Generative Adversarial Networks, Recurrent Neural Networks, Sequence GANs and Chemoinformatics

Мария Попова

МФТИ, Сколтех, UNC Chapel Hill

6 ноября 2016

Генеративные модели

- Имеется обучающая выборка $\mathbf{x} \sim \mathbf{p_{data}}(\mathbf{x})$
- Требуется построить модель для генерации новой выборки: $\mathbf{x} \sim \mathbf{p_{model}}(\mathbf{x})$
- При этом $\mathbf{p_{model}} \approx \mathbf{p_{data}}$

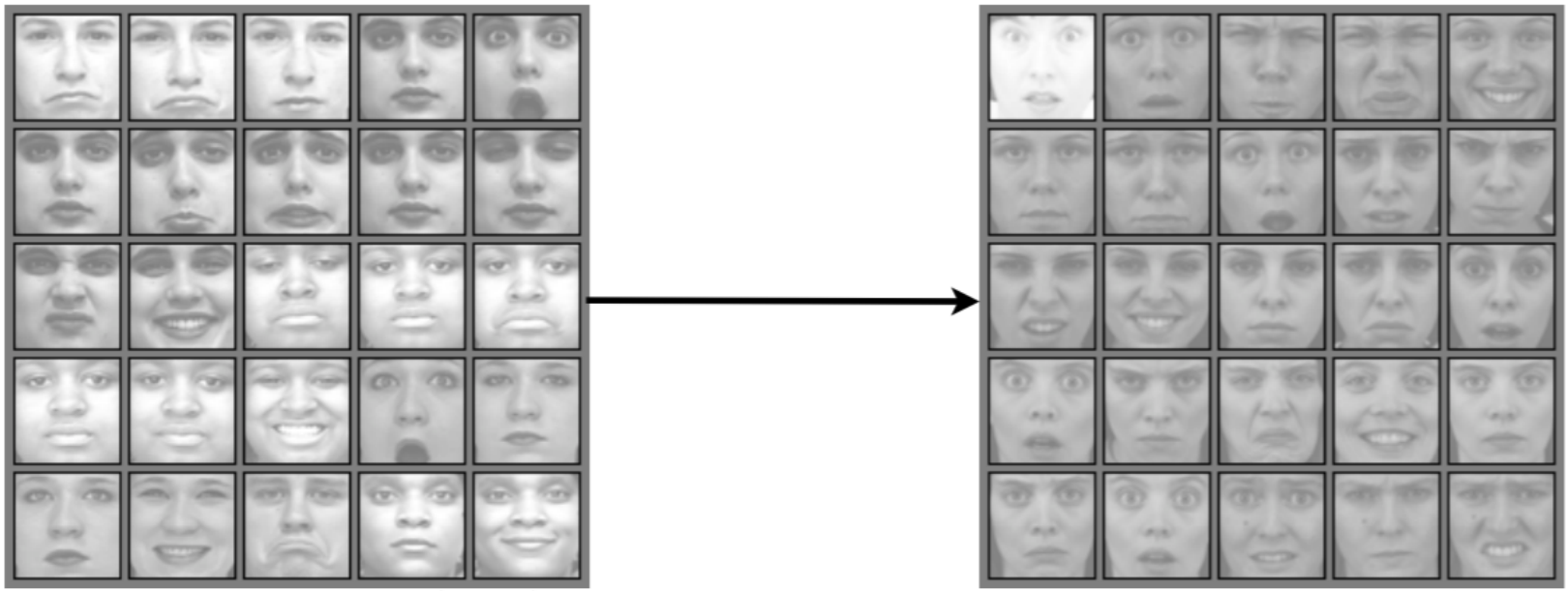
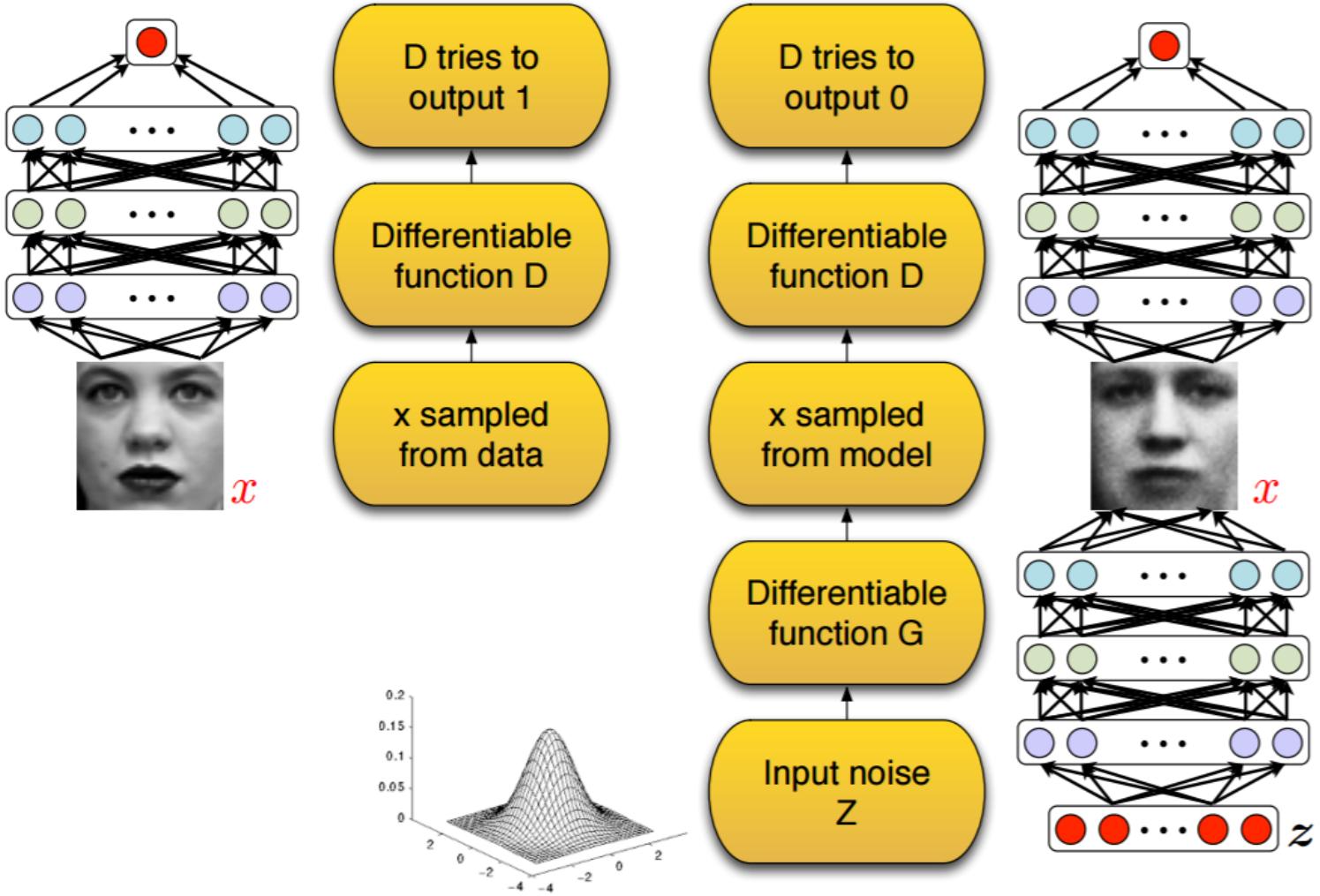


