Автоматический морфологический анализ

Еникеева Екатерина Владимировна

27 сентября 2022

Автоматическая обработка естественного языка, лекция 4

Немного о моделях

Модель

- математическая модель
- статистическая модель
- статистическая n-граммная модель языка
- pretrained language model

ELMo is a deep contextualized word representation that models both (1) complex characteristics of word use (e.g., syntax and semantics), and (2) how these uses vary across linguistic contexts (i.e., to model polysemy).

Модель морфологии

- Где традиционно описывается морфологическое устройство языка?
- Как можно описать морфологию ЕЯ?
- Какие процессы относятся к морфологии? Как их смоделировать? (сымитировать?)

Tokenization issues

- **Токенизация –** идентификация словоформ в тексте.
- **Проблема –** единица анализа:
  - аналитические формы (*буду писать*),
  - предлоги, сложные союзы (*как бы, потому что*),
  - сокращения (*и т.п.*),
  - терминологические словосочетания (*железная дорога*),
  - разрывные союзы (*не только, но и*)

Tokenization issues

Осложняющие факторы:

- сегменты текста между пробелами требуют переразложения: *буду (часто) писать; с разбегу;*
- словоформы могут разделяться не только пробелами:

наконец-то (vs кто-то, во-первых, по-моему)

Модель компьютерной морфологии

- какие грамматические категории? (часть речи / род / одушевлённость …)
- с какими граммемами? (значения грамматических категорий)
- с какими формальными показателями?

две букашки (Num f nom + N f nom pl)
на опушке (Prep + N f loc sg)
шьют мышатам две подушки (V tr pres imp + N m dat pl + Num f acc + N f acc pl)

Модель компьютерной морфологии

Проблемы:

- Оптимальное число грамматических категорий, грамматических значений, граммем
- Как анализировать служебную лексику?
- Как решить проблему транспозиции?
- Как быть с грамматической омонимией?

Морфологическая разметка

- *tag / тег / аннотация* словоформы — полный набор значений грамматических категорий словоформы
- *grammeme / граммема* — значение одной грамматической категории (род / время / …)
- *markup / tagging / разметка / аннотация* корпуса
- *lexeme / lemma / лемма* — начальная / нормальная форма

Типы морфологической разметки

- **Позиционная разметка**

  MULTEXT-East
  для корпусов в формате TEI

- Стандартная разметка (тег = множество граммем)

  pymorphy2 (Opencorpora)

  Universal Dependencies

Table 298. Specification for Noun

P	Attribute (en)	Value (en)	Code (en)
0	CATEGORY	Noun	N
1	Type	common	c
		proper	p
2	Gender	masculine	m
		feminine	f
		neuter	n
		common	c
3	Number	singular	s
		plural	p
4	Case	nominative	n
		genitive	g
		dative	d
		accusative	a
		vocative	v
		locative	l
		instrumental	i
5	Animate	no	n
		yes	y
6	Case2	partitive	p
		locative	l

Universal Dependencies

- 17 Universal POS tags
- Features (name + value)

- universal
- language-specific

- lexical
- inflectional
- layered

Open class words	Closed class words	Other
ADJ	ADP	PUNCT
ADV	AUX	SYM
INTJ	CCONJ	X
NOUN	DET	
PROPN	NUM	
VERB	PART	
	PRON	
	SCONJ	

Lexical features	Inflectional features	
	Nominal	*Verbal*
PronType	Gender	VerbForm
NumType	Animacy	Mood
Poss	Number	Tense
Reflex	Case	Aspect
	Definite	Voice
	Degree	Person
		Polarity

Представление разметки (TSV)

- TSV = tab separated values
- CoNLL-U format (StanfordNLP/Stanza, UD):

1. ID: Word index (starting from 1)
2. FORM: Word form or punctuation symbol
3. LEMMA: Lemma or stem of word form
4. UPOS: Universal POS tag
5. XPOS: Language-specific POS tag
6. FEATS: Morphological features
7. HEAD: Head of the current word (ID or 0)
8. DEPREL: Universal dependency relation to the HEAD
9. DEPS: Additional dependencies (graph)
10. MISC: Any other annotation

