

# Лингвистическая разметка

## Морфология. Синтаксис. Семантика

Александра Ивойлова

ИОД

1 апреля 2024 г.



# Содержание

1 Лингвистическая разметка

2 Морфологическая разметка

3 Синтаксическая разметка

4 Семантическая разметка

5 Разметка дискурса

# Лингвистическая разметка

# Язык как система

## Уровни языка

Самая маленькая единица естественного языка - **фонема**, но минимальная единица, обладающая семантикой (смыслом) - **морфема**

Единицы языка (разных уровней) все обладают какими-то свойствами, которые можно им приписать в качестве категорий



# Для чего нужна разметка

- В корпусах: для теоретических исследований
- В качестве дополнительных признаков для машинного обучения
- Для решения некоторых практических задач (н-р, лемматизация и поиск)

А POS=CCONJ а  
 что POS=PRON,Animacy=Inan,Case=Acc,Gender=Neut,Number=Sing что  
 ты POS=PRON,Case=Nom,Number=Sing,Person=2 ты

прогой POS=NOUN,Animacy=Inan,Case=Ins,Gender=Fem,Number=Sing прога

делаешь POS=VERB,Aspect=Imp,Mood=Ind,Number=Sing,Person=2,Tense=Pres,VerbForm=Fin,Voice=Act делать  
 ?! POS=PUNCT ?!

Пример разметки из корпуса ГИКРЯ

# Виды разметки

- Морфологическая разметка
- Синтаксическая разметка
- Семантическая разметка
- Разметка дискурса: выделение минимальных дискурсивных единиц, речевых пауз и т.д.
- ...

| Реплики, поделенные на ЭДЕ |   |
|----------------------------|---|
| 1.                         | A: ... еление и сколько там заболело<br>ну сколько скажите<br>сколько население в Приморье→ |
| 2.                         | B: В Приморье сейчас между прочим [ (неразборчиво)]<br>A: [сколько население] в Приморье? → |
| 3.                         | B: А вы хотите чтобы там захлебнулось   |
| 4.                         | A: → [ я спрашиваю сколько население ]<br>B: (→ )[ какая процентная составляющая] →         |

## Морфологическая разметка

# Что можно размечать

По степени востребованности:

- Леммы (lemmatization)
- Части речи (PoS-tagging)
- Морфологические характеристики (morphological analysis)

```

text="Потом"><tfr rev_id="3498393" t="Потом"><v><l id="263319" t="потом"><g v="ADVB"/></l></v></tfr></token>
text="проект"><tfr rev_id="3408581" t="проект"><v><l id="279545" t="проект"><g v="NOUN"/><g v="inan"/><g v="m
/token>
text="переехал"><tfr rev_id="835123" t="переехал"><v><l id="228592" t="переехал"><g v="VERB"/><g v="perf"/><g
indc"/></l></v></tfr></token>
text="о"><tfr rev_id="835124" t="о"><v><l id="311151" t="о"><g v="PREP"/></l></v></tfr></token>
text="«"><tfr rev_id="2420264" t="«"><v><l id="0" t="«"><g v="PNCT"/></l></v></tfr></token>
text="Культуры"><tfr rev_id="2632828" t="Культуры"><v><l id="144838" t="культура"><g v="NOUN"/><g v="inan"/><
/token>
text="»"><tfr rev_id="2420265" t="»"><v><l id="0" t="»"><g v="PNCT"/></l></v></tfr></token>
text="на"><tfr rev_id="835128" t="на"><v><l id="166264" t="на"><g v="PREP"/></l></v></tfr></token>
text="HTB"><tfr rev_id="4489375" t="HTB"><v><l id="192522" t="htb"><g v="NOUN"/><g v="inan"/><g v="neut"/><g
ing"/><g v="accs"/></l></v></tfr></token>
text=". "><tfr rev_id="835130" t=". "><v><l id="0" t=". "><g v="PNCT"/></l></v></tfr></token>
```

Пример морфологической разметки OpenCorpora

# Как это размечать

- Части речи - token classification
- Морфологические характеристики (грам. категории) - token classification
- Лемматизация?

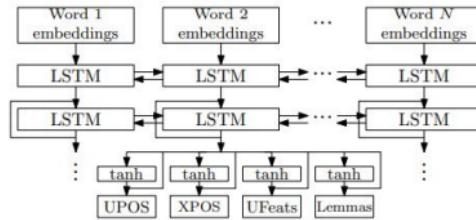


Figure 2: Tagger and lemmatizer model.

В современных SOTA-решениях лемматизацию тоже рассматривают как задачу классификации: только это классификация по правилам генерации лемм, которые вычисляются при обучении. Такой подход использует популярная архитектура UDPipe ([Straka \(2018\)](#)): в приведенной статье есть описание этого алгоритма.

# Тагсеты

Тагсет (tagset) - принятый в данном формате разметки набор тегов.  
Существует множество различных форматов морфологической разметки, основанных на разных лингвистических теориях, например:

- **Brown Corpus Tagset** (используется в Брауновском корпусе)
- **UPenn tagset** (используется в Penn Treebank)
- **Multext-East** (русский вариант используется в ГИКРЯ 1.0)
- **Тагсеты SketchEngine** (большой набор лингвоспецифичных тагсетов)
- **Tagset НКРЯ**
- **Universal Dependencies** (самый популярный универсальный формат на сегодня)
- Дополненный формат Universal Dependencies для НКРЯ: с 2023 года корпус постепенно переходит на разметку в стиле UD, свои дополнения они описали в [Lyashevskaya et al. \(2023\)](#)

# Подходы

Как в большинстве NLP-задач, имеем следующие варианты:

- **Правиловые**

Один из самых известных теггеров - Brill Tagger ([Brill \(1992\)](#)). На сегодня практически не используются; исключения - малоресурсные языки с высоким индексом агглютинации. Например, есть парсер для хакасского языка ([Дыбо and Шеймович \(2014\)](#)).

- **Статистические**

Были особенно популярны до 2010. Известные - [TreeTagger](#) ([Schmid \(1999\)](#)), [TnT Tagger](#) ([Brants \(2000\)](#)). Русскоязычной модификацией последнего размечен ГИКРЯ 1.0. Многие корпуса SketchEngine размечены TreeTagger.

- **На нейронных сетях**

SOTA. Для разметки в формате Universal Dependencies используется UDPipe ([Straka et al. \(2016\)](#)) или [SpaCy](#), также есть [stanza](#) (Stanford NLP)

- **Гибридные**

Как правило, к нейронной сети подключается словарь: часто используется для улучшения лемматизации.

## State of the Art

В последние годы задачу морфологического анализа обычно не рассматривают отдельно: она входит в задачу парсинга в формате Universal Dependencies

Самые последние достижения в парсинге в этом формате - например, [Levi and Tsarfaty \(2024\)](#): это интегральный морфосинтаксический парсер (авторы концентрируются на иврите)

Последняя на сегодняшний день публикация для русского языка - [Lyashevskaya et al. \(2023\)](#): модель Rubic, но они не опубликовали код.

# Парсеры для русского языка

Самые известные правиловые (сегодня используются для быстрого анализа, когда качество не принципиально):

- Mystem ([Segalovich \(2003\)](#)). Использует тагсет НКРЯ
- PyMorphy2. Использует тагсет OpenCorpora

На нейронных сетях (все, за исключением RNNMorph, используют тагсет UD):

- RNNMorph - SOTA 2017 года (ранняя версия тагсете UD)
- Joint Morpho-syntactic Parser - SOTA 2020 года; на этой архитектуре базируется модель Rubic
- Разумеется, русскоязычные модели есть у UDPipe, SpaCy, stanza.

# Joint Morpho-syntactic Parser

Парсер одновременно предсказывает теги и для морфологии, и для синтаксиса

Часть речи и грамматические категории склеиваются в одну строку: таким образом парсер не должен предсказывать неверные категории для части речи, н-р, одушевленность для глагола

Лемматизация происходит по такому же принципу, какой используется в UDPipe

Архитектура была обучена с разными вариантами эмбеддеров и энкодеров.

| Model            | POS           | MorphoFeats   | Lemma         | LAS           | Overall       |
|------------------|---------------|---------------|---------------|---------------|---------------|
| chars            | 94.7% / 91.7% | 92.2% / 90.9% | 95.5% / 93.6% | 41.1% / 38.1% | 80.9% / 78.6% |
| chars_lstm       | 97.2% / 94.1% | 96.9% / 94.6% | 97.3% / 95.0% | 87.2% / 77.8% | 94.7% / 90.4% |
| chars_morph_lstm | 97.5% / 94.4% | 97.7% / 95.2% | 98.1% / 95.6% | 89.6% / 77.0% | 95.7% / 90.6% |
| elmo             | 97.4% / 95.4% | 96.2% / 95.8% | 93.1% / 92.8% | 80.3% / 74.1% | 91.8% / 89.5% |
| elmo_lstm        | 97.9% / 95.9% | 97.5% / 95.9% | 97.0% / 95.3% | 88.9% / 80.3% | 95.3% / 91.9% |
| elmo_morph_lstm  | 97.8% / 95.7% | 97.7% / 96.1% | 97.3% / 95.3% | 89.5% / 79.6% | 95.6% / 91.7% |
| bert             | 98.4% / 96.2% | 98.3% / 96.4% | 98.6% / 96.5% | 93.1% / 84.6% | 97.1% / 93.4% |
| bert_lstm        | 98.6% / 95.8% | 98.4% / 96.3% | 98.5% / 96.2% | 93.2% / 83.5% | 97.2% / 92.9% |
| bert_morph_lstm  | 98.4% / 95.9% | 98.4% / 96.4% | 98.5% / 96.4% | 93.3% / 84.1% | 97.2% / 93.2% |
| bert_random      | 97.0% / 92.3% | 96.7% / 93.3% | 97.1% / 93.1% | 88.0% / 73.1% | 94.7% / 88.0% |

# Лемматизация

Цифры, как видим, очень высокие, качество лемматизации для некоторых моделей достигает 98.5%. Но что в действительности скрывается за цифрами?

| Wordform | Right lemma | BERT       | ELMo      |
|----------|-------------|------------|-----------|
| потерь   | потеря      | потерь     | потерь    |
| подсел   | подсесть    | подйти     | подсеть   |
| льдах    | лед         | льер       | льд       |
| прилечу  | прилететь   | прилестить | прилечуть |
| пою      | петь        | повать     | поть      |
| берите   | брать       | беть       | берить    |
| бегите   | бежать      | бяться     | бегять    |
| шипящими | шипеть      | шипить     | шипить    |
| стань    | стать       | станть     | стть      |
| зажги    | зажечь      | зжечь      | зажгть    |

## Гибридный подход к лемматизации

Обнаруживаем, что даже 98% качества означает, что 1-2 слова из 100 получают несуществующее слово вместо леммы (парсер галлюцинирует)

Потенциальное решение - гибридный подход, т.е. подключение словаря.

В ГИКРЯ 2.0 к этому парсеру подключили грамматический словарь Compreno: после разметки парсером пост-обработка проверяет все леммы по словарю с учетом другой морфологической информации ([Michurina et al. \(2021\)](#)).

Но полный словарь парадигм - дорогое удовольствие; в 2023 году было предложено решение Топ-k: парсер генерирует не единственный вариант леммы, а k самых вероятных, после чего они проверяются по обычному словарю лемм, и наиболее вероятный найденный в словаре вариант принимается за ответ.

## Shared Tasks

Только для русского языка на конференции “Диалог” проводилось четыре соревнования по автоматической лингвистической разметке:

- Dialogue Evaluation 2010: Морфология
- Dialogue Evaluation 2017: Морфологический анализ
- Dialogue Evaluation 2020: GramEval2020 (разметка морфосинтаксиса)
- Dialogue Evaluation 2023: SEMarkup (разметка морфологии, синтаксиса и семантики)

Аналогичные соревнования проводятся и для других языков, но в последнее время есть тенденция проводить соревнования по одновременной автоматической разметке морфологии и синтаксиса (и иногда также семантики).

# Метрики

Какие используются метрики для оценки качества морфологической разметки:

- PoS - Accuracy
- Лемматизация - Accuracy
- Грамматические категории - сумма всех правильно предсказанных категорий делится на сумму категорий в ground truth

Но ведь лемматизировать предлоги с наречиями в разы проще, чем существительные и глаголы (у них самая богатая парадигма)... да и с категориями не все так просто: ведь получается, при такой метрике выгодно предсказывать все существующие категории вообще

# Метрики

Поэтому на соревновании SEMarkup (Petrova et al. (2023)) были предложены модифицированные варианты метрик.

