі великого тлумачного словника російської мови і деяких інших словників. Станом на 2007 рік словник В.А. Тузова містив 164 тисячі слів [16].

Всі слова діляться на 2 великі групи: базові і похідні. Базові слова в рамках даного опису є самостійними та не виражаються через інші. Для них вказується ієрархічний клас, наприклад, слово дитина отримує клас: іменник, фізичний об’єкт, жива істота, людина, дитина. Ієрархія класів базових понять містить понад 1500 класів. Похідні слова описуються не тільки через клас, але і за допомогою базових функцій. Наприклад, дієслово «оценивать» визначається як робити так, щоб щось отримало ціну.

Крім того в словнику представлена валентність дієслів. Вона визначається через синтактико-семантичну формулу слова, яка включає класи і відмінки можливих синтаксичних актантів вихідного дієслова. Приклади синтактико-семантичну формул слова «идти» представлені на рисунку 2.2.

```
ИДТИ

Caus(ЧЕЛОВЕК$1241~!Им,IncepOper01(!Им,ДЕЙСТВИЕ$15~!Куда))
Caus(#,IncepLab(МЕСТО$121~!Им,ДЕЙСТВИЕ$15~!Куда))
Caus(ЖЕНЩИНА$12413130~!Им,IncepCopul(!Им,ЖЕНА$1241355(!заВин)))
Func(ДЕЙСТВИЕ$15\ОСАДКИ$122/15~!Им)
IncepFunc(СОБЫТИЕ$111\ВРЕМЯ$16\СИЛА$15422\ПУТЬ$122431~!Им)
Oper00(#,СОГЛАСИЕ$13127)
Oper00(ОДЕЖДА$12136~!Им,СООТВЕТСТВИЕ$11017(!Дат))
Caus(ЧЕЛОВЕК$1241\НЕКТО$702~!Им,IncepCopul(#,СПЕЦ$124132~!вКого))
Caus(#,IncepUsor(ДЕНЬГИ$1214~!Им,!наВин))
Caus(Oper01(ЧЕЛОВЕК$1241~!Им,ХОД$15304(!поДат,!Откуда,!Куда)),!заТв\Incep(!Инфин))
Caus(КОРИДОР$12326~!Им,Func(ВОЗМОЖНОСТЬ$11081(ХОД$15304(!Где,!поДат,!Откуда,!Куда))))
Oper01(МЕСТО$121~!Им,ХОД$15304(!Где,!поДат,СЛЕДОМ$703(!заТв!),!Откуда,!Куда))
Oper01(МЕСТО$121~!Им,ХОД$15304(!поДат,!Откуда,ПРИРОДА$122\ПОСЕЛЕНИЕ$123~!Куда))
Oper01(ПРИРОДА$122~!Им,ХОД$15304(!поДат,!Откуда,!Куда))
Oper01(ЧАСЫ$1213231~!Им,РАБОТА$1511)
```

Рис. 2.2 – Приклад синтактико-семантичних формул для дієслова «идти»

В основі створення словника було покладено наступні твердження. Слово будь-якої природної мови – це ім’я функції, яка зв’язується з цим словом і називається його семантикою. Кожне конкретне значення слово отримує тільки після підстановки конкретних значень. А речення – це єдина закінчена суперпозиція функцій. Сенс речення обчислюється в процесі виконання суперпозиції.

2.2.3 RussNet

Робота над проектом WordNet була розпочата в 1985 р Дж. Міллером в Прінстонському університеті. В оригінальній версії проекту інформація про валентність дієслова відображалася у вигляді фреймів речень (sentence frames). Вони використовуються тільки для автоматичного породження ілюстративного матеріалу, якщо він не надається лексикографом [17]. Таким чином показується кількість актантів у предиката і варіанти їх заповнення.

WordNet став зразком для створення подібних словників для безлічі інших мов, в тому числі і для російської мови. З усіх російськомовних версій найбільш повно валентності дієслів російської мови представлені в словнику RussNet. Це одна з російських версій проекту WordNet. У RussNet у дієслівного сінсета є список рамок валентностей з вказівкою на те, яка рамка відповідає якому члену сінсета. Відзначається граматична форма аргументів, їх порядок і факультативность, наявність або відсутність прийменника, семантичні характеристики. Більш того, валентність допомагає розрізняти значення багатозначних дієслів. На рисунку 2.3 представлене xml-уявлення рамки валентностей.

```
<VALENCY_FRAME active="yes" main_segment="proposition">
    <VALENCY obligatory="yes" role="arg0" sem_type="object">
        <VARIANTS>
            <VARIANT>
                <morph_data POS="noun|pron" CASE="nom"/>
                <sem_data TYPE="group" ID="RUS-nAnimate"/>
            </VARIANT>
        </VARIANTS>
    </VALENCY>
    <VALENCY obligatory="no" role="arg3" sem_type="object">
        <VARIANTS>
            <VARIANT>
                <morph_data POS="noun|pron" PREP="по" CASE="dat" />
                <sem_data TYPE="top" ID="RUS-nLocation"/>
            </VARIANT>
        </VARIANTS>
    </VALENCY>
</VALENCY_FRAME>
```

Рис. 2.3 – Приклад xml-уявлення рамки валентностей

Рольова характеристика валентності (role) регулярно зустрічається в різних концепціях опису валентності. Для дерева дієслів руху, виділяються наступний набір аргументів: одухотворений або неживий суб'єкт руху; кінцева точка руху; початкова точка руху; перетинати простір; засіб транспорту, яке використовується для руху; особа, яка направляється на своєму шляху; об'єкт, який переноситься при русі і т. п.