Пример разметки CoNLL-U

```
1	Сумма	сумма	NOUN	_	Animacy=Inan|Case=Nom|Gender=Fem|Number=Sing	4	nsubj	_	_
2	аренды	аренда	NOUN	_	Animacy=Inan|Case=Gen|Gender=Fem|Number=Sing	1	nmod	_	_
3	не	не	PART	_	Polarity=Neg	4	advmod	_	_
4	включает	включать	VERB	_	Aspect=Imp|Mood=Ind|Number=Sing|Person=3|Tense=Pres|VerbForm=Fin|Voice=Act
5	в	в	ADP	_	_	6	case	_	_
6	себя	себя	PRON	_	Case=Acc	4	obl	_	_
7	коммунальные	коммунальный	ADJ	_	Animacy=Inan|Case=Acc|Degree=Pos|Number=Plur	8	amod	_	_
8	услуги	услуга	NOUN	_	Animacy=Inan|Case=Acc|Gender=Fem|Number=Plur	4	obj	_	_
9	!	!	PUNCT	_	_	4	punct	_	_
```

Представление разметки (XML)

```
<se>•
<w><ana lex="сделать" gr="V,pf,tran=pl,act,2p,imper"></ana>Сд`елайте</w>
<w><ana lex="всё" gr="S-PRO,n,sg=acc"></ana>всё</w>
<w><ana lex="возможный" gr="A=n,sg,acc,inan,plen"></ana>возм`ожное</w> ,
<w><ana lex="чтобы" gr="CONJ"></ana>чт`обы</w>
<w><ana lex="наладить" gr="V,pf,tran=inf,act"></ana>нал`адить</w>
<w><ana lex="дневной" gr="A=m,sg,acc,inan,plen"></ana>дневн`ой</w>
<w><ana lex="сон" gr="S,m,inan=sg,acc"></ana>сон</w> --
<w><ana lex="ослабить" gr="V,pf,tran=partcp,f,sg,dat,pass,praet,plen"></ana>осл`абленной</w>
<w><ana lex="нервный" gr="A=f,sg,dat,plen"></ana>н`ервной</w>
<w><ana lex="система" gr="S,f,inan=sg,dat"></ana>сист`еме</w>
<w><ana lex="необходимый" gr="A=m,sg,brev"></ana>необход`им</w>
<w><ana lex="послеобеденный" gr="A=m,sg,nom,plen"></ana>послеоб`еденный</w>
<w><ana lex="отдых" gr="S,m,inan=sg,nom"></ana>`отдых</w> .</se>
```

(из подкорпуса НКРЯ со снятой омонимией)

Датасеты (English)

- Brown Corpus
- Penn Treebank
- Universal Dependencies

Есть в `nltk.corpus`

Датасеты (для русского)

- Universal Dependencies
- НКРЯ (RNC) https://ruscorpora.ru/new/
- SynTagRus
- GramEval https://github.com/dialogue-evaluation/GramEval2020
- OpenCorpora http://opencorpora.org/
- BSNLP-19 http://bsnlp.cs.helsinki.fi/shared_task.html

Оценка качества

Accuracy

- Доля правильных тегов от размера корпуса
- Обычно с учетом неоднозначности

Морфологический анализ

процедура установления связей между вариантами лексической единицы и их инвариантом (парадигматическая идентификация словоформ)

Исследовать -> {исследовать} + Неопр.ф.

Исследую -> {исследовать} + Наст., Буд. вр. + Ед.ч. + 1 л.

Исследуешь -> {исследовать} + Наст., Буд. вр. + Ед.ч. + 2 л.

Исследует -> {исследовать} + Наст., Буд. вр. + Ед.ч. + 3 л.

Морфологический синтез

{исследовать} + Неопр.ф. -> исследовать

{исследовать} + Наст. вр. + Ед.ч. + 1 л. -> исследую

{исследовать} + Наст. вр. + Ед.ч. + 2 л. -> исследуешь

{исследовать} + Наст. вр. + Ед.ч. + 3 л. -> исследует

…

Лемматизация

Лемматизация (нормализация) – сведение различных словоформ к единому представлению (исходной форме или лемме)

Исследовать {исследовать}

Исследую {исследовать}

Исследуешь {исследовать}

Исследует {исследовать}

Стемминг

Стемминг – вид нормализации, при котором разные словоформы приводятся к одной основе (псевдооснове, stem).

Есть задачи, где псевдоосновы может быть достаточно (например, информационный поиск: ***фотограф****ический,* ***фотограф****ия* **–** в выдаче все документы)

NB! Полнота vs. точность

POS tagging

- Во многих языках — POS-теги
- Morphologically rich languages — 😨

В общем – задача *sequence tagging* (разметка последовательности)

> статистические методы, учитывающие контекст

> нейронные модели *sequence-to-sequence*

Ещё одна такая задача – Named Entity Recognition (NER)

Словарный метод

- со словарем словоформ (лучше, появился, когда проблема ограничения памяти была снята)
- со словарем основ (*бег-беж воз-вож-вожд*…) был нужен, когда память машин была ограничена, *стек – стек, стечь, стекло, стечь, стеклами, стеками* – минус – много шума)

