

Управляемая генерация естественного языка

AI 4 Humanities: ruGPT-3

Управляемая генерация естественного языка

Крупные генеративные модели на базе архитектуры Трансформер, обученные на крупном корпусе, способны производить примеры естественного языка высокого качества, даже для комплексных доменов.

Начальная последовательность, сформированная автоматически или вручную, способна задать отправную точку и направить генерацию в сторону желаемого стиля и контекста.

Такой механизм обеспечивает лишь грубый контроль. Фактически, генерацией управляют атрибуты, которые были сформированы из обучающих примеров во время обучения.

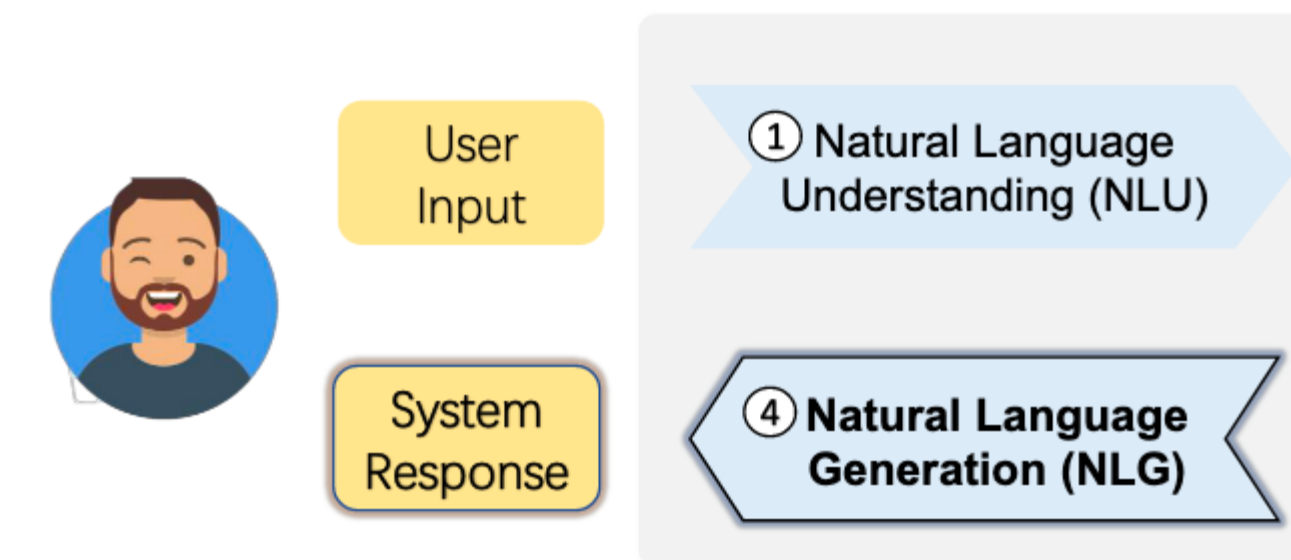
В этой работе, рассматриваются существующие подходы и предлагается решение для стабильной управляемой генерации естественного языка с использованием шаблонов.

Бизнес-применение

Одна из сфер применения управляемой генерации естественного языка – генерация ответа в **целеориентированных диалоговых системах** (таких, как Alexa, Google Assistant, Siri, Алиса, Салют и другие). Системные возможности генерации могут быть так же расширены непосредственно в **умных приложениях** для обеспечения ещё более яркого пользовательского опыта.

Интерес к целеориентированным диалоговым системам стремительно растёт с появлением всё большего количества умных приложений, способных помочь пользователям в повседневных задачах: организовать день, дать рекомендацию фильма, рассказать сказку, забронировать билеты и т.п.

В подобных системах, задачей модуля генерации (NLG) является не только обеспечение естественного ответа (chit-chat диалоговые системы), но и заполнение слотов шаблона значениями характерными для выполняемой задачи (т.е. название ресторана, время бронирования и т.п.).



