Text mining

2. Синтаксисический анализ

Дмитрий Ильвовский, Екатерина Черняк

dilvovsky@hse.ru, echernyak@hse.ru

Национальный Исследовательский Университет – Высшая Школа Экономики
НУЛ Интеллектуальных систем и структурного анализа

February 1, 2017

Синтаксический анализ

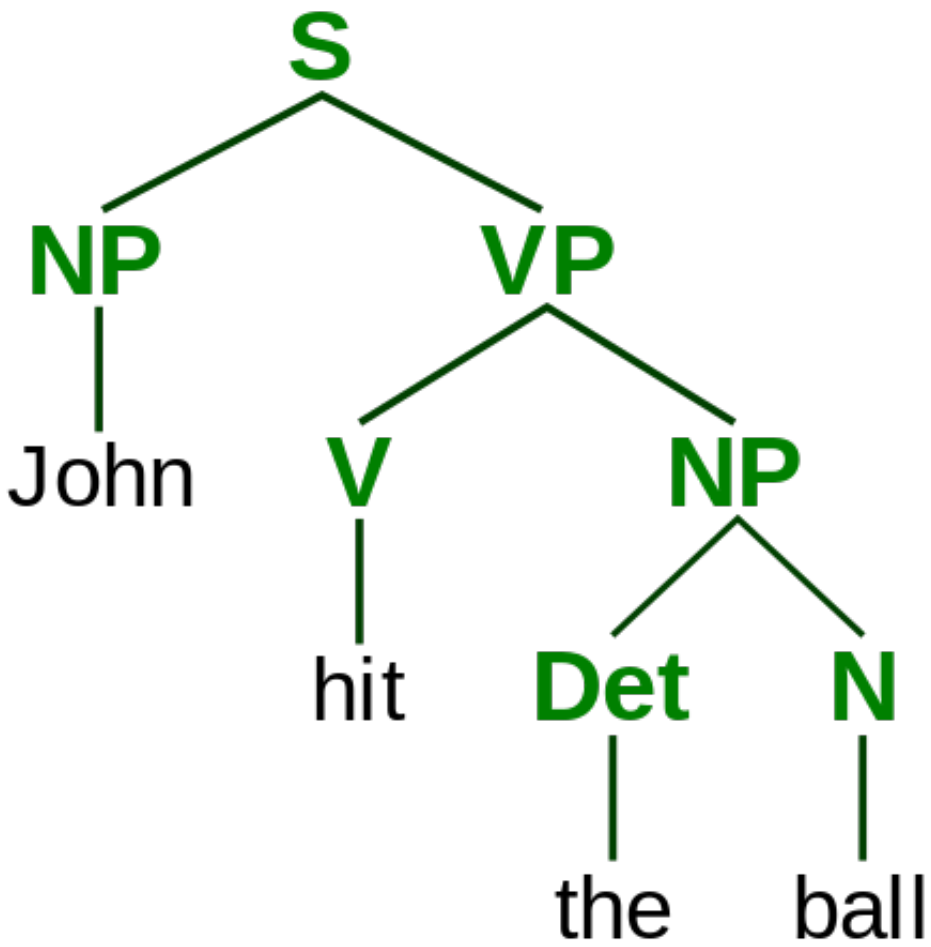
- Определение синтаксических связей между словами
  - Частичный синтаксический анализ: выделяют связи определенного вида
  - Полный синтаксический анализ: каждое предложение представляется в виде дерева
- Выделяют связи двух видов:
  - составляющие (constituency) – фразовая структура предложения
  - зависимости (dependency) – иерархическая структура предложения

	Составляющие	Зависимости
Частичный разбор	Группы (ИГ, ГГ, П) (chunking)	Определение семантических ролей (semantic role labelling)
Полный разбор	Дерево зависимостей (constituency tree)	Дерево составляющих (dependency tree)