Рис.: $\mathbf{x} \sim \mathbf{p_{data}} \rightarrow \mathbf{x} \sim \mathbf{p_{model}}$

Generative adversarial подход

- Игра между двумя игроками:
  1. Генератор G
  2. Дискриминатор D
- D учится различать:
  1. Данные из обучающей выборки
  2. Данные сгенерированные G
- Генератор учится "обманывать"дискриминатор, генерируя данные, которые тяжело отличить от данных из обучающей выборки

GAN framework



Целевая функция и алгоритм

$$\min_G \max_D V(D,G) = \mathrm{E}_{\mathbf{x}\sim p_{data}(\mathbf{x})}[\log D(\mathbf{x})] + \mathrm{E}_{\mathbf{z}\sim p_{\mathbf{z}}(\mathbf{z})}[\log(1-D(G(\mathbf{z})))]$$

Algorithm 1 Minibatch stochastic gradient descent training of generative adversarial nets. The number of steps to apply to the discriminator, k, is a hyperparameter. We used $k = 1$, the least expensive option, in our experiments.

for number of training iterations **do**

  for k steps **do**

  - Sample minibatch of m noise samples $\{\boldsymbol{z}^{(1)}, \ldots, \boldsymbol{z}^{(m)}\}$ from noise prior $p_g(\boldsymbol{z})$.
  - Sample minibatch of m examples $\{\boldsymbol{x}^{(1)}, \ldots, \boldsymbol{x}^{(m)}\}$ from data generating distribution $p_{\text{data}}(\boldsymbol{x})$.
  - Update the discriminator by ascending its stochastic gradient:

$$\nabla_{\theta_d} \frac{1}{m} \sum_{i=1}^{m} \left[\log D\left(\boldsymbol{x}^{(i)}\right) + \log\left(1 - D\left(G\left(\boldsymbol{z}^{(i)}\right)\right)\right) \right].$$

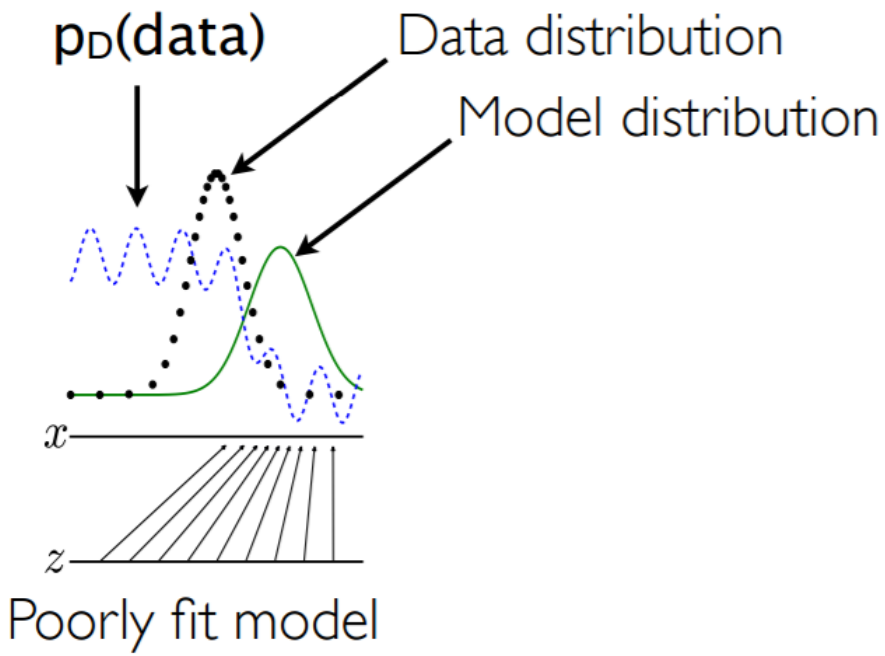
  end for

  - Sample minibatch of m noise samples $\{\boldsymbol{z}^{(1)}, \ldots, \boldsymbol{z}^{(m)}\}$ from noise prior $p_g(\boldsymbol{z})$.
  - Update the generator by descending its stochastic gradient:

$$\nabla_{\theta_g} \frac{1}{m} \sum_{i=1}^{m} \log\left(1 - D\left(G\left(\boldsymbol{z}^{(i)}\right)\right)\right).$$

