

Рекомендательные системы



Проверить, идет ли запись

Меня хорошо видно & слышно?



Тема вебинара

Нейросетевые методы рекомендаций



Бажмин Сергей

ex Intel, Sber, Sbermarket. Сейчас Senior ML Engineer в Research отделе по улучшению поиска и рекомендаций в hh.ru
Магистр по прикладной математике и информатике



bizhman322



Правила вебинара



Активно
участвуем



Off-topic обсуждаем
в учебной группе **OTUS RecSys-2024-10**

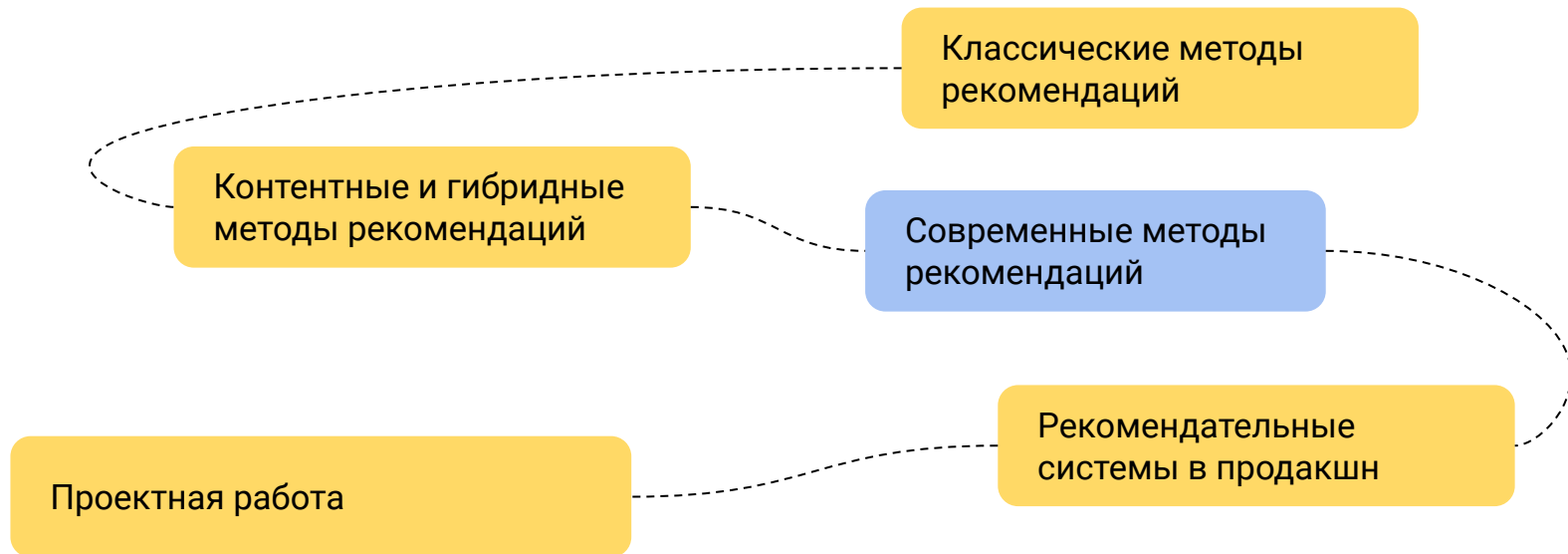


Задаем вопрос
в чат или голосом



Вопросы вижу в чате,
могу ответить не сразу

Карта курса



Маршрут вебинара



Neural CF (NCF)

Wide & Deep learning

DSSM

Deep & Cross Network

Обучение нейросети

Цели вебинара

К концу занятия вы сможете

1. Изучить какие нейросетевые подходы применяют на практике
2. Понимать основные тренды развития нейросетевых моделей для рекомендаций

Смысл

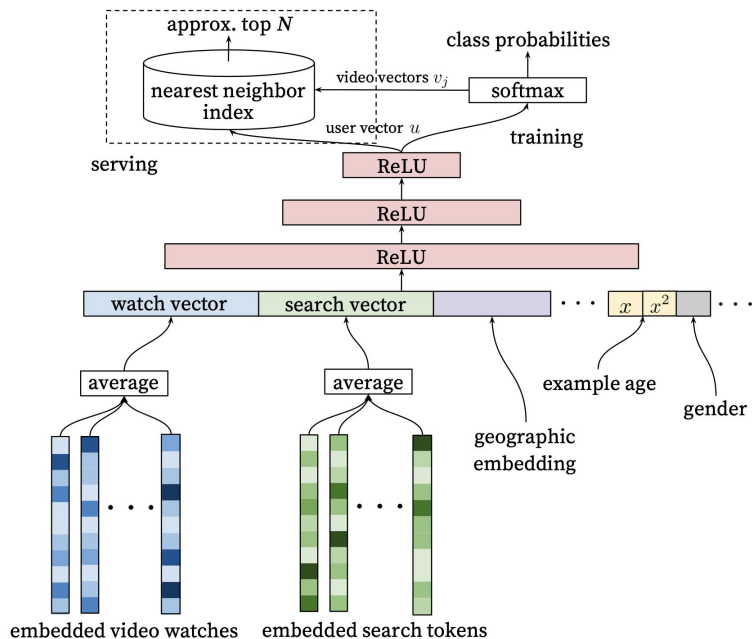
Зачем вам это изучать

1. Знать какие нейросетевые модели лучше всего работают на практике
2. Понимать в чем отличие рекомендательных нейросетевых моделей от нейросетей в других доменах (изображения, тексты и т.д)

