



Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное
учреждение высшего образования
«Московский государственный технический университет
имени Н. Э. Баумана
(национальный исследовательский университет)»
(МГТУ им. Н. Э. Баумана)

ФАКУЛЬТЕТ «Информатика и системы управления»

КАФЕДРА «Программное обеспечение ЭВМ и информационные технологии»

Отчет по рубежному контролю №1 по курсу «Анализ алгоритмов»

Тема Граф синтаксических зависимостей в тексте

Студент Пронина Л.Ю.

Группа ИУ7-54Б

Оценка (баллы) _____

Преподаватель Волкова Л. Л.

Москва — 2023 г.

Содержание

Введение	3
1 Аналитическая часть	4
1.1 Граф синтаксических зависимостей текста	4
2 Конструкторская часть	6
2.1 Разработка алгоритмов	6
3 Технологическая часть	10
3.1 Средства реализации	10
3.2 Листинги кода	10
4 Исследовательская часть	12
4.1 Демонстрация работы программы	12
Заключение	13
Список используемых источников	14

Введение

Целью данной работы является описание построения графа синтаксических зависимостей в тексте при помощи библиотек: синтаксического анализа (*spacy*) и морфологического анализатора (*py morphology2*).

Для достижения поставленной цели необходимо выполнить следующие задачи:

- 1) описать понятие графа синтаксических зависимостей в тексте;
- 2) описать алгоритм построения рассматриваемого графа;
- 3) реализовать программу, выполняющую построение графа синтаксических зависимостей в тексте.

1 Аналитическая часть

В этом разделе будет представлена информация по поводу графа синтаксических зависимостей в тексте.

1.1 Граф синтаксических зависимостей текста

Дерево синтаксических зависимостей предложения представляет собой структуру, которая отображает связи между словами в предложении и их синтаксические роли.

В таком графе каждое слово представлено узлом, а связи между словами представлены направленными ребрами. Каждое ребро указывает зависимость между двумя словами, где одно слово является зависимым (дочерним), а другое - главным (родительским).

Синтаксическое дерево помогает визуализировать структуру предложения и понять связи между словами. Оно может быть использовано для различных задач обработки естественного языка, таких как синтаксический анализ, машинный перевод, именованное сущностное распознавание и другие.

Граф синтаксических зависимостей текста строится на основе деревьев синтаксических зависимостей предложений. В нем каждое слово представлено узлом (возможно, с дополнительной информацией, такой как лемма или часть речи), а связи между словами являются направленными ребрами, причем ребра помечаются количеством таких связей между соответствующими словами.

Библиотека *spacy* предоставляет инструменты для анализа синтаксических зависимостей и визуализации деревьев синтаксических зависимостей, которые помогают исследователям и разработчикам в понимании структуры текста и автоматическом анализе естественного языка.

А библиотека *py morphology2* предоставляет функции для получения информации о словах, такой как лемма (нормальная форма слова), грамматическая информация, часть речи и т.д. Она основана на морфологической

разметке и обучена на большом количестве русских текстов.

Вывод

В данном разделе было рассмотрено понятие графа синтаксических зависимостей.

2 Конструкторская часть

В этом разделе будут представлены схемы алгоритма построения графа синтаксических зависимостей в тексте.

2.1 Разработка алгоритмов

На рисунках 2.1 и 2.2 представлены схемы алгоритма построения графа синтаксических зависимостей в тексте.