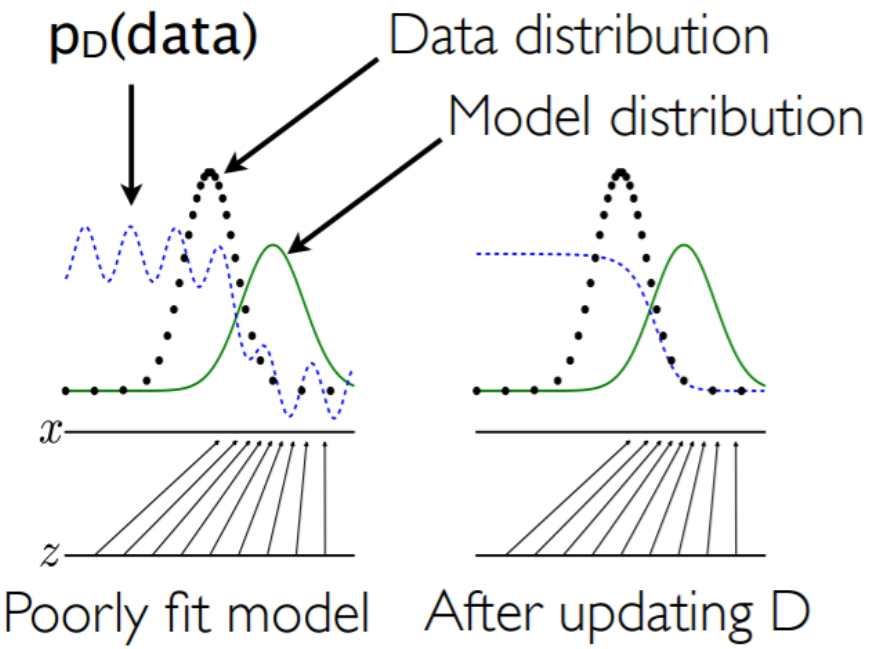
end for

Процесс обучения GAN

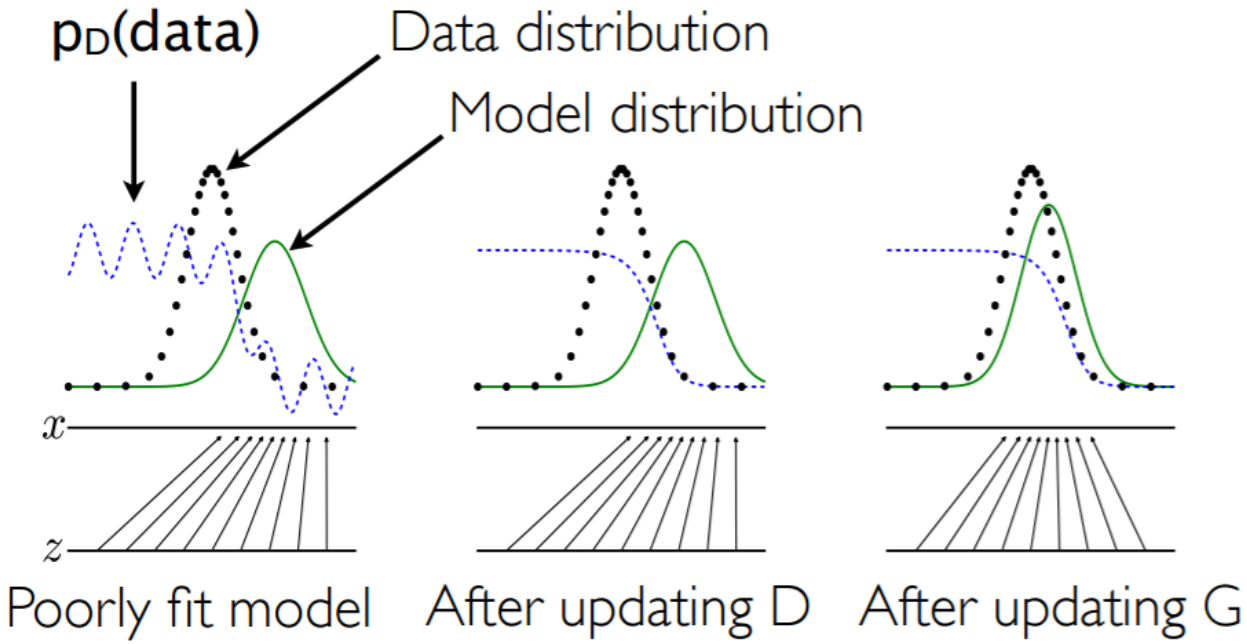


Poorly fit model

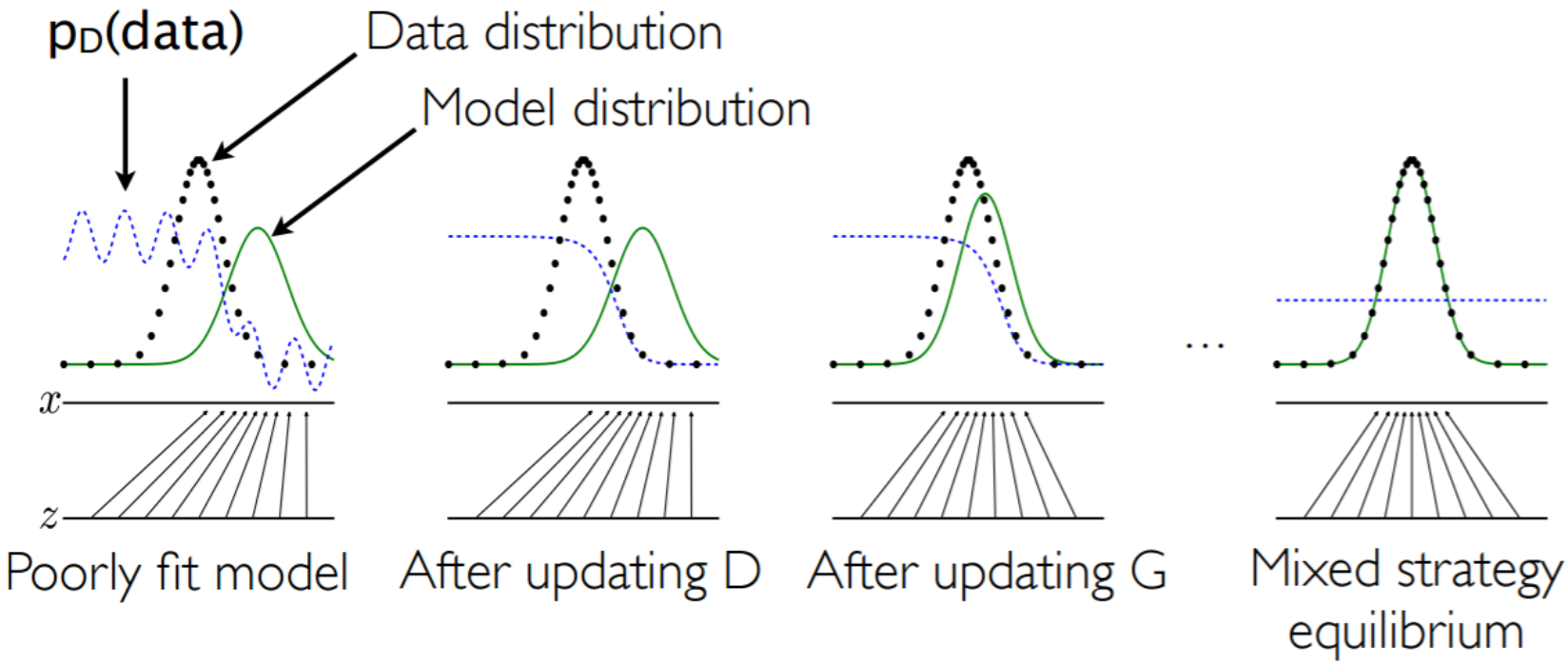
Процесс обучения GAN



Процесс обучения GAN



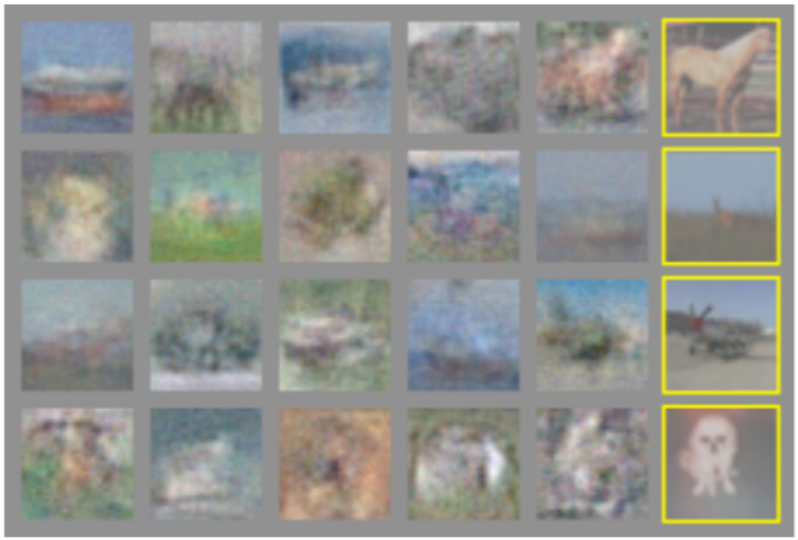