Peng et. al, 2020

Постановка задачи, модель и данные

В данной работе, **шаблон генерации** обеспечивает возможности для:

- **управления контекстом** генерируемого текста;
- **заполнения слота шаблона** заданным при генерации значением.

По причине отсутствия размеченных данных для целеориентированных диалоговых систем для русского языка, используются данные из SentiRuEval 2016 (комментарии в Twitter) и RuCoS (новости). Далее в примерах участвуют две модели, которые раздельно обучены на этих данных.

- В SentiRuEval 2016 контекст задает эмоциональный окрас твитов, т.е. “позитивный”, “нейтральный”, “негативный”.
- В RuCoS контекст был добавлен по средствам разметки статей по категориям: “спорт”, “политика”, “бизнес”, “мир”.

Формат шаблона для генерации

В этой работе, используется формат шаблона представленный в Tsung-Hsien Wen et. al, 2015.

- Пример для данных **RuCoS**

News{inform(**topic**=спорт, **agency**=РИА Новости)}

Где

- **topic**, определяет контекст: “спорт”, “политика”, “бизнес”, “мир”.
- **agency**, задает значение, которое будет встроено в генерируемый текст.
- **News**, домен.

- Пример для данных **SentiRuEval 2016**

Bank{inform(**sentiment**=позитивный, **subject**=Сбер)}

Где

- **sentiment**, определяет контекст: “позитивный”, “нейтральный”, “негативный”.
- **subject**, задает значение, которое будет встроено в генерируемый текст.
- **Bank** или **Telecom**, домен.

$$\mathcal{A} = \left[\underbrace{\mathbf{I}}_{\text{Intent}}, \underbrace{(s_1, v_1), \dots, (s_P, v_P)}_{\text{Slot-value pairs}} \right]$$

Peng et. al, 2020

Примеры: управление контекстом – RuCoS

```
▶ output_idsequences = generate(model, "News{inform(topic=бизнес)}[BOS]")  
gen.process_outputs(output_idsequences, tokenizer)[0]["text"]
```

'Норвегия не намерена вводить ограничения на количество кораблей для экспорта товаров из-за **Brexit**. Об этом в четверг, 12 января, пишет ТАСС. Как заявил министр экономики страны Ларс Лёкке Расмуссен, такой шаг повлияет на показатели Норвегии по экспорту. «Тот факт, что **Brexit** произошел, не означает, что мы должны ставить крест на экспорте. У нас сейчас есть очень хорошие возможности», — отметил глава норвежского кабинета. Ранее 12 января газета **The Guardian** сообщила, что британское правительство планирует ввести ограничения на число кораблей для экспорта в связи с **Brexit**.'

```
▶ output_idsequences = generate(model, "News{inform(topic=спорт)}[BOS]")  
gen.process_outputs(output_idsequences, tokenizer)[0]["text"]
```

'Фигуристка Анфиса Резцова поделилась мнением о критике Международного союза конькобежцев (**ISU**) за применение допинга российскими спортсменами. Об этом сообщает ТАСС. Резцова подчеркнула, что в мире недопустимы любые проявления дискриминации. По ее словам, стоит ли из-за этого запрещать ее катание, спортсменка не знает. «Если так дальше пойдет, то я буду вынуждена уехать. Мне пишут из стран Европы, из США, из Азии, откуда угодно.'

Примеры демонстрируют способность модели генерировать текст в соответствии с указанным в шаблоне контекстом (topic).

