Параллельная (не авторегрессионная) языковая модель

https://github.com/awant/non_autoregressive_lm

План

- Проблемы генерации текста, языковых моделей
- Постановка задачи
- Метрики
- Бейзлайн
- Эксперименты
- Выводы

Проблемы генерации текста

- Для текста пространство дискретно, а генерируемое латентное пространство - непрерывно

- Текст - последовательность с длинными зависимостями и сильными зависимостями каждого следующего слова от предыдущего

- Текст, как правило, генерируется последовательно (медленный инференс)

- Малая вариативность примеров. Проблема "ухода с пути" при генерации текста

- Сложность измерения качества полученного текста

Постановка задачи

- Определить метрики, характеризующие качество языковой модели
- Построить параллельную языковую модель, лишенную перечисленных недостатков
- Сравнить качество получившейся модели с бейзлайном

- Устранение прямых зависимостей между генерируемыми токенами
  - увеличивает разнообразие примеров
  - позволяет получать токены параллельно
  - может быть использовано как аналог beam search

Метрики

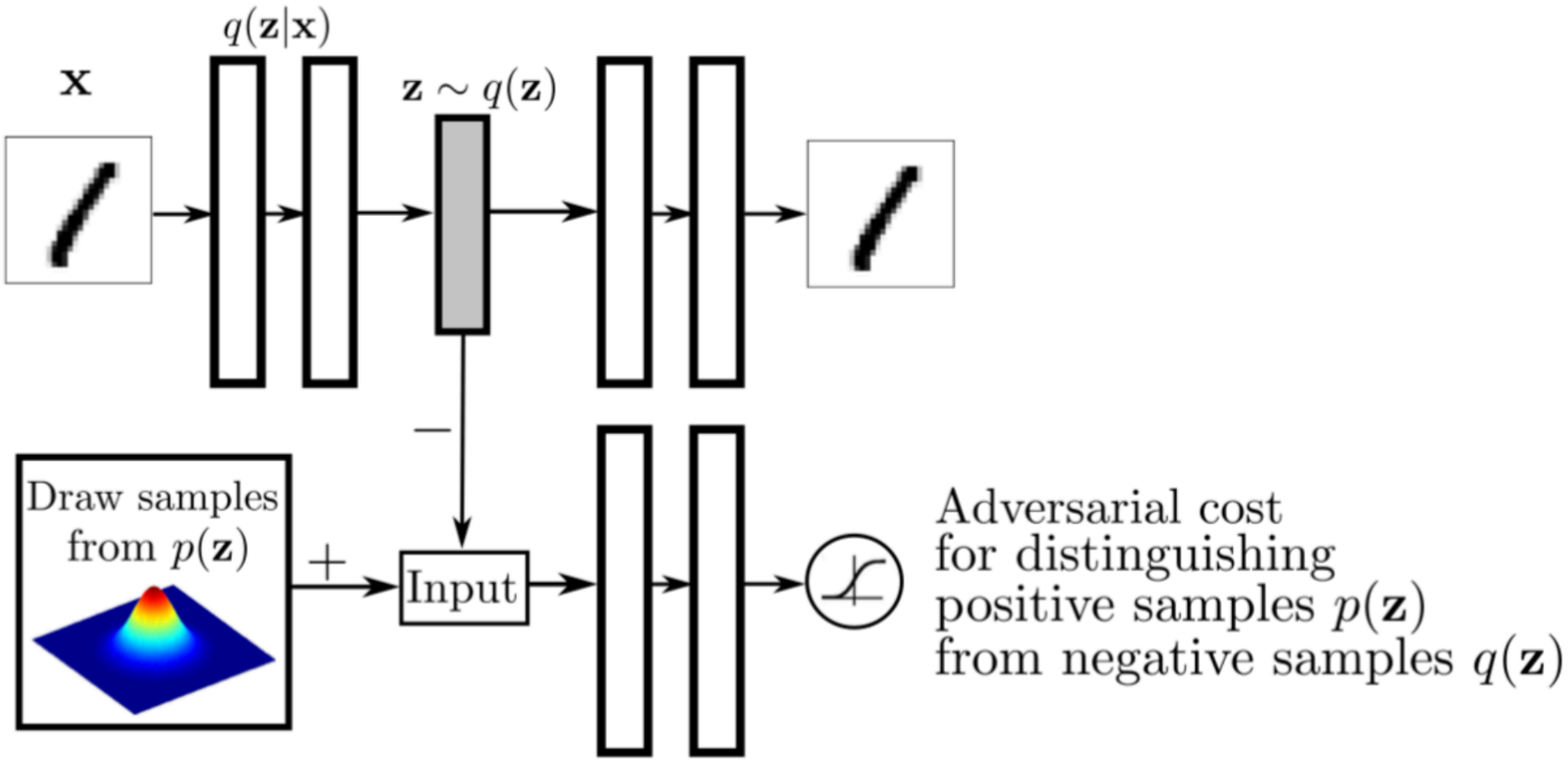
- Подсчет forward ppl, reverse ppl (меньше - лучше).
- Для подсчета perplexity выбрана хорошо известная языковая модель: **kenlm** (Kneser–Ney, https://kheafield.com/code/kenlm/)
- Обученная языковая модель: **model**
- В таком виде **forward ppl** показывает то, насколько генерируемые предложения адекватны, а **backward ppl** - их разнообразие (в случае backward ppl мы тренируем kenlm на сгенерированных предложениях и вычисляем ppl на реальных предложениях (oof sentences)