Процесс обучения GAN



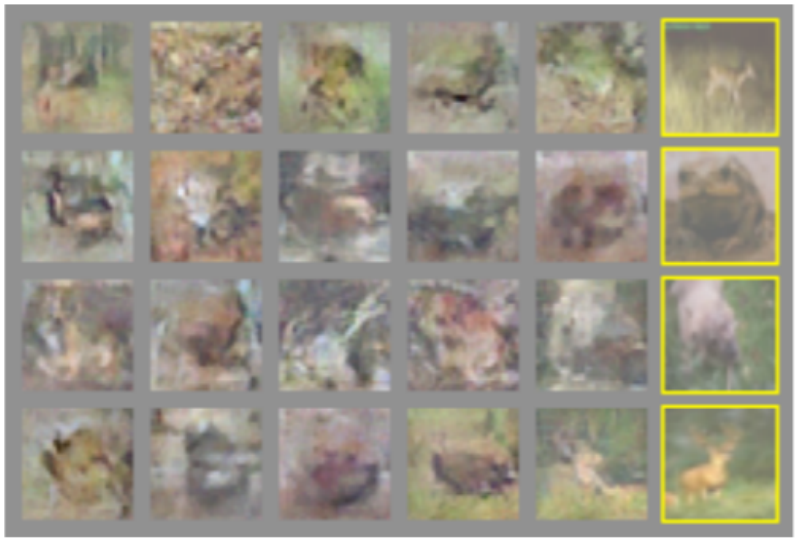
Примеры

MNIST

TFD



CIFAR-10 (fully connected)



CIFAR-10 (convolutional)

Рекуррентные нейронные сети

$$\mathbf{h}_t = \sigma(\mathbf{x}_t \mathbf{W}_x + \mathbf{h}_{t-1} \mathbf{W}_h + \mathbf{b})$$

