Управляемая генерация текста с использованием механизма внимания

Беляев Станислав
Научный руководитель: Николенко Сергей Игоревич

Санкт-Петербургский Академический Университет

18 июня 2018

Введение

Задача генерации

Нейронным сетям удается эффективно обобщать зависимости для данных, имеющих **непрерывное** представление в $\mathbb{R}^n$ (картинки, видео).
В **дискретном** же пространстве операции теряют свойство дифференцируемости, что ведет к трудностям при оптимизации.

Задача генерации (порождения)

По подвыборке $X_{\text{train}} \subset X$ генеральной совокупности X, распределенной по p_{data}, построить (явно или неявно) распределение p_{model}, приближающее реальное.

Возьмем в качестве X текстовые данные (последовательность символов из конечного алфавита). Недостатки существующих порождающих моделей для текста:

- Низкая связность или вариативность при генерации длинных примеров
- Невозможность эффективно использовать неразмеченные данные при генерации с условием
- Отсутствие интерпретируемости

Введение

Цель

Целью данной работы является разработка генеративной модели, позволяющей производить эффективную, управляемую и интерпретируемую генерацию текстовых данных с увеличенной длиной в условиях данных с частичной разметкой, поддерживая *связность*, *правдоподобие* и *разнообразие* генерируемых примеров. Решение будет основываться на применении идей механизма **внимания** (attention) из глубокого обучения.

Задачи:

- Проанализировать предметную область и существующие модели.
- Выбрать и предобработать данные для обучения и тестирования.
- Выбрать метрики для оценки результата.
- Придумать и реализовать способы, позволяющие эффективно справляться существующими проблемами.
- Произвести сравнение подходов и анализ результатов.

Данные

Описание

Каждый $x \in X$ - цельный законченный отрывок длиной в пару предложений с частично размеченным свойством.
Для размеченной части мы возьмем Стэнфордский датасет (**SST**), основанный на базе данных отзывов о фильмах. Неразмеченная часть взята из той же области, но из другого набора данных (**IMDB**).

- Разметка - бинарное категориальное свойство эмоциональной окраски
- Ограничение на длину предложения - 30 слов
- 30000 сэмплов для обучения, по 3000 для валидации и тестирования
- BPE encoding для токенизации, позволяет абстрагироваться от языка.
- Ограничение на размер словаря - 15000
- 4 служебных слова - $\langle bos \rangle$, $\langle eos \rangle$, $\langle unk \rangle$ и $\langle pad \rangle$ - символы начала, конца, пустоты и отступа. Используются для обрамления начала/конца и объединения примеров в батч для обучения.

Метрики

Описание

Автоматические метрики для $W \in X_{\text{test}}$ и новых сэмлов $W \in X_{\text{gen}}$

Perplexity (связность, правдоподобие)

$$PP(W) = P(w_1 w_2 w_3 \dots w_{|W|})^{-\frac{1}{|W|}} = \left[\prod_{i=1}^{|W|} \frac{1}{P(w_i | w_1 \dots w_{i-1})} \right]^{\frac{1}{|W|}}$$

BLEU (правдоподобие)

