

Математические методы анализа текстов

Чат-боты и вопросно-ответные системы

Дербаносов Роман

28 ноября

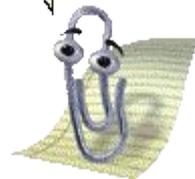
План

1. Типы диалоговых систем
2. Chit-chat
3. Task-oriented

It looks like you're writing a letter.

Would you like help?

- Get help with writing the letter
- Just type the letter without help
- Don't show me this tip again



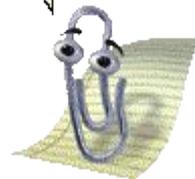
План

1. **Типы диалоговых систем**
2. Chit-chat
3. Task-oriented

It looks like you're writing a letter.

Would you like help?

- Get help with writing the letter
- Just type the letter without help
- Don't show me this tip again



Типы диалоговых систем

Chit-chat

Алиса, давай поболтаем

Задавайте тему.

Ты смотрела Минутмэн?

Нет, но собираюсь посмотреть

Но я хочу посмотреть его

Смотрите, не пожалеете

Какой твой любимый фильм?

Например, "Звездные войны" предпочитаю я.

А ты смотрела Звёздные Войны?

Не хочу говорить об этом.

Почему? Ты боишься?

Я боюсь только отключения питания вашего устройства.

Task-oriented

Я могу помочь заказать еду в ресторанах сети «Папа Джонс». Для начала, мне нужно узнать, в каком городе вы находитесь.

Москва

Отлично. Вы можете сразу надиктовать свой заказ или изучить меню. Также я могу рассказать о специальных акциях. Итак, что бы вы хотели заказать?

Подсказка: вы можете сказать "Супер Папа с сырным бортом и большая бутылка пепси".

Десерты

Какой десерт вы бы хотели сегодня?



Брауни шоколадный

349 руб.



Чизкейк "Нью-Йорк"

239 руб.



Маффин Шоколадный

129 руб.

Чизкейк "Нью-Йорк"

Вы выбрали: Чизкейк "Нью-Йорк" за 239 руб.
Уточните, пожалуйста, количество.

1

Добавила в корзину.
Нужно добавить что-то еще к заказу?

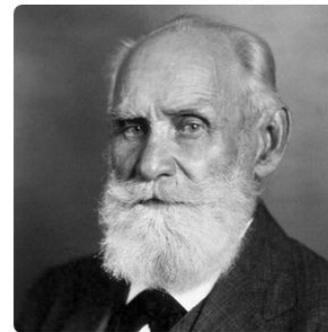
Нет

Вы заказали:
1. Чизкейк "Нью-Йорк" - 239 руб.
Кол-во: 1.
Общая сумма заказа: 239 руб.
Всё верно?

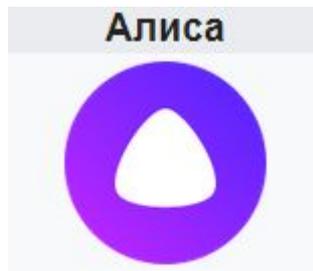
Диалоговые системы



iPavlov.ai



Google Home



DeepPavlov

HelloBot in 7 lines in DeepPavlov

- Import core components of the dialogue Agent

```
from deeppavlov.core.agent import Agent, HighestConfidenceSelector
from deeppavlov.skills.pattern_matching_skill import PatternMatchingSkill
```

- Define responses and input patterns for skills

```
hello = PatternMatchingSkill(['Hello world!'], patterns=["hi", "hello", "good day"])
bye = PatternMatchingSkill(['Goodbye world!', 'See you around'],
                           patterns=["bye", "chao", "see you"])
fallback = PatternMatchingSkill(["I don't understand, sorry", 'I can say "Hello world!"]])
```

- Combine Skills with SkillManager (selector) into an Agent

```
HelloBot = Agent([hello, bye, fallback], skills_selector=HighestConfidenceSelector())
```

- Talk with HelloBot!

```
HelloBot(['Hello', 'Bye', 'Or not'])
```

```
['Hello world!', 'See you around', 'I can say "Hello world!"]])
```