Примеры: управление контекстом – SentiRuEval

```
▶ output_idsequences = generate(model, "Bank{inform(sentiment=позитивный)}[BOS]")  
gen.process_outputs(output_idsequences, tokenizer)[0]["text"]
```

'Сбербанк открыл уникальный центр по обслуживанию корпоративных'

```
▶ output_idsequences = generate(model, "Bank{inform(sentiment=негативный)}[BOS]")  
gen.process_outputs(output_idsequences, tokenizer)[0]["text"]
```

'США ввели санкции против Банка Москвы, ВТБ и Россельхозбанка'

```
▶ output_idsequences = generate(model, "Telecom{inform(sentiment=позитивный)}[BOS]")  
gen.process_outputs(output_idsequences, tokenizer)[0]["text"]
```

'Компания Мобильные ТелеСистемы (МТС) объявила о запуске в коммерческую эксплуатацию сети четвертого поколения (4G) LTE в Саратовской об'

```
▶ output_idsequences = generate(model, "Telecom{inform(sentiment=негативный)}[BOS]")  
gen.process_outputs(output_idsequences, tokenizer)[0]["text"]
```

'Сотовый оператор МТС на Украине остался без лицензии'

Примеры: заполнение слота шаблона значением – RuCoS

```
output_idsequences = generate(model, "News{inform(topic=мир, agency=РИА)}[BOS]")
gen.process_outputs(output_idsequences, tokenizer, ["РИА"])[0]["text"]
```

'Новая глава Еврокомиссии Урсула фон дер Ляйен, избранная на этот пост 14 июля, рассказала журналистам о своих внешнеполитических приоритетах и программе работы на ближайшие пять лет. Об этом сообщает **РИА Новости**. «Я намерена поддерживать верховенство закона и верховенство права во всех странах, и в Евросоюзе в том числе», — заявила фон дер Ляйен. По ее словам, приоритет будет состоять в укреплении европейского единства и в возвращении смертной казни. Кроме того, глава Еврокомиссии намерена отстаивать европейскую интеграцию и права человека. «Европа должна стать надежным домом для беженцев, и я намерена сделать все для этого», — отметила фон дер Ляйен.'

```
output_idsequences = generate(model, "News{inform(topic=политика, agency=BBC)}[BOS]")
gen.process_outputs(output_idsequences, tokenizer, ["BBC"])[0]["text"]
```

'Партии «Единая Россия» и КПРФ раскритиковали идею сформировать Госдуму на базе всех фракций. Об этом сообщает **BBC News**. По мнению членов партии, необходимо принимать во внимание национальные программы партий, а также рекомендации парламентских комитетов по фракциям. Парламентарии предлагают отказаться от разделения на «спецфракции» и «максимально широкую» группу депутатов, чтобы все фракции могли принять участие в работе нижней палаты парламента. «Мы знаем, что у нас в парламенте очень много различных фракций. Мы знаем, что у нас есть Жириновский, у нас есть все, кто хочет набрать себе очки в Думе, и мы знаем, как они проводят свои выходные», — отметил зампреда фракции ЛДПР Михаил Дегтярев.'

Примеры демонстрируют присутствие заданного для слота (agency) значения в финальном тексте.

Примеры: заполнение слота шаблона значением – SentiRuEval

```
▶ output_idsequences = generate(model, "Bank{inform(subject=Альфа, sentiment=позитивный)}[BOS]")  
gen.process_outputs(output_idsequences, tokenizer, verify("Альфа"))[0]["text"]
```

'Альфабанк снизил ставку по ипотеке'

```
▶ output_idsequences = generate(model, "Telecom{inform(subject=ТЕЛЕ2, sentiment=позитивный)}[BOS]")  
gen.process_outputs(output_idsequences, tokenizer, verify("ТЕЛЕ2"))[0]["text"]
```