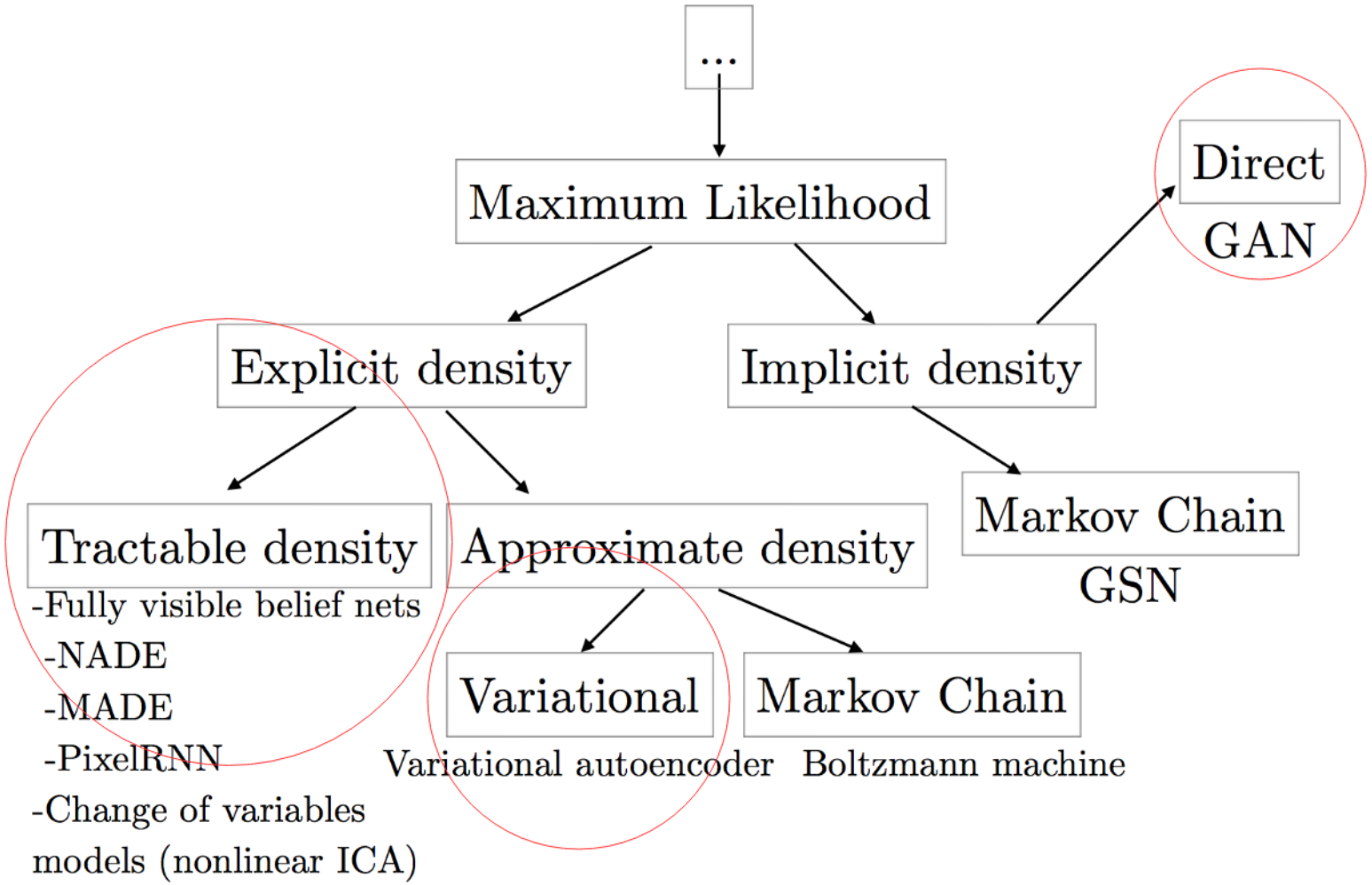
$BLEU(W_1, W_2) \in [0, 1]$, N-gram'ая схожесть, усредненная по примерам

Self-BLEU (разнообразие)

$$Self\text{-}BLEU(S) = \frac{1}{|S|} \sum_{i=1}^{|S|} BLEU(\{S_i\}, S \setminus \{S_i\})$$