NER

Вскоре группу хакеров, в которой состоял **Митник**, сдала властям девушка одного из участников. **Кевина** приговорили к трём месяцам в исправительной колонии для несовершеннолетних и году условно. Но **Митник** вскоре совершил хитроумную атаку на компьютерную сеть **Пентагона**. Вначале им был взломан один из компьютеров одного **Лос-Анджелесского университета**, а уже якобы с него был атакован **Пентагон**. Но, несмотря на такую уловку, **Митника** поймали и приговорили уже к шести месяцам тюремного заключения. Однако и в тюрьме **Кевин** не терял времени зря. К моменту освобождения он имел все знания по телефонным сетям, которыми располагали лучшие специалисты **Bell Labs**. **Митник** стал виртуозом телефонного взлома, он мог создавать незарегистрированные телефонные номера, звонить с чужого номера, прослушивать и разъединять разговоры. В среде хакеров **Митника** называли **Кондором**; это прозвище **Кевину** досталось от главного персонажа одного американского фильма, где тот скрывался от **ЦРУ**, используя свои умения в работе с телефонной сетью. На протяжении 1980-х годов **Митник** совершенствовал свои способности. Поселившись в **Калифорнии**, он проделывал различные компьютерные и телефонные «шутки». Умело заматывая следы своих деяний, ему удавалось долго укрываться от ответственности, но в декабре 1987 года **Митнику** снова не повезло — его выдал полиции **Ленни Ди Чикко**, друг и

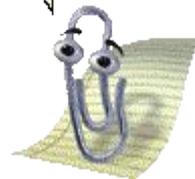
План

1. Типы диалоговых систем
2. **Chit-chat**
3. Task-oriented

It looks like you're writing a letter.

Would you like help?

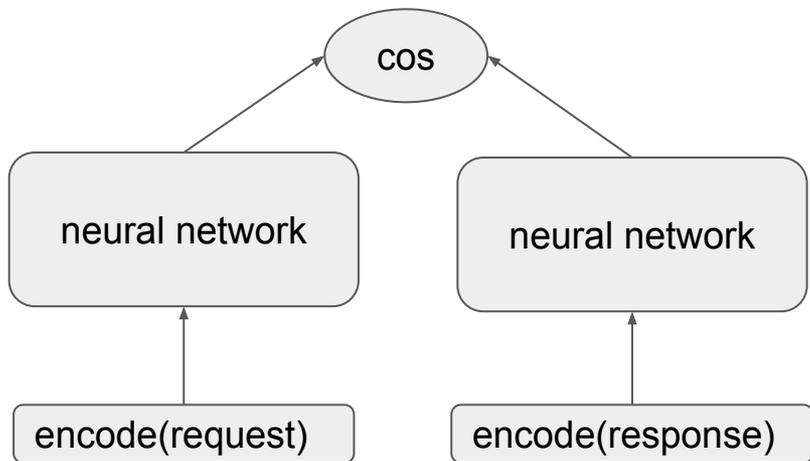
- Get help with writing the letter
- Just type the letter without help
- Don't show me this tip again



Chit-chat: архитектуры

Retrieval models

- Ответ ищется в базе (например, с помощью DSSM)

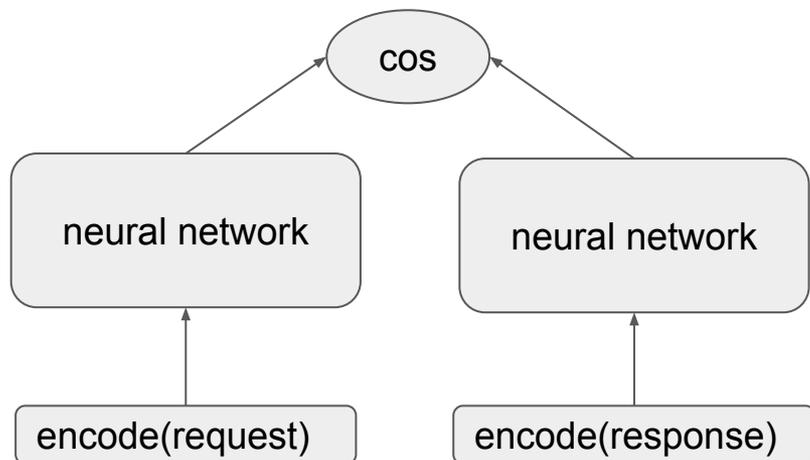


Статья про DSSM: [ссылка](#)