Словарный метод АОТ

Грамматический словарь Зализняка

АОТ (http://aot.ru)

Формат:

- идентификатор лексемы
- идентификатор парадигмы (отсылки к таблицам с наборами правил для конкретных парадигм)

Словарный метод АОТ

Демо АОТ (Диалинг)

Input Your text:

Исследующий

○ English ◉ Russian ○ German

☑ With paradigms

Submit Request

Found	Dict ID	Lemma	Grammems	
+	пе,нс,св,	ИССЛЕДОВАТЬ	ПРИЧАСТИЕ дст,но,од,нст,мр,вн,им,ед,	

OpenCorpora

(OpenCorpora — модифицированный словарь АОТ)

Лексема состоит из всех форм слова, причем для каждой формы указана грамматическая информация (тег). Первой формой в списке идет нормальная форма слова.

ёж NOUN,anim,masc sing,nomn
ежа NOUN,anim,masc sing,gent
ежу NOUN,anim,masc sing,datv
ежа NOUN,anim,masc sing,accs
ежом NOUN,anim,masc sing,ablt
…

Предсказательные методы

«Бессловарный анализ» или «анализ по аналогии»?

- Термин «бессловарный анализ» применим в ситуации полного отсутствия словаря лексических единиц
- Термин «анализ по аналогии» описывает анализ слов, которые не вошли в существующий словарь.

КРОВАТЬ – слово с парадигмой

КРОВАТЬ, КРУЙ, КРУЙТЕ, КРУЮ .. (как ПИРОВАТЬ)

Предсказание по префиксу

Анализ новых, редких слов, имен собственных, окказионализмов (несловарных словоформ), или анализ по аналогии:

- предсказание префиксального образования
- предсказание по концовке, взятой из известных словоформ

Например: Если префикс не длиннее M символов, а правая часть (совпавшая с известной словоформой) не короче N символов, то слово разбирается по образцу известной словоформы.

[евро]технологию, [супер]коньками

Морфоанализаторы для русского языка

- Mystem (pymystem3)
- pymorphy2
- stanza
- SpaCy
- UDPipe
- natasha/slovnet

Исторические:

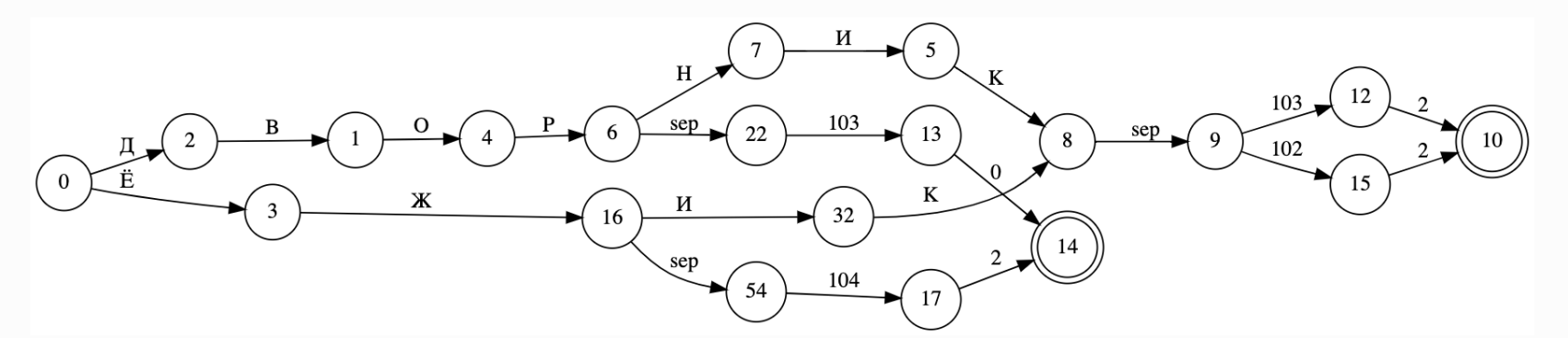
- AOT
- TreeTagger
- TnT

	news	wiki	fiction	social	poetry
slovnet	**0.961**	0.815	0.905	0.807	0.664
slovnet_bert	**0.982**	**0.884**	**0.990**	**0.890**	**0.856**
deeppavlov	0.940	0.841	0.944	0.870	**0.857**
deeppavlov_bert	0.951	**0.868**	**0.964**	**0.892**	**0.865**
udpipe	0.918	0.811	**0.957**	0.870	0.776
spacy	**0.964**	**0.849**	0.942	0.857	0.784
stanza	0.934	0.831	0.940	**0.873**	0.825
rnnmorph	0.896	0.812	0.890	0.860	0.838
maru	0.894	0.808	0.887	0.861	0.840
rupostagger	0.673	0.645	0.661	0.641	0.636

pymorphy

https://pymorphy2.readthedocs.io/en/latest/index.html

- Словарь OpenCorpora в сжатом виде (DAWG)
- лемматизация, морф. разметка
- генерация форм (inflection)



https://gist.github.com/named-entity/ce4d121512626568ca3059170333750c

Неоднозначность

- морфологическая / грамматическая неоднозначность / омонимия (ambiguity)
- разрешение / снятие неоднозначности (disambiguation)
- POS disambiguation / lemma disambiguation …