Генерация

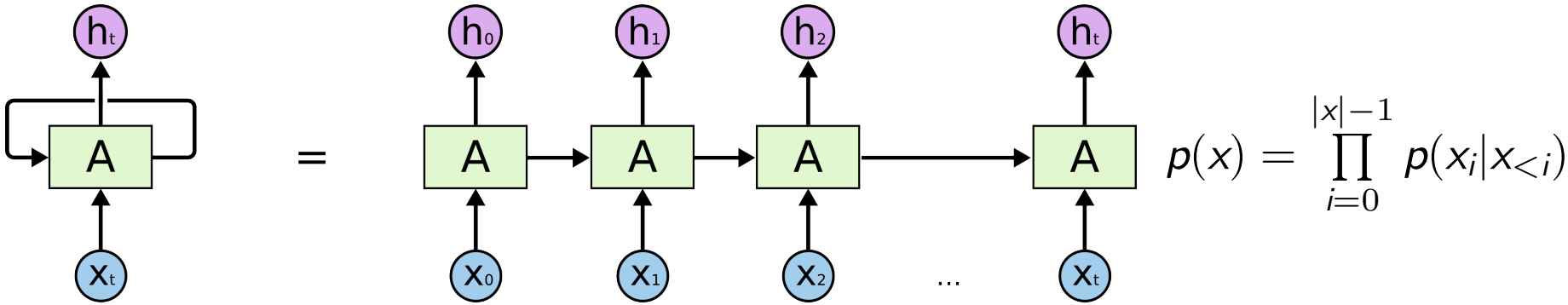
Таксономия генеративных моделей



Ian Goodfellow "NIPS 2016 Tutorial: Generative Adversarial Networks", 2017

Генерация

Рекуррентные нейронные сети (RNN)



$$p(x) = \prod_{i=0}^{|x|-1} p(x_i|x_{<i})$$

Преимущества:

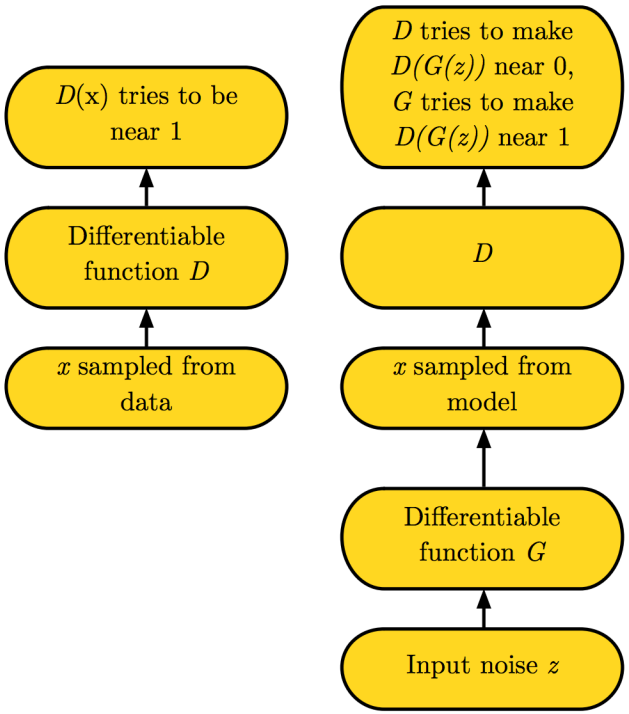
- Эффективное и простое обучение
- Расширяемая и простая реализация
- Эффективное сэмплирование и оценивание совместной вероятности

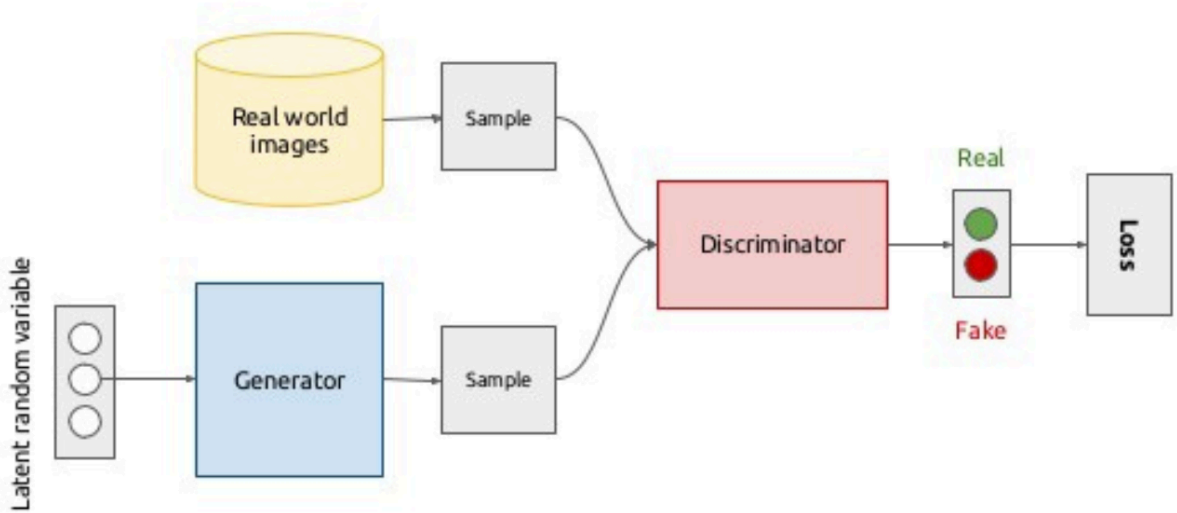
Недостатки (Bengio, 2017):