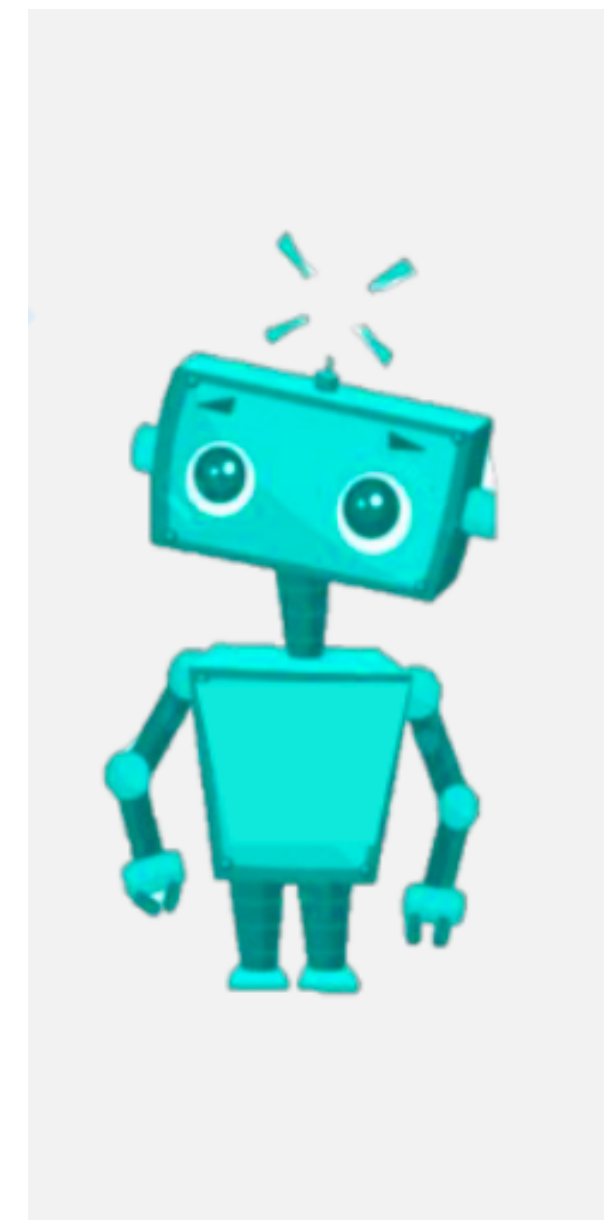
'Tele2 расширяет границы международного роуминга'

Эмоциональное вовлечение и социальная значимость

Голосовые технологии в значительной степени изменяют пользовательский опыт. Умные ассистенты и приложения способны **упростить повседневные задачи, качественно улучшить и сделать более ярким опыт взаимодействия** с технологическими продуктами. Для людей старшего поколения и людей с ограниченными возможностями, голосовые технологии **открывают ранее не доступные возможности**. Для бизнеса, это новый свободный для заполнения рынок.

Целью этой работы, является привлечение внимания к русскоязычному сегменту голосовых технологий.

Открытый доступ к размеченным данным для русского языка и раскрытие обученных моделей способны дать импульс для инновационной деятельности в профессиональном сообществе и развитию русскоязычного сегмента голосовых технологий.



Peng et. al, 2020

Инновации

Наиболее распространенными архитектурами для управления генерацией естественного языка, в контексте авторегрессивных моделей, являются:

- **SC-GPT** (Peng et. al, 2020)

Управляющий шаблон задается в виде текстовой последовательности, Это позволяет внедрить контролирующие атрибуты непосредственно в данные на которых совершается fine-tuning.

- **PPLM** (Dathathri et. al, 2019)

Языковая модель не изменяется и дополнительно не обучается. Для управления генерацией над существующей моделью надстраивается дополнительная модель.

- **CTRL** (Keskar et. al, 2019)

Для обучения модели используется корпус размеченный управляющими кодами. Может быть использован fine-tuning для обучения модели новым кодам.

Инновации

В данной работе, используется архитектура представленная в Peng et. al, 2020. Отличительной особенностью которой является, то что после обучения генерации на шаблонах одного домена, модель способна адаптироваться под новые, ранее не известные домены.