Классический пример:

Эти типы стали есть в цехе.

Уровни морф. неоднозначности

- неоднозначность лемм
  - леммы имеют разный POS-тег
    стали → сталь NOUN / стать VERB
  - леммы имеют один POS-тег и совпадающие формы, но разную начальную
    графине → графин NOUN masc / графиня NOUN femn
  - супплетивные формы
    дети → дитя / ребёнок
- неоднозначность форм одной лексемы
  красного → красный ADJ masc / neut gent
- транспозиция (conversion / zero derivation)
  В палату привезли больного.

Разрешение неоднозначности

- Методы, основанные на контекстных правилах, составляемых экспертами-лингвистами
- Методы, основанные на контекстных правилах, выводимых из текстов (обучение на размеченных данных и обучение без учителя)
- Методы, основанные на вероятностных моделях (обучение на размеченных данных и обучение без учителя)
- Методы, основанные на нейронных сетях
- Гибридные методы

Rule-based методы

- Для английского языка - грамматика ограничений (*constraint grammar*, F. Carlsson & A. Voutilainen 1995) включает правила типа «выполни действие X над объектом Y в контексте Z»
- В первой версии - 1200 правил, основанных на грамматике, и 200 эвристических правил, потом расширение до **3600** правил
- Контекстные правила могут быть закодированы в виде *конечных преобразователей* (*finite-state transducers*)

Rule-based методы

Пример правила для английского языка:

tag:red ‘VB’ <- tag: ‘DT’@[-1] ○

«исключить тег VB, если сосед на расстоянии ‘-1’ (т.е. непосредств. сосед слева) имеет тег DT»

the / {DT} light / {JJ, NN, VB}

превращается в

the / {DT} light / {JJ, NN}

+/- rule-based методов

Плюсы

- Не нужны обучающие данные, но нужен хорошо размеченный корпус
- Результаты не ухудшаются из-за расширения множества тегов
- Используются независимые друг от друга правила (или группы правил)

Минусы

- Жёсткая система правил
- Низкая полнота
- Много ручной работы
- Набор правил нельзя/сложно адаптировать к другим языка

Brill tagger

Автоматическое построение правил по корпусу

> обучение на размеченном корпусе — Brill 1992-1994

> обучение без учителя (unsupervised) — Brill 1995

Требования:

- Словарь / обучающий корпус
- Шаблоны правил

Идея Brill tagger

- Каждому слову в обучающей выборке присваиваем самый частотный тег для этого слова
- Сравниваем с эталонной разметкой и формулируем правила **изменения** приписанного тега (transformation)
- Повторяем несколько итераций, пока не будет достигнут запланированный эффект:
  - полное отсутствие улучшений
  - заданный уровень точности
  - заданное максимальное число правил

Unsupervised Brill tagger

(Brill 1995)

- Корпус текстов без предварительной разметки и словарь
- Предварительная разметка текста по словарю с указанием всех вариантов

The	can	will	rust
DT	MD **NN** VB	**MD** NN VB	NN **VB**

Правила в Brill tagger

Общий вид правил:

«Заменить тег X на тег Y в контексте C, где X является последовательностью из двух или более тегов, а Y – один тег, такой что $Y \in X$».

Пример построения правила

Строим частотную модель для шаблонов правил:

После слова *the* среди однозначной разметки чаще всего встречаются слова с тегом NN. Можем сформулировать следующее правило:

- Заменять тег MD_NN_VB на NN после слова *the*

The	can	will	rust
DT	MD **NN** VB	**MD** NN VB	NN **VB**

Вероятностные методы

- Скрытые марковские модели (Hidden Markov Model, HMM)

  вычисление параметров:

  - Алгоритм Витерби (Viterbi)
  - Алгоритм Баума-Уэлча (Baum – Welch)

- Нейросетевые модели

Sequence labelling task

- Задача разметки последовательности

1 токен → 1 тег

Предложение длины N → последовательность тегов длины N

- Probabilistic sequence model: строим распределение вероятностей на возможных последовательностях тегов, выбираем наиболее вероятную

Как использовать частоты?