- Небольшое правдоподобие и связность
- Работает только в условиях полной разметки данных
- Неудачные механизмы управляемой генерации
  - Расширение данных (in-band)
  - Расширение архитектуры (out-of-band)

Генерация

Генеративные состязательные сети (GAN)





$$\min_G \max_D V(D,G) = \mathbb{E}_{q(\mathbf{x})}[\log(D(\mathbf{x}))] + \mathbb{E}_{p(\mathbf{z})}[\log(1 - D(G(\mathbf{z})))]$$

Для дискретных значений (Goodfellow, 2017):

- REINFORCE (SeqGAN, LeakGAN)
- GumbelSoftmax (GSGAN)
- Embeddings ($\mathbb{N} \Rightarrow \mathbb{R}^n$)

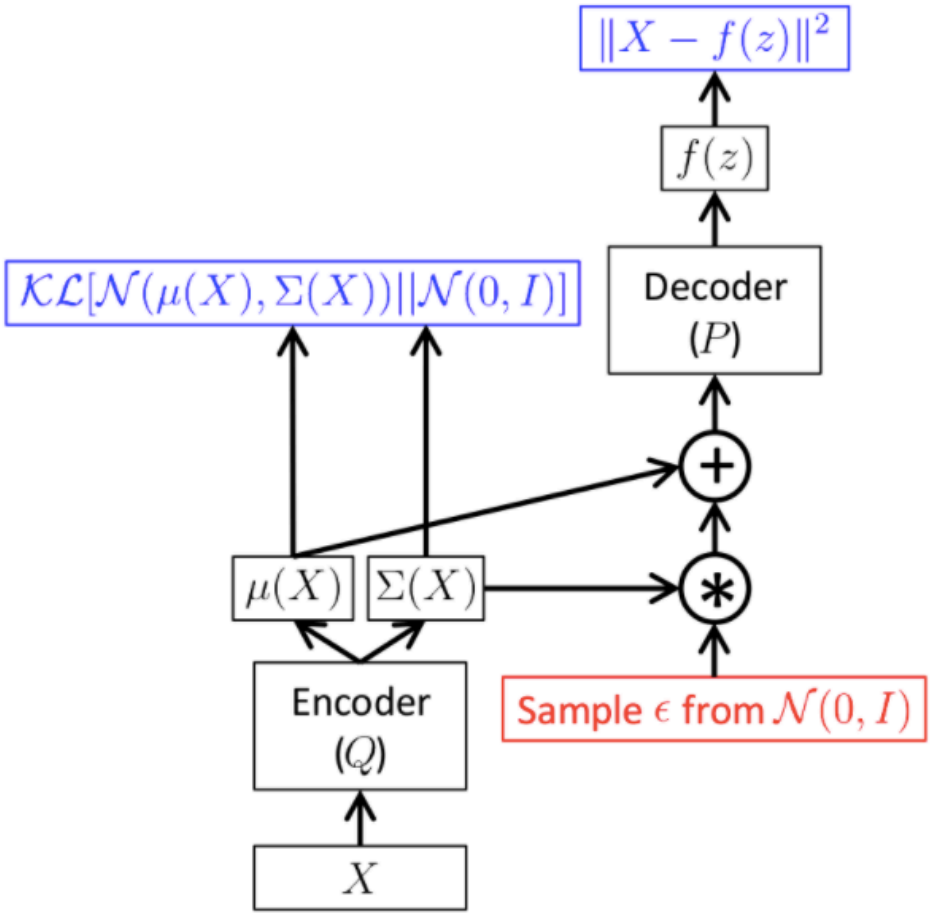
Генерация

Вариационный автоэнкодер (VAE)

$$E_{\mathbf{x}\sim p_d(\mathbf{x})}[-\log p(\mathbf{x})] < E_{\mathbf{x}}[\mathrm{E}_{q(\mathbf{z}|\mathbf{x})}[-\log(p(\mathrm{x}|\mathrm{z})]] + E_{\mathbf{x}}[\mathrm{KL}(q(\mathbf{z}|\mathbf{x})\|p(\mathbf{z}))]$$