Forward ppl	**Backward ppl**
model -> sentences	model -> sentences
kenlm(sentences) -> ppl	train kenlm on sentences -> kenlm
	kenlm(oof sentences) -> ppl

Бейзлайн

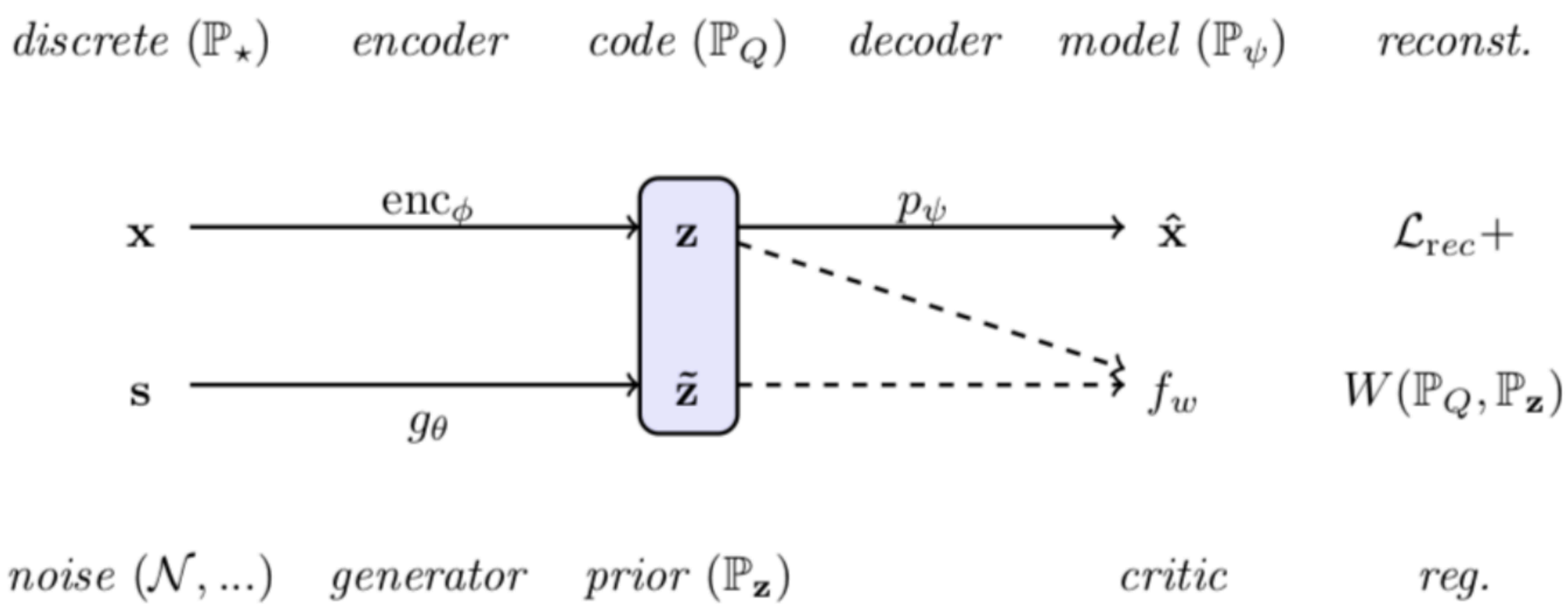
- За бейзлайн была выбрана авторегрессионная ARAE модель (https://arxiv.org/abs/1706.04223)

- ARAE модель обучается генерировать предложения из шума. Компоненты: Autoencoder, GANs.



Бейзлайн. ARAE

- Обучение модели (сети: encoder, decoder, generator, critic)
- Генератор учится отображать шум на распределение **Z** (внутреннее представление текста)
- Критик / дискриминатор старается отличить реальный Z от сгенерированного Z