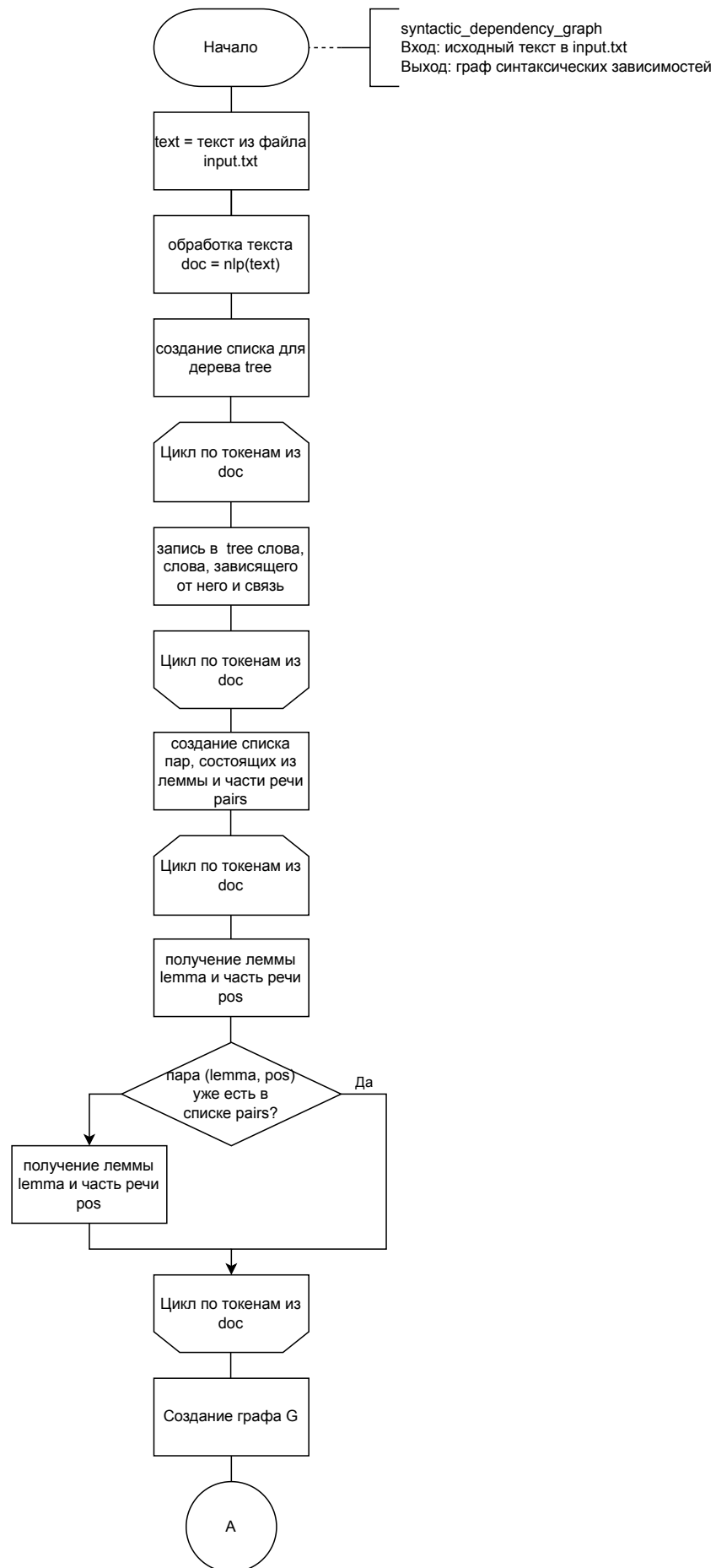


Рисунок 2.1 – Схема алгоритма построения графа синтаксических зависимостей в тексте (1 часть)

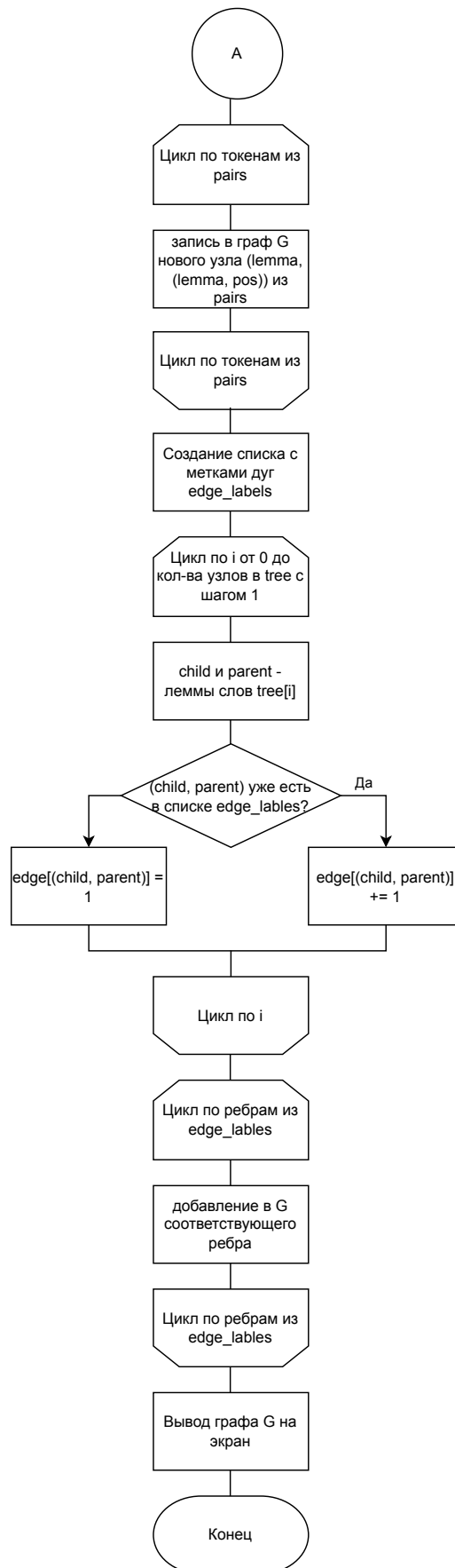


Рисунок 2.2 – Схема алгоритма построения графа синтаксических зависимостей в тексте (2 часть)

Вывод

В данном разделе были представлены схемы алгоритма построения графа синтаксических зависимостей в тексте.

3 Технологическая часть

В данном разделе будут представлены средства реализации и листинги алгоритмов построения графа синтаксических зависимостей в тексте.

3.1 Средства реализации

Для реализации был выбран язык программирования *Python* [1].

3.2 Листинги кода

В листинге 3.1 представлена реализация алгоритма построения деревьев синтаксических зависимостей предложений, а также построения пар, состоящих из леммы и части речи слова. А в листинге 3.2 представлена реализация алгоритма построения графа синтаксических зависимостей и вывода его на экран.

Листинг 3.1 – Алгоритм построения деревьев синтаксических зависимостей

```
1 import spacy
2 import pymorphy
3 nlp = spacy.load('ru_core_news_sm')
4 morph = pymorphy2.MorphAnalyzer()
5 with open('input.txt', 'r', encoding='utf-8') as file:
6     text = file.read()
7 doc = nlp(text)
8 tree = []
9 for token in doc:
10     if token.dep_ != 'punct' and token.dep_ != 'ROOT':
11         tree.append((token.head.text, token.text, token.dep_))
12 pairs = []
13 for token in doc:
14     parsed_token = morph.parse(token.text)[0]
15     lemma = parsed_token.normal_form
16     pos = parsed_token.tag.POS
17     if pos and (lemma, pos) not in pairs:
18         pairs.append((lemma, pos))
```

Листинг 3.2 – Алгоритм построения графа синтаксических зависимостей и вывода его на экран

```
1 import networkx as nx
2 import matplotlib.pyplot as plt
3 G = nx.DiGraph()
4 for token in pairs:
5     G.add_node(token[0], label=f"{token[0]}_{token[1]}")
6 edge_labels = {}
7 for i in range(len(tree)):
8     parsed_token = morph.parse(tree[i][0])[0]
9     child = parsed_token.normal_form
10    parsed_token = morph.parse(tree[i][1])[0]
11    parent = parsed_token.normal_form
12    if (child, parent) not in edge_labels:
13        edge_labels[(child, parent)] = 1
14    else:
15        edge_labels[(child, parent)] += 1
16 for edge in edge_labels:
17     G.add_edge(edge[0], edge[1])
18 plt.figure(figsize=(10, 8))
19 pos = nx.spring_layout(G, seed=42)
20 nx.draw(G, pos, labels=nx.get_node_attributes(G, 'label'),
21         node_color='lightblue', node_size=2000, arrows=False)
21 nx.draw_networkx_edge_labels(G, nx.spring_layout(G, seed=42),
22                             edge_labels=edge_labels)
22 plt.show()
```

Вывод

Были представлены листинги реализации алгоритма построения графа синтаксических зависимостей текста.

4 Исследовательская часть

В данном разделе будет приведен пример работы программы.

4.1 Демонстрация работы программы

На рисунке 4.1 представлен пример работы программы для текста «Я люблю лето. Летом светит солнце ярко. Я люблю солнце. Солнце светит ярко на цветы. Цветы летом красивые.».

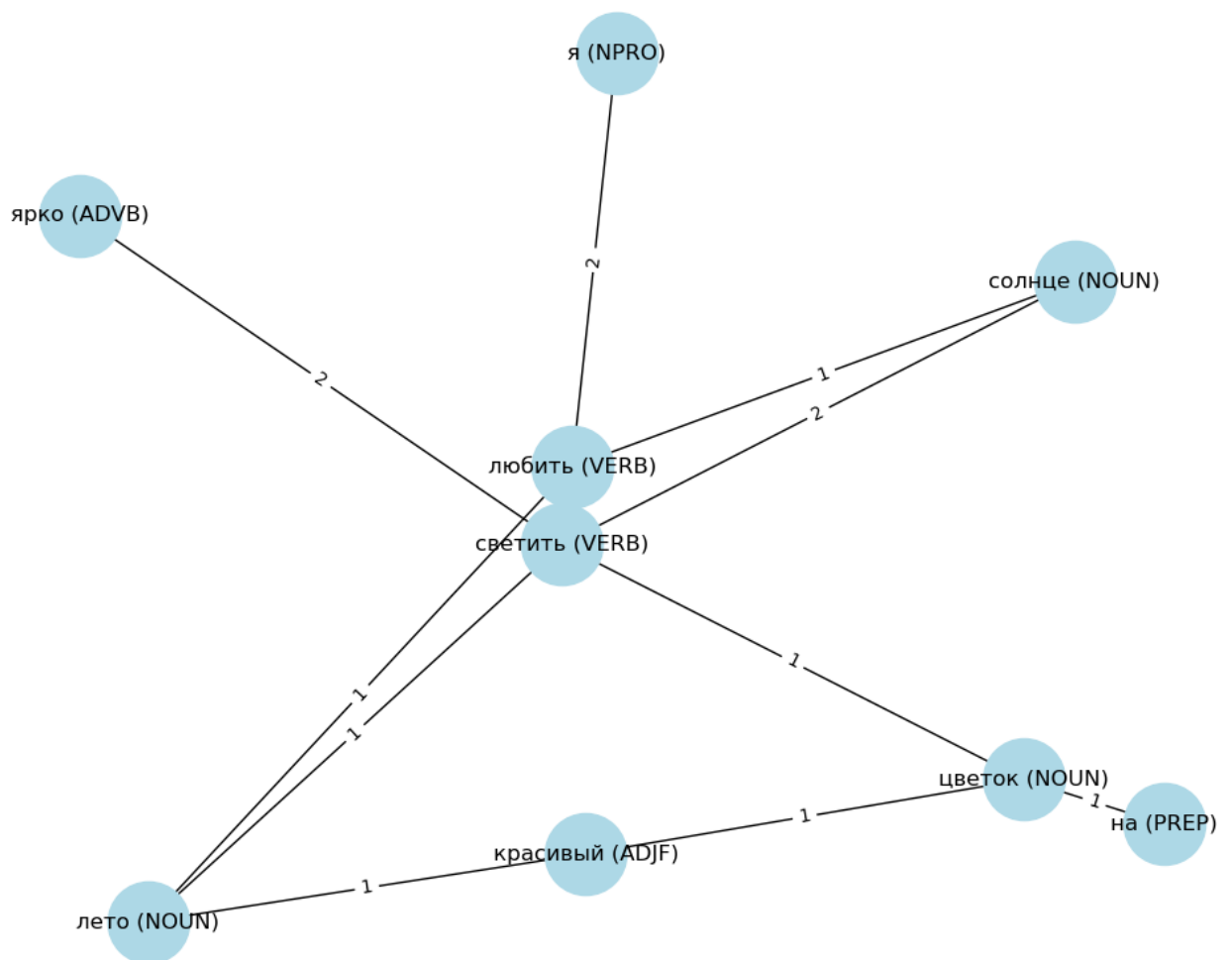


Рисунок 4.1 – Пример работы программы

Заключение

В ходе выполнения работы были выполнены следующие задачи:

- 1) описано понятие графа синтаксических зависимостей в тексте;
- 2) описан алгоритм построения рассматриваемого графа;
- 3) реализована программа, выполняющая построение графа синтаксических зависимостей в тексте;

Таким образом, все задачи были выполнены и, следовательно, поставленная цель была достигнута.

Список используемых источников

1. Welcome to Python [Электронный ресурс]. Режим доступа:
<https://www.python.org> (дата обращения: 15.12.2023).