- Для грамматических категорий:

$$ScoreFeats(test, gold) = Penalty(test_{feats}, gold_{feats})$$

$$* \frac{\sum_{(cat, gram) \in gold_{feats}} CatWeight(cat) * [gram = test_{feats}^{cat}]}{\sum_{(cat, gram) \in gold_{feats}} CatWeight(cat)},$$

где

$$Penalty(f_{test}, f_{gold}) = \begin{cases} \frac{1}{1 + (Size(f_{test}) - Size(f_{gold}))} & \text{if } Size(f_{test}) > Size(f_{gold}), \\ 1 & \text{otherwise.} \end{cases}$$

- Для лемм (вес неизменяемых частей речи, например, предлогов - 0.3, вес сущ. и гл. - 0.7):

$$ScoreLemma(test, gold) = LemmaWeight(gold_{POS}) * [Norm(test_{lemma}) = Norm(gold_{lemma})].$$

# Морфоразметка и малоресурсные языки: XLT

Наилучшее

качество сегодня дают алгоритмы  
DL - но для них требуется большое  
количество размеченных данных.

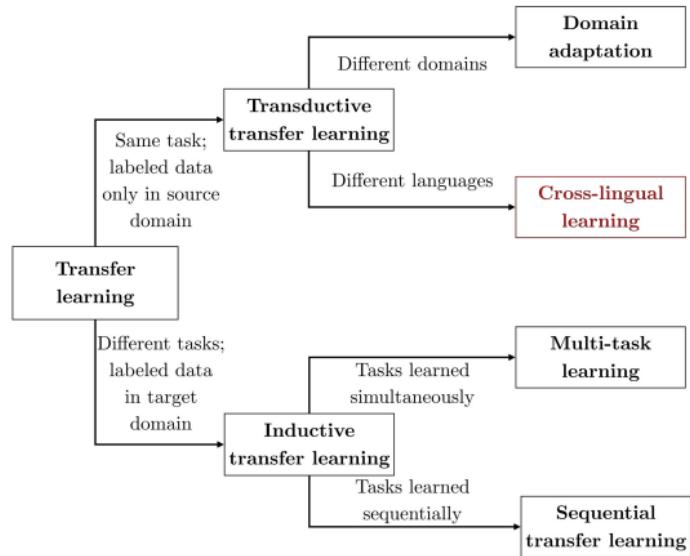
Данные постоянно

размечаются: в формате UD, например,  
размечены датасеты для >100 языков.

Однако многие эти датасеты  
крошечные, да и языков около 5К

Возможное решение - использование  
Cross-Lingual Transfer (XLT)

В  
последние годы появляется все больше  
работ, посвященных этому методу,  
н-р, Kim et al. (2017), de Vries et al. (2022).



# Challenges

- **Омонимия:** 'я плачу' - какая лемма у 'плачу'?
- OOV-слова
- Слова, которых не было в обучающих данных
- Различные теории: 'надо' - это предикатив или наречие?
- Разные тагсеты: как сравнивать качество?

```

ОTake POS=X отаке
воЕнE POS=NOUN,Animacy=Inan,Case=Dat,Gender=Neut,Number=Sing воЕна
!! POS=PUNCT !!
11 POS=NUM 11
<s />пържоли POS=SYM пържоли
<s />уг POS=NOUN,Animacy=Inan,Case=Nom,Gender=Masc,Number=Sing уг
Буде POS=ADV,Degree=Pos буде
разуплотНЕH POS=INTJ разуплотНЕH
! POS=PUNCT !
11<s /> POS=NUM 11

```

Что делает SOTA-парсер, когда встречает OOV

## Разрешение омонимии

Также известно как дизамбигуация, Word Sense Disambiguation (WSD)  
Подходы к решению:

- Ручная проверка: посадить лингвистов-разметчиков проверять за автоматическим парсером
- Смотреть по частотности на данных какого-либо корпуса (это точно дизамбигуация?)
- Использование контекста (алгоритм Леска, статистические алгоритмы, контекстуальные эмбеддинги)

Однако насколько современные контекстуальные эмбеддинги справляются с омонимией? Существует довольно много исследований на эту тему, например, [Garcia \(2021\)](#); общий вывод - они это делают не безукоризненно.

# OOV

Вроде бы использование BPE-токенизации и контекстуальных эмбеддингов решает и эту проблему, но на практике оказывается, что OOV-слова все равно размечаются парсерами с низким качеством, особенно тяжело дается их лемматизация (Таблица взята из [Michurina et al. \(2021\)](#): для сравнения, те же модели для словарных слов дают качество в районе 98%).

|  | <b>BERT</b> | <b>ELMo</b> |
|--|-------------|-------------|
| Lemmatization  | 85.26       | 80.34       |
| PoS-tagging  | 91.88       | 89.1        |
| Different lemmas of the same lexemes<br>(based on lemmatization of lexemes with several occurrences) | 17.58       | 18.68       |

Table 9. Percentage of right lemmas of out-of-vocabulary occurrences

## Конвертация тагсетов

Для сравнения качества разных инструментов морфологического анализа необходимо иметь общий набор тегов.

Для соревнования по морфологии 2017 года М. Копотев разработал **конвертер**.

Многие датасеты в формате UD сконвертированы из других форматов, например, **SyntagRus**.

Конвертация из стандарта Compreno в UD была представлена в **Ivoylova et al. (2023)**.

Однако всякий, кто занимался конвертацией форматов, знает, что это занятие неблагодарное: слишком много несостыковок, любая конвертация привносит собственные ошибки.

# Universal Dependencies: морфология

Форматов разметки очень много и многие из них лингвоспецифичны;  
 Хочется иметь какой-то общий формат;  
 Так появился стандарт Universal Dependencies.

Он был разработан на основе Stanford Typed Dependencies ([De Marneffe and Manning \(2008\)](#)) и сперва назывался Universal Stanford Dependencies ([De Marneffe et al. \(2014\)](#)).

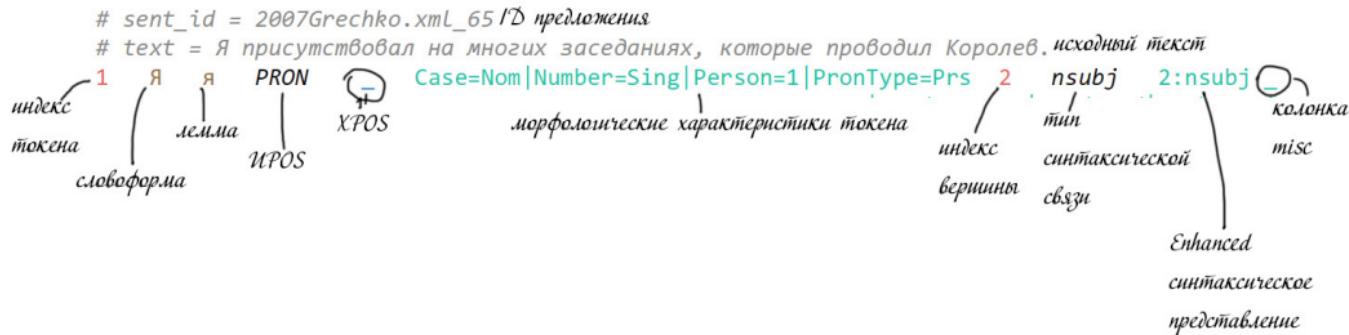
В этом формате размечается не только морфология, но и синтаксис.

```
# sent_id = 2013Algorithm.xml_15
# text = Формальные свойства алгоритмов.
1  Формальные   формальный   ADJ   _   Case=Nom|Degree=Pos|Number=Plur 2   amod   2:amod   _   root   0:root   _
2  свойства     свойство     NOUN  _   Animacy=Inan|Case=Nom|Gender=Neut|Number=Plur 0   root   0:root   _
3  алгоритмов   алгоритм   NOUN  _   Animacy=Inan|Case=Gen|Gender=Masc|Number=Plur 2   nmod   2:nmod:gen   SpaceAfter=No
4  .   .   PUNCT   _   _   2   punct   2:punct   _
```

Пример разметки UD (взят из Синтагруса).

Стандарт UD использует формат файлов **CONLL-U** или **CONLL-U Plus**: это табличный формат, где в каждой колонке содержится определенный тег.

# Universal Dependencies: морфология



Это стандартные 10 колонок формата CONLL-U. UPOS - универсальный PoS-тег, XPOS - тег исходного формата, если корпус был сконвертирован. В формате CONLL-U Plus можно определять собственные колонки, написав строку в самом начале файла, например:

```
# global.columns = ID FORM LEMMA UPOS XPOS FEATS HEAD DEPREL DEPS MISC SEMSLLOT SEMCLASS
```

Такая запись в начале позволит нам создавать 12 колонок вместо 10.

# Universal Dependencies: основные принципы в морфологии

- Основной элемент разметки - **слово**. Это означает, что в формате, например, составные предлоги ('в течение') анализируются как два разных токена
- Выделяется 17 универсальных частей речи; считается, что эти части речи могут быть приписаны словам во всех языках мира
- Грамматические категории могут различаться для разных языков; считается, что любая категория может сочетаться с любой частью речи (потенциально)

# Universal Dependencies: PoS-tags и русский язык

В универсальных тегах UD отсутствует тег 'предикатив', который есть в стандарте НКРЯ.

Частица 'бы' размечается как AUX (вспомогательный глагол).

Трудности вызывает тег PROPN, потому что не всегда ясно, что считать за имя собственное.

Тег DET вызовет у теоретиков споры с пеной у рта: русский язык - безартиклевый, считать ли, что у нас есть детерминант в языке? (Для теоретических лингвистов это известный спор между Ж. Башковичем с одной стороны и А. Перельцвайг и Е. Лютиковой с другой)

| Open class words | Closed class words | Other        |
|------------------|--------------------|--------------|
| <u>ADJ</u>       | <u>ADP</u>         | <u>PUNCT</u> |
| <u>ADV</u>       | <u>AUX</u>         | <u>SYM</u>   |
| <u>INTJ</u>      | <u>CCONJ</u>       | <u>X</u>     |
| <u>NOUN</u>      | <u>DET</u>         |              |
| <u>PROPN</u>     | <u>NUM</u>         |              |
| <u>VERB</u>      | <u>PART</u>        |              |
|                  | <u>PRON</u>        |              |
|                  | <u>SCONJ</u>       |              |

# Universal Dependencies: универсальность?

Universal Dependencies вводит одинаковые теги для всех языков, и данные всех языков размечаются по одинаковому принципу.

Однако языки очень разнообразны, тем более в морфологии: в китайском языке ее почти нет, зато в языках типа турецкого или корейского вообще не всегда понятно, что же считать за лемму, например.

Для стандартных европейских языков нормально деление на прилагательные и глаголы; в тех же турецком и корейском прилагательные и глаголы ведут себя часто одинаково, и их принято выделять в одну часть речи в традиционных грамматиках (о корейском языке и UD можно почитать в [этой статье](#)).

Из-за таких особенностей часто ошибаются инструменты автоматической разметки: идентичное поведение двух вроде бы разных частей речи не дает установить закономерность.