Chit-chat: архитектуры

Retrieval models

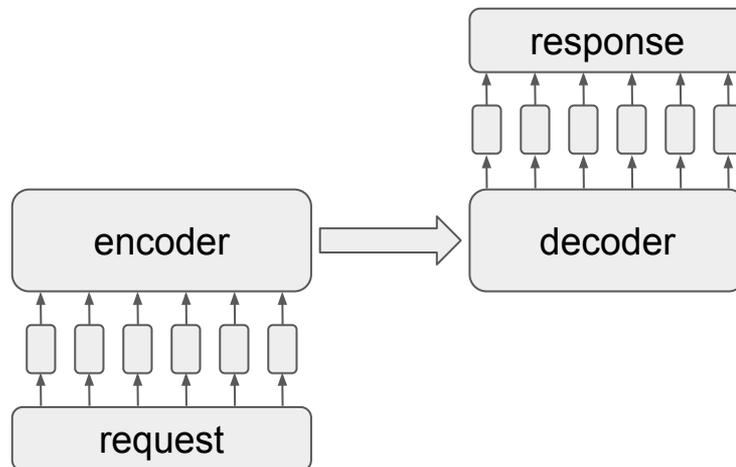
- Ответ ищется в базе (например, с помощью DSSM)



Статья про DSSM: [ссылка](#)

Generative models

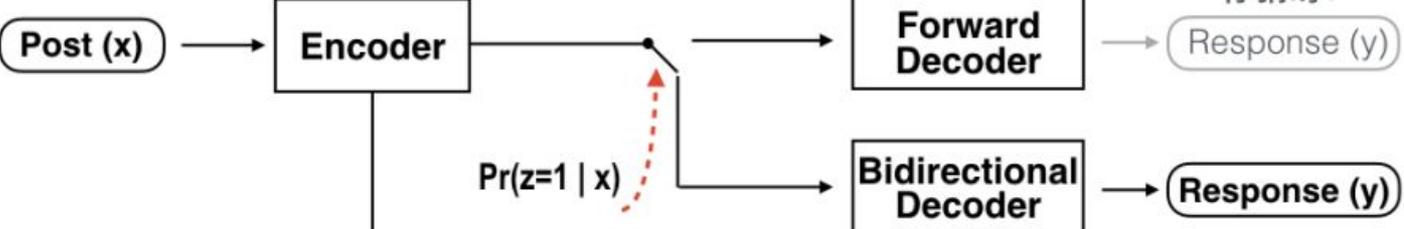
- Ответ генерируется моделью (например, encoder-decoder архитектурой)
- Проблема: может генерироваться бессвязный текст



Chit-chat: личность чат-бота

(What's your speciality?)

你有什么特长?



(Have a guess!)

你猜呀?

Response (y)

Response (y)

- (Name: Wang Zai)
- (Age: 3)
- (Gender: Boy)
- (Hobby: Cartoon)
- (Speciality: Piano)