2.2.4 Framebank

Словник Framebank створювався як ресурс, де фахівці могли б отримати дані про типи дієслівного управління, їх розподілі і варіативності в різних лексичних одиниць [14]. Основним зразком для Framebank став FrameNet, але при складанні словника враховувалися особливості російської лексичної семантики.

Словник орієнтований лексично, тобто пошук ведеться по лексемі, проте в прикладах вживання розмічаються всі предикатні слова і пов'язані з ними конструкції. Список лексем формувався за їх частотності і різноманітності управління. Базою прикладів з реальних текстів для даного проекту став Національний корпус російської мови. Вибірка була збалансована щодо характеристик пропозицій (довжини, складності), місця предиката в реченні, метатекстових ознак (датування, автора, жанру). Для частотних дієслів також перевірялися контекстні маркери, щоб забезпечити більшу різноманітність вживань. Інформація про моделі управління міститься як в словнику, так і в текстовому вживанні. Приклад моделі управляння наведено на рисунку 2.4.

В результаті з'являється можливість простежити, як реалізуються активні і пасивні валентності дієслів, реляційних імен, прикметників, прислівників і т.д., як впливають на це «нелексичні» граматичні конструкції, наприклад, інфінітивна або компаративна, як це пов'язано з порядком слів, пунктуацією і так далі. Тим самим,

створюються передумови для прориву в ще одній важливій і малодослідженій галузі лінгвістики – у вивченні взаємодії різних конструкцій на просторі тексту.

1.2_Все восторгались картинами молодого художника. Snom V Sins примеры (59)

Все восторгались картинами молодого художника. Он восторгался природой юга. Мы восторгаемся концертами Рихтера.

L	Morph	Rank	Expl	Sem	Status	Comments
X	Snom	Субъект	говорящий - субъект психологического состояния	лицо	v	
-	восторгаться	Предикат	-	-	c	
Z	Sins	Периферия	причина	-	v	

Рис 2.4 – Приклад моделі управління для дієслова «восторгаться»

Кожній моделі управління присвоюється умовна назва, вони містять скорочену назву учасника, стандартний спосіб морфосинтаксичного вираження, синтаксичний ранг, експлікацію учасника, семантичні обмеження на заповнення валентності». Під експлікацією учасника розуміється семантична роль. Також пишеться пояснення, що ототожнює учасника.

3 МЕТОД СИНТАКСИЧНОГО АНАЛІЗУ СЛОВОСПОЛУЧЕНЬ

3.1 Граматика словосполучень для російської мови

Словосполучення в російській мові – це синтаксична одиниця, що складається з двох і більше слів, пов’язаних за змістом і граматично, і будується на основі підрядного зв’язку.

Загальновизнано, що до словосполученням формуються на основі підрядного зв’язку (зв’язку головного і залежного членів). Деякі дослідники визнають також інші види зв’язків: сурядні словосполучення – поєднання однорідних членів речення, а також предикативні словосполучення – підмет і присудок [11]. У даній роботі словосполучення будуть розглядатися з класичної точки зору: зв’язок виконується за допомогою тільки підрядного зв’язку.

Існує три види підрядного зв’язку: узгодження, управління та примикання.

Узгодження – вид підрядного зв’язку, при якому залежне слово узгоджується з головним в роді, числі, відмінку. Головне слово зазвичай іменник, може бути займенником та кількісним числівником. Залежне може бути: прикметником, дієприкметником, займенником або числівником. Обидва слова є змінними. Узгодження є слабким зв’язком. Приклади словосполучень зі зв’язком узгодження: красное платье, моё пальто, второй день, опавший лист.

Управління – вид підрядного зв’язку, при якому залежне слово вживається в тому непрямому відмінку, якого вимагає головне слово. Залежно від того, якими частинами мови виражені головне і залежне слово, розрізняємо види словосполучень зі зв’язком керування [11]:

– дієслово + іменник (поймать мяч);

– прикметник + займенник (всем интересный);

– іменник + іменник (работник фабрики);

– прислівник + іменник (выше дерева);

– числівник + іменник (две страницы);

– прикметник + іменник (понятный детям);

– дієприслівник + іменник, займенник (перепрыгнув яму).

Примикання – вид підрядного зв’язку, при якому залежне слово зв’язується з головним смислом, порядком слів і інтонацією, без застосування службових слів або морфологічних змін. У якості залежного слова виступають незмінні частини мови і їх форми: наріччя, інфінітиви, дієприслівник, незмінні прикметники, форма порівняльного ступеня прикметника. Приклади словосполучень: час пик, дышать спокойной, наука слушать, бежать босиком.

Крім виду підрядного зв’язку, словосполучення ще ділять за частиною речі головного слова. Розрізняють дієслівні, іменні і прислівникові словосполучення.