# Морфологическая разметка: подводя итоги

- Морфология - хорошо изученная область теоретической лингвистики, морфологическим анализом в NLP занимались дольше и больше всего
- Однако даже такая вроде бы простая задача, как POS-tagging, сталкивается с рядом проблем, многие из которых вызваны тем, что морфология - наиболее лингвоспецифический уровень языка
- Задача лемматизации вызывает больше всего сложностей; хотя цифры показывают очень высокое качество, при ручном анализе это качество может оказаться совсем не таким хорошим
- Существенно осложняет морфологический анализ омонимия в языке
- Наиболее популярный формат разметки сегодня - Universal Dependencies, однако он тоже не идеальный
- SOTA-решения сегодня - DL-архитектуры и гибридные архитектуры с использованием LLM

# Синтаксическая разметка

# Синтаксическая разметка

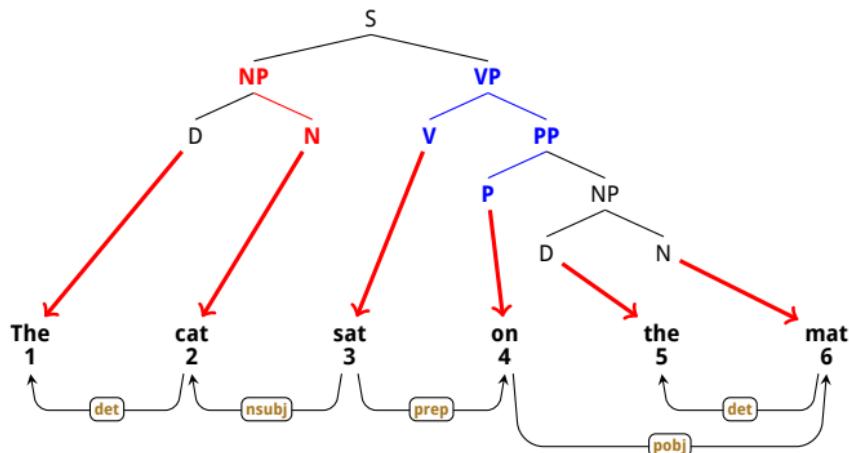
В теоретической лингвистике существует довольно много различных синтаксических теорий:

- Функциональные: нельзя рассматривать синтаксис в отрыве от остальных уровней языка, особенно семантики и прагматики.  
Например:
  - Cognitive Grammar ([Langacker \(1995\)](#))
  - Systemic Functional Grammar ([Matthiessen and Halliday \(2009\)](#))
  - Lexical Functional Grammar ([Börjars et al. \(2019\)](#))
  - Construction Grammar ([Goldberg and Suttle \(2010\)](#))
- Формальные:
  - теории Ноама Хомского в развитии - от его идей трансформационной генеративной грамматики ([Chomsky \(1965\)](#)) к самой последней версии генеративизма - минимализму ([Chomsky \(1995\)](#))
  - Head Driven Phrase Structure Grammar (HPSG), ([Pollard and Sag \(1994\)](#))
  - Combinatory Categorial Grammar (CCG) ([Steedman and Baldridge \(2011\)](#))

# Синтаксическая разметка

В большинстве теорий в любом случае рассматриваются синтаксические связи между словами, и визуально они часто представляются как направленные ациклические графы.

Две наиболее известные разновидности таких графов - деревья составляющих (сверху) или деревья зависимостей (снизу):

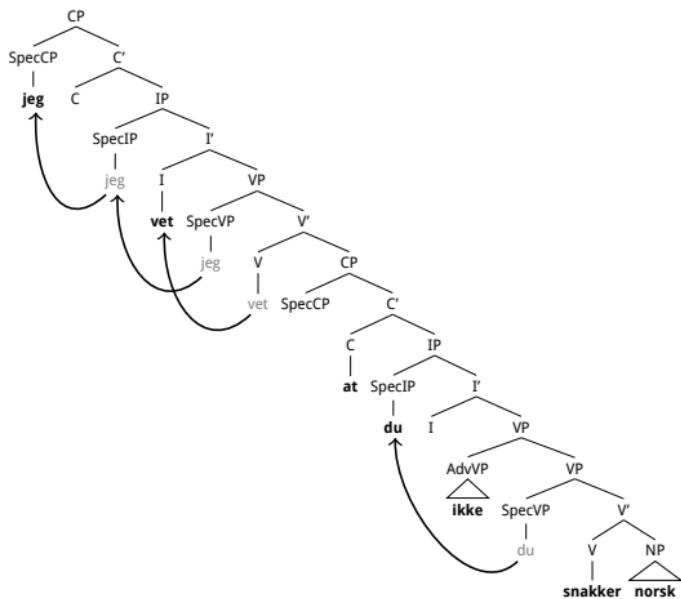


# Деревья составляющих

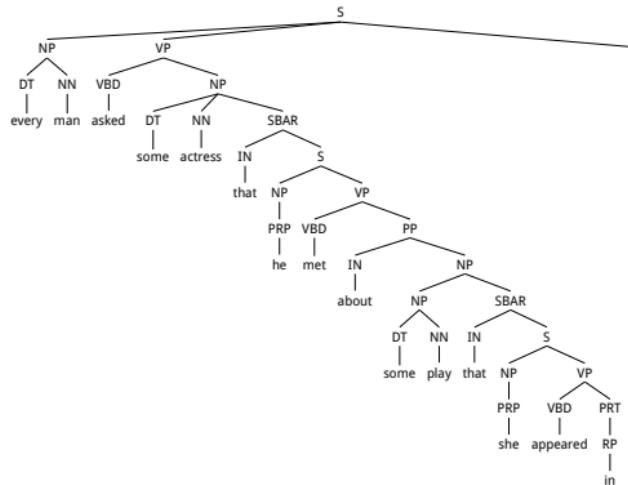
Деревья составляющих могут выглядеть достаточно разнообразно.

В современных публикациях по теоретическому синтаксису обычно можно видеть что-нибудь подобное.

Как можно заметить, любые перестановки слов в предложении ломают все дерево, и лингвистам приходится идти на сложные ухищрения при анализе.



# Деревья составляющих



На практике часто используют формат Penn Treebank - он более простой.

Узлы в дереве составляющих - сами составляющие, их еще называют группами (phrase).

Листья дерева (терминалы) - сами слова + их части речи.

Цель парсинга составляющих - правильно найти все группы и установить связи между ними.

# Парсинг составляющих

В качестве метрики обычно используется F1-score.

Основные подходы к автоматическому построению деревьев составляющих:

- Transition-based
- Span-based (chart-based)
- Sequence-based

# Transition-based constituency parsing

Используется shift-reduce алгоритм:

- Вначале создается пустой стек  $S$  и очередь  $W$  со всеми словами предложения.
- **shift** достает первое слово из очереди и размещает его наверх  $S$
- **reduce** имеет унарный и бинарный вариант: при унарном слово наверху  $S$  вытесняется новым словом, а при бинарном вытесняются два объекта и превращаются в узел с двумя листьями.

Используются как правиловые, так и нейронные варианты.

Более подробные описания и связанные с этим работы:

[Sagae and Lavie \(2005\)](#), [Watanabe and Sumita \(2015\)](#), [Liu and Zhang \(2017\)](#), [Yang and Deng \(2020\)](#) - SOTA

# Span-based constituency parsing

- В основе идеи - трансформационные правила контекстно-свободных грамматик.
- Дерево составляющих рассматривается как совокупность размеченных спанов в предложении.
- Парсер должен предсказать границы спанов и их категории.
- Этот подход пользуется большой популярностью; SOTA - Yang and Tu (2023)

# Sequence-based constituency parsing

- В этом подходе используются обычные Seq2Seq модели
- На вход такой модели подается исходное предложение, а сгенерировать она должна дерево составляющих в скобочной форме записи, например, "<s> (S (NP XX XX )NP (VP XX (NP XX XX )NP )VP )S </s>"
- SOTA для такого подхода - [Kamigaito et al. \(2017\)](#)

# LLM и деревья составляющих

Все вышеперечисленные методы могут быть реализованы с помощью Large Language Models: совсем недавно такие эксперименты провели [Bai et al. \(2023\)](#).

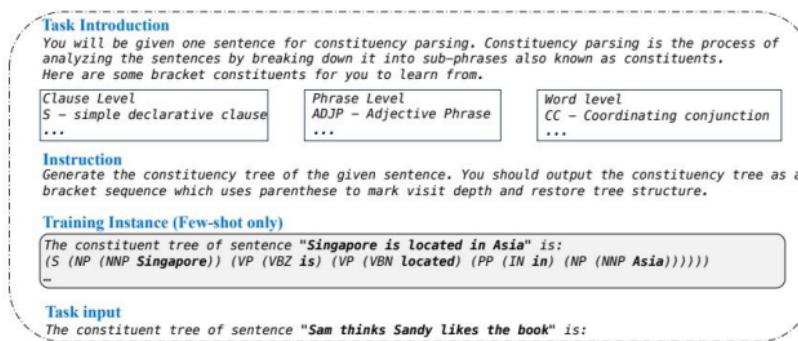


Figure 3: Illustration of prompts for zero/few-shot learning.

# Составляющие: известные датасеты

- Самый известный - **Penn Treebank**
- **FLAU**: французский бенчмарк
- **Potsdam Commentary Corpus**: для немецкого языка
- **MASC**: англоязычный корпус
- На **SPMRL Shared Task** были предложены датасеты для нескольких крупных языков, но они больше не доступны для скачивания
- Датасеты в формате **CoBaLD** конвертируются из разметки в виде деревьев составляющих Compreno; потенциально можно получить их разметку и в виде деревьев составляющих.

# Деревья составляющих для русского языка

Как можно заметить, в открытом доступе датасетов с разметкой составляющих для русского языка нет.

Существует правиловый парсер ABBYY Compreno, который умеет строить деревья составляющих, но это проприетарный инструмент.

```

"#NonexclamatonClause DECLARATIVE_MAIN_CLAUSE"
$Verb, Predicate: "готовить;готовить;PREPAREDNESS"
$AdjunctTime, Time: "обычно;#frequentative_adverbs_adi:FREQUENTATIVE"
$Subject, Experiencer_Metaphoric: "бюджет;бюджет;BUDGET"
$Object_Indirect_K, Object_Situation: "чтение;READING_OF_THE_DRAFT_LAW"
$Preposition: "к;#preposition;PREPOSITION"
$Ordinal, OrderInTimeAndSpace: "второй;TWO_ORDINAL"
$Adjunct_Locative, Locative: "дума дума;DUMA"
$QuantitativeAdverb, DegreeApproximative: "непосредственный;DIRECT_OBLIQUE"
$Preposition: "в; Prepositional:#preposition;PREPOSITION"
$SpecificationClause_Colon, Specification_Clause: "корректировать;корректировать;TO_CORRECT"
$Subject, Agent: "депутат;депутат;DEPUTY"
$Object_Direct, Object_Situation: "план;план;SCHEDULE_FOR_ACTIVITY"
$Modifier_Attributive, Agent: "правительство;правительство;GOVERNMENT"

```

## Shared Tasks

Начиная с 2010 года под эгидой NAACL (North American Chapter of the Association for Computational Linguistics) проводились воркшопы Statistical Parsing of Morphologically Rich Languages (SPMRL) ([Tsarfaty et al. \(2010\)](#)).

На этих воркшопах в 2013 и 2014 проводились соревнования по парсингу деревьев составляющих; на сегодня это последние известные такие соревнования.

- Shared Task 2013
- Shared Task 2014

# Составляющие или зависимости?

Очевидно, у деревьев составляющих есть свои недостатки:

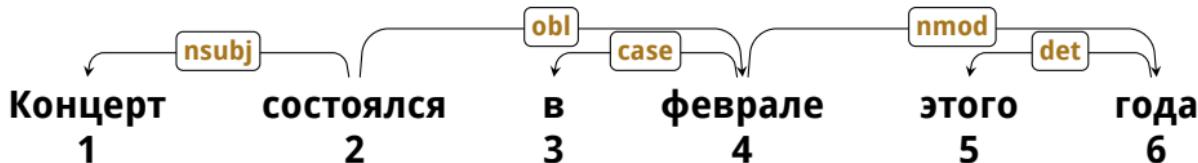
- Они довольно сложно устроены, особенно в генеративизме
- Языки с более свободным порядком слов вызывают огромные трудности для анализа
- Их не очень удобно обрабатывать автоматически

Зачем же они вообще нужны?

- С позиции теоретического синтаксиса они объясняют гораздо больший круг явлений
- В современных формальных теориях деревья составляющих довольно универсальны (хотя от этого выглядят еще более жуткими)

Для практических же задач удобнее использовать деревья зависимостей.

# Деревья зависимостей



Самый популярный формат синтаксической разметки - это формат Universal Dependencies, базирующийся на Stanford Typed Dependencies.

В отличие от составляющих, деревьям зависимостей странный порядок слов не помеха: существует только понятие непроективности (когда стрелки зависимостей пересекаются).

У каждого слова в предложении должна быть вершина, к которой оно присоединяется каким-либо видом синтаксической связи, например, **nsubj** - подлежащее. Единственное слово в примере, у которого вершины вроде бы нет - 'состоялся' - на самом деле имеет вершиной невидимый **root** - корень предложения.