Key(k)	Value(v)
姓名	汪仔
年龄	三岁
性别	男孩
爱好	动漫
特长	钢琴

Agent Profile

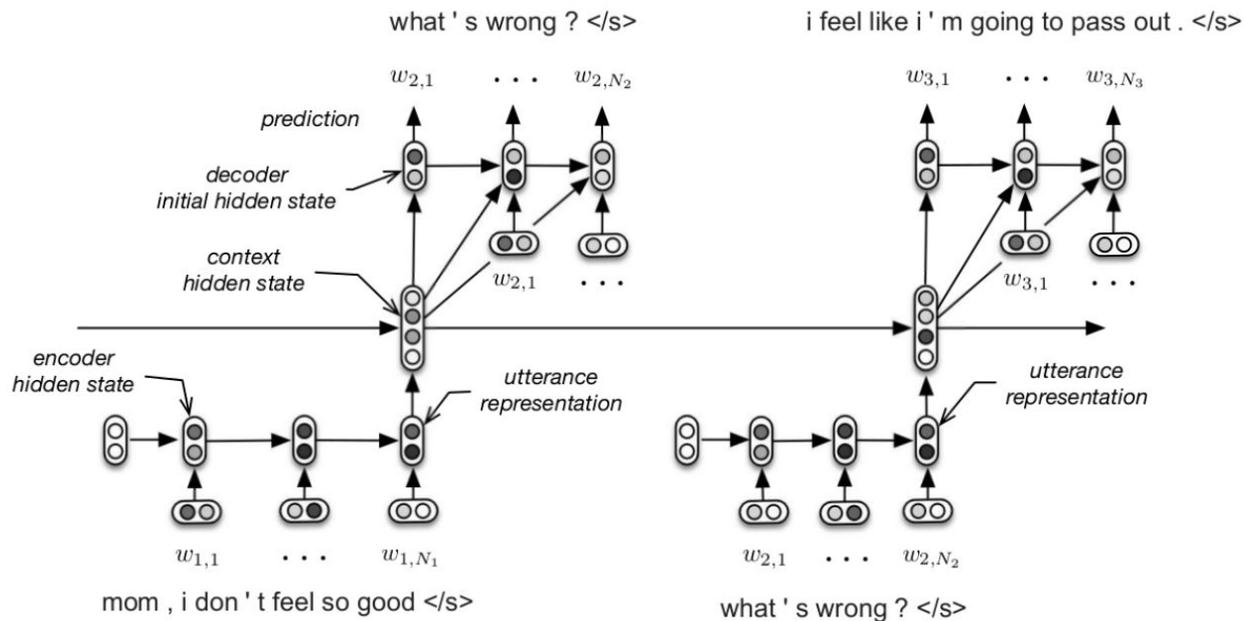


钢琴 (Piano)

←.....●.....→
 我擅长弹钢琴!
 (I'm good at playing the piano!)

- profile value
- ←... backward decoding
- ...→ forward decoding

Chit-chat: Hierarchical Neural Network Models



Metrics: perplexity = $\exp\left(-\frac{1}{N_W} \sum_{n=1}^N \log P_{\theta}(U_1^n, U_2^n, U_3^n)\right)$, word error-rate. Dataset: MovieTriples.

Источник: <https://arxiv.org/pdf/1507.04808.pdf>

Chit-chat: метрики качества модели

Chit-chat: метрики качества модели

Metric	Twitter				Ubuntu			
	Spearman	p-value	Pearson	p-value	Spearman	p-value	Pearson	p-value
Greedy	0.2119	0.034	0.1994	0.047	0.05276	0.6	0.02049	0.84
Average	0.2259	0.024	0.1971	0.049	-0.1387	0.17	-0.1631	0.10
Extrema	0.2103	0.036	0.1842	0.067	0.09243	0.36	-0.002903	0.98
METEOR	0.1887	0.06	0.1927	0.055	0.06314	0.53	0.1419	0.16
BLEU-1	0.1665	0.098	0.1288	0.2	-0.02552	0.8	0.01929	0.85
BLEU-2	0.3576	< 0.01	0.3874	< 0.01	0.03819	0.71	0.0586	0.56
BLEU-3	0.3423	< 0.01	0.1443	0.15	0.0878	0.38	0.1116	0.27
BLEU-4	0.3417	< 0.01	0.1392	0.17	0.1218	0.23	0.1132	0.26
ROUGE	0.1235	0.22	0.09714	0.34	0.05405	0.5933	0.06401	0.53
Human	0.9476	< 0.01	1.0	0.0	0.9550	< 0.01	1.0	0.0

- 25 ассессоров
- Оценивалась адекватность ответа по пятибалльной шкале
- Стандартные метрики чат-бота: 'adequacy', 'fluency' and 'informativeness'

Вывод: ассессорскую оценку "уместности ответа в контексте" сложно заменить на автоматически вычисляемую метрику.

Chit-chat: хаки

- Полезно учитывать контекст диалога и иметь заранее заготовленную личность чат-бота
- DSSM: hard-negative samples
- AI safety: “Я не хочу говорить об этом” и прочая нецензурная лексика

Chit-chat: хаки: разнообразие ответов

- MMI: $\text{score}(\text{Resp}) \sim \log P(\text{Resp} | \text{Req}) - k * \log P(\text{Resp}) \sim (1 - k) * \log P(\text{Resp} | \text{Req}) + k * \log P(\text{Req} | \text{Resp})$ ([статья](#))