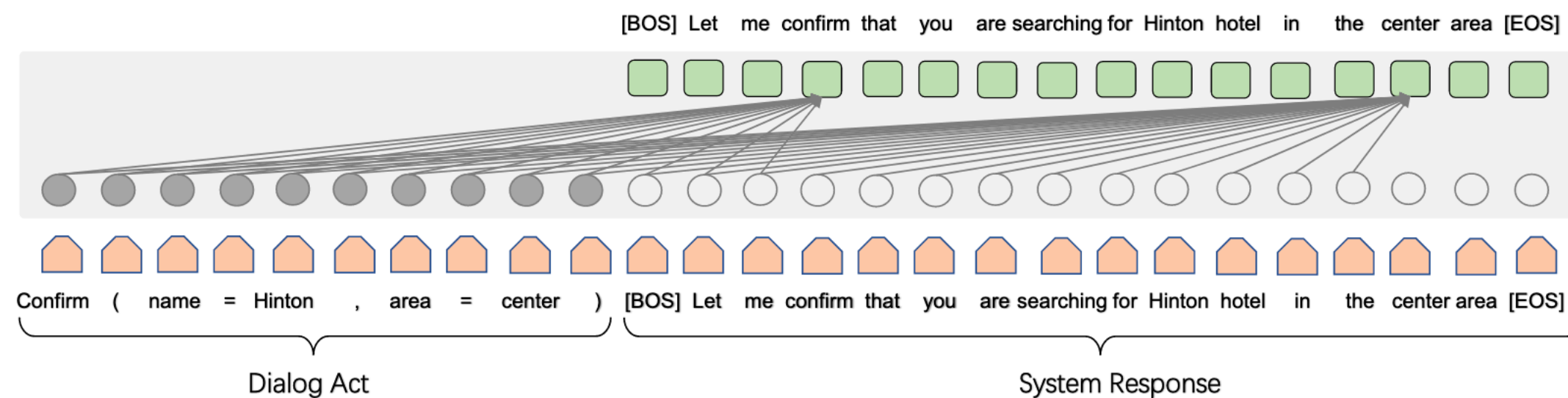
Для целеориентированных диалоговых систем это свойство играет важную роль, потому что:

- **требует меньшего количества размеченных данных**, которые для ряда доменов могут быть не доступны в принципе;
- **решает проблему масштабирования**, связанную с ростом количества комбинаций диалоговых актов и слотов шаблонов.



Технические детали: обучение

- Основой для обеих моделей выступает ruGPT-3 Large. Fine-tuning выполнялся для последних 3х слоёв модели, в течении 5 эпох.
- Формат примеров обучения повторяет представленный в Tsung-Hsien Wen et. al, 2015. Для ограничения генерируемой последовательности используются специальные символы [BOS] и [EOS].



Peng et. al, 2020

- Loss рассчитывается только для генерируемой последовательности.

Технические детали: генерация

- Для выбора кандидатов используется семплирование со значением температуры 0.75, что ограничивает вариативность генерируемой последовательности;
- Для каждого кандидата рассчитывается ROUGE score и slot error rate (если слот задан);
- Кандидаты сортируются в порядке убывания ROUGE score и получают негативное пенальти если slot error rate отличен от нуля;
- slot error rate рассчитывается как $(p + q) / m$, где p и q – количество пропущенных и избыточных значений слота, m – общее количество слотов (в данной работе, $m=1$).

Технические детали: ресурсы

- **Few-shot Natural Language Generation for Task-Oriented Dialog** (Peng et. al, 2020)
<https://arxiv.org/abs/2002.12328>
- **Plug and Play Language Models: A Simple Approach to Controlled Text Generation** (Dathathri et. al, 2019)
<https://arxiv.org/abs/1912.02164>
- **CTRL: A Conditional Transformer Language Model for Controllable Generation** (Keskar et. al, 2019)
<https://arxiv.org/abs/1909.05858>
- **Semantically Conditioned LSTM-based Natural Language Generation for Spoken Dialogue Systems** (Tsung-Hsien Wen et. al, 2015)
<https://arxiv.org/abs/1508.01745>