# Universal Dependencies: синтаксис

Все слова в предложении индексируются, начиная с единицы (0 - тот самый незримый root)

Каждое слово получает две категории:

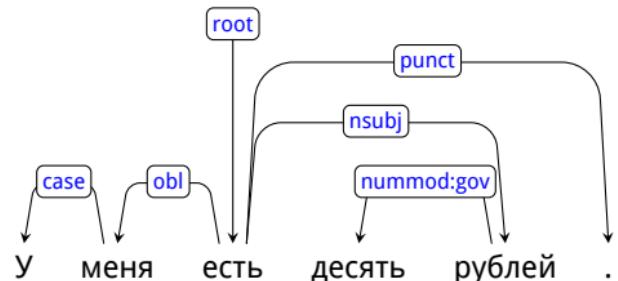
- 1) индекс своей вершины;
- 2) тип синтаксической связи

Всего в UD **37 синтаксических связей**. Связи могут иметь какие-то уточнения: уточнения пишутся через двоеточие

|                          | Nominals  | Clauses                                      | Modifier words   | Function Words                            |
|--------------------------|---|--|--|---|
| Core arguments           | <u>nsubj</u><br><u>obj</u><br><u>iobj</u>                         | <u>csubj</u><br><u>ccomp</u><br><u>xcomp</u> |  |   |
| Non-core dependents      | <u>obl</u><br><u>vocative</u><br><u>excl</u><br><u>dislocated</u> | <u>advcl</u>                                 | <u>advmod*</u><br><u>discourse</u>                                     | <u>aux</u><br><u>cop</u><br><u>mark</u>   |
| Nominal dependents       | <u>nmod</u><br><u>appos</u><br><u>nummod</u>                      | <u>acl</u>                                   | <u>amod</u>  | <u>det</u><br><u>clf</u><br><u>rank</u>   |
| Coordination             | Headless  | Loose  | Special  | Other                                     |
| <u>conj</u><br><u>cc</u> | <u>fixed</u><br><u>flat</u>                                       | <u>list</u><br><u>parataxis</u>              | <u>compound</u><br><u>orphan</u><br><u>deposited</u><br><u>renamed</u> | <u>punct</u><br><u>root</u><br><u>den</u> |

# Universal Dependencies: синтаксис

Например, отношение `nummod:gov` уточняет, что это не просто связь между числительным и существительным, но числительное определяет падеж существительного.



Знаки препинания считаются в UD полноправными членами предложения и должны зависеть от своих вершин (вершины определяются по своим правилам, описанным [здесь](#)).

# Universal Dependencies: основные принципы в синтаксисе

Подробное описание теоретических основ и принципов формата есть в статье [De Marneffe et al. \(2021\)](#).

- Формат стремится к языковой универсальности: поэтому вершинами всегда выступают только 'content words' - слова, несущие смысл. Функциональные слова (н-р, копула) вершинами синтаксических связей быть не могут. Поэтому же существительное - вершина предлога, хотя в теоретическом синтаксисе наоборот.
- Различаются номиналы, клаузы и модификаторы.
- Предполагается, что во всех языках выделяются как минимум объект и субъект.
- Размечаются только выраженные словами связи: эллипсис не размечается.

# Подходы

Существует два основных (но не единственных) подхода к парсингу деревьев зависимостей:

- Graph-based
- Transition-based

Для обоих подходов могут использоваться как правиловые и статистические, так и алгоритмы глубинного обучения.

# Graph-based dependency parsing

В основе этого подхода - использование алгоритма maximum spanning tree (MST).

Дерево раскладывается на части, например, на ребра зависимостей.

В зависимости от того, сколько включается ребер в минимальную часть, парсеры называют N-порядковыми (N-order)

Самые простые парсеры - first-order; парсеры более высокого порядка могут использовать более сложные признаки, но имеют большую вычислительную сложность (от  $O(n^3)$  до  $O(n^4)$ ).

Статистический бейзлайн - **MST Parser**.

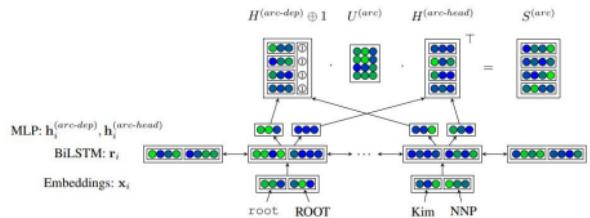
В числе последних предложенных моделей - **Altıntaş and Tantuğ (2023)**

# Биаффинный парсер Дозата-Маннинга

В 2016 [Dozat and Manning \(2016\)](#) предложили новую архитектуру парсера на графах, которая серьезно улучшила метрики.

Их модель использует трехслойную Bi-LSTM в качестве энкодера и биаффинное внимание.

Эта архитектура стала очень популярной, существует много ее улучшений.



# Transition-based dependency parsing

Первоначально этот подход был придуман именно для деревьев зависимостей и потом стал применяться к деревьям составляющих.

Transition-based модели могут сравняться по качеству с графовыми, при этом они гораздо шустрее и могут достигать линейной вычислительной сложности.

В отличие от графовых моделей, transition-based могут использовать больше признаков при обучении.

Статистический бейзлайн - [Malt Parser](#).

В числе последних предложенных моделей - [Le-Hong and Cambria \(2024\)](#)

# Метрики

В качестве метрик обычно используются следующие:

- UAS (unlabeled attachment score) - процент токенов, которым правильно приписан номер вершины;
- LAS (labeled attachment score) - процент токенов, которым правильно приписан и номер вершины, и тип синтаксической связи.
- В [Nivre and Fang \(2017\)](#) предложена также метрика CLAS (content labeled attachment score) - она вычисляется как F1 для всех синт. связей, кроме связей для пунктуации и функциональных слов (н-р, [aux](#)).

# Shared Tasks

В числе самых известных проведенных соревнований по автоматической разметке в формате UD соревнования CONLL:

- CoNLL 2017 Shared Task
- CoNLL 2018 Shared Task

Для русского языка также проводились два уже упоминавшихся выше соревнования:

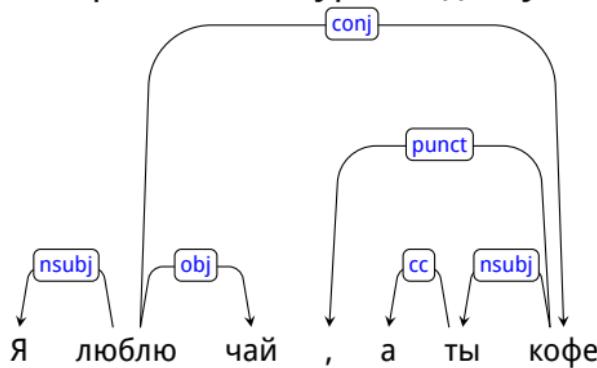
- GramEval 2020
- SEMarkup 2023 (основная цель - семантическая разметка)

# Universal Dependencies: синтаксис?

Формат UD стремится к межъязыковой универсальности, но из-за этого за бортом могут оказаться лингвоспецифические особенности.

Для большинства задач это не принципиально, но может возникнуть необходимость учитывать такие вещи, например, тип предлога при связи *obl/nmod*.

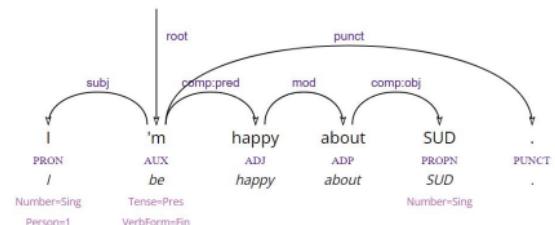
Также для стандартных UD эллипсис и гэппинг - серьезная проблема, ведущая к странной разметке и бурным дискуссиям среди теоретиков.



# Surface Universal Dependencies

В качестве альтернативы команда исследователей предложила свой подход к деревьям зависимостей на базе UD: *Surface Syntactic Universal Dependencies*.

- Стандартные UD исторически опирались на генеративизм и HPSG, но постепенно все больше стали склоняться к прагматическим соображениям; SUD опирается на теории И. Мельчука и ставит целью именно описание синтаксиса.
- Поэтому в SUD вершинами могут быть функциональные слова, например, копула или предлог может быть вершиной;
- Изменены и виды отношений, SUD не различают клаузы и номиналы в роли комплемента;
- При этом формат может быть сконвертирован в UD и обратно



# Enhanced Universal Dependencies

С другой стороны, уже несколько лет разрабатывается усовершенствованный формат Enhanced Universal Dependencies: именно он содержится в девятой колонке файла CONLL-U.

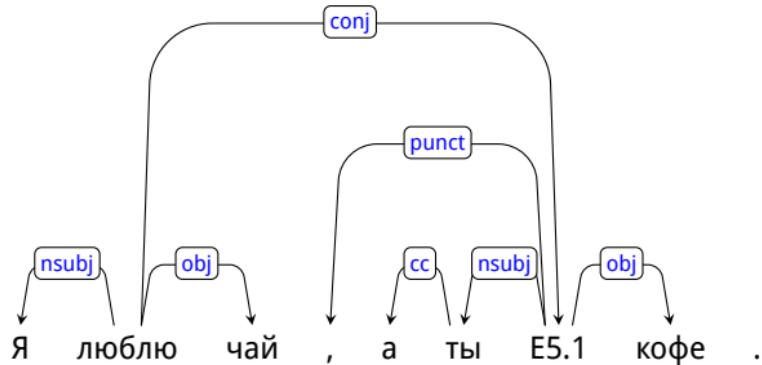
Его основные особенности:

- восстанавливается эллипсис;
- по-другому размечаются конъюнкты;
- появляются дополнительные отношения `xsubj` для конструкций подъема и контроля;
- размечается референтная связь в относительных придаточных: `ref`;
- дополнительная информация о падежах и предлогах добавляется к тегам типа `obl`.

Описания этого формата есть в [Schuster and Manning \(2016\)](#) и на [официальном сайте UD](#).

# Enhanced Universal Dependencies

Теперь эллипсис выглядит более логично:



Правда, возникают другие проблемы - для парсинга: ведь теперь одно слово может иметь больше двух вершин, и приходится предсказывать пустые узлы для эллипсиса.

# Метрики

Для оценки качества автоматической разметки в формате E-UD используются такие метрики:

- ELAS (labeled attachment score on enhanced dependencies) - F1-мера на множестве Enhanced зависимостей. Такие теги, как, например, `obl` и `obl:on`, считаются разными!
- EULAS - то же, что ELAS, но учитывается только "универсальная" часть тега - то есть, `obl` и `obl:on` будут считаться одним и тем же.

# Shared Tasks

Есть два известных соревнования, проведенных на International Conference on Parsing Technologies:

- IWPT 2020 Shared Task
- IWPT 2021 Shared Task

Победитель 2021 года - [Shi and Lee \(2021\)](#), однако они не предоставили код; [код](#) есть только у занявших третье место участников - [Grünewald et al. \(2021\)](#).

Также в настоящее время готовится соревнование в том числе для русского языка (CoBaLD Parsing), срок его проведения - предположительно этой осенью.

## Известные датасеты

- Большое количество датасетов выложено в открытом доступе на **официальном гитхабе UD**, часть этих датасетов дополнительно размечена в E-UD. Репозиторий постоянно пополняется; например, в 2023 году появился первый treebank для сингальского языка (один из официальных языков Шри-Ланки).
- Также в формате, совместимом с UD, выкладывает датасеты команда CoBaLD на своем **гитхабе**. Пока доступны только русский и английский датасеты (английский - в E-UD). Готовятся маленькие сербский и венгерский датасеты.

# Синтаксическая разметка и XLT

Синтаксис менее лингвоспецифичен, чем морфология, для него есть универсальный формат разметки и много датасетов на самых разных языках, так что задача dependency parsing - одна из самых распространенных даунстимовых задач для межъязыкового переноса.

Поэтому ежегодно появляются новые исследования, посвященные переносу разметки зависимостей, например, только за 2023 год: [Sun et al. \(2023\)](#), [Choudhary and O'riordan \(2023\)](#), и даже есть целый обзор по тематике [Das and Sarkar \(2020\)](#).

# Синтаксическая разметка: подводя итоги

- Два основных вида представления синтаксических связей - деревья зависимостей и составляющих, причем первые популярнее в NLP;
- Синтаксис - менее лингвоспецифический уровень языка, поэтому относительно легко переносится с помощью XLT;
- Самый популярный формат для деревьев зависимостей - Universal Dependencies, но из-за стремления к универсальности его разметка не чисто синтаксическая, а скорее семантическая;
- Две более “синтаксических” версии - SUD и Enhanced UD; последняя набирает популярность, но парсить деревья в этом формате довольно сложно из-за эллипсиса и множественных вершин.
- Размеченных датасетов для UD - очень много и на разных языках; для деревьев составляющих это число заметно ограничено.