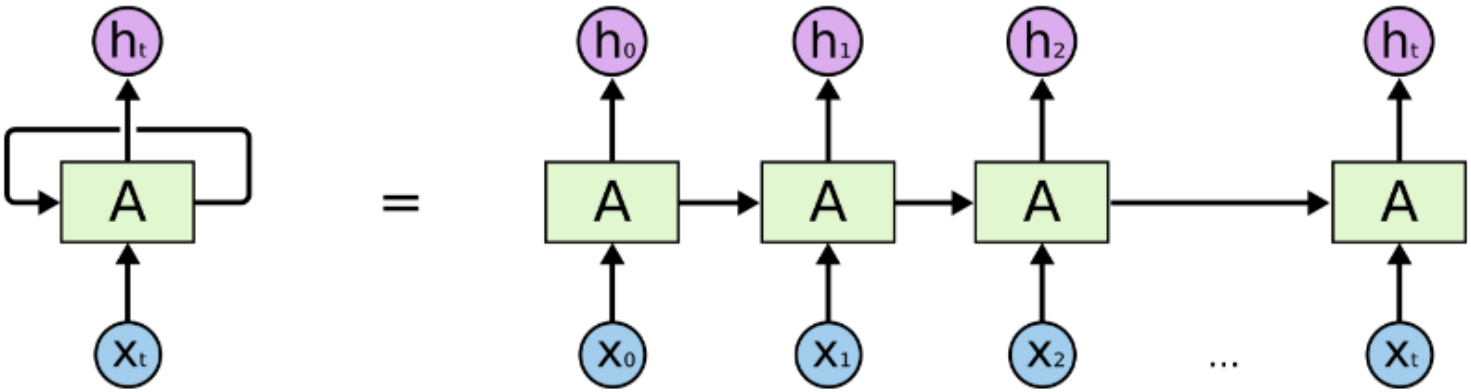
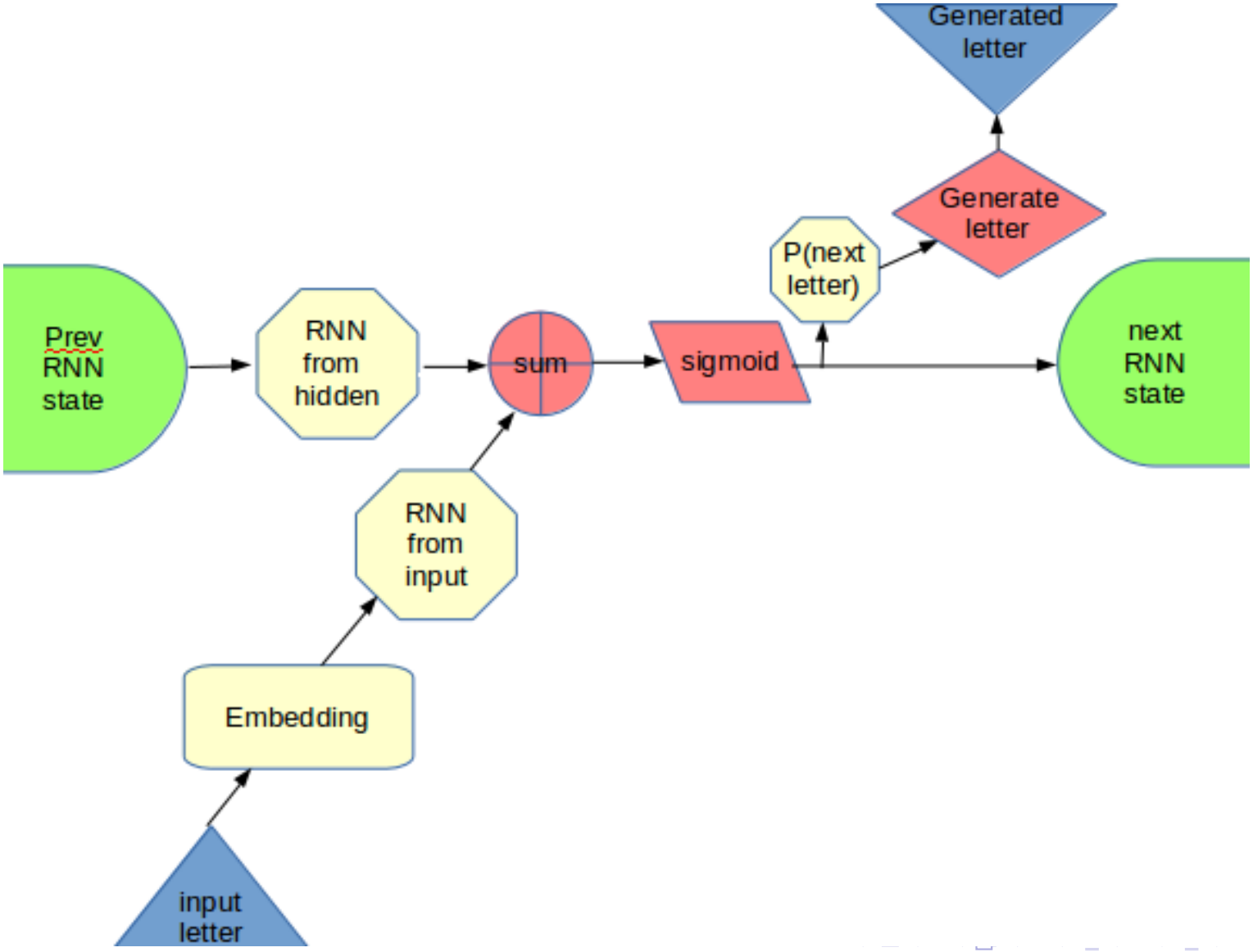