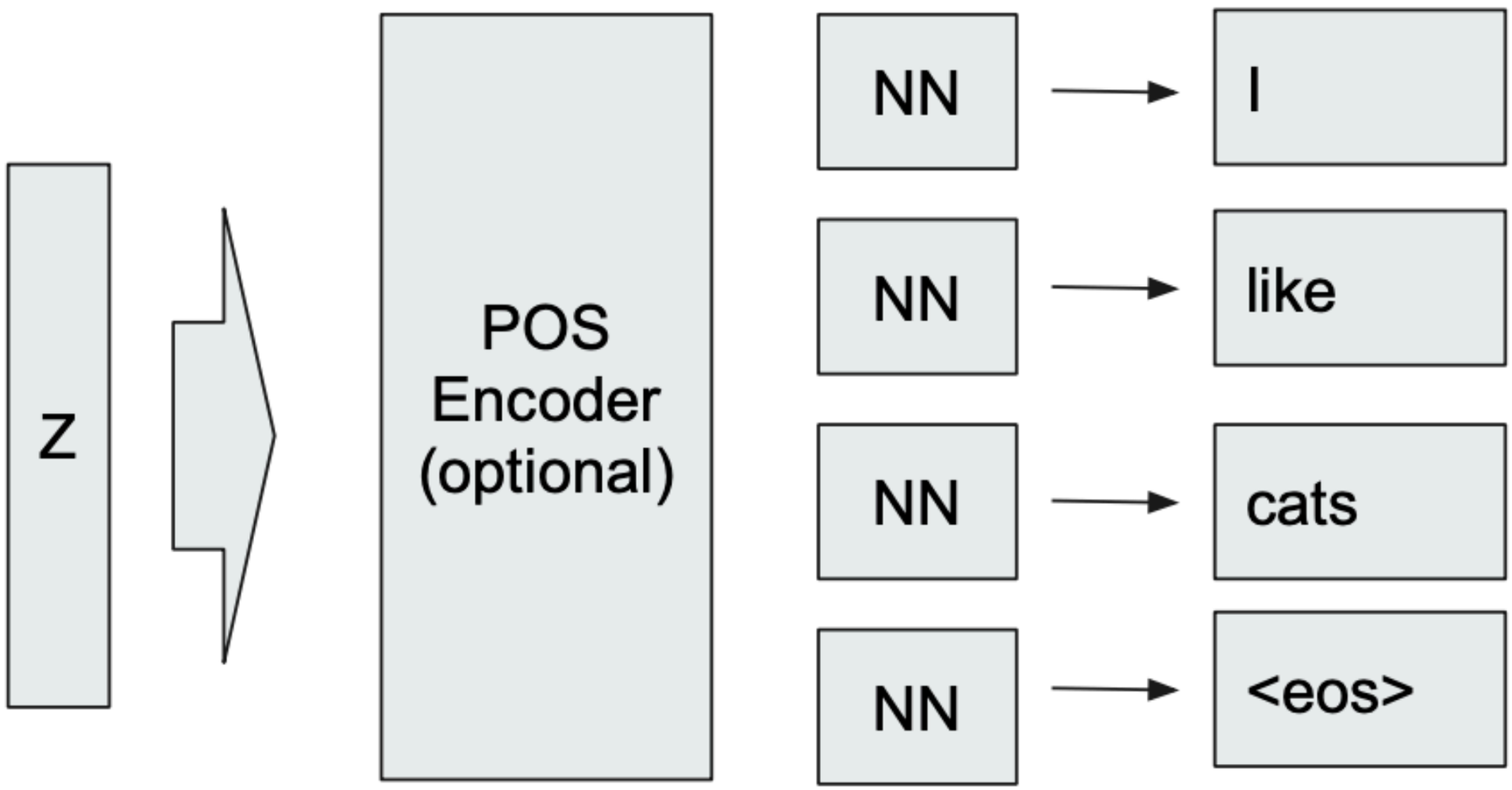


ParallARAE

- Вход декодера - вектор **z**, выход - распределение вероятностей токенов на каждой позиции

- Для получения параллельной модели из авторегрессионной выбрано несколько архитектур декодера:
  - Dense (своя Dense NNet сеть на каждое из позиций. Позиций - фиксированное количество)
  - Dense + PosEnc (добавление слоя Positional encoding - из архитектуры трансформера. суммирование/конкатенация)
  - Dense с добавлением индекса позиции (конкатенация)
  - Conv (convolutional NN)

Parall Dense/PosEnc



Результаты

Model	Autoencoder Accuracy, %	Forward PPL	Reverse PPL
ARAE	93	44.3	82.2
Parall ARAE (Dense)	86	110	108
Parall ARAE (Dense, pos_idx)	87	113	105
Parall ARAE (Dense, Pos enc, sum)	86	108	103
Parall ARAE (Dense, Pos enc, concat)	**91**	**94**	**97**
Parall ARAE (Conv)	70	181	254

Примеры сгенерированных предложений

Модель:

Parall ARAE (Dense, Pos enc, concat)

Предложения:

the man is on a slide outdoors .
the couple is sharing the bat on his friends .
the people are sitting down a river .
an old girl is with a dog .
two people walking at the field .

Выводы

- Исследовано латентное пространство
- Реализована ARAE модель
- Выполнены эксперименты по параллельной генерации
- Выбрана лучшая архитектура
- Написание статьи (?)