Дієслівні словосполучення – це синтаксичні одиниці, що складаються з дієслова, дієприкметники або дієприслівників, які з’єднані з іншим словом підрядним зв’язком.

Іменні словосполучення – це поєднання лексем, головним в яких є слово іменний частини мови: іменник, прикметник, числівник або займенник. Словосполучення з іменниками у якості головного слова називають субстантивними, а з прикметниками – ад’єктивними.

Прислівникові словосполучення складають прислівники з залежними словами, між якими існує підрядний зв’язок управління або примикання [11].

Усі види словосполучень за їх головним словом розташовані у таблиці 2.1.

Таблиця 2.1 – Види словосполучень

Вид словосполучення	Підвид	Головне слово	Залежне слово	Тип звя’зка
Дієслівні	Дієслівні	Дієслово	Іменник	Узгодження
			Інфінітив	Примикання
			Займенник	Примикання
			Прислівник	Примикання
			Дієприслівник	Примикання

Кінець таблиці 2.1

Вид словосполучення	Підвид	Головне слово	Залежне слово	Тип звя’зка
Іменні	Субстантивні	Іменник	Іменник	Управління
			Прикметник	Узгодження
			Займенник	Узгодження
			Числівник	Узгодження
			Причастя	Узгодження
			Інфінітив	Примикання
	Ад’єктивні	Прикметник	Іменник	Управління
			Займенник	Управління
			Прислівник	Примикання
			Інфінітив	Примикання
	Кількісні	Числівник	Іменник	Управляння
	Займенникові	Займенник	Іменник	Узгодження
Прислівникові	Прислівникові	Прислівник	Іменник	Управління
			Займенник	Управління
			Прислівник	Примикання

За своїм складом словосполучення діляться на прості, складні та комбіновані. Просте словосполучення утворюється має один підрядний зв’язок. Складне словосполучення утворюється на основі двох або більше підрядних зв’язків, що виходять від одного головного слова. Комбіноване словосполучення утворюється на основі зв’язків, що виходять від різних головних слів, тобто складається з декількох простих або складних словосполучень. Наявність такої ієрархії дозволяє представити будь-яке словосполучення у вигляді дерева. Можливість побудови такого дерева є необхідною і достатньою умовою синтаксичної коректності словосполучення.

На основі створеної таблиці виведемо правила знаходження словосполучень. Оскільки кількість можливих видів словосполучень кінцеве, то зручно використовувати атрибутивну контекстно-вільну граматику. Формальна граматика це спосіб опису формальної мови, тобто виділення деякого підмножини з безлічі всіх слів деякого кінцевого абетки. Сенс терміну «контекстно-вільна» полягає в тому, що є можливість застосувати продукцію до нетерміналу, причому незалежно від контексту цього нетермінала. Граматика складається з списка правил, які називають продукціями. В контекстно-вільній граматиці ліві частини всіх продукцій є поодинокими нетерміналом [20]. Продукції граматики мають такий вигляд:

$$A_{i_1,\dots,i_n} \rightarrow \alpha_{i_1,\dots,i_n} \tag{3.1}$$

де A – символ нетермінального алфавіту;

α – рядок граматичних символів;

$i_1, \dots, i_n$ – атрибути, які описують морфологічні характеристики слів.

Атрибутивна КВ-граматика в загальному випадку має наступний вигляд:

$$G = (V_t, V_n, P, S) \tag{3.2}$$

де V_t – абетка термінальних символів. В даному випадку – це множина частин мови, які формують словосполучення;

V_n – алфавіт нетермінальних символів. Він містить типи словосполучень в залежності від головного слова, які можуть формувати символи термінального алфавіту;

P – система продукцій. Вона містить список правил формування словосполучень;

S – аксіома. Це символ нетермінального алфавіту. Якщо поточна вхідний рядок згортається в даний символ, то синтаксичний аналіз вважається завершеним. У даній граматиці це словосполучення.

Для наочності будемо позначати термінальні символи малими літерами, а нетермінальні великими.

В граматиці необхідно врахувати що словосполучення утворюють ієрархію від простого до комбінованого словосполучення. Для цього необхідно, щоб було правило, ліва частина якого сполучення, а права – словосполучення, де в ролі головного слова виступає слово або словосполучення з таким головним словом, граматичні характеристики якого представляє права частина, а в ролі залежного слова – інші слова або словосполучення, граматичні характеристики яких описують інші символи правій частині продукції.

Окремо варто відзначити термінальний символ, який не є видом словосполучення, а являє собою прийменник. У лівій частині правила він розкривається як нетермінальниий символ прийменник або як порожній символ, за допомогою чого можна організовувати словосполучення як з прикметниками, так і без них.

Опишемо КВ-граматику, що описує синтаксис словосполучень російської у формальному виді. Множина термінальних символів (V_t)буде містити:

– verb (дієслово);

– noun (іменник);

– adj (прикметник);

– adv (прислівник);

– inf (інфінітив);

– pron (займенник);

– part (причастя);

– numeral (числівник);

– gerund (дієприслівник);

– prepos (прийменник).