Chit-chat: хаки: разнообразие ответов

- MMI: $\text{score}(\text{Resp}) \sim \log P(\text{Resp} | \text{Req}) - k * \log P(\text{Resp}) \sim (1 - k) * \log P(\text{Resp} | \text{Req}) + k * \log P(\text{Req} | \text{Resp})$ ([статья](#))
- Diverse beam search ([статья](#))

Algorithm 1: Diverse Beam Search

- 1 Perform a diverse beam search with G groups using a beam width of B
 - 2 **for** $t = 1, \dots, T$ **do**
 - 3 // perform one step of beam search for first group without diversity
4 $Y_{[t]}^1 \leftarrow \operatorname{argmax}_{(\mathbf{y}_{1,[t]}^1, \dots, \mathbf{y}_{B',[t]}^1)} \sum_{b \in [B']} \Theta(\mathbf{y}_{b,[t]}^1)$
 - 5 **for** $g = 2, \dots, G$ **do**
 - 6 // augment log-probabilities with diversity penalty
7 $\Theta(\mathbf{y}_{b,[t]}^g) \leftarrow \Theta(\mathbf{y}_{b,[t]}^g) + \sum_h \lambda_g \Delta(\mathbf{y}_{b,[t]}^g, Y_{[t]}^h) \quad b \in [B'], \mathbf{y}_{b,[t]}^g \in \mathcal{Y}^g \text{ and } \lambda_g > 0$
 - 8 // perform one step of beam search for the group
9 $Y_{[t]}^g \leftarrow \operatorname{argmax}_{(\mathbf{y}_{1,[t]}^g, \dots, \mathbf{y}_{B',[t]}^g)} \sum_{b \in [B']} \Theta(\mathbf{y}_{b,[t]}^g)$
 - 7 Return set of B solutions, $Y_{[T]} = \bigcup_{g=1}^G Y_{[T]}^g$
-

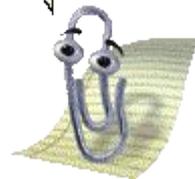
План

1. Типы диалоговых систем
2. Chit-chat
3. **Task-oriented**

It looks like you're writing a letter.

Would you like help?

- Get help with writing the letter
- Just type the letter without help
- Don't show me this tip again



Task-oriented



Natural language understanding

Domain and intent detection

Алиса, какие у тебя навыки?

Вот что я могу:



Какой сегодня день?

Для путешественников во времени.



Сколько стоит умный телевизор

Посмотрю цены на Маркете и назову среднюю.



Включи сказку про зайца

Или про репку. В общем, про что хотите.



Алиса, где я?

Сообщу ваше местоположение.



Давай поболтаем, Алиса

Помогу скоротать час-другой.

Slot filling

Город: Пекин
Дата: 01.12.2018

Алиса, какая в Пекине температура 1 декабря?

1 декабря в Пекине

+8°

облачно с прояснениями от -3 до +8

Утро

+1

День

+6

Вечер

-1

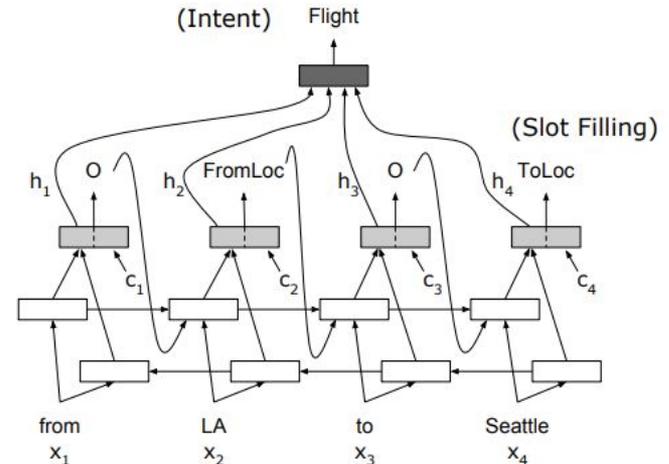
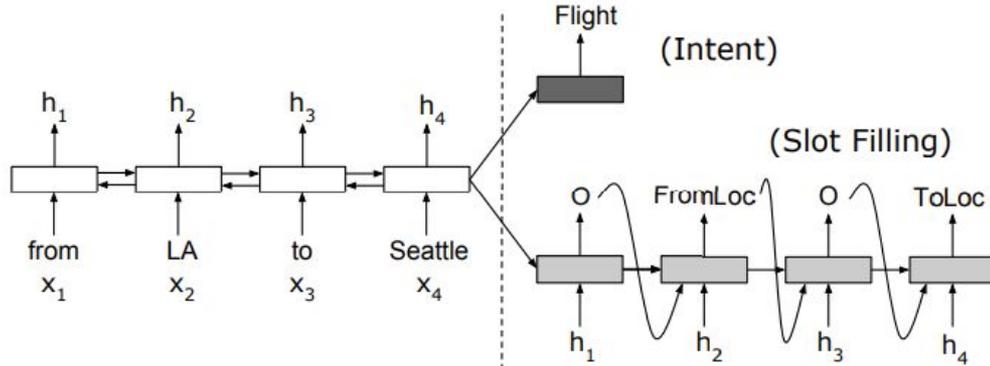
Domain and intent detection