Генеративная модель языка

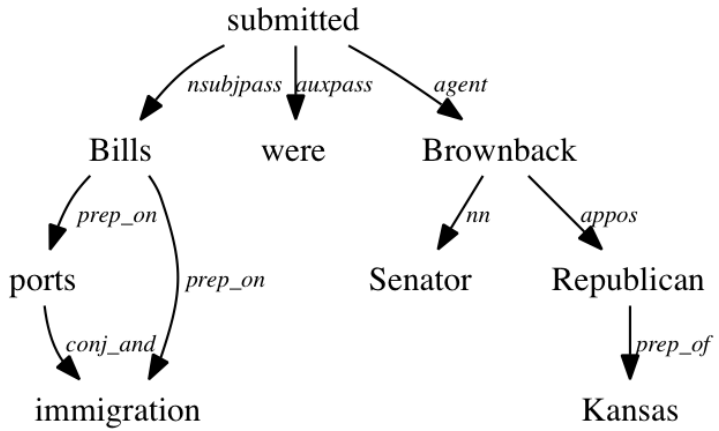
- Язык – множество цепочек слов
- Правила порождения цепочек описываются формальными грамматиками Хомского
- Грамматика: правила вида $[aAbB] \rightarrow [aBc]$, слева и справа цепочки терминальных и нетерминальных символов
- 4 вида грамматик:
  - Неограниченные грамматики
  - Контекстно-зависимые и неукорачивающие грамматики
  - Контекстно-свободные грамматики
  - Регулярные грамматики
- Для естественных языков используются контекстно-свободные грамматики вида $A \rightarrow aBa$
  - Слева – ровно один нетерминальный символ
  - Справа – произвольная цепочка
- Дерево вывода цепочки-предложения – дерево составляющих

Пример дерева составляющих



- S, NP, VP – нетерминальные символы
- V, N, Det – терминальные символы

Пример дерева зависимостей



- Все слова в предложении связаны отношением типа “хозяин-слуга”, имеющим различные подтипы
- Узел дерева – слово в предложении
- Дуга дерева – отношение подчинения

Деревья зависимостей

1. Правила (rule-based)
   - Набор шаблонов, схем, правил вывода, использующих лингвистические сведения
   - Зависит от языка
   - ЭТАП-3
2. Машинное обучение
   - Корпуса с морфологической и синтаксической разметкой
   - Не требуется знание специфики языка
   - MaltParser
3. Предложение с проективными связями может быть преобразовано в дерево составляющих

Демо

- Berkley Tomcat constituency parser `http://tomato.banatao.berkeley.edu:8080/parser/parser.html`
- Stanford CoreNLP dependency parser `http://nlp.stanford.edu:8080/corenlp/`
- ARK dependency parser (Carnegie Melon) `http://demo.ark.cs.cmu.edu/parse`

Универсальные зависимости

Universal Dependencies

Universal Dependencies (UD) is a project that is developing cross-linguistically consistent treebank annotation for many languages, with the goal of facilitating multilingual parser development, cross-lingual learning, and parsing research from a language typology perspective. The annotation scheme is based on an evolution of (universal) Stanford dependencies (de Marneffe et al., 2006, 2008, 2014), Google universal part-of-speech tags (Petrov et al., 2012), and the Interset interlingua for morphosyntactic tagsets (Zeman, 2008). The general philosophy is to provide a universal inventory of categories and guidelines to facilitate consistent annotation of similar constructions across languages, while allowing language-specific extensions when necessary.

`http://universaldependencies.org/`

Универсальные зависимости

- Части речи (POS tags): 6 открытых классов ЧР (ADJ, ADV, INTJ, PROPN, VERB), 8 закрытых (ADP, AUX, CCONJ, DET, NUM, PART, PRON, SCONJ), 3 других (PUNC, SYM, X)
- Грамматические категории: 6 лексических признаков, 6 признаков словоизменения существительных, 9 признаков словоизменения глаголов
- 37 универсальных синтаксических зависимостей

`http://universaldependencies.org/`

Универсальные зависимости

	Nominals	Clauses	Modifier words	Function Words
Core arguments	nsubj obj iobj	csubj ccomp xcomp		
Non-core dependents	obl vocative expl dislocated	advcl	advmod* discourse	aux cop mark
Nominal dependents	nmod appos nummod	acl	amod	det clf case
Coordination	**MWE**	**Loose**	**Special**	**Other**
conj cc	fixed flat compound	list parataxis	orphan goeswith reparandum	punct root dep

SyntaxNet

Parsey McParseface

At Google, we spend a lot of time thinking about how computer systems can read and understand human language in order to process it in intelligent ways. We are excited to share the fruits of our research with the broader community by releasing SyntaxNet... Our release includes all the code needed to train new SyntaxNet models on your own data, as well as Parsey McParseface, an English parser that we have trained for you, and that you can use to analyze English text.