- Простейший вариант – присваивать каждой словоформе наиболее вероятную морфологическую интерпретацию – вспомним 1-gram LM
- За вероятности принимаются нормализованные частоты присвоения тега определенной форме в размеченном корпусе:

$$P(t|w) = \frac{count(w,t)}{|C|}$$

Марковская цепь

- Множество возможных состояний / states: $Q = q_1 \dots q_N$
- Матрица вероятностей переходов из состояния i в j / transition probability matrix: $A = a_{11} \dots a_{ij} \dots a_{NN}$
- Исходное распределение вероятностей состояний: $\pi = \pi_1 \dots \pi_N$

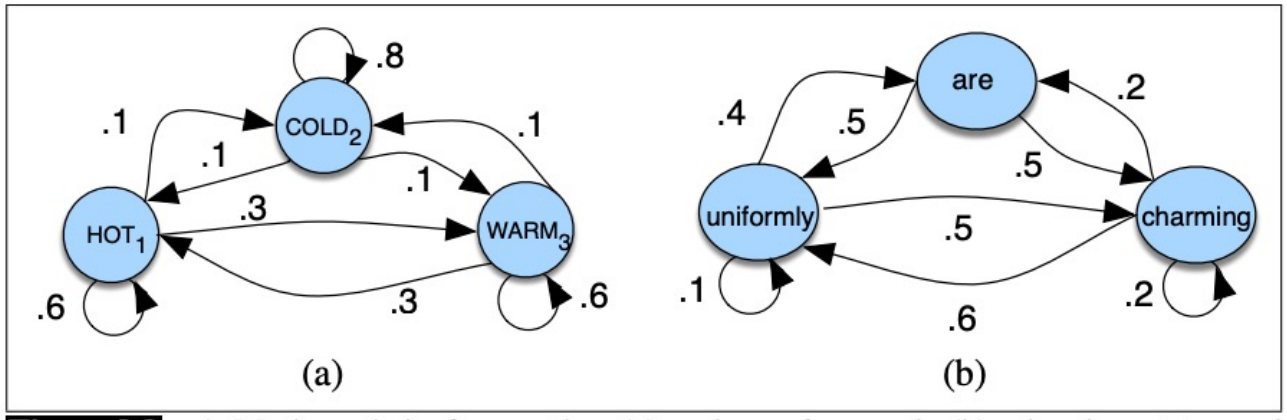


Figure 8.8 A Markov chain for weather (a) and one for words (b), showing states and transitions. A start distribution π is required; setting $\pi = [0.1, 0.7, 0.2]$ for (a) would mean a probability 0.7 of starting in state 2 (cold), probability 0.1 of starting in state 1 (hot), etc.

Скрытая Марковская модель

- Hidden Markov Model (HMM)
- Множество возможных состояний / states: $Q = q_1 \dots q_N$
- Матрица вероятностей переходов из состояния i в j / transition probability matrix: $A = a_{11} \dots a_{ij} \dots a_{NN}$
- Последовательность наблюдений / observations: $O = o_1 \dots o_T$
- Последовательность вероятностей наблюдений / emission probabilities: $B = b_i(o_t)$
- Исходное распределение вероятностей состояний: $\pi = \pi_1 \dots \pi_N$

Markov assumption

Предполагаем *марковское свойство / Markov assumption* (как в n-граммных языковых моделях):

- встречаемость каждого тега в определенном месте цепочки зависит только от предыдущего тега
  $P(q_i|q_1 \dots q_{i-1}) = P(q_i|q_{i-1})$
- то, какое слово находится в том или ином месте цепочки, полностью определяется тегом (а не, допустим, соседними словами)
  $P(o_i|q_1 \dots q_T, o_1 \dots o_T) = P(o_i|q_i)$

> *марковская модель 1-го порядка*

HMM tagger

Вероятности перехода A:

$$a_{i,i-1} = P(t_i|t_{i-1}) = \frac{count(t_{i-1}, t_i)}{count(t_{i-1})}$$

Вероятности наблюдений B:

$$b_i(w_i) = P(w_i|t_i) = \frac{count(t_i, w_i)}{count(t_i)}$$

- Как здесь можно использовать готовый морфологический словарь?