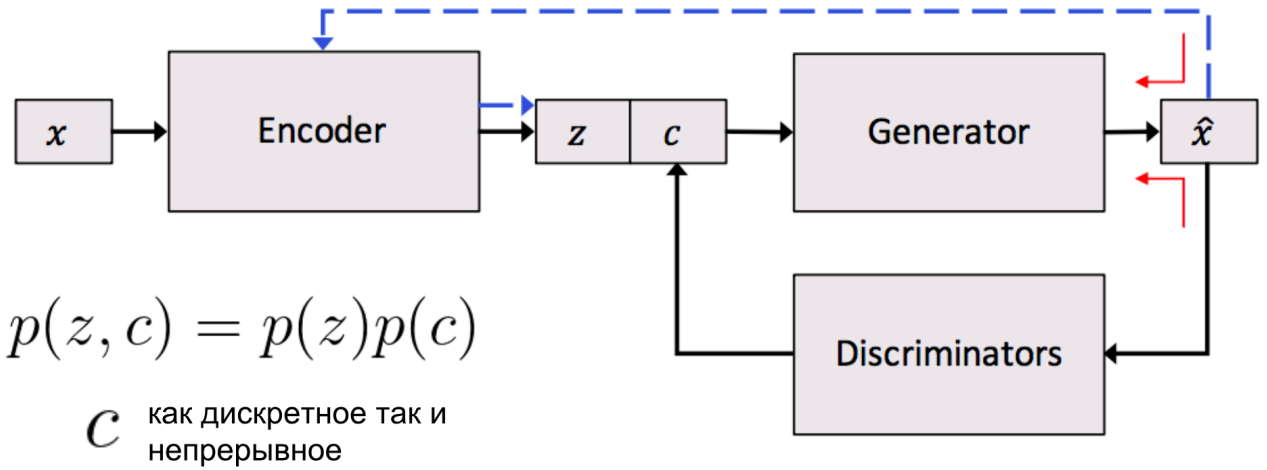
VAE - вариационное продолжение автоэнкодера для генерации.
Чтобы правильно обучить VAE для текста (Bowman et al., 2014), нужно:

- Постепенно плавно увеличивать вес ошибки kl-терма.
- Реализовать дропаут для декодера, чтобы тот не обучался быстрее энкодера.