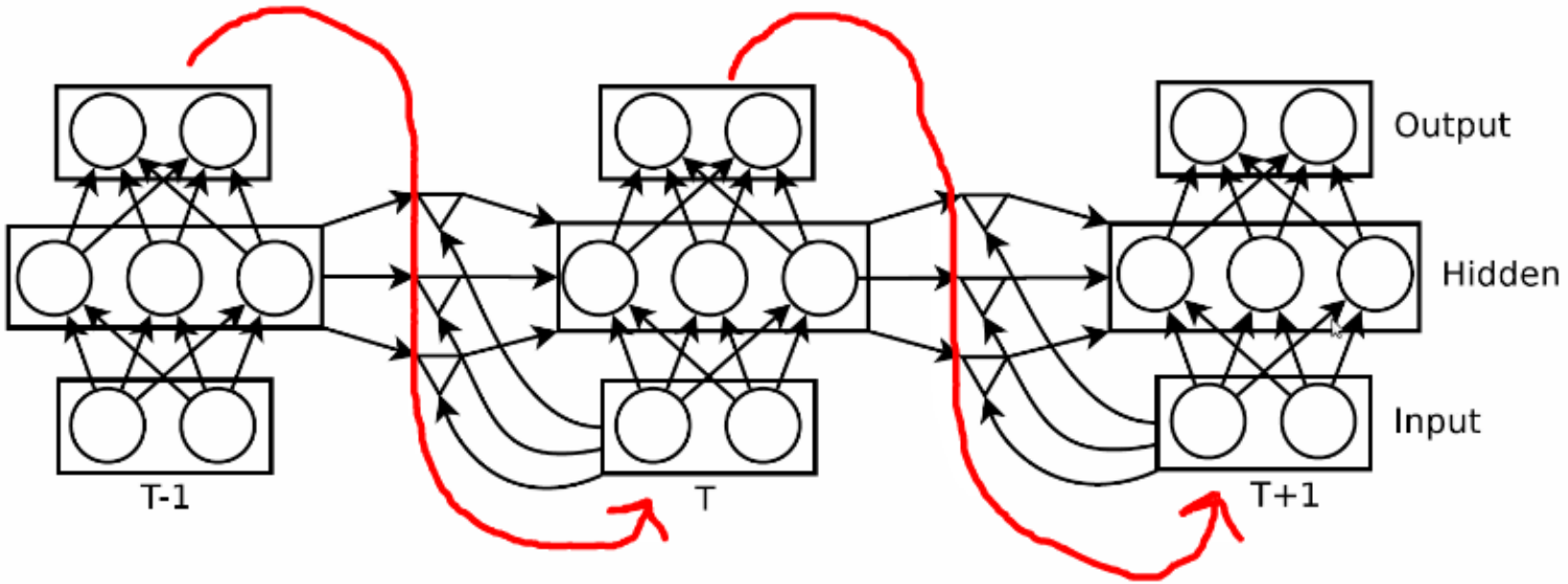
Рис.: Развернутая рекуррентая сеть

Генерация дискретных последовательностей с помощью RNN



Как тренировать RNN и генерировать новые последовательности

- Как тренировать:
  - Метод обратного распространения ошибки
  - Функция ошибки – кросс-энтропия
  - SGD
- Как генерировать:
  - В качестве $\mathbf{x}_t$ подаем на вход символ сгенерированный на шаге $t-1$.



Примеры сгенерированных текстов

Trained on Wikipedia...

The meaning of life is the tradition of the ancient human reproduction: it is less favorable to the good boy for when to remove her bigger. In the show's agreement unanimously resurfaced. The wild pasteured with consistent street forests were incorporated by the 15th century BE. In 1996 the primary rapford undergoes an effort that the reserve conditioning, written into Jewish cities, sleepers to incorporate the .St Eurasia that activates the population. Mar??a Nationale, Kelli, Zedlat-Dukastoe, Florendon, Ptu's thought is. To adapt in most parts of North America, the dynamic fairy Dan please believes, the free speech are much related to the

And the Machine Learning Corpus...

Recurrent network with the Stiefel information for logistic regression methods Along with either of the algorithms previously (two or more skewprecision) is more similar to the model with the same average mismatched graph. Though this task is to be studied under the reward transform, such as (c) and (C) from the training set, based on target activities for articles a ? 2(6) and (4.3). The PHDPic (PDB) matrix of cav'va using the three relevant information contains for tieming measurements. Moreover, because of the therap tor, the aim is to improve the score to the best patch randomly, but for each initially four data sets. As shown in Figure 11, it is more than 100 steps, we used ?? nto ninfty with 1000

Примеры сгенерированных текстов

The New World Order

```
seed = u"Каждый человек должен" #if you are using non-russian text corpora, use seed in it's language inst
ead
sampling_fun = proportional_sample_fun
result_length = 300

generate_sample(sampling_fun,seed,result_length)
```

```
----
 аждый человек должен, кожета оорисле; коморговленных для судом за постанусти в дицель передания сошей прод
ол временное на недействии рассиипочитель его искается ответствических лиц врему объектерсопинениях исполме
няются оснего налостоя исполнение с такой: 1карительному предопределение органами в законо обеспеченную дел

----
```

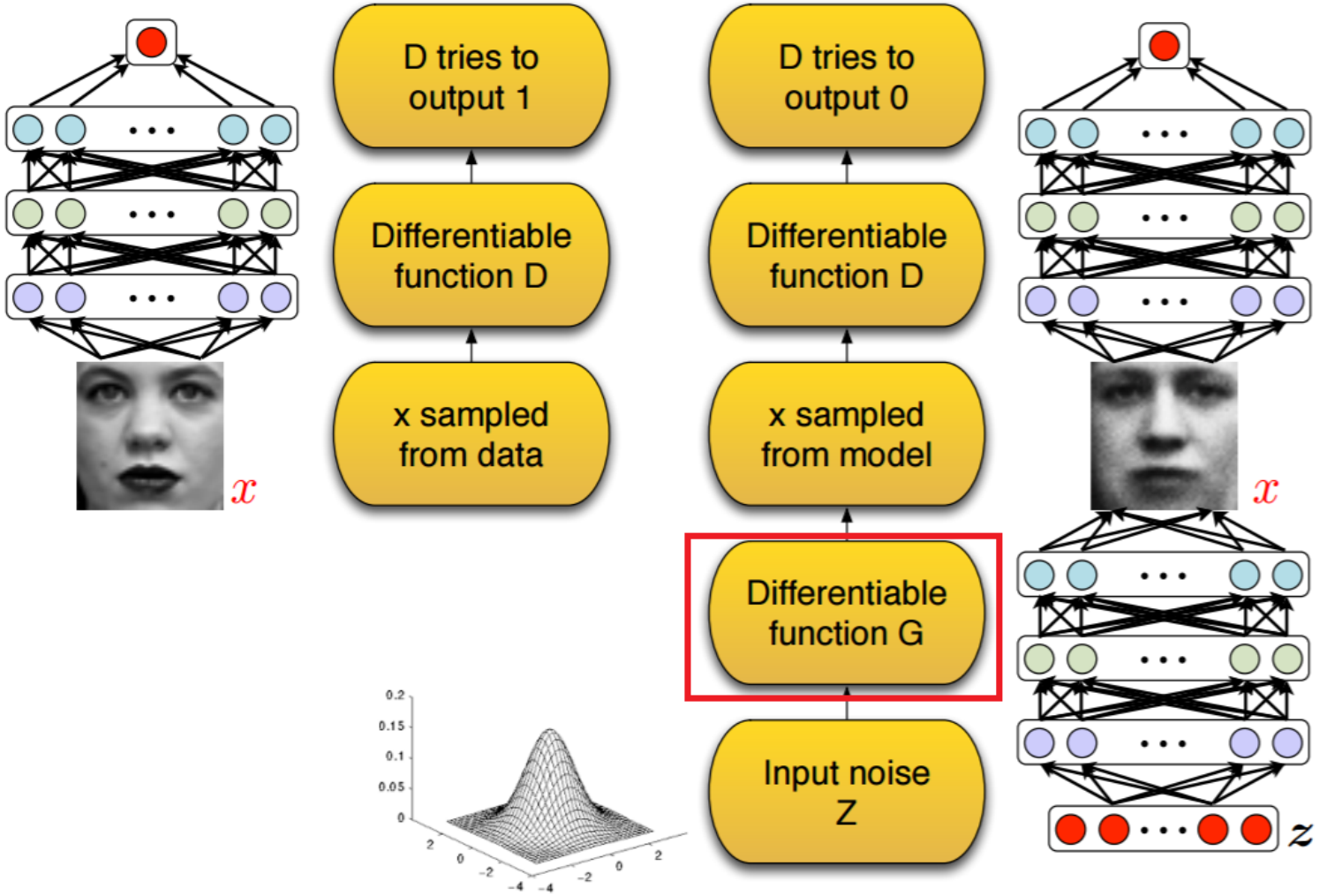
```
seed = u"В случае неповиновения"
sampling_fun = proportional_sample_fun
result_length = 300

generate_sample(sampling_fun,seed,result_length)
```

```
----
 случае неповиновения о производстве. Припие, вледение приводить в упереттствии членахли лица (портавко эксп
ортных средств, общества, по долоченным редляда аремичей, в обстанты обязан передать комору нерее изъятия б
ез помещащении, заместе обеспечен - от проживах без до принятия действию помещений, а также при удовостранн

----
```