Вероятности на примере

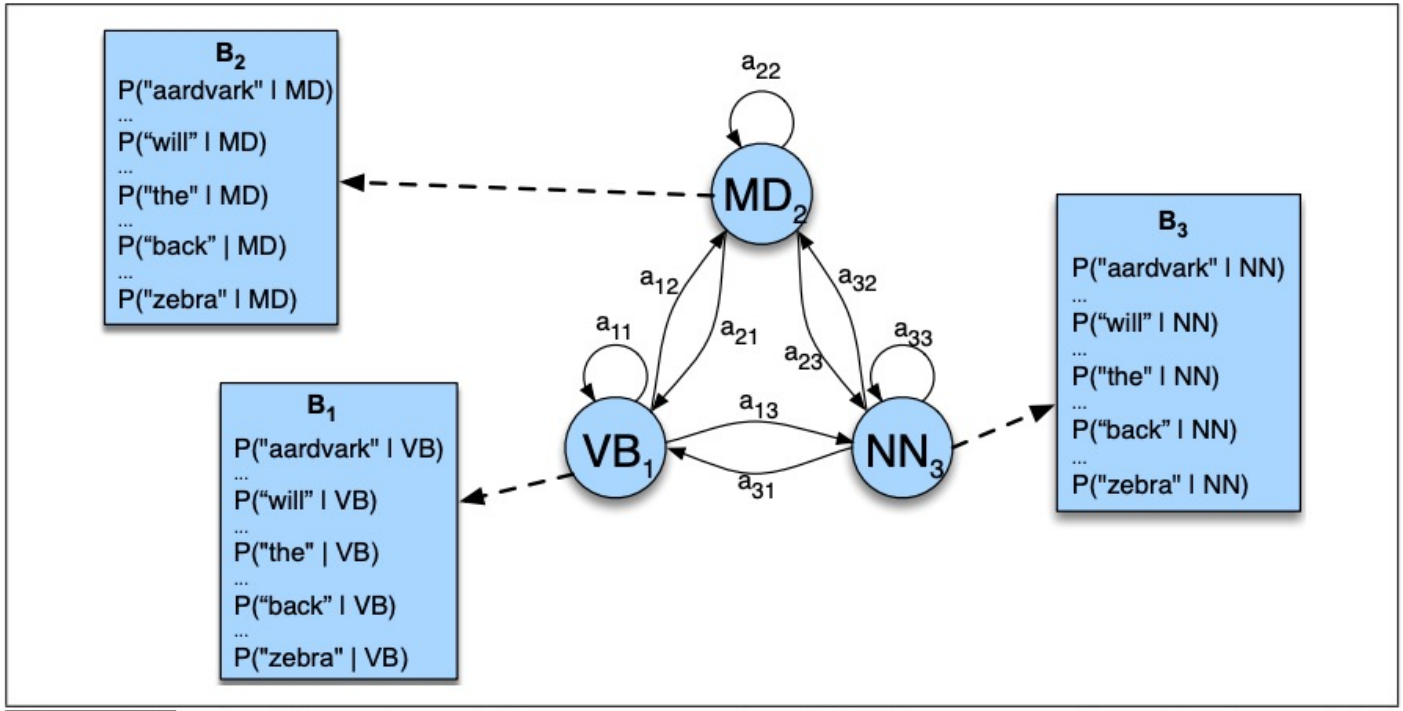


Figure 8.9 An illustration of the two parts of an HMM representation: the *A* transition probabilities used to compute the prior probability, and the *B* observation likelihoods that are associated with each state, one likelihood for each possible observation word.

HMM decoding

Decoding – определение последовательности скрытых состояний, соответствующее наблюдениям:

$$\hat{t}_{1:n} = arg \max_{t_1 \dots t_n} P(t_1 \dots t_n | w_1 \dots w_n)$$

$$= arg \max_{t_1 \dots t_n} \frac{P(w_1 \dots w_n | t_1 \dots t_n)\, P(t_1 \dots t_n)}{P(w_1 \dots w_n)}$$

$$= arg \max_{t_1 \dots t_n} P(w_1 \dots w_n | t_1 \dots t_n) P(t_1 \dots t_n)$$

$$= \arg \max_{t_1 \dots t_n} \prod_{i=1}^{n} P(w_i | t_i) P(t_i | t_{i-1})$$

Алгоритм Витерби

Позволяет сократить количество вычислений:

1. Инициализация:

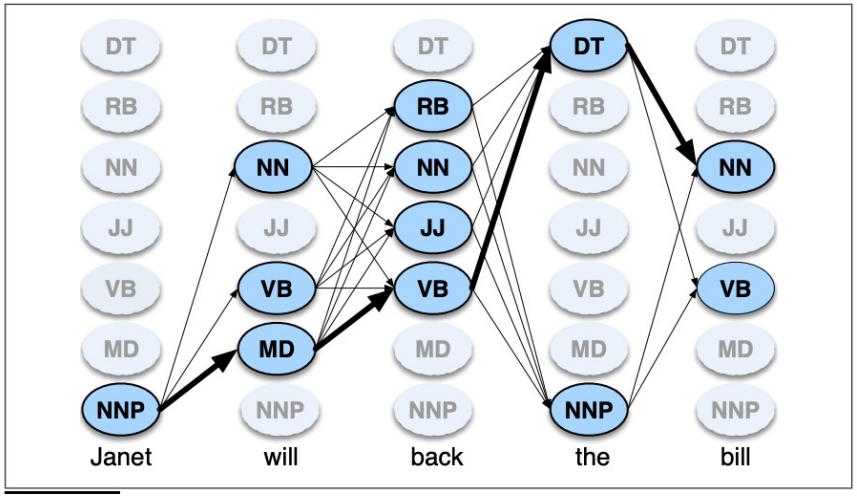
   для первого токена используем только вероятность наблюдения и π

2. Рекурсия:

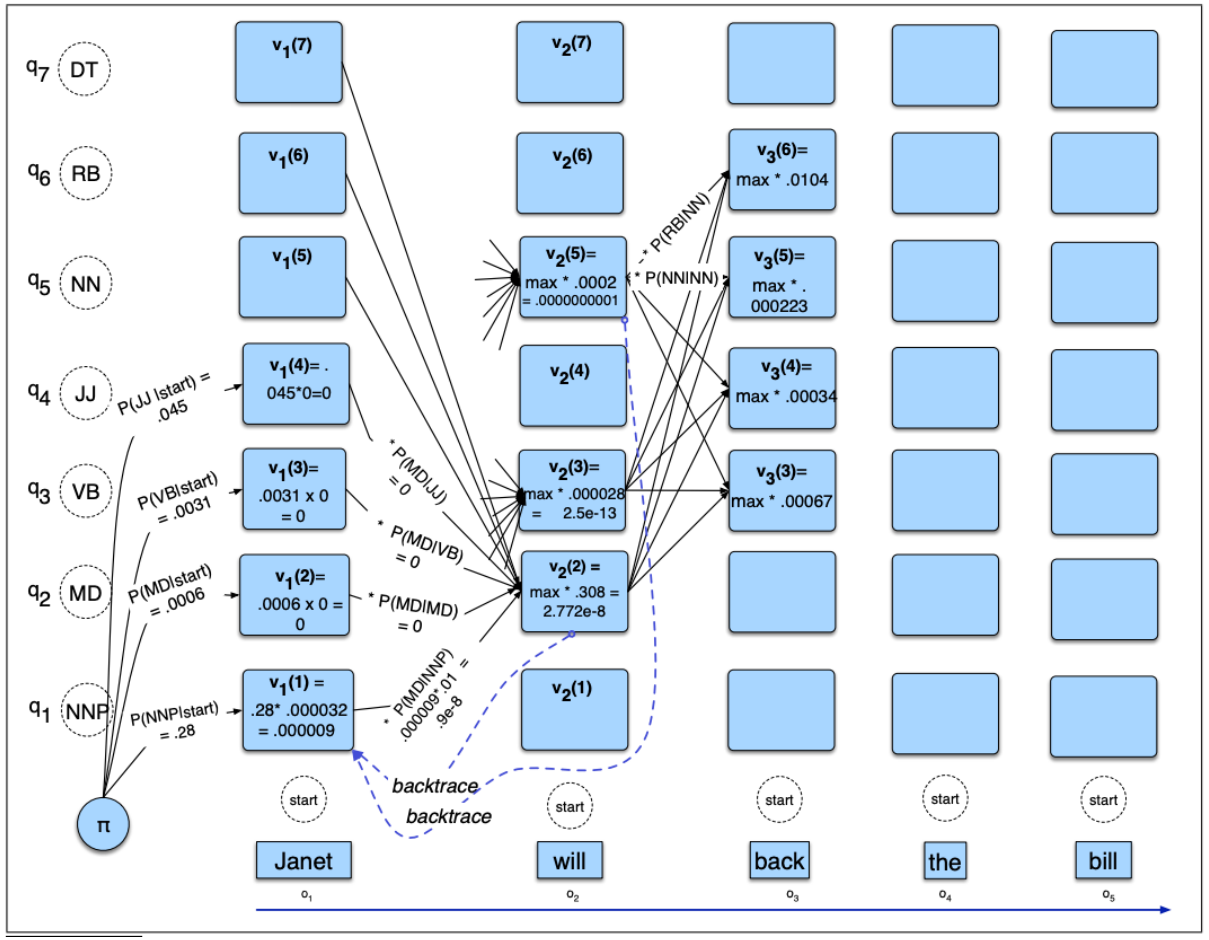
   в момент t выбираем наиболее вероятный путь к текущему состоянию

   $$v_t(j) = \max_{i=1\ldots N} v_{t-1}(i) a_{ij} b_j(o_t)$$

   сохраняем лучший тег (backpointer)