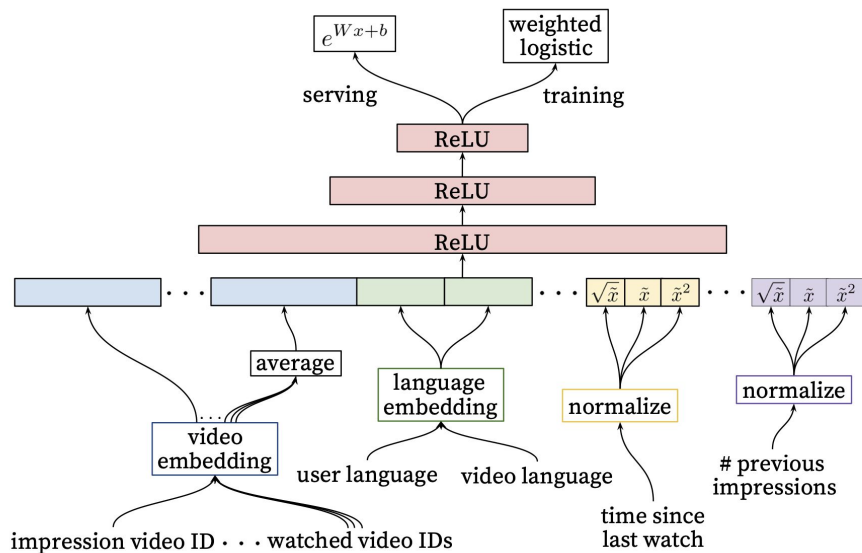
Подходы к моделированию

Пример модели YouTube (2016)

модель, отбирающая кандидатов



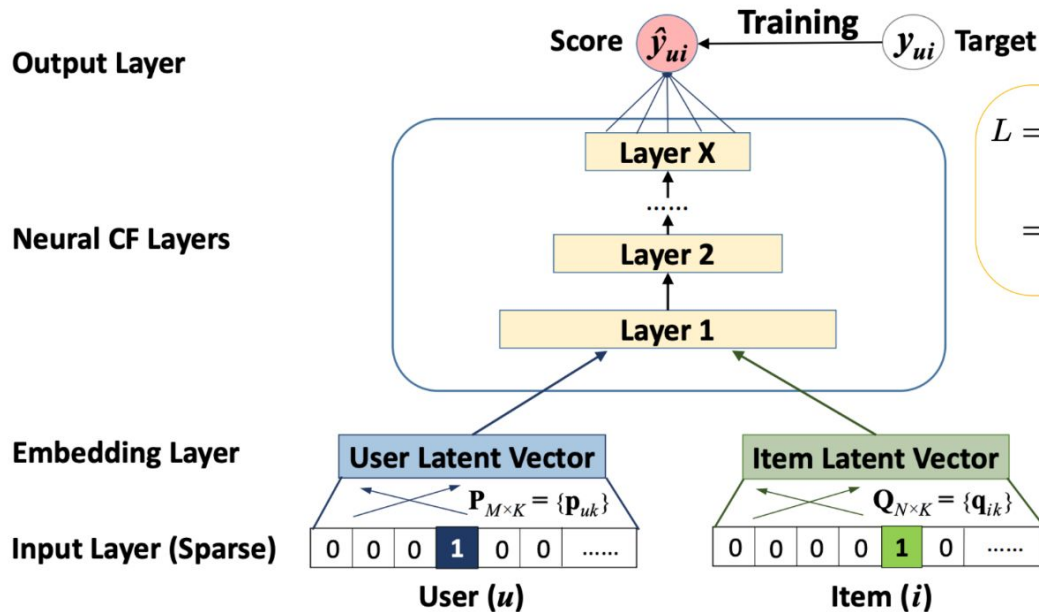
ранкер



Neural CF (NCF)

К обычному MF добавим башню из MLP:

[статья](#)



$$L = - \sum_{(u,i) \in \mathcal{Y}} \log \hat{y}_{ui} - \sum_{(u,j) \in \mathcal{Y}^-} \log(1 - \hat{y}_{uj})$$

$$= - \sum_{(u,i) \in \mathcal{Y} \cup \mathcal{Y}^-} y_{ui} \log \hat{y}_{ui} + (1 - y_{ui}) \log(1 - \hat{y}_{ui}).$$

$$\mathbf{z}_1 = \phi_1(\mathbf{p}_u, \mathbf{q}_i) = \begin{bmatrix} \mathbf{p}_u \\ \mathbf{q}_i \end{bmatrix},$$

$$\phi_2(\mathbf{z}_1) = a_2(\mathbf{W}_2^T \mathbf{z}_1 + \mathbf{b}_2),$$

$$\dots\dots$$

$$\phi_L(\mathbf{z}_{L-1}) = a_L(\mathbf{W}_L^T \mathbf{z}_{L-1} + \mathbf{b}_L),$$

$$\hat{y}_{ui} = \sigma(\mathbf{h}^T \phi_L(\mathbf{z}_{L-1})),$$

Wide & Deep learning

[статья](#)

По сути на вход в нейросеть отдается большое кол-во фичей, а над какими-то из них строим большую башню из MLP.