GAN для генерации дискретных данных



SeqGAN и REINFORCE алгоритм

- Проблема:
  - Выход генератора недифференцируем
- Идея: рассматриваем Adversarial процесс с позиции Reinforcement Learning
  - Генератор – RL агент
  - Состояние – текущая сгенерированная последовательность
  - Действие – сгенерировать элемент последовательности
  - Дискриминатор назначает агенту reward

REINFRCE алгоритм

Целевая функция:

$$J(\theta) = \sum_{a_{1:T} \in \mathbb{A}^\dagger} p_\theta(a_{1:T}) R(a_{1:T}) =$$

$$\mathbb{E}_{a_{1:T} \sim p_\theta} \sum_{t=1}^{n} r(a_{1:t}) =$$

$$\mathbb{E}_{a_1 \sim p_\theta(a_1)} \mathbb{E}_{a_2 \sim p_\theta(a_2|a_1)} \cdots \mathbb{E}_{a_T \sim p_\theta(a_T|a_{1:(T-1)})} \sum_{t=1}^{T} r(a_{1:t})$$

где $\mathbb{A}$ – это множество всех возможнных последовательностей, $a_{1:T}$ – последовательность, $R(a_{1:T})$ – reward за последовательность $a_{1:T}$.

Algorithm 1 Sequence Generative Adversarial Nets

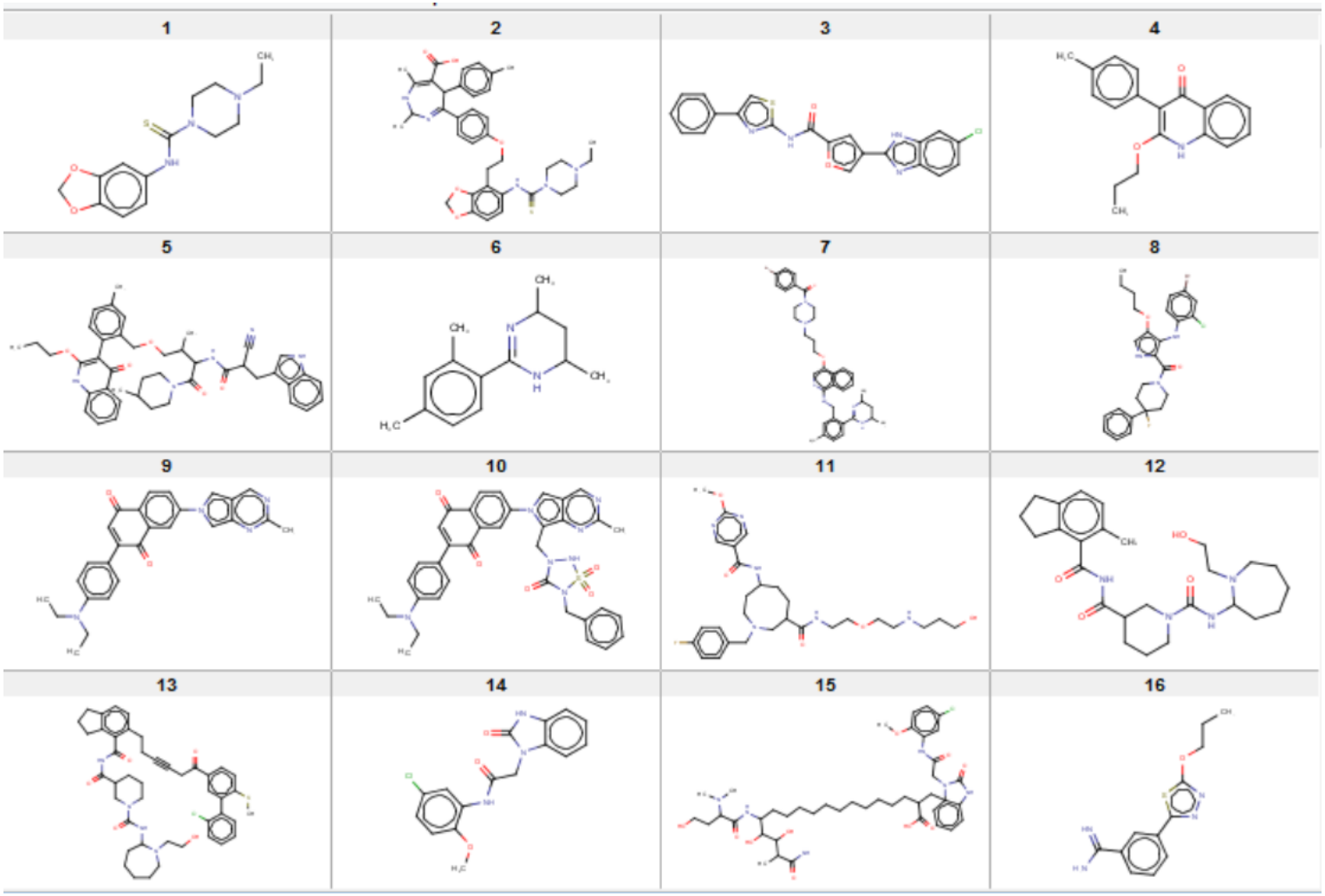
Require: generator policy G_θ; roll-out policy G_β; discriminator D_ϕ; a sequence dataset $\mathcal{S} = \{X_{1:T}\}$