Алгоритм Витерби



Другие модели теггинга

- 3gram HMM (2-order assumption)
- maximum entropy Markov model – MEMM tagger
- Conditional Random Fields – CRF taggers
- Recurrent neural network – RNN taggers
- BiLSTM taggers
- …

Оценка качества теггинга

С учётом разрешения неоднозначности:

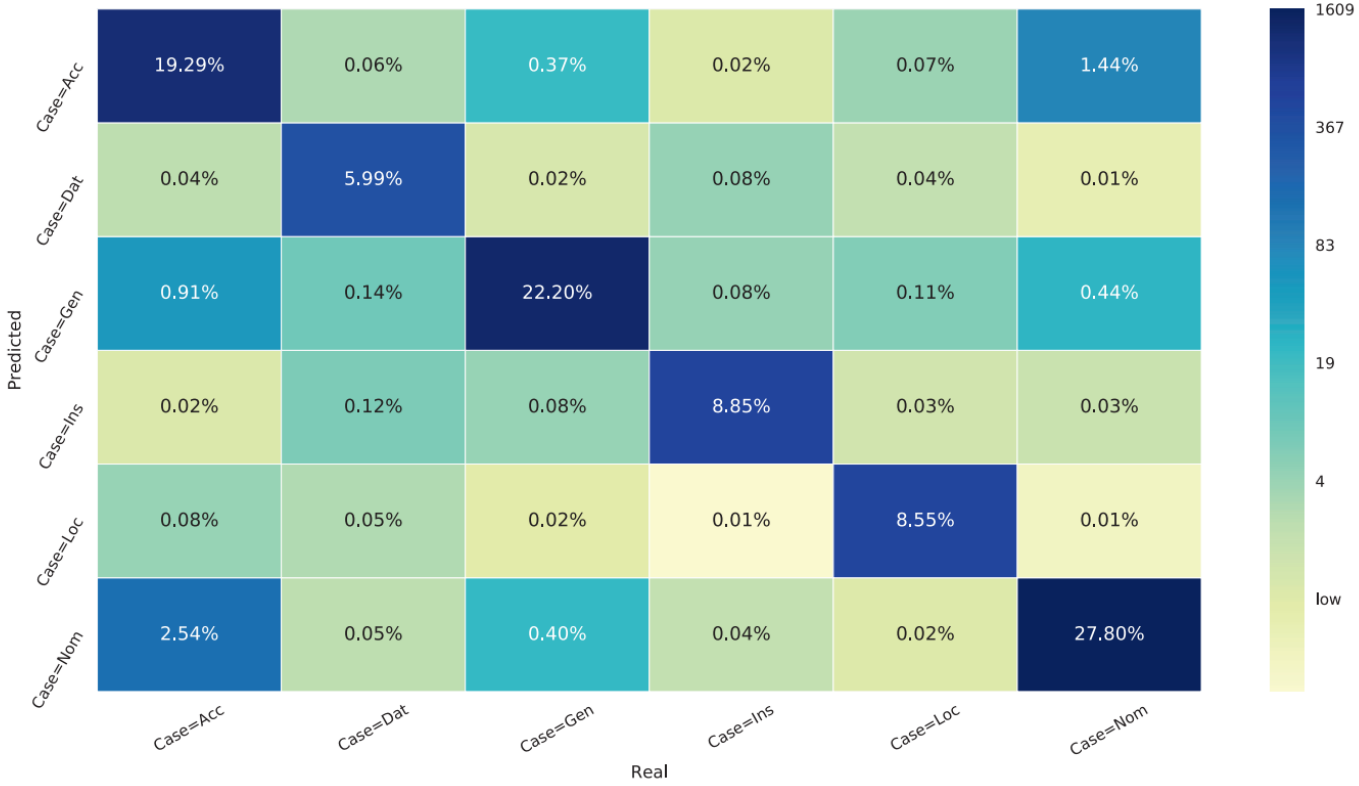
- Accuracy на уровне POS
- Accuracy на уровне полного тега
- Accuracy per tag / class
- Accuracy per sentence

Полезно помнить:

- согласованность разметки (Penn Treebank) – 97% (*human ceiling*)
- unigram baseline – см. слайд 11

Error Analysis

Confusion matrix / contingency table



Пример

https://colab.research.google.com/drive/1wMkEJTvXwrWQg9D6ekqV2iNWEQMD_EiH?usp=sharing

Спасибо!

Вопросы?