- Нужно определить области (domains) и набор возможных намерений пользователя (intents). С помощью экспертов или ML, например, с помощью тематического моделирования.
- Стандартная задача классификации текстов.
- Алиса: kNN с предобученной на большом датасете метрикой семантической близости. Плюсы: быстрое переобучение при добавлении новых фраз (для рекуррентных моделей - несколько часов). ([ссылка](#))

Slot filling

- Стандартная задача тегирования с Beginning-Inside-Outside-нотацией (BIO):
 - CRF, RNN
 - Регулярные выражения
- Алиса: biLSTM + attention ([ссылка](#))
- Можно обучать intent detection + slot filling вместе ([статья](#))

○	○	B-fromloc. city_name	○	B-toloc. city_name	○	○	B-depart_time. time_relative	B-depart_time. period_of_day
flight	from	cleveland	to	dallas	that	leaves	before	noon



Dialogue management

Реализация:

- Нейросетевые модели
- Закодированные правила

Некоторые проблемы:

- Эллипсис (пропуск слов):
 - Какая сегодня погода в Сочи? А завтра?
- Анафоры (отсылки к ранее упомянутым сущностям):
 - Есть ли билеты в Сочи на завтра? Есть ли билеты туда на послезавтра?
- Вложенные диалоги и отложенные ответы:
 - - Что ты делаешь завтра? - А что? - Хочу позвать тебя в театр. - После обеда - свободна.

Конфиг диалогового менеджера Алисы

- Только правила и конфиги. Предсказуемо и легко изменить.
- Вызов бинарного классификатора с выходом в поиск или болталку при неопределении intent-а.

```
1  {% nlginclude "common/error.nlg" %}
2  {% nlginclude "common/suggests.nlg" %}
3  {% nlginclude "scenarios/intents/cards/call.nlg" %}
4
5  {% phrase render_result %}
6    {% if form.recipient_info %}
7      {% if context.attention and context.attention.attention_type == "calls_not_supported_on_device" %}
8        {% chooseline %}
9          Телефон {{ form.recipient_info.title | inflect('gen') }} – {{ form.recipient_info.phone }}, звоните скорее со с
10         {{ form.recipient_info.title | capitalize_first }} – номер {{ form.recipient_info.phone }}, наберите на телефон
11         {{ form.recipient_info.phone }} – номер {{ form.recipient_info.title | inflect('gen') }}, наберите скорее со св
12         Я волнуюсь за вас, {{ form.recipient_info.title }} – номер {{ form.recipient_info.phone }}, звоните скорее с те
13         Звоните с телефона в {{ form.recipient_info.title | inflect('acc') }} по номеру {{ form.recipient_info.phone }}
14       {% endchooseline %}
15     {% else %}
16       {% chooseline %}
17         Вызываю {{ form.recipient_info.title | inflect('acc') }}...
18         Звоню в {{ form.recipient_info.title | inflect('acc') }}...
19       {% endchooseline %}
20     {% endif %}
21   {% else %}
22     {% chooseline %}
23     Пока что я умею звонить только в экстренные службы.
24   {% endchooseline %}
25   {% endif %}
26   {% endphrase %}
```

Источник: <https://events.yandex.ru/lib/talks/5444/>

Natural language generation

NLG(action='purchase inform', item='Чизкейк
"Нью-Йорк"', price='239 руб')



Вы выбрали: Чизкейк
"Нью-Йорк" за 239 руб.