Множина нетермільних символів (V_n) містить:

– VerbP (дієслівне словосполучення);

– NounP (субстантивне словосполучення);

– PronP (займенникове словосполучення);

– NumP (кількісне словосполучення);

– AdjP (ад’єктивне словосполучення);

– AdvP (прислівникове словосполучення);

– Prep (прийменник);

– Phrase (словосполучення).

Система продукцій (P) складається з десяти правил та зазначена на рисунку 3.1.

P = {
(1) Phrase → <VerbP> | <NounP> | <PronP> | <NumP> | <AdjP> | <AdvP> ,
(2) VerbP → <VerbP> <NounP> | <VerbP> <AdjP> | <VerbP> <PronP> | <VerbP> <AdvP>,
(3) VerbP → <VerbP> gerund | <VerbP> inf | <Prep> verb,
(4) NounP → <AdjP> <NounP> | <NounP> <NounP> | part <NounP> | <NumP> <NounP> | <NounP> inf | <PronP> <NounP> ,
(5) NounP → <Prep> noun,
(6) PronP → <PronP> <PronP> | <PronP> <AdjP> | <Prep> pron,
(7) NumP → <NumP> <NounP> | <NumP> <AdjP> | <NumP> part | <Prep> numeral,
(8) AdjP → <AdjP> <NounP> | <AdjP> inf | <AdjP> <AdvP> | <Prep> adj,
(9) AdvP → <AdvP> <NounP> | <AdvP> <AdjP> | <AdvP> <AdvP> | <Prep> adv,
(10) Prep → prepos | ε
};

Рисунок 3.1 – Система продукцій для граматики словосполучень

Аксіома (S) дорівнює термінальному символу Phrase, тобто словосполученню.

Описана система правил описує синтаксис словосполучень в загальному вигляді [18]. Вона описуємо всі можливі схеми формування словосполучень тільки на основі комбінування частин мови, але не враховує можливість існування зв’язків між головним і залежним словом. Тому необхідно описати підрядні зв’язків тих частин мови, слова яких виступають в ролі головного слова словосполучення. Такий опис виконується за допомогою завдання семантичних правил для кожної продукції, яка описує структурну схему словосполучення. Семантичне правило виконує перевірку значень атрибутів слів, що з’єднуються. Вид семантичного

правила залежить від виду підрядного зв’язку між головним і залежним словом і від частин мови, до яких належать ці слова.

Далі мається на увазі: X –головне слово словосполучення, c – морфологічна характеристика, n – кількість морфологічних характеристик, Y – залежне слово.

Узгодження. Повинні співпадати морфологічні характеристики слів, тому в загальному випадку правило буде виглядати так:

```
if (X.cᵢ = Y.cᵢ,   i = 1,..,n)
    return true;
else
    return false;
```

Управління. Головне слово вимагає непрямого відмінка від залежного. В загальному вигляді правило виглядає так:

```
if (X.відмінок == «іменний» && Y.відмінок !=
«іменний»)
    return true;
else
    return false;
```

Примикання не має семантичних правил тому що головне слово є незмінним і в нього немає морфологічних характеристик.

Правила такого виду описують підрядні зв’язки тільки в найзагальнішому випадку і не враховують окремі випадки, які виникають при підрядних зв’язках у слів різних частин мови. Вони враховуються при описі правил для конкретних продукцій. Нижче наведено приклад опису семантичних правил для субстантивного словосполучення на основі іменника (четверта продукція) [19]. Вона складається з шести правил. Розглянемо перше правило:

$$NounP \rightarrow < AdjP > < NounP > \quad (3.3)$$

де $NounP$ – субстантивне словосполучення;

$AdjP$ – ад’єктивне словосполучення.

Тип зв’язку узгодження. В загальному випадку повинні співпадати морфологічні характеристики слів. В даному випадку якщо співпадає число і воно має значення однини, то повинні співпадати роди та відмінки, якщо число має значення множини, то повинні співпадати відмінки. Приклад «блестящее кольцо». У коді це правило буде виглядати так:

```
if(AdjP.число == NounP.число){
     if(AdjP.число == «однина»){
          if(AdjP.рід == NounP.рід && AdjP.відмінок ==
          NounP.відмінок)
               return true;
     else
          return false;
     }
     if(AdjP.число == «множина»){
          if(AdjP.відмінок == NounP.відмінок)
               return true;
     else
          return false;}}
else
     return false;
```

Друге правило складається з двох субстантивних словосполучень:

$$NounP \rightarrow < NounP > < NounP > \quad (3.4)$$

де $NounP$ – субстантивне словосполучення.

Тип зв’язку управляння. Головне слово вимагає від залежного певного відмінку. Рід та число можуть не співпадати. В даному випадку головне слово йде першим и вимагає родового відмінку. Приклад «учитель физики». У коді це правило буде виглядати так:

```
if(NounP2.відмінок == «родовий»)
     return true;
else
     return false;
```

Третє правило складається із з’єднання причастя та субстантивного словосполучення:

$$NounP \rightarrow part < NounP > \quad (3.5)$$

де $NounP$ – субстантивне словосполучення;

$part$ – причастя.