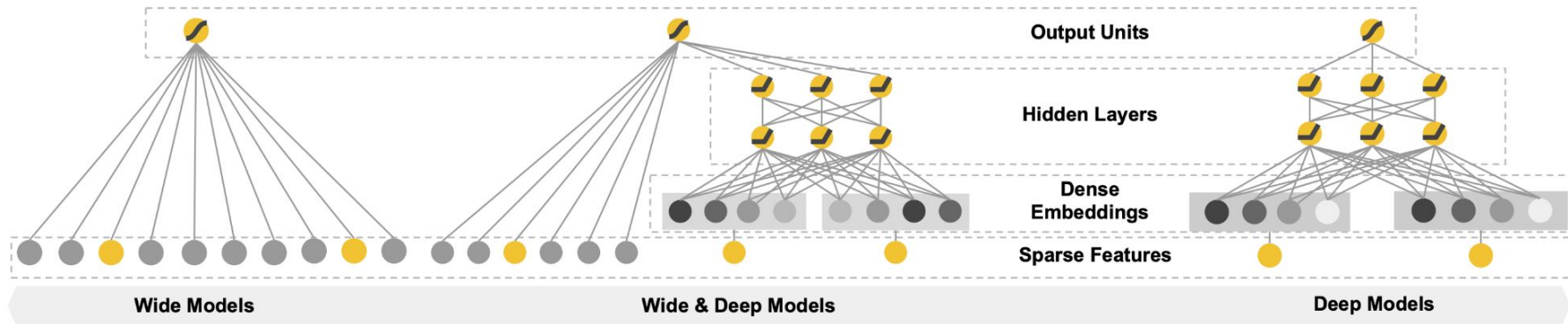


Figure 1: The spectrum of Wide & Deep models.

Deep Similarity Similarity Model (DSSM)

[статья](#)

Posterior probability
computed by softmax

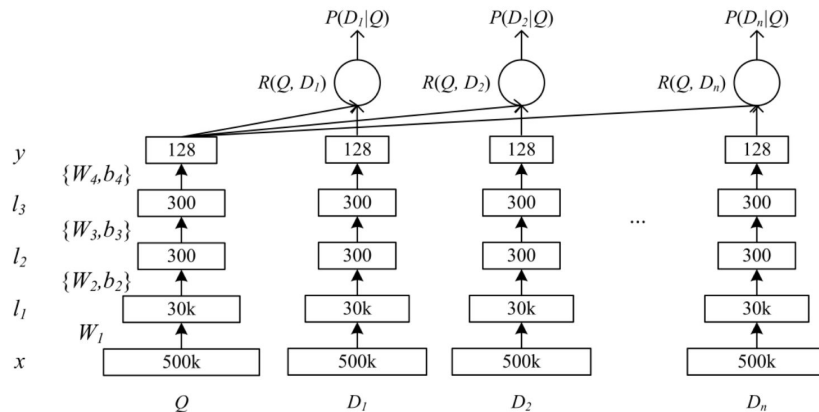
Relevance measured
by cosine similarity

Semantic feature

Multi-layer non-
linear projection

Word Hashing

Term Vector

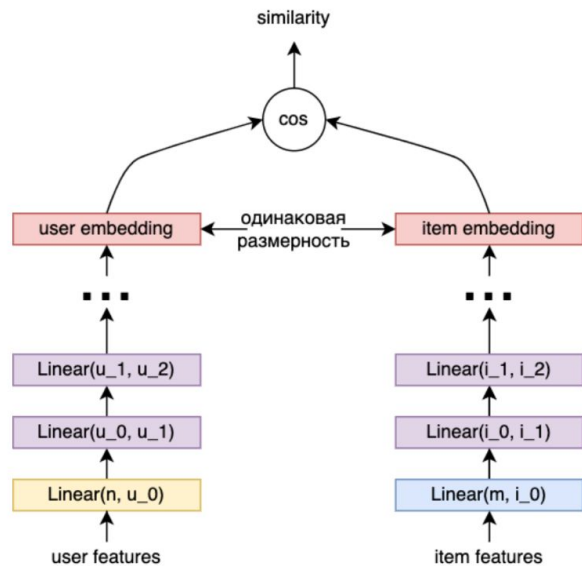


$$R(Q, D) = \text{cosine}(y_Q, y_D) = \frac{y_Q^T y