Решение

Conditional VAE

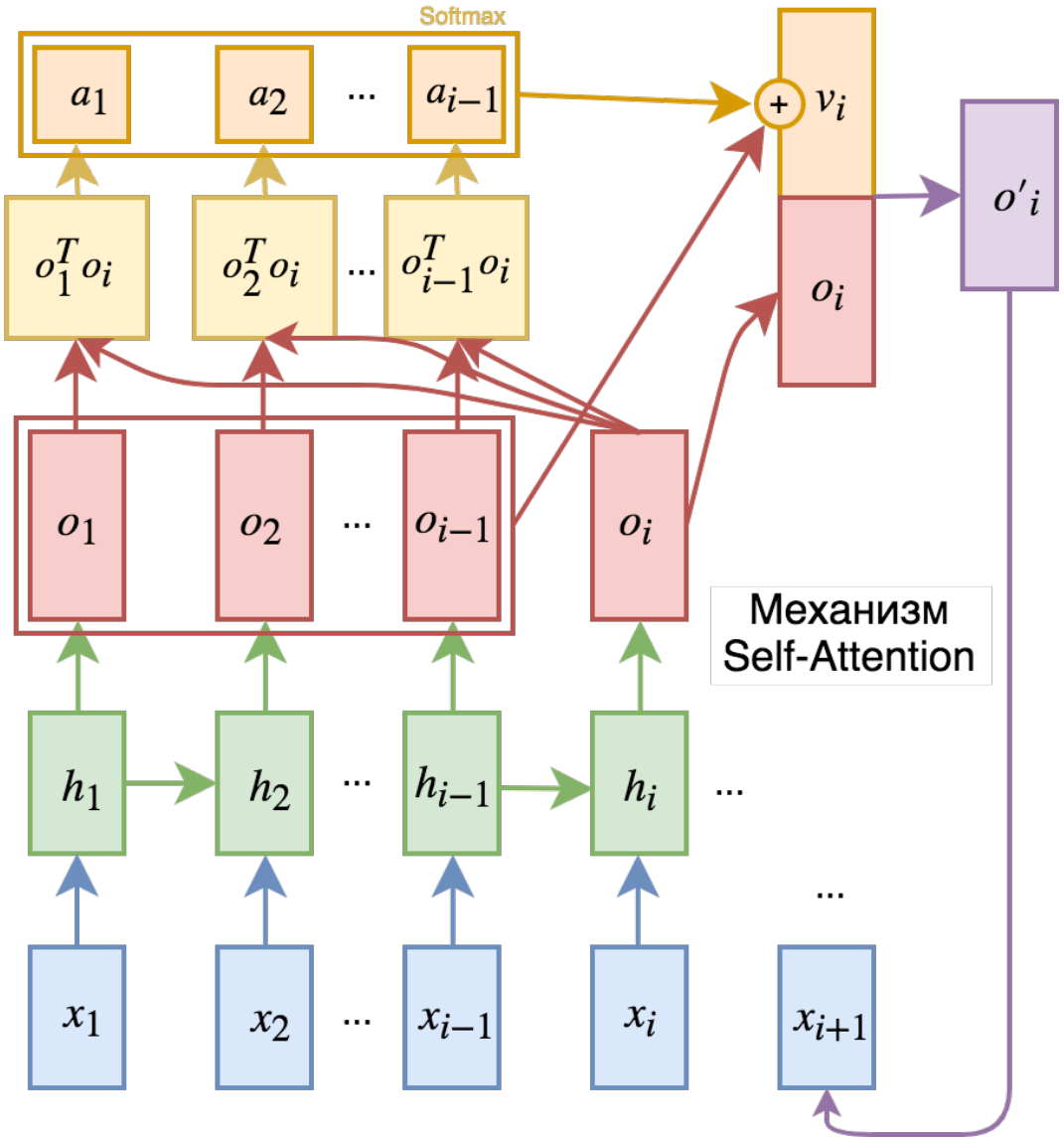


Реализация, основанная на CVAE (Hu et al., 2018):

- В качестве дискриминатора взят CNNEncoder (Zhange et al., 2016)
- Вектора слов - GLoVE размерностью 100 без заморозки
- WordDropout и плавное увеличение веса kl-терма по *tanh*
- 3-layers SRU в качестве энкодера и стохастический beam search с векторными операциями на графическом процессоре ($\sim 6x$ скорость)
- SGDR на Adam с 3 рестартами для оптимизации
- PyTorch 0.4

Решение

Self-Attention



Расширение механизма внимания из seq2seq моделей для задач генерации:

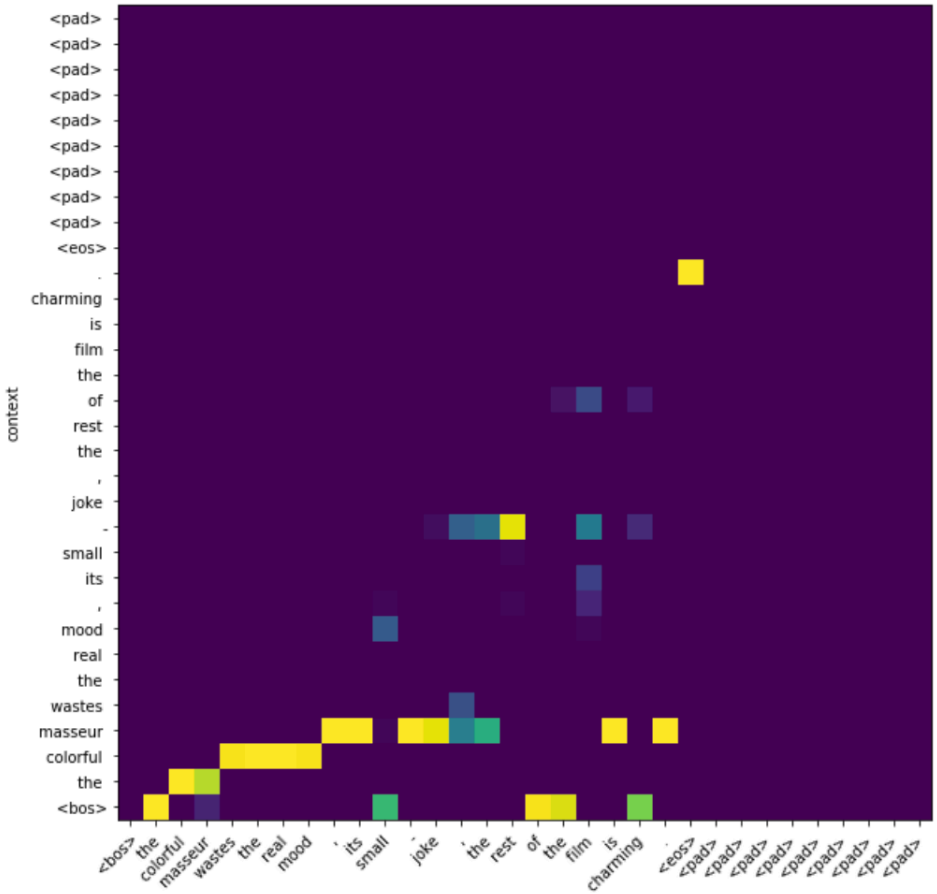
- Используем информацию о корреляции с предыдущими представлениями на очередном шаге **декодера**.
- Линейный слой до и после (general), SELU активация после
- Зависимости можно визуализировать в виде heatmap
- PyTorch 0.4

Решение

Attention penalty

Decoded	It's no use going back to yesterday.⟨eos⟩
Expected	It's no use going back to yesterday, because I was a different person then.⟨eos⟩

```
sent: 'the colorful masseur wastes the real mood , its small-joke , the rest of the film is charming .'
```



Вывод может повторяться или быть недостаточно длинным, решение:

- Каждый кандидат beam search имеет вероятность и матрицу внимания
- Регуляризация по весам матрицы:

$$cp(A) = \sum_{j=1}^{|x|} \log \left[min(\sum_{i=1}^{|x|} a_{ij}, 1) \right]$$

- $cp(A)$ - аддитивная добавка к log-правдоподобию

Результаты

Сравнение моделей и подходов

Metrics	SeqGAN	MaliGAN	RankGAN	LeakGAN	TextGAN	MLE
BLEU	18.0	15.9	15.6	**23.0**	20.7	8.1
Self-BLEU	48.9	43.7	61.8	78.0	74.6	**10.6**

Таблица: BLEU5 * 100 для 500 сгенерированных сэмплов

- D_{ACC} - точность классификации после генерации
- $CVAE_{++}$ - улучшенная реализация CVAE
- SA - Self-Attention, CP - штраф на матрице внимания

Metrics	VAE	CVAE	$CVAE_{++}$ + SA	$\mathbf{CVAE_{++}}$ + **SA** + **CP**
D_{ACC}	-	83.483	**84.263**	**84.263**
Perplexity	150	104	84	**83**
BLEU	8.5	9.3	18.2	**20.5**
Self-BLEU	9.0	8.7	45.8	**44.2**

Таблица: Метрики для расширений *VAE*

Заключение

Результаты и будущая работа

Результаты:

- Изучены принципы и особенности работы генеративных моделей с дискретными значениями, намечены основные сложности, проблемы и границы применимости разных подходов.
- Придуманы и описаны метрики для комплексной оценки качества
- Придуманы и реализованы способы, позволяющие справляться с существующими проблемами и потерей качества при увеличении длины генерации (до 30 слов).
- Эффективная, интерпретируемая и гибкая модель, основанная на CVAE.

Будущая работа:

- Преодоление ограничения на выбор априорного и апостериорного распределения в моделях, основанных на VAE (AAE, α-GAN).
- Запустить предложенную модель на дискретных не строковых данных.

Ссылки

Статьи, код и контакты

Спасибо за внимание!

1. https://github.com/stasbel/text-gen (Генерация)
2. https://github.com/stasbel/bachelor-thesis (Презентация)
3. stasbelyaev96@gmail.com
4. https://t.me/stasbel