Тип зв’язку узгодження. Якщо співпадає число і воно має значення однини, то повинні співпадати роди та відмінки, якщо число має значення множини, то повинні співпадати відмінки. Приклад «брошенный камень». У коді це правило буде виглядати так:

```
if(part.число == NounP.число)
{
    if(part.число == «однина»)
    {
        if(part.рід == NounP.рід && part.відмінок ==
    NounP.відмінок)
            return true;
        else
            return false;
    }if(part.число == «множина»){
        if(part.відмінок == NounP.відмінок)
            return true;
        else
            return false;}
}else
    return false;
```

Наступне правило складається з субстантивне словосполучення та кількісного словосполучення.

$$NounP \rightarrow < NumP > < NounP > \quad (3.6)$$

де $NounP$ – субстантивне словосполучення;

$NumP$ – кількісне словосполучення.

Тип зв’язку узгодження. Якщо числівник порядковий та співпадає число і воно має значення однини, то повинні співпадати роди та відмінки, якщо число має значення множини, то повинні співпадати відмінки. Приклад «первый этаж». Якщо числівник кількісний, то повинен співпадати відмінок (наприклад, «семь стульев»), за виключення числівників «два» та «оба», де повинні співпадати відмінок та рід (наприклад, «две шляпы», «обе сосны»), та числівника «один», де повинні співпадати род, число та відмінок (наприклад, «одни штаны»)[]. У додатку А приведено код.

Наступне правило містить субстантивне словосполучення разом з інфінітивом. Тип зв’язку в такому словосполученні примикання, тому семантичних правил нема.

$$NounP \rightarrow < NounP > \ inf \tag{3.7}$$

де $NounP$ – субстантивне словосполучення;

inf – інфінітив.

Останнє правило містить займенникове словосполучення разом з субстантивним.

$$NounP \rightarrow < PropP >< NounP > \tag{3.8}$$

де $NounP$ – субстантивне словосполучення;

$PropP$ – займенникове словосполучення.

Тип зв’язку узгодження. Якщо співпадає число і воно має значення однини, то повинні співпадати роди та відмінки, якщо число має значення множини, то повинні співпадати відмінки. Приклад «мой дом». Код для цього правила приведено в додатку А.

3.2 Синтаксичний парсер

Однією з найбільш ефективних технологій синтаксичного аналізу, які працюють з великим класом КС-граматик, є технологія LR-аналізу. Синтаксичний аналіз за допомогою даної технології виконується LR-аналізатором, що виконує такий аналіз за допомогою алгоритму LR-аналізу.

Букву R в назві аналізатора означає right – напрямок роботи аналізатору. Це значить, що ми йдемо від крайніх правих частин правил до аксіоми, тобто з більш простих правил збираємо вихідне правило.

Аналізатору потрібно кілька речей для його роботи:

– вхідний потік, який будемо аналізувати;

– стек (структура даних, яка відповідає правилу Last In First Out), для станів парсеру;

– таблиця дій (action), яка підказує що нам робити в поточному стані і з поточним символом на вході;

– таблиця переходів (goto).

Загальна схема LR-аналізатора наведена на рисунку 3.2.

Опишемо, як буде працювати аналізатор. Поточним станом вважається стан на вершині стека. Дивимося в таблицю дій (індексами є поточний вхідний символ і поточний стан). У цій таблиці можуть бути 4 типи записів:

– accepted – вхідний рядок належить даній граматиці;

– error – немає ніяких дій, ми в глухому куті, користувач помилився з поточним символом;

– shift – на вершину стека кладемо стан яке відповідає вхідному символу, читаємо наступний;

– reduce – у нас в стеці набралися стану, які ми можемо замінити одним, використовуючи правило граматики, тут використовується значення з таблиці переходів.

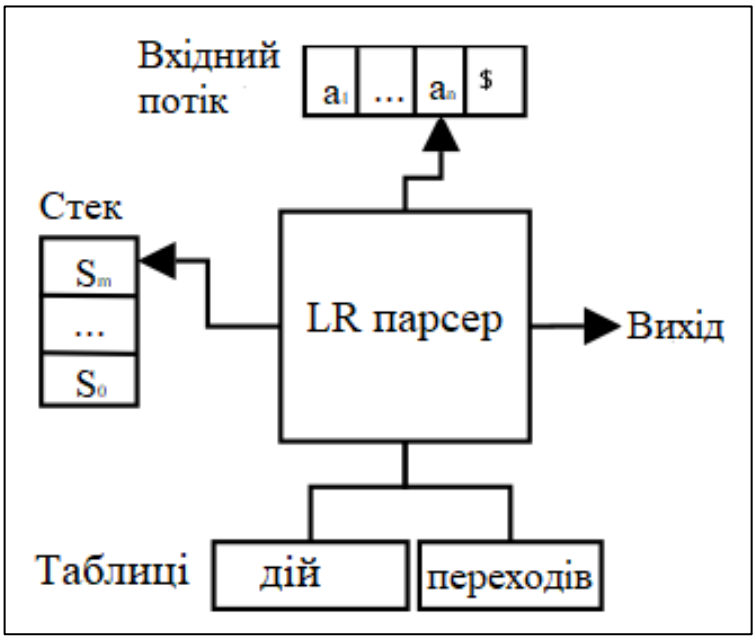

Рисунок 3.2 – Загальна схема роботи LR-аналізатору

Для побудови таблиць LR-аналізу існує кілька методів: SLR, canonical LR, LALR. Вони відрізняються складністю побудови таблиць і класом граматик, з якими дані методи працюють [].