## Семантическая разметка

# Семантическая разметка

В теоретической лингвистике существует много различных теорий, связанных с семантикой (глубинные падежи Филлмора, семантические примитивы Вежбицкой...). В NLP семантической разметке посвящены две задачи:

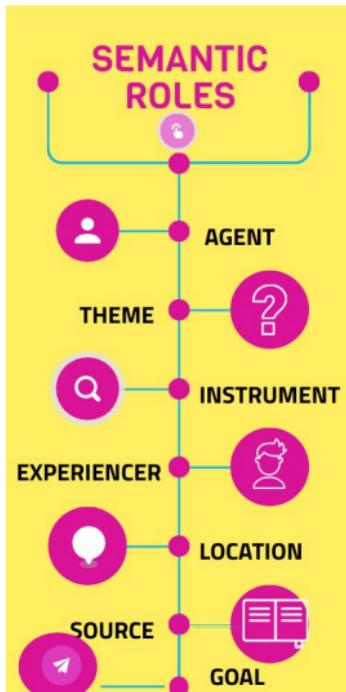
- Semantic Role Labeling (SRL) - разметка семантических ролей
- Semantic Parsing (SP) - полная разметка семантических связей в предложении/тексте

# Semantic Role Labeling

Семантические роли - известное понятие в теоретической лингвистике; еще в прошлом веке были разработаны влиятельные теории, связанные с ним и повлиявшие на большинство сегодняшних форматов разметки:

- Глубинный падеж Филлмора ([Fillmore \(1967\)](#))
- Role and Reference Grammar ван Валина ([Van Valin Jr \(1990\)](#))
- Семантические примитивы Джекендоффа ([Jackendoff \(1975\)](#))

# Семантическая роль: понятие

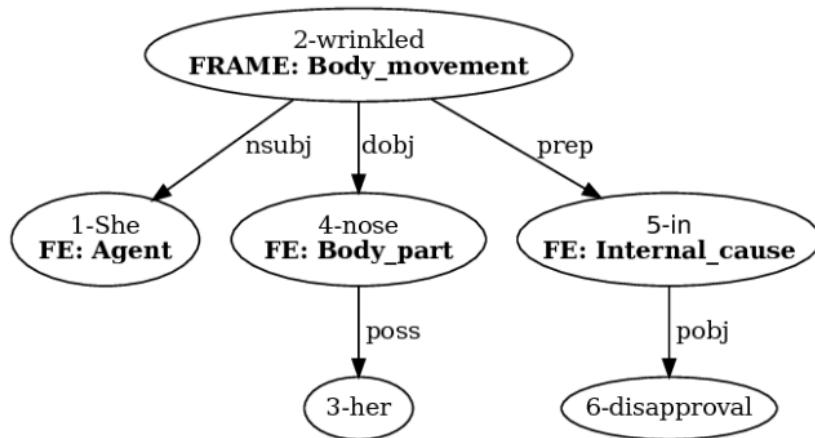


Вне зависимости от теории, обычно семантическая роль - это определенная позиция при глаголе (как правило), например, у глагола “купить” есть потенциальный агент - тот, кто покупает, и объект - то, что покупают.

Набор семантических ролей может отличаться для разных теорий, но, как правило, всегда есть Agent, Recipient, Experiencer, Object, Instrument и подобные.

# Фреймы Филлмора

Огромное влияние, в частности, на развитие компьютерной семантики оказали теории Ч. Филлмора: именно благодаря им появились такие ресурсы, как **FrameNet**, **WordNet**, **VerbNet**, **PropBank**. Эти базы данных очень часто используются при решении задачи SRL и связанных задач (например, для обогащения эмбеддингов: [Ponti et al. \(2018\)](#)).

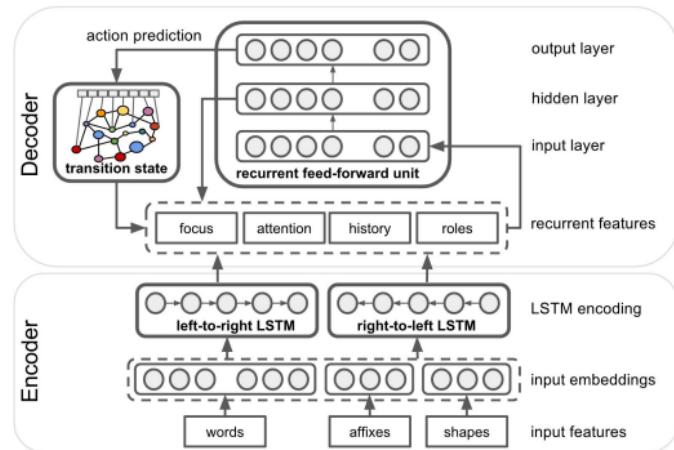


# Frame Semantics

Поскольку FrameNet

- изначально компьютерный ресурс, существуют различные инструменты для автоматической разметки в его формате. Самый известный - это парсер **SLING**.

Он использует transition-based подход и базируется на LSTM-архитектуре.



# Semantic Parsing

Семантическая разметка вообще - это перевод предложения естественного языка в (машиночитаемое) представление на формальном языке смысла. Таким образом, семантическая разметка может включать в себя:

- семантические роли;
- референтные связи;
- смыслы слов (семантические категории слов);
- модальность;
- именованные сущности;
- связность текста и другое.

Что включается - очень зависит от конкретной теории и формата разметки.

# Universal Networking Language

Подавляющее большинство стандартов семантической разметки первоначально задумывались с целью автоматического перевода (как и теория Смысл-Текст И. Мельчука). Один из ранних стандартов - **Universal Networking Language**.

Формат различает универсальные слова (UW), бинарные связи и атрибуты.

UW - концепты, или значения слов человеческого языка;

Связи - семантические роли, их всего 42;

Атрибуты - грамматические признаки (время, модальность, число...);

Все это представляется в форме графов;

У UW есть еще ограничения на присоединение сем. ролей;

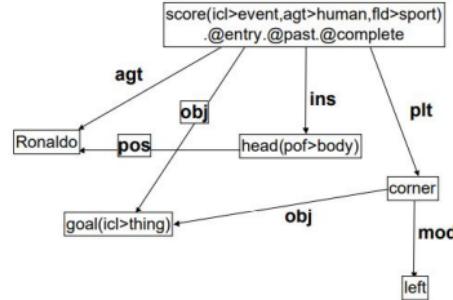
UNL принципиально абстрагируется от синтаксической структуры предложения.

# Universal Networking Language

Статьи, посвященные UNL и парсингу в этом формате, появляются до сих пор (например, [Ali et al. \(2021\)](#)), хотя сам формат давно перестал активно разрабатываться.

На этом формате базируется правиловая система автоматического перевода ЭТАП, разрабатываемая в ИППИ им. Харкевича (и по сей день).

Есть мнение, что UNL на самом деле не такой уж универсальный, и на него влияет английский. Известных размеченных в UNL датасетов нет.



A possible UNL graph for “Ronaldo has headed the ball into the left corner of the goal”

# ЭТАП

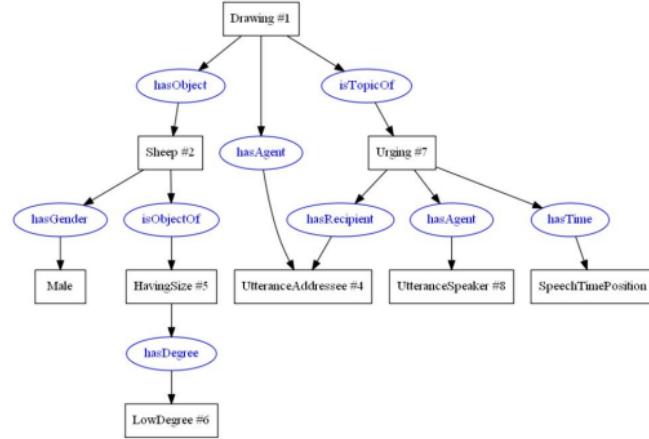
С 1980-х гг. в ИППИ им. Харкевича разрабатывался многоцелевой лингвистический процессор **ЭТАП**.

## ЭТАП

в основном был предназначен для правилового машинного перевода, но также он выполняет полную лингвистическую разметку текстов.

Теория, на которую он опирается - модель Смысл-Текст Мельчука, а также в его основе - Universal Networking Language.

Размечаются семантические роли и семантические классы, последние берутся из онтологии OntoETAP.



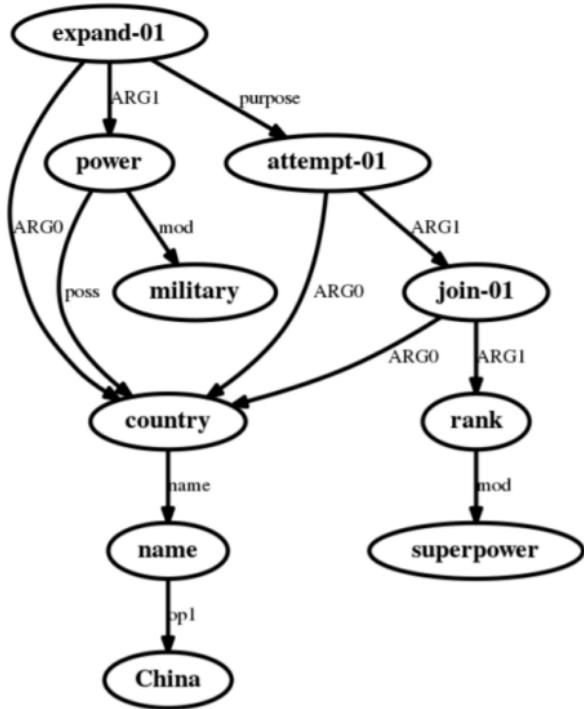
# ЭТАП

Лаборатория 15 под руководством И. Богуславского, которая занимается разработкой ЭТАПа, представила несколько связанных проектов:

- собственно ЭТАП (работа сейчас ведется над версией 4)
- Семантический анализатор SemETAP - правиловая разметка семантики
- корпус SynTagRus первоначально был размечен в этом формате
- Семантический корпус русского языка SemOntoCor ([Boguslavsky et al. \(2023\)](#)) - один из двух существующих корпусов с семантической разметкой для русского языка

При этом корпус содержит только текст “Маленького принца” Сент-Экзюпери и в открытый доступ не выложен.

# Abstract Meaning Representations



AMR (Banarescu et al. (2013)) очень похож на UNL: он тоже стремится к универсальным представлениям семантики предложений.

Точно так же представления AMR - это направленные ациклические графы, в узлах которых - концепты, а ребра представляют собой отношения.

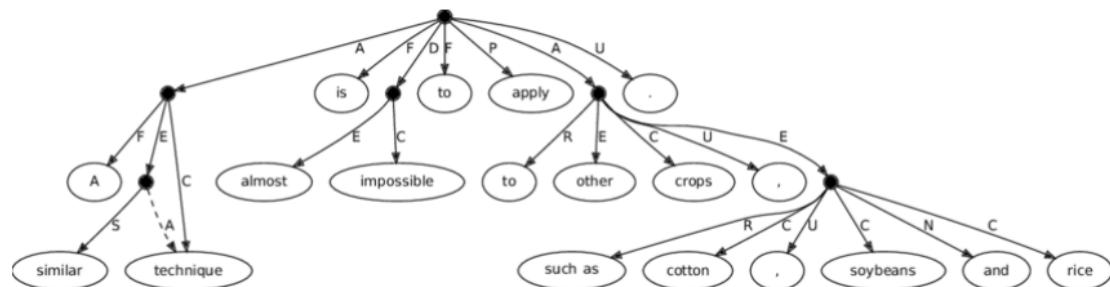
AMR довольно популярны: уже четыре раза проводился International Workshop on Designing Meaning Representations. Для AMR существуют парсеры и датасеты: на английском и китайском.

# Abstract Meaning Representations

- всего в AMR около 100 отношений - семантических ролей;
- концепты AMR могут браться из PropBank, если отсутствуют - то берутся просто английские слова;
- AMR максимально абстрагируется от синтаксиса: например, заменяет местоимения в исходном предложении на переменные;
- формат не способен представлять флексивную морфологию для числа и времени, артикли;
- не имеет универсальных кванторных слов (all, every...);
- точно так же, как UNL, находится под влиянием английского языка.

# Universal Conceptual Cognitive Annotation

UCCA (Abend and Rappoport (2013)) - сравнительно молодой стандарт разметки, появившийся в 2013 году.

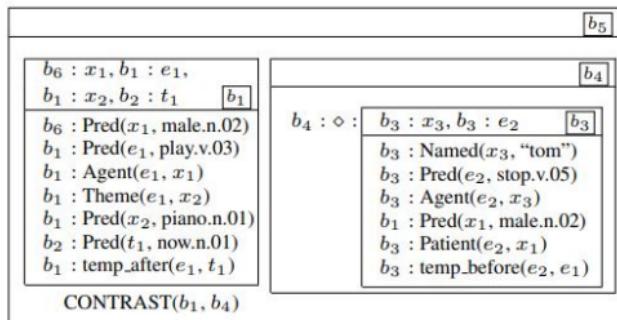


Он довольно отличается по концепции от AMR или UNL: центральное понятие UCCA - это сцена. Сцена - это описание действия, движения или состояния, имеющее одно-единственное отношение, которое определяет тип сцены. Типов два: State (S) и Process (P). Сцена содержит участников, также размечаются несценные единицы.