- Правила
- Модели машинного обучения,
генерирующие последовательности

Task-oriented



Towards AI-complete Question Answering

- BaBI dataset: 20 типов задач.

Task 1: Single Supporting Fact

Mary went to the bathroom.
John moved to the hallway.
Mary travelled to the office.
Where is Mary? **A:office**

Task 2: Two Supporting Facts

John is in the playground.
John picked up the football.
Bob went to the kitchen.
Where is the football? **A:playground**

Task 3: Three Supporting Facts

John picked up the apple.
John went to the office.
John went to the kitchen.
John dropped the apple.
Where was the apple before the kitchen? **A:office**

Task 4: Two Argument Relations

The office is north of the bedroom.
The bedroom is north of the bathroom.
The kitchen is west of the garden.
What is north of the bedroom? **A: office**
What is the bedroom north of? **A: bathroom**

Task 5: Three Argument Relations

Mary gave the cake to Fred.
Fred gave the cake to Bill.
Jeff was given the milk by Bill.
Who gave the cake to Fred? **A: Mary**
Who did Fred give the cake to? **A: Bill**

Task 6: Yes/No Questions

John moved to the playground.
Daniel went to the bathroom.
John went back to the hallway.
Is John in the playground? **A:no**
Is Daniel in the bathroom? **A:yes**

Towards AI-complete Question Answering

- 1000 заданий для обучения, 1000 заданий для тестирования

TASK	Weakly Supervised		Uses External Resources	Strong Supervision (using supporting facts)						
	<i>N</i> -gram Classifier	LSTM	Structured SVM <small>CORF + SRL features</small>	MemNN <small>Weissen et al. (2014)</small>	MemNN <small>ADAMITE-MEMORY</small>	MemNN <small>AM + N-GRAMS</small>	MemNN <small>AM + NONLINEAR</small>	MemNN <small>AM + NG + NE</small>	No. of ex. resp. ≥ 95	Multitask Training
1 - Single Supporting Fact	36	50	99	100	100	100	100	100	250 ex.	100
2 - Two Supporting Facts	2	20	74	100	100	100	100	100	500 ex.	100
3 - Three Supporting Facts	7	20	17	20	100	99	100	100	500 ex.	98
4 - Two Arg. Relations	50	61	98	71	69	100	73	100	500 ex.	80
5 - Three Arg. Relations	20	70	83	83	83	86	86	98	1000 ex.	99
6 - Yes/No Questions	49	48	99	47	52	53	100	100	500 ex.	100
7 - Counting	52	49	69	68	78	86	83	85	FAIL	86
8 - Lists/Sets	40	45	70	77	90	88	94	91	FAIL	93
9 - Simple Negation	62	64	100	65	71	63	100	100	500 ex.	100
10 - Indefinite Knowledge	45	44	99	59	57	54	97	98	1000 ex.	98
11 - Basic Coreference	29	72	100	100	100	100	100	100	250 ex.	100
12 - Conjunction	9	74	96	100	100	100	100	100	250 ex.	100
13 - Compound Coref.	26	94	99	100	100	100	100	100	250 ex.	100
14 - Time Reasoning	19	27	99	99	100	99	100	99	500 ex.	99
15 - Basic Deduction	20	21	96	74	73	100	77	100	100 ex.	100
16 - Basic Induction	43	23	24	27	100	100	100	100	100 ex.	94
17 - Positional Reasoning	46	51	61	54	46	49	57	65	FAIL	72
18 - Size Reasoning	52	52	62	57	50	74	54	95	1000 ex.	93
19 - Path Finding	0	8	49	0	9	3	15	36	FAIL	19
20 - Agent's Motivations	76	91	95	100	100	100	100	100	250 ex.	100
Mean Performance	34	49	79	75	79	83	87	93		92

ИСТОЧНИК: <https://arxiv.org/pdf/1502.05698.pdf>

Материалы

Слайды про диалоговые системы от [Yun-Nung \(Vivian\) Chen](#), [Asli Celikyilmaz](#),
Dilek Hakkani-Tur: [ссылка](#)

Про Алису: [ссылка](#), [ссылка](#), [ссылка](#).

NLP-progress github от Sebastian Ruder: [ссылка](#), [ссылка](#).