Simple LR (SLR) – найслабший з методів за кількістю граматик, з якими він працює, однак найпростіший в реалізації.

Етап розпізнавання будь-якої продукції представляється у вигляді пункту – продукції граматики з точкою в правій частині, яка показує розпізнану (зліва від точки) і очікувану (праворуч від точки) частину продукції. Стани в таблиці синтаксичного аналізу будуються на основі канонічної системи пунктів – системи множин, в якій кожна множина складається з даних пунктів і відповідає певному стану в таблиці аналізу. Для побудови даної системи необхідно визначити розширену граматику G', яка є вихідною граматикою з новим стартовим символом S' і продукцією $S' \rightarrow S$. Кожна множина в даній системи формується за допомогою операції замикання – функції $closure(I)$, де I – множина пунктів граматики G. Результатом виконання даної функції є множина пунктів. Кожна множина пунктів в канонічній системі пунктів відповідає стану, в яке здійснюється перехід з будь-

якого стану при будь-якому вхідному символі, і є результатом виконання операції goto. Результатом виконання даної функції є замикання всіх пунктів.

Для побудови таблиці SLR-аналізу необхідно побудувати систему множин пунктів $C = \{I_0, I_1, \ldots, I_n\}$. Значення в комірках рядка таблиці синтаксичного аналізу (стан) і визначаються на основі I_i за формулою 3.9:

$$\begin{gathered} action\,[i,a] = "shift\ j", [A \rightarrow \alpha \cdot a\beta] \in I_i, goto\,(I_i,a) = I_j, \\ action\,[i,a] = "reduce\ A" \rightarrow \alpha [A \rightarrow \alpha \cdot] \in I_i, \forall a \in FOLLOW\,(A), \\ action\,[i,\$] = "accepted", [S' \rightarrow S] \in I_i, \\ goto\,[i,A] = j, goto\,(I_i,A) = I_j, \forall A \end{gathered} \tag{3.9}$$

де i – поточний елемент;

I – множина пунктів граматики;

\$ – символ кінця строки;

A – нетермінальний символ;

a – термінальний символ.

Всі інші записи в таблиці вказуються як «error», тобто попадання на такий запис означає неможливість зробити замикання.

3.3 Синтаксичний аналіз у складі семантичного аналізу

Описаний вище метод синтаксичного аналізу може використовуватися як частина в модулі семантико-семантичного аналізу. Коротко опишемо особливості методу синтаксичного аналізу, реалізованого в даному модулі, і їх вплив на ефективність його роботи. При цьому будемо оперувати поняттям «узагальнений базовий елемент», поняттям сегмент.

Сегмент визначається наступними значеннями: послідовність елементів, що становлять «тіло» сегмента, кожен з яких є або узагальненим базовим елементом або сегментом.

Зауважимо, що сегменті за структурою дуже схожі на словосполучення. Тому тут можна застосувати алгоритм, описаний в попередньому розділі. Завдяки синтаксичному парсеру можна розбити речення на словосполучення та замінити кожне словосполучення його головним словом.

Структури даних, що описують сегмент і узагальнений базовий елемент, також містять поля для зберігання альтернативної інтерпретації сегмента. Альтернативна інтерпретація (написання та його морфологічна інформація) може «представляти» весь сегмент в батьківському сегменті. При написання альтернативної інтерпретації використовуємо займенникові слова: тоді-то; тому що; там-то, такий-то; той-то; так то; тому-то; то-то, щось. Це потрібно, якщо узагальнений базовий елемент являє собою ідіоматичний вираз або сегмент відповідає неморфологізірованому члену речення.

Наприклад, візьмемо речення: «Участливая ученица старших классов немедленно предложила аккуратно сложенный платок».

Розіб’ємо це речення на словосполучення та виділимо в кожному головне слово. Результат показано на рисунку 3.3:

(Участливая ученица старших классов) (немедленно предложила) (аккуратно сложенный платок).

Рисунок 3.3 – Розбиття речення на словосполучення та виділення головних слів

Останнім етапом замінимо кожне словосполучення його головним словом. Таким чином виконаємо спрощення речення. Результат зображено на рисунку 3.4. Такий підхід дозволяє скоротити кількість елементів, які підлягають аналізу на наступних етапах.

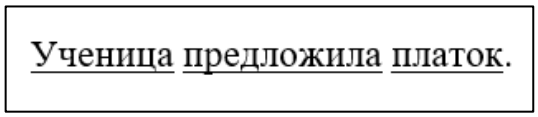

Рисунок 3.4 – Спрощення речення

На наступних етапах проводиться розбиття речень на сегменти по знакам пунктуації та визначення типу кожного сегмента, пошук кордонів простих речень у складі складнопідрядних речення та визначення типу зв’язку між ними, встановлення зв’язків між однорідними членами і в кінці отримання предикатной структури речення, представленою фрагментом дерева залежності, в якому дугам приписані типи семантичного зв’язку, а для дієслова відзначений його семантичний клас.