# Universal Conceptual Cognitive Annotation

- формат задумывался уже в “нейронную” эпоху и предполагает, что все данные будут размечаться сперва вручную (в отличие от UNL и AMR, опирающихся на банки данных);
- принципиально игнорирует синтаксис, разметка идет поверх токенизированного текста;
- количество категорий относительно невелико, отсутствует разметка смыслов слов (только семантические роли);
- восстанавливается эллипсис;
- могут размечаться кореферентные связи;
- есть довольно много [датасетов](#): на английском, немецком, французском, недавно появился на [турецком](#);
- есть и парсеры, в числе последних - [Bölükü et al. \(2023\)](#).

# Discourse Representation Structures



The man is going to play the piano. Tom may stop him.

**Groningen Meaning Bank** (GMB) - датасет с семантической разметкой в собственном формате, который опирается на Discourse Representation Theory (Kamp and Reyle (2013)).

Этот формат значительно отличается от вышеперечисленных: он гораздо ближе к формальной семантике и использует неодавидсонианскую логику.

Верхний уровень разметки в DRS - весь текст.

# Discourse Representation Structures

- идея DRT - уровень ментальных представлений: говорящий строит такое ментальное представление и дополняет его в течение разговора;
- разметка идет поверх синтаксической разметки в парадигме Combinatory Categorial Grammar (деревья составляющих);
- все представляется как события: события - это сущности первого порядка, характеризуемые одноместными символами предикатов;
- события совмещаются со своими семантическими аргументами через инвентарь тематических ролей (который берется из VerbNet);
- таким образом, размечаются:
  - семантические роли (тематические)
  - смыслы слов (по WordNet)
  - кореференция
  - значения истинности
  - именованные сущности
  - отношения между фрагментами дискурса (текста)

# Discourse Representation Structures

Поскольку таблички неудобно парсить, существует конвертация в DAGs:  
[Abzianidze et al. \(2020\)](#).

Есть несколько парсеров ([Van Noord et al. \(2018\)](#), [Liu et al. \(2018\)](#)), но в последнее время значительных работ по парсингу DRS нет.

В DMB представлены английский, немецкий, итальянский и нидерландский языки.

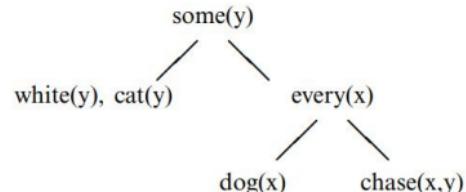
# Minimal Recursion Semantics

Этот формат семантической разметки базируется на HPSG; описание есть в [Copestate et al. \(2005\)](#).

Размечаются элементарные предикации (EPs): единичные отношения с аргументами; точно так же, как и DRS, формат близок к формальной семантике и логике.

Эта разметка представлена в [LinGO Redwoods Treebank](#). Активно сегодня ей не занимаются, но есть некоторые описания ресурсов в [Copestate et al. \(2016\)](#).

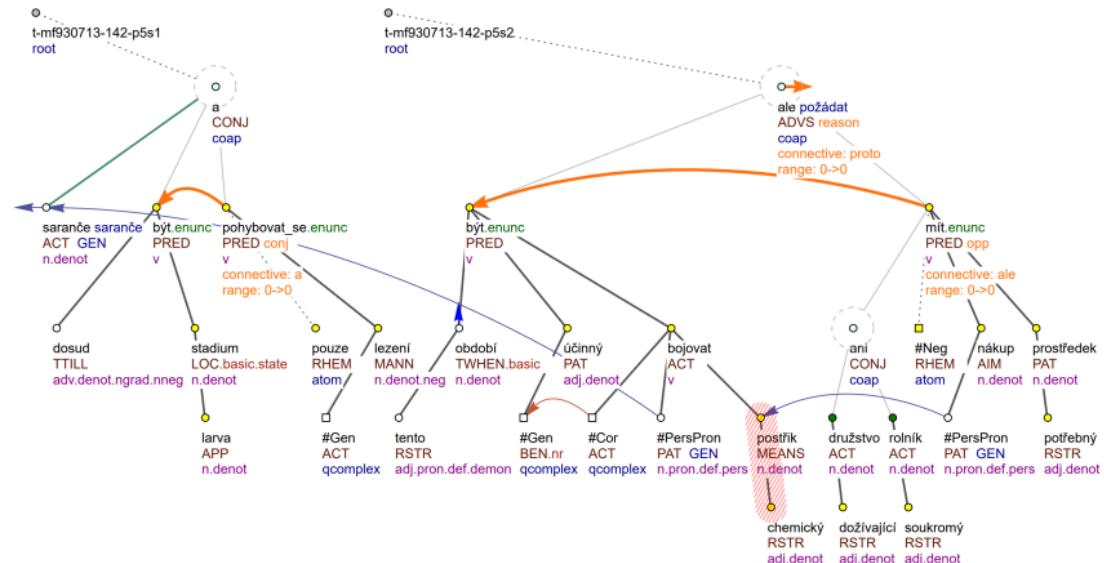
- a.  $\text{some}(y, \text{white}(y) \wedge \text{cat}(y), \text{every}(x, \text{dog}(x), \text{chase}(x, y)))$
- b.



- c.  $h1: \text{every}(x, h3, h4), h3: \text{dog}(x), h7: \text{white}(y), h7: \text{cat}(y), h5: \text{some}(y, h7, h1), h4: \text{chase}(x, y)$

# Prague Tectogrammatical Graphs

**Prague Dependency Treebank** - старый и хорошо известный корпус с полной лингвистической разметкой чешских, английских и арабских текстов.



# Prague Tectogrammatical Graphs

- теория, на которую опирается - Functional Generative Description ([Sgall and Hajíčová \(1971\)](#)).
- размечает текст слоями: токенизированный, морфологический, аналитический (синтаксис), тектограмматический (семантика).
- базовый элемент разметки - предложение.
- восстанавливает эллипсис;
- есть референтные связи;
- размечает актуальное членение предложения (тему и рему);
- не размечает семантические классы слов, только семантические роли.
- существуют **парсеры**, а также попытки создавать новые датасеты с помощью XLT ([Novák et al. \(2021\)](#)).

# MRP Tasks

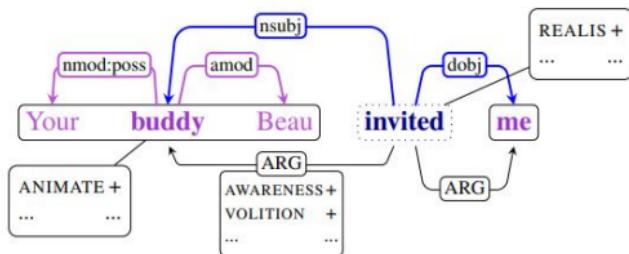
В 2019 и 2020 году на конференции Conference for Computational Language Learning (CoNLL) проводились два соревнования: Meaning Representation Parsing (MRP).

Целью этих соревнований была автоматическая разметка в различных форматах.

- **MRP 2019 Shared Task:** DM (DELPH-IN MRS BiLexical Dependencies), PSD (Prague Semantic Dependencies), EDS (Elementary Dependency Structures), UCCA и AMR, язык - только английский;
- **MRP 2020 Shared Task:** DRS, EDS, UCCA, AMR, языки - английский + один дополнительный язык для каждого формата.

DM и EDS - преобразования над Minimal Recursion Semantics; PSD - упрощенная разметка Prague Tectogrammatical Graphs.

# Universal Decompositional Semantics



Как можно заметить, форматов семантической разметки много, и в основном они различны между собой.

Однако хочется иметь какой-то универсальный трехуровневый формат разметки: как Universal Dependencies объединил в себе морфологию и синтаксис, так же

добавить туда семантику.

С этой целью в 2016 году был предложен формат Universal Decompositional Semantics (UDS, [White et al. \(2016\)](#)).

В этом формате каждое слово получает набор характеристик - ответов на вопросы, размеченные датасеты есть только для **английского**.

# ABBYY Compreno

Один из двух проектов, занимающихся семантикой русского языка - это Compreno ([Anisimovich et al. \(2012\)](#)). Compreno - проприетарная модель, однако в 2023 году компания ABBYY выложила в открытый доступ **некоторые материалы** для нее.

В модели размечается:

- морфология: части речи, леммы, грамматические характеристики
- синтаксис: поверхностные позиции (подлежащее, дополнение...) и границы составляющих
- синтаксис: синтаксические характеристики, такие, как способность присоединять зависимые определенных типов
- семантика: семантические роли (глубинные позиции) и семантические классы
- семантика: кореферентность
- восстанавливается эллипсис (Compreno вообще любит анализировать с помощью нулей)

## ABBYY Compreno

Обычно бюджет ко второму чтению готовится непосредственно в Думе: депутаты корректируют правительственные планы. 'Usually the budget is prepared for the second reading directly in the Duma: the deputies update the government plans'.

```
#[[Time: Обычно"обычно:#frequentative_adverbs_adj:FREQUENTATIVE"] [Experiencer_Metaphoric:  
бюджет"бюджет:BUDGET"] [[ко"к:#preposition:PREPOSITION"] [OrderInTimeAndSpace:  
второму"второй:TWO_ORDINAL"] Object_Situation: чтению "чтение:READING_OF_THE_DRAFT_LAW"] Predicate:  
готовится"готовить:готовить:PREPAREDNESS" [[DegreeApproximative:  
непосредственно"непосредственный:DIRECT_OBLIQUE"] [в"в_Prepositional:#preposition:PREPOSITION"] Locative:  
Думе"дума:дума:DUMA"]#: [[Agent: депутаты"депутат:депутат:DEPUTY"] Specification_Clause:  
корректируют"корректировать:корректировать:TO_CORRECT" [[Agent:  
правительственные"правительство:правительство:GOVERNMENT"] Object_Situation:  
планы"план:план:SCHEDULE_FOR_ACTIVITY"]]]|
```

Деревья составляющих Compreno можно записывать в плоском скобочном виде (тогда морфологию и подробные характеристики не видно). Красным цветом выделены семантические роли, зеленым цветом - семантическая разметка: заглавными буквами написан семантический класс, левее от него через двоеточие идет лексический класс (конкретное языковое наполнение) и может быть лемма.

# ABBYY Compreno

У Compreno своя развитая семантическая иерархия из более чем 200 тысяч семантических классов, при которых описывается все остальное:

**Navigation**

- > GRAMMATICAL\_ELEMENTS
- IDIOMATICAL\_ELEMENTS
- DISCOURSIVE\_UNITS
- ✓ LEXICAL\_ELEMENTS
  - ✓ ENTITY\_LIKE\_CLASSES
    - > COGNITIVE\_CATEGORIES
  - ✓ ENTITY
    - > ABSTRACT\_SCIENTIFIC\_OBJECTS
    - > ADMINISTRATIVE\_AND\_TERRITORIAL\_UNIT
    - > AGGREGATE
    - > COMMUNICATIONS
    - > ENTITY\_BY\_FUNCTION\_AND\_PROPERTY
    - > ENTITY\_GENERAL
    - > FOOD
  - > INFORMATION\_AND\_SOCIAL\_OBJECTS
  - MENTAL\_OBJECT
  - > ORGANIZATION
  - > PART\_OR\_PORTION\_OF\_ENTITY
  - > PHYSICAL\_OBJECT
  - SUBSTANCE
  - OBJECTS\_BY\_FORM\_OF\_MANIFESTATION
  - > SPACE\_AND\_SPATIAL\_OBJECTS
  - TIME
- > ENTITY\_OR\_SITUATION\_PRONOUN
- > SITUATIONAL\_AND\_ATTRIBUTIVE\_CLASSES

**Class "FOOD", language "en"**

**Languages**

Type a language tag:  OR choose from existing:

**Comments**

Food:  
*pasta, salad, cutlet, pizza*

**Examples**

1. This has been brought about by eating the right **foods** (FOOD) and cutting out the **snacks** (FOOD).
2. A further 200 jobs at the Department of the Environment, **Food** (FOOD) and Rural Affairs have been earmarked to be cut.
3. Payment levels vary from area to area, with some carers getting just £50 a week for clothes, **food** (FOOD) and other costs.

# CoBaLD Annotation Project

Модель Compreno представляет очень подробное описание семантики; хотя категорий много, сама их структура довольно просто устроена. Однако из-за своей чрезмерной подробности модель сложна в использовании (разметчики учатся по полгода и сдают экзамен!), не говоря уже о том, что в открытый доступ выложить ее нельзя.

