

15. Практическое занятие: Синтаксические зависимости и синтаксический анализ. Грамматика взаимосвязей

Цель занятия

Изучить способы автоматического анализа структуры предложения, научиться строить деревья зависимостей (Dependency Parsing) и извлекать грамматические связи между словами (субъект, предикат, объект).

Теоретический минимум

Синтаксический анализ позволяет понять, «кто что сделал», а не просто какие слова есть в тексте.

1. **Дерево зависимостей (Dependency Tree):** граф, где узлы — это слова, а ребра — типы связей между ними. В основе всегда лежит корень предложения (обычно глагол-сказуемое).
2. **Грамматика взаимосвязей (Link Grammar):** подход, определяющий правила, по которым слова могут соединяться друг с другом (например, существительное требует определения, а глагол — дополнения).
3. **Типы связей:** * `nsubj` — номинальный субъект (подлежащее).
 - o `obj` — прямой объект (дополнение).
 - o `root` — корень предложения.

Задание 1. Построение дерева зависимостей с помощью Spacy

Библиотека `sраСу` (с моделью для русского языка) предоставляет мощный инструмент для извлечения синтаксических связей.

Инструкция:

Проанализируйте предложение и выведите отношения «родитель — ребенок» для каждого слова.

Python

```
import spacy

# Загрузка модели (требуется предварительная установка: python -m spacy
download ru_core_news_sm)
nlp = spacy.load("ru_core_news_sm")

text = "Маленький мальчик быстро читает интересную книгу."
doc = nlp(text)

print(f"{'Токен':<12} | {'Связь':<10} | {'Родитель':<10} | {'Дети'}")
for token in doc:
    print(f"{'token.text':<12} | {'token.dep_':<10} | {'token.head.text':<10} |
    {[child for child in token.children]}")
```

Задание 2. Извлечение триплетов (Субъект — Действие — Объект)

Синтаксический анализ позволяет автоматически извлекать факты из текста в формате семантических триплетов (SVO).

Инструкция:

Напишите функцию, которая находит главное действие и участников этого действия.

Python

```
def extract_svo(sentence):
    doc = nlp(sentence)
    subject = ""
    verb = ""
    obj = ""

    for token in doc:
        if token.dep_ == "nsubj":
            subject = token.text
        if token.pos_ == "VERB" and token.dep_ == "ROOT":
            verb = token.text
        if token.dep_ == "obj":
            obj = token.text

    return (subject, verb, obj)

text_example = "Студенты успешно выполняют сложную лабораторную работу."
print("Извлеченный факт (SVO):", extract_svo(text_example))
```

Задание 3. Визуализация синтаксической структуры

Графическое представление дерева помогает быстро отладить правила извлечения информации.

Инструкция:

Используйте инструмент displacy для отрисовки графа зависимостей.

Python

```
from spacy import displacy

# Если вы работаете в Jupyter Notebook, используйте 'jupyter': True
# В обычном скрипте это создаст локальный сервер для просмотра
displacy.render(doc, style="dep", options={'distance': 100})
```

Задание 4. Синтаксическая неоднозначность

Лингвистическая проблема: предложение «Я видел человека с телескопом» может иметь два дерева зависимостей.

1. Телескоп был у меня (инструмент наблюдения).
2. Телескоп был у человека (признак человека).

Инструкция:

Попробуйте подать подобное предложение в модель.

1. Какую связь выбрал алгоритм для слова «с телескопом»?
2. Как изменение порядка слов может помочь алгоритму понять смысл правильно?

Задание 5. Анализ согласования (Грамматика связей)

Изучите атрибут `token.morph` в `sraCy`. Он показывает грамматические свойства, по которым слова «связываются» (род, число, падеж).

Инструкция:

Проверьте, согласуются ли в вашем предложении из Задания 1 «мальчик» и «маленький» по падежу и числу через программный код.

Python

```
noun = doc[1] # мальчик
adj = doc[0] # маленький

print(f"Мальчик: {noun.morph.get('Case')}, {noun.morph.get('Number')}")
print(f"Маленький: {adj.morph.get('Case')}, {adj.morph.get('Number')}")
```

Контрольные вопросы

1. Какое слово в предложении всегда является корнем (ROOT) согласно правилам Dependency Grammar?
2. Почему синтаксический анализ важен для понимания смысла предложения в языках со свободным порядком слов (как русский)?
3. Чем синтаксический анализ отличается от морфологического?

Итог работы

Обучающиеся научились «видеть» каркас предложения. Синтаксический анализ позволяет не просто считать слова, а выстраивать логические цепочки событий, что необходимо для создания сложных вопросно-ответных систем и автоматического извлечения знаний.