Parsey's Cousins

Parsey models are now available for 40 languages trained on Universal Dependencies datasets, with support for text segmentation and morphological analysis.

`https://github.com/tensorflow/models/tree/master/syntaxnet`

Архитекстура SyntaxNet

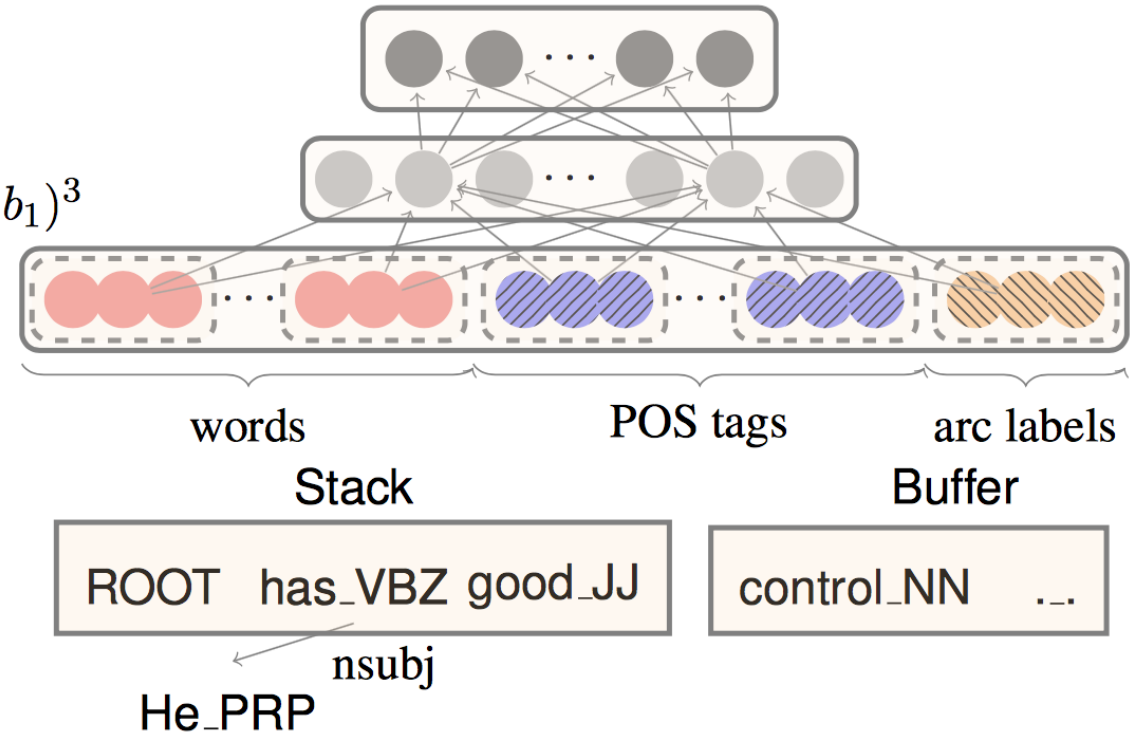
Softmax layer:

$$p = \texttt{softmax}(W_2 h)$$

Hidden layer:

$$h = (W_1^w x^w + W_1^t x^t + W_1^l x^l + b_1)^3$$

Input layer: $[x^w, x^t, x^l]$



Danqi Chen and Christopher D. Manning. A Fast and Accurate Dependency Parser using Neural Networks. EMNLP. 2014.

SyntaxNet для русского

https://github.com/mnvx/syntax-tree