Однако на базе модели в 2023 году был разработан формат CoBaLD (Compreno-Based Linguistic Data).

Одна из целей формата - совместить семантику Compreno и морфосинтаксис UD, сделать это все легким и удобным в использовании.

CoBaLD добавляет в CONLL-U две колонки: семантическая роль и семантический класс.



# CoBaLD Annotation Project

Семантические роли и классы при этом значительно упрощены: берутся только обобщающие классы-гиперонимы, так что количество классов оказывается всего около тысячи, а ролей - около 150 (из 1000+).

## Sentence № 3

A globalizációról és annak a magyar gazdaságra gyakorolt hatásairól Békesi László volt pénzügyminiszter tartott előadást a Pázmány Péter Katolikus Egyetem Jogtudományi Karán, a Barankovics és Faludi Akadémia, valamint a Pázmány Pódium által közösen szervezett vitasorozat keretében.

## TRANSLATION

Former Minister of Finance László Békesi gave a lecture on globalization and its effects on the Hungarian economy at the Faculty of Law of Pázmány Péter Catholic University, the Barankovics and Faludi Academy, and the Pázmány Pódium.

| ID  | FORM             | LEMMA            | UPOS  | HEAD | DEPREL    | DEPS | SEMSLOT                  | SEMCLASS  |
|-----|------------------|------------------|-------|------|-----------|------|--------------------------|---|
| 1.  | Á                | á                | DET   | 2    | det       | -    | =                        | PREPOSITION                                       |
| 2.  | globalizációról  | globalizáció     | NOUN  | 14   | obl       | -    | Theme                    | CH_SPHERE_OF_COVERAGE                             |
| 3.  | és               | és               | CCONJ | 9    | cc        | -    | =                        | COORDINATING_CONJUNCTIONS                         |
| 4.  | annak            | az               | PRON  | 9    | nom:att   | -    | Object_Situation         | =   |
| 5.  | á                | á                | DET   | 7    | det       | -    | =                        | PREPOSITION                                       |
| 6.  | magyar           | magyar           | ADJ   | 7    | amod:att  | -    | =                        | COUNTRY_AS_ADMINISTRATIVE_UNIT                    |
| 7.  | gazdaságra       | gazdaság         | NOUN  | 8    | obl       | -    | Experiencer              | ECONOMY   |
| 8.  | gyakorolt        | gyakorolt        | ADJ   | 9    | amod:att  | -    | ParticipleRelativeClause | TO_COMMIT   |
| 9.  | hatásairól       | hatás            | NOUN  | 2    | conj      | -    | Theme                    | CH_POWER_AND_EFFECT                               |
| 10. | Békesi           | Békesi           | PROPN | 14   | nsubj     | -    | Name_Title               | BEING   |
| 11. | László           | László           | PROPN | 10   | flat:name | -    | Name_Title               | BEING   |
| 12. | volt             | van              | VERB  | 10   | amod:att  | -    | OrderInTimeAndSpace      | CH_REFERENCE_AND_QUANTIFICATION                   |
| 13. | pénzügyminiszter | pénzügyminiszter | NOUN  | 10   | appos     | -    | Agent                    | HUMAN   |
| 14. | tartott          | tart             | VERB  | 0    | root      | -    | Predicate                | TO_COMMIT   |
| 15. | előadást         | előadás          | NOUN  | 14   | obj       | -    | Object_Situation         | VERBAL_COMMUNICATION                              |
| 16. | á                | á                | DET   | 21   | det       | -    | =                        | PREPOSITION                                       |
| 17. | Pázmány          | Pázmány          | PROPN | 14   | obl       | -    | Name_Title               | BEING   |
| 18. | Péter            | Péter            | PROPN | 17   | flat:name | -    | =                        | BEING   |
| 19. | Katolikus        | Katolikus        | PROPN | 17   | flat:name | -    | Characteristic           | WORLD_OUTLOOK                                     |
| 20. | Rektátor         | Rektátor         | PROPN | 17   | flat:name | -    | Ent_General              | CULTURAL_EDUCATIONAL_AND_EDUCATIONAL_INSTITUTIONS |

## CoBaLD Annotation Project: датасеты

Существует две версии формата: 1.0 - основана на базовых UD; 2.0 - основана на E-UD.

На [гитхабе проекта](#) опубликованы русский датасет в версии 1.0 и английский датасет в версии 2.0. Готовятся венгерский и сербский датасеты версии 1.0.

Русский и английский датасеты были сконвертированы в формат, совместимый с UD (E-UD), автоматически из разметки Compreno, поэтому особенно в E-UD может отличаться восстановление эллипсиса: Compreno восстанавливает эллипсис по-своему, например, считает, что в фразе 'This is a sentence' есть эллиптикованное подлежащее после 'this'.

На сегодняшний день CoBaLD Rus - единственный опубликованный русскоязычный датасет с полной семантической разметкой.

# CoBaLD Annotation Project: соревнования

- SEMarkup 2023 - первое соревнование, посвященное разметке в формате CoBaLD (трехуровневая разметка).  
Уже бейзлайн этого соревнования на ruBERT-tiny показал качество семантической разметки F1 в районе 90%.
- CoBaLD Parsing 2024 - это соревнование готовится к осени 2024 года и будет включать две дорожки:
  - E-Parse - разработка трехуровневого парсера для формата версии 2.0.
  - CLT - few-shot языковой перенос с русского и английского на сербский и венгерский язык.

# CoBaLD Annotation Project: XLT

Формат автоматически переносился на данные сербского, венгерского, чешского, турецкого и корейского языков.

Предварительные оценки лингвистов показывают, что качество переноса очень высокое: это свидетельствует о том, что формат действительно отражает универсальную семантику языка.

```
# text = Dışisleri Bakanı Davutoğlu, Yunanistan ile Türkiye arasındaki farllilikların ortak vizyon ile çözülebileceğini söyledi.
# text = Foreign Minister Davutoğlu said that the differences between Greece and Turkey can be resolved with a common vision.
1 Dışisleri dışisleri NOUN _ Case=Nom|Number=Plur|Person=3 2 nmod Object STATE_AUTHORITIES
2 Bakanı bakan NOUN _ Case=Nom|Number=Sing|Number[psor]=Sing|Person=3|Person[psor]=3 3 nmod Agent HUMAN
3 Davutoğlu Davutoğlu PROPN _ Case=Nom|Number=Sing 14 nsubj Name_Title BEING
4 , PUNCT _ 14 punct -
5 Yunanistan yunanistan PROPN _ Case=Nom|Number=Sing 8 nmod Correlative COUNTRY_AS_ADMINISTRATIVE_UNIT
6 ile ile CCONJ _ 7 cc COORDINATING_CONJUNCTIONS
7 Türkiye türkiye PROPN _ Case=Nom|Number=Sing 5 conj Correlative COUNTRY_AS_ADMINISTRATIVE_UNIT
8 arasındaki ara ADJ _ 9 amod CH_OF_CONNECTIONS
9 farllilikların farllilik NOUN _ Case=Gen|Number=Plur|Person=3 13 nsubj Object_Situation CH_OF_CONNECTIONS
10 ortak ortak ADJ _ 11 amod StaffOfPossessors CH_TYPE_OF_POSSESSION_AND_PARTICIPATION
11 vizyon vizyon NOUN _ Case=Nom|Number=Sing|Person=3 13 obl Instrument MENTAL_OBJECT
12 ile ile CCONJ _ 11 case PREPOSITION
13 çözülebileceğini çöz NOUN _ Case=Acc|Number=Sing|Number[psor]=Sing|Person=3|Person[psor]=3 14 ccomp Object_Situation MANAGE_FAIL_CONDITION
14 söyledi söyle VERB _ Aspect=Perf|Mood=Ind|Number=Sing|Person=3|Polarity=Pos|Tense=Past|VerbForm=Fin 0 root Predicate VERBAL_COMMUNICATION
15 . . PUNCT _ 14 punct -
```

# Семантическая разметка: подводя итоги

- Задача SRL - разметка только семантических ролей (глаголов); SP - полная семантическая разметка, может включать в себя самые разные вещи.
- Для семантики существует самое большое количество разных форматов, причем какого-то общепринятого наподобие UD пока нет.
- Форматы разметки могут опираться на традиционные лингвистические теории (семантические роли) или на формальную семантику и логику.
- Форматы разметки могут использовать банки данных (UNL, Etap, AMR, CoBaLD) или не использовать (UCCA). Чаще используют.
- В качестве метрик обычно используется F1, CoBaLD использует адаптированные под семантику UAS и LAS.
- Семантика считается универсальным уровнем языка, и языковой перенос обычно дает очень высокие результаты.

## Разметка дискурса

## Разметка дискурса

Дискурс в лингвистике - это человеческая речь в pragматическом контексте; иными словами, это уже реализованные тексты (устные или письменные). В рамках дискурса обычно изучают такие вещи, как связи внутри текста и способы реализации (как человек строит предложения, какие использует средства для связи: например, дискурсивные маркеры).

Основное понятие в дискурсе - элементарная дискурсивная единица (ЭДЕ, EDU).

С дискурсом также довольно тесно связано актуальное членение предложения (тема-рема).

Разметка дискурса может быть полезной для самых разных задач NLP.

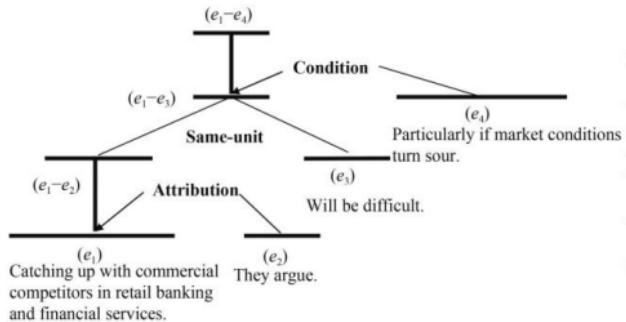
Обзор по теме: [Li et al. \(2022\)](#)

# Разметка дискурса

На сегодня тексты могут быть размечены в основном в двух форматах:

- RST (Rhetorical Structure Theory): ЭДЕ представляются в виде листьев дерева;
- PDTB (Penn Discourse TreeBank): связи выписываются в виде короткого текста.

RST discourse parsing



PDTB discourse parsing

## Explicit relation

**Connective:** if

**Arg1:** catching up with commercial competitors in retail banking and financial services will be difficult.

**Arg2:** market conditions turn sour.

**Sense:** contingency.Condition.Hypothetical.

# Разметка дискурса: парсеры

- RST-Parsing:

Из числа последних работ - [Hu and Wan \(2023\)](#), [Li and Huang \(2023\)](#). Также недавно появился парсер с использованием LLM: [Maekawa et al. \(2024\)](#).

- PDTB-Parsing:

PDTB в последнее время пользуется меньшей популярностью; из недавних работ - [Zhao and Webber \(2022\)](#), [Kutlu et al. \(2023\)](#).

# Разметка дискурса: датасеты

- Penn Discourse Treebank - английский
- RST Discourse Treebank - английский
- Для русского языка теоретические лингвисты размечают два мультиканальных датасета - "Рассказы о сновидениях" и "Рассказы о груше". В датасетах размечаются границы ЭДЕ и способы невербальной коммуникации.

# Лингвистическая разметка: итоги

- Три базовых уровня разметки - морфологический, синтаксический, семантический.
- Общепринятый стандарт разметки морфосинтаксиса - формат Universal Dependencies. Для семантики такого стандарта пока нет, хотя несколько форматов стремятся им стать.
- С технической точки зрения сложнее всего парсить синтаксические связи.
- Для всех видов задач в последнее время активно используется метод XLT: особенно хорошо переносится семантика.
- Для некоторых видов разметки существует огромное количество датасетов (UD), для других датасетов почти нет (деревья составляющих, разметка дискурса).
- Лингвистическая разметка может использоваться как вспомогательное средство для решения практических задач; например, для обогащения эмбеддингов.
- В корпусах лингвистическая разметка нужна для теоретических исследований. Во всех корпусах есть морфологическая разметка, во многих появляется синтаксическая и семантическая (н-р, в НКРЯ).

# References I

- Abend, O. and Rappoport, A. (2013). Universal conceptual cognitive annotation (ucca). In Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 228–238.
- Abzianidze, L., Bos, J., and Oepen, S. (2020). Drs at mrp 2020: Dressing up discourse representation structures as graphs. arXiv preprint arXiv:2012.14837.
- Ali, M. N. Y., Rahman, M. L., Chaki, J., Dey, N., and Santosh, K. (2021). Machine translation using deep learning for universal networking language based on their structure. International Journal of Machine Learning and Cybernetics, 12(8):2365–2376.
- Altıntaş, M. and Tantuğ, A. C. (2023). Improving the performance of graph based dependency parsing by guiding bi-affine layer with augmented global and local features. Intelligent Systems with Applications, 18:200190.
- Anisimovich, K., Druzhkin, K. Y., Zuev, K., Minlos, F., Petrova, M., and Selegei, V. (2012). Syntactic and semantic parser based on abbyy comprehendo linguistic technologies. In Компьютерная лингвистика и интеллектуальные технологии, pages 91–103.
- Bai, X., Wu, J., Chen, Y., Wang, Z., and Zhang, Y. (2023). Constituency parsing using llms. arXiv preprint arXiv:2310.19462.
- Banerescu, L., Bonial, C., Cai, S., Georgescu, M., Griffitt, K., Hermjakob, U., Knight, K., Koehn, P., Palmer, M., and Schneider, N. (2013). Abstract meaning representation for sembanking. In Proceedings of the 7th linguistic annotation workshop and interoperability with discourse, pages 178–186.
- Boguslavsky, I., Inshakova, E., Lazursky, A., Timoshenko, S., Dikonov, V., Iomdin, L., Rygaev, I., and Frolova, T. (2023). Constructing a semantic corpus for russian: Semontocor. In Proceedings of the International Conference Dialogue, volume 2023.

## References II

- Bölükü, N., Can, B., and Artuner, H. (2023). A siamese neural network for learning semantically-informed sentence embeddings. *Expert Systems with Applications*, 214:119103.
- Börjars, K., Nordlinger, R., and Sadler, L. (2019). Lexical-functional grammar: An introduction. Cambridge University Press.
- Brants, T. (2000). Tnt-a statistical part-of-speech tagger. arXiv preprint cs/0003055.
- Brill, E. (1992). A simple rule-based part of speech tagger. In *Speech and Natural Language: Proceedings of a Workshop Held at Harriman, New York, February 23-26, 1992*.
- Chomsky, N. (1965). Aspects of the theory of syntax. Special technical report. Research laboratory of electronics. Massachusetts institute of technology (.
- Chomsky, N. (1995). The minimalist program. MIT Press.
- Choudhary, C. and O'riordan, C. (2023). Multilingual end-to-end dependency parsing with linguistic typology knowledge. In *Proceedings of the 5th Workshop on Research in Computational Linguistic Typology and Multilingual NLP*, pages 12–21.
- Copestake, A., Emerson, G., Goodman, M. W., Horvat, M., Kuhnle, A., and Muszyńska, E. (2016). Resources for building applications with dependency minimal recursion semantics. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 1240–1247.
- Copestake, A., Flickinger, D., Pollard, C., and Sag, I. A. (2005). Minimal recursion semantics: An introduction. *Research on language and computation*, 3:281–332.
- Das, A. and Sarkar, S. (2020). A survey of the model transfer approaches to cross-lingual dependency parsing. *ACM Transactions on Asian and Low-Resource Language Information Processing (TALLIP)*, 19(5):1–60.

## References III

- De Marneffe, M.-C., Dozat, T., Silveira, N., Haverinen, K., Ginter, F., Nivre, J., and Manning, C. D. (2014). Universal stanford dependencies: A cross-linguistic typology. In LREC, volume 14, pages 4585–4592.
- De Marneffe, M.-C. and Manning, C. D. (2008). The stanford typed dependencies representation. In Coling 2008: proceedings of the workshop on cross-framework and cross-domain parser evaluation, pages 1–8.
- De Marneffe, M.-C., Manning, C. D., Nivre, J., and Zeman, D. (2021). Universal dependencies. Computational linguistics, 47(2):255–308.
- de Vries, W., Wieling, M., and Nissim, M. (2022). Make the best of cross-lingual transfer: Evidence from pos tagging with over 100 languages. In The 60th Annual Meeting of the Association for Computational Linguistics, pages 7676–7685. Association for Computational Linguistics (ACL).
- Dozat, T. and Manning, C. D. (2016). Deep biaffine attention for neural dependency parsing. arXiv preprint arXiv:1611.01734.
- Fillmore, C. J. (1967). The case for case.
- Garcia, M. (2021). Exploring the representation of word meanings in context: A case study on homonymy and synonymy. arXiv preprint arXiv:2106.13553.
- Goldberg, A. and Suttle, L. (2010). Construction grammar. Wiley Interdisciplinary Reviews: Cognitive Science, 1(4):468–477.
- Grünewald, S., Oertel, F. T., and Friedrich, A. (2021). Robertnlp at the iwpt 2021 shared task: Simple enhanced ud parsing for 17 languages. In Proceedings of the 17th International Conference on Parsing Technologies and the IWPT 2021 Shared Task on Parsing into Enhanced Universal Dependencies (IWPT 2021), pages 196–203.

## References IV

- Hu, X. and Wan, X. (2023). Rst discourse parsing as text-to-text generation. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*.
- Ivoylova, A., Dyachkova, D., Petrova, M., and Michurina, M. (2023). The problem of linguistic markup conversion: the transformation of the compreno markup into the ud format. In International Conference on Computational Linguistics and Intellectual Technologies «Dialog».
- Jackendoff, R. (1975). A system of semantic primitives. In *Theoretical issues in natural language processing*.
- Kamigaito, H., Hayashi, K., Hirao, T., Takamura, H., Okumura, M., and Nagata, M. (2017). Supervised attention for sequence-to-sequence constituency parsing. In Kondrak, G. and Watanabe, T., editors, *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 7–12, Taipei, Taiwan. Asian Federation of Natural Language Processing.
- Kamp, H. and Reyle, U. (2013). From discourse to logic: Introduction to modeltheoretic semantics of natural language, formal logic and discourse representation theory, volume 42. Springer Science & Business Media.
- Kim, J.-K., Kim, Y.-B., Sarikaya, R., and Fosler-Lussier, E. (2017). Cross-lingual transfer learning for pos tagging without cross-lingual resources. In *Proceedings of the 2017 conference on empirical methods in natural language processing*, pages 2832–2838.
- Kutlu, F., Zeyrek, D., and Kurfali, M. (2023). Toward a shallow discourse parser for turkish. *Natural Language Engineering*, pages 1–26.
- Langacker, R. W. (1995). Cognitive grammar. In *Concise History of the Language Sciences*, pages 364–368. Elsevier.
- Le-Hong, P. and Cambria, E. (2024). Integrating graph embedding and neural models for improving transition-based dependency parsing. *Neural Computing and Applications*, 36(6):2999–3016.

# References V

- Levi, D. Y. and Tsarfaty, R. (2024). A truly joint neural architecture for segmentation and parsing. arXiv e-prints, pages arXiv-2402.
- Li, J., Liu, M., Qin, B., and Liu, T. (2022). A survey of discourse parsing. *Frontiers of Computer Science*, 16(5):165329.
- Li, M. and Huang, R. (2023). Rst-style discourse parsing guided by document-level content structures. arXiv preprint arXiv:2309.04141.
- Liu, J., Cohen, S. B., and Lapata, M. (2018). Discourse representation structure parsing. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 429–439.
- Liu, J. and Zhang, Y. (2017). In-order transition-based constituent parsing. *Transactions of the Association for Computational Linguistics*, 5:413–424.
- Lyashevskaya, O., Afanasev, I., Rebrikov, S., Shishkina, Y., Suleymanova, E., Trofimov, I., and Vlasova, N. (2023). Disambiguation in context in the russian national corpus: 20 years later. In *Proceedings of the International Conference “Dialogue”, volume 2023*.
- Maekawa, A., Hirao, T., Kamigaito, H., and Okumura, M. (2024). Can we obtain significant success in rst discourse parsing by using large language models? arXiv preprint arXiv:2403.05065.
- Matthiessen, C. M. and Halliday, M. A. K. (2009). Systemic functional grammar: A first step into the theory.
- Michurina, M., Ivoylova, A., Kopylov, N., and Selegey, D. (2021). Morphological annotation of social media corpora with reference to its reliability for linguistic research. *Komp'juternaja Lingvistika i Intellektual'nye Tehnologii*, pages 492–504.

## References VI

- Nivre, J. and Fang, C.-T. (2017). Universal dependency evaluation. In Proceedings of the NoDaLiDa 2017 Workshop on Universal Dependencies (UDW 2017), pages 86–95.
- Novák, A., Novák, B., and Novák, C. (2021). Zero-shot cross-lingual meaning representation transfer: Annotation of hungarian using the prague functional generative description. In Proceedings of the Joint 15th Linguistic Annotation Workshop (LAW) and 3rd Designing Meaning Representations (DMR) Workshop, pages 1–11.
- Petrova, M., Ivoylova, A., Bayuk, I., Dyachkova, D., and Michurina, M. (2023). The cobald annotation project: the creation and application of the full morpho-syntactic and semantic markup standard. In Proceedings of the International Conference "Dialogue, volume 2023.
- Pollard, C. and Sag, I. A. (1994). Head-driven phrase structure grammar. University of Chicago Press.
- Ponti, E. M., Vulić, I., Glavaš, G., Mrkšić, N., and Korhonen, A. (2018). Adversarial propagation and zero-shot cross-lingual transfer of word vector specialization. arXiv preprint arXiv:1809.04163.
- Sagae, K. and Lavie, A. (2005). A classifier-based parser with linear run-time complexity. In Proceedings of the Ninth International Workshop on Parsing Technology, pages 125–132.
- Schmid, H. (1999). Improvements in part-of-speech tagging with an application to german. In Natural language processing using very large corpora, pages 13–25. Springer.
- Schuster, S. and Manning, C. D. (2016). Enhanced english universal dependencies: An improved representation for natural language understanding tasks. In Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16), pages 2371–2378.
- Segalovich, I. (2003). A fast morphological algorithm with unknown word guessing induced by a dictionary for a web search engine. MLMTA, 2003:273.

## References VII

- Sgall, P. and Hajičová, E. (1971). A „functional” generative description: background and framework.
- Shi, T. and Lee, L. (2021). Tgif: Tree-graph integrated-format parser for enhanced ud with two-stage generic-to individual-language finetuning. arXiv preprint arXiv:2107.06907.
- Steedman, M. and Baldridge, J. (2011). Combinatory categorial grammar. Non-Transformational Syntax: Formal and Explicit Models of Grammar, pages 181–224.
- Straka, M. (2018). Udpipe 2.0 prototype at conll 2018 ud shared task. In Proceedings of the CoNLL 2018 shared task: Multilingual parsing from raw text to universal dependencies, pages 197–207.
- Straka, M., Hajic, J., and Straková, J. (2016). Udpipe: trainable pipeline for processing conll-u files performing tokenization, morphological analysis, pos tagging and parsing. In Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC’16), pages 4290–4297.
- Sun, K., Li, Z., and Zhao, H. (2023). Cross-lingual universal dependency parsing only from one monolingual treebank. IEEE Transactions on Pattern Analysis and Machine Intelligence.
- Tsarfaty, R., Seddah, D., Goldberg, Y., Kübler, S., Versley, Y., Candito, M., Foster, J., Rehbein, I., and Tounsi, L. (2010). Statistical parsing of morphologically rich languages (spmrl) what, how and whither. In Proceedings of the NAACL HLT 2010 First Workshop on Statistical Parsing of Morphologically-Rich Languages, pages 1–12.
- Van Noord, R., Abzianidze, L., Toral, A., and Bos, J. (2018). Exploring neural methods for parsing discourse representation structures. Transactions of the Association for Computational Linguistics, 6:619–633.
- Van Valin Jr, R. D. (1990). Semantic roles and grammatical relations.

## References VIII

- Watanabe, T. and Sumita, E. (2015). Transition-based neural constituent parsing. In Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers), pages 1169–1179.
- White, A. S., Reisinger, D., Sakaguchi, K., Vieira, T., Zhang, S., Rudinger, R., Rawlins, K., and Van Durme, B. (2016). Universal decompositional semantics on universal dependencies. In Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, pages 1713–1723.
- Yang, K. and Deng, J. (2020). Strongly incremental constituency parsing with graph neural networks. Advances in neural information processing systems, 33:21687–21698.
- Yang, S. and Tu, K. (2023). Don't parse, choose spans! continuous and discontinuous constituency parsing via autoregressive span selection. In Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 8420–8433.
- Zhao, Z. and Webber, B. (2022). Revisiting shallow discourse parsing in the pdtb-3: handling intra-sentential implicits. arXiv preprint arXiv:2204.00350.
- Дыбо, А. and Шеймович, А. (2014). Автоматический морфологический анализ для корпусов хакасского и древнетюркского языков. Научное обозрение Саяно-Алтая, pages 9–30.