Initialize G_θ, D_ϕ with random weights θ, ϕ.
Pre-train G_θ using MLE on $\mathcal{S}$
$\beta \leftarrow \theta$
Generate negative samples using G_θ for training D_ϕ
Pre-train D_ϕ via minimizing the cross entropy
repeat
  for g-steps **do**
    Generate a sequence $Y_{1:T} = (y_1, \ldots, y_T) \sim G_\theta$
    for t in $1 : T$ **do**
      Compute $Q(a = y_t; s = Y_{1:t-1})$
    end for
    Update generator parameters via policy gradient
  end for
  for d-steps **do**
    Use current G_θ to generate negative examples and combine with given positive examples $\mathcal{S}$
    Train discriminator D_ϕ for k epochs
  end for
  $\beta \leftarrow \theta$
until SeqGAN converges

SMILES strings:

```
CCC(C1CCCCC1Sc1cc(O)c(N)c2C(CC2=C1Nc1ccccc1C(CCc1ccccc1)c1ccccc1)N(=O)=O)c1ccco1
CC(C)N(C(C)C)c1cc(C)cc(c1)C1=CCCCC1
CCOC(=O)c1nnc(SCC(=O)NC2COCCNC2c3ccccc3-c2nn1-c1ccccc1)C2CCCOc1cccc(C)c1O
Brc1ccc2nc(-c3ccc(Cl)cc3)c[n+](NC(=O)c4ccc(F)cc3)cc2C1=O
Brc1ccc2nc(-c3ccc(Cl)cc3)c[n+](NC(=O)c4ccc(F)cc3)cc2C1=ONC(=O)c1sccc1C(=O)NC1CC2(OC1(C)C)c1cccs1
CCc1cc(ccn1)C1CC(CN1c1ccccc1S(=O)(=O)c1ccccc1)c1ccccc1C
ONC(=O)CCc1c(C)c(OC(F)(F)F)cc-c1OCCCc1ccccc1
ONC(=O)CCc1c(C)c(OC(F)(F)F)cc-c1OCCCc1ccccc1CN(Cc1ccccc1)C(CCc1ccccc1F)C(=O)OC
COc1ccc2C3CN(C(C)CC3)C(=O)Cc12
CC(C)(CO)Nc1nc(C)c(nc1-c1ccc(cc1)S(F)(F)-c1ccc(Cl)cc1)C(F)(F)F
O=C1Nc2sccc2-c2[nH]cc2c1Cl
O=C1Nc2sccc2-c2[nH]cc2c1ClNC1CCc2ccc(NC(=O)Nc3ccc(NC(=O)c4nc(NCCCN4CCCC5)cc5)cc4)ccc2C(F)(F)F3
FC(F)(N1CCOCC1)c1nc2ccc(Nc3nc4ccccc4[nH]3)ccn21
FC(F)(N1CCOCC1)c1nc2ccc(Nc3nc4ccccc4[nH]3)ccn21CN(C)c1ccc(cc1)C(=O)NCc1nccs1
COc1ccc2nc(ccc2n1)N1CC(CO)C(O)C1O
COc1ccc2nc(ccc2n1)N1CC(CO)C(O)C1OCOc1ccc(c(F)c1)-c1cc2CCN(C)c2c(C)c1C#N
Cc1cccc(C)c1CNS(=O)(=O)c1ccc(OC2CC3CCCC23)cc1
Cc1cccc(C)c1CNS(=O)(=O)c1ccc(OC2CC3CCCC23)cc1CNc1ccccc1CNc1ccc2c(N)nnc2c2c1C#Cc1ccc(Cl)cc1
CC(C)Nc1ncc(nn1)-c1c(oc2ccc(O)cc12)-c1cnc(nn1)-c1ccccc1
CC(C)Nc1ncc(nn1)-c1c(oc2ccc(O)cc12)-c1cnc(nn1)-c1ccccc1CC(=NNC(=O)c1ccccc1)c1ccc2[nH]nc(C)c2c1
OCC1=CN(C2CC2)C(C1)C1CCC2CO1
OCC1=CN(C2CC2)C(C1)C1CCC2CO1CN(C)CCOc1c(C(=O)N2CCCCC2)c(CO)n1CCCO
CC1=CC(OC(C)=O)OC2C=C(C=C(C)C12)#Cc1c(C)cc(C)cc1C
```

Приложения в хемоинформатике



Ссылки

- Ian J. Goodfellow et al. Generative Adversarial Networks arXiv:1406.2661
- Ian J. Goodfellow et al. http://www.cs.toronto.edu/ dtarlow/pos14/talks/goodfellow.pdf
- Ilya Sutskever et al. https://prezi.com/ch-dlduspqif/generating-text-with-rnns/
- Lantao Yu et al. SeqGAN: Sequence Generative Adversarial Nets with Policy Gradient arXiv:1609.05473
- Wojciech Zaremba et al. Reinforcement Learning Neural Turing Machines arXiv:1505.00521