3.4 Програмна реалізація

По описам алгоритмів в розділах 3.1 та 3.2 була реалізована програмна система.

Система написана на мові програмування C# з використанням технологій ASP.NET та Entity Framwork. В якості програмної среди використовувалось MS Visual Studio.

Програмна система складається з декількох класів:

– клас, який виконує морфологічний аналіз. Він повертає морфологічні ознаки слів. Для цього використовується зовнішня бібліотека AOT;

– клас, який зберігає морфологічні правила для контекстно-вільної граматики та таблиці action, goto;

– клас, який виконує LR-алгоритм;

– клас, який зберігає отриманий результат у базу даних;

– клас, який відповідає за зв’язок зовнішнього інтерфейсу з класами, які проводять аналіз даних. Цей клас збирає текст, введений користувачем та відображає результат аналізу.

Програма дозволяє користувачу ввести текст, який він хоче проаналізувати. Аналіз можна запустити кнопкою під вікном введення тексту. Пісня аналізу користувач отримує розбиття кожного речення на словосполучення. В разі коли у реченні була помилка або програма не змогла зробити аналіз, видається повідомлення про це, та вказується на якому етапі трапилась помилка: під час морфологічного аналізу (це значить програма не змогла розпізнати окреме слово, в слові помилка чи цього слова нема у словнику), чи під час синтаксичного аналізу (проблема в узгодженні слів в реченні).

3.5 Можливості використання отриманих результатів

Отриманий в попередньому розділі семантико-синтаксичний аналізатор можна застосовувати для чорнової розмітки корпусів тексту. Для створення корпусу тексту необхідна безпосередня участь людини, що робить створення корпусу ресурсовитратним. Семантико-синтаксичний аналізатор допоможе визначити такі основні частини речення, як об’єкт, який виконує дію, тобто підмет, саму дію, тобто присудок та суб’єкт, над яким виробляється дія. Це основні характеристики в реченні, визначення яких автоматично призведе до зниження витрат часу. Однак можуть знадобиться і інші характеристики в реченні, які необхідно визначати людині. До того ж для правильного навчання систем потрібні правильно розмічені корпуси, тому необхідна людська перевірка результатів роботи.

Іншим можливим способом використання отриманої системи є перевірка узгодження частин мови між собою. Речення повинне повністю розбиватись на словосполучення. Якщо цього не відбувається, то в реченні є помилка. Помилку в

узгодженні слів можна визначити по тому, що граматика знаходить ланцюжок слів, який потенційно може бути словосполученням, але морфологічні правила не пропускають цей ланцюжок як словосполучення, і після цього деякі слова в реченні не відносяться ні до якого словосполучення. Використовуючи граматику і морфологічні правила можна припустити які морфологічні характеристики повинно мати слово без помилки. Такого роду помилки часто виникають при перекладах між мовами зі схожою структурою граматики, але різними морфологічними характеристиками в деяких словах, наприклад, при перекладах між українською та російською мовою, в яких роди багатьох слів відрізняються. Наприклад, слово мова в українській мові має жіночий рід, а в російській – чоловічий.

ВИСНОВКИ

Під час написання атестаційної роботи магістра було проведено аналіз предметної галузі, розглянуто існуючи аналоги та проведено аналіз методів синтаксичного аналізу для систем автоматичної обробки текстової інформації.

Спочатку було розглянуто яке місце займає автоматична обробка текстів на даний час серед актуальних напрямків дослідження та для вирішення яких задач вона може застосовуватися. Був зроблений висновок, що це сучасне і актуальне рішення для великої кількості задач, дослідження в якому зараз ведуться та потребують нових напрацювань. Було прийнято рішення розглядати обробку текстів на російській мові. Таке рішення обумовлено широкою розповсюдженістю мови в світі та її використання в Інтернеті. Були розглянуті основні перешкоди, які з’являються при обробці натуральних мов в взагалі и в російський мові зокрема.

Оскільки за автоматичну обробку тесту відповідає лінгвістичний процесор, то було розглянуто його типову внутрішню будову. Він має декілька рівнів, за які відповідають окремі програмні блоки. Таке розділення обумовлено наявністю в природних мовах також розділення на рівні. Були розглянути рівні лексичного, морфологічного, синтаксичного, семантичного аналізу. При розгляді було встановлено основні задачі кожного етапу, їх особливості та труднощі, що виникають. Особливо ретельно були розглянути існуючи підходи до синтаксичного аналізу.

Також було відзначено, що для нового способу обробки природник мов – машинного навчання – існує проблема нехватки матеріалу для навчання – розмічених корпусів тексту. Було прийнято рішення застосовувати синтаксичний аналіз текстів для напівавтоматичної розробки корпусів. Була поставлена задача для атестаційний роботи. Було вирішено в рамках синтаксичного аналізу проводити пошук словосполучень замість побудови дерева речення.

Наступний етапом розглядалися існуючи аналоги для синтаксичних парсеров. Для огляду були вибрані ті парсери, що показали найкращі результати.

Треба зазначити, то парсери були реалізовані різними підходами. Були зазначені сильні сторони цих підходів, труднощі під час реалізації та допущення під час обробки речень.

Також було розглянуто поняття валентності та словники валентності для російській мови.

Проводилось побудова математичної моделі. Оскільки для реалізації синтаксичного аналізу було вибрано використовувати знаходження словосполучень в реченні та поняття валентності для дієслів, в першу чергу була побудована модель словосполучення в російській мові.

Для цього було розглянуто які види словосполучень існують в російській мові та яким чином в них зв’язуються слова. Для кожного з трьох зв’язків були розглянуті морфологічні засоби, які реалізують ці зв’язки у мові. Для зручності була створена зведена таблиця з типами словосполучень та видами зв’язків. Пізніше по цій таблиці було створено атрибутивну контекстно-вільну граматику для опису словосполучення. Оскільки граматика базувалася тільки на з’єднанні частин мови, цього було не достатньо, тому були написані ще семантичні правила, які використовували морфологічні характеристики слів.

Було реалізовано додаток, який використовував написану граматику та синтаксичні правила. У наданому реченні додаток знаходить словосполучення.

У кінці було описано яким чином можна використовувати отриманий синтаксичний парсер як частину в семантико-синтаксичному парсері, який працює на наступному рівні. Також були запропоновані два інших способи використання. Перший – використовувати парсер для створення напівчорнової розмітки для корпусів тексту, а другий – для пошуку помилок узгодження частин мови в реченні.

ПЕРЕЛІК ПОСИЛАНЬ

1. Ингерсолл Г. Обработка неструктурированных текстов. Поиск, организация и манипулирование / Г. Ингерсолл– М.: ДМК Пресс, 2015. – 428 с.

2. Белоногов Г. Языковые средства автоматизированных информационных систем. / Г. Белоногов, Б. Кузнецов – М.: Наука. Главная редакция физико-математической литературы, 1983. – 288 с.

3. Маккьюин К. Дискурсивные стратегии для синтеза текста на естественном языке / К. Маккьюин – М.: Прогресс, 1989. – с.311-356.

4. Большакова Е. Автоматическая обработка текстов на естественном языке и анализ данных / Е.И. Большакова, К.В Воронцов. та ін. – НИУ ВШЭ, 2017. – 269 с.

5. Мельчук И. Опыт теории лингвистических моделей «СМЫСЛ ⇔ ТЕКСТ» / Мельчук И. А. – М.: Наука, 1974. – 492 с.

6. Лингвистический энциклопедический словарь под ред. В. Н. Ярцевой / М.: Советская энциклопедия, 1990. – 685 с.

7. Леонтьева Н. Н. Автоматическое понимание текстов: Системы, модели, ресурсы: навч. посіб. – М.: Академия, 2006. – 629 с.

8. Открытый корпус (OpenCorpora). [Електронний ресурс] URL http://opencorpora.org/ (дата звернення: 01.12.19)

9. Национальный корпус русского языка [Електронний ресурс] URL:http:// www.ruscorpora.ru/ (дата звернення: 01.12.19)

10. ГИКРЯ: генеральный интернет-корпус русского языка. [Електронний ресурс] URL: http://www.webcorpora.ru/ (дата звернення: 01.12.19)

11. Синтаксис русского языка [Електронний ресурс] URL: https://russkiiyazyk.ru/ (дата звернення: 01.12.19)

12. Anisimovich K. V., Druzhkin K. Ju., Minlos F. R., Petrova M. A., Selegey V. P., Zuev K. A. Syntactic and semantic parser based on ABBYY Compreno linguistic

technologies / Вып. 11 (18): В 2 т. Т. 2: Доклады специальных секций – М.: Изд-во РГГУ, 2012. – с. 91-103.

13. И.М. Ножов Морфологическая и синтаксическая обработка текста (модели и программы) [Електронний ресурс] URL: http://www.aot.ru/docs/Nozhov/msot.pdf (дата звернення: 01.12.19)

14. Оценка методов автоматического анализа текста: морфологические парсеры русского языка Ляшевская О. Н. и др.// Международная конференция "Диалог" (2010) Т. 9. Вып. 16. М.: РГГУ, 2010, С. 318-326

15. Семантико-синтаксический анализатор SemSin [Електронний ресурс] URL: http://www.dialog21.ru/digests/dialog2012/materials/pdf/Kanevsky.pdf (дата звернення: 01.12.19)

16. Тузов, В.А. Компьютерная семантика русского языка. СПб, 2003. – с. 45-51.

17. WordNet. Verb Frames. [Електронний ресурс] URL: http://wordnet.princeton.edu/man/wninput.5WN.html (дата звернення: 01.12.19)

18. Валенда Н.А., Кобзарь А.В. Синтаксический анализатор простых конструкций русского языка на основе атрибутных LR-грамматик. Восточно-Европейский журнал передовых технологий. – 2008. –№ 4/2 (34). – с. 8–12.

19. Валенда Н.А., Самофалов Л.Д., Павленко Н.А. Анализ словосочетаний русского языка на основе LR-грамматик Системы обработки информации – 2016. – № 8(145), с. 69-72

20. Захарія Л. М., Заяць М. М. Формальні мови, граматики та автомати: навч. посіб. – Львів : Видавництво Львівської політехніки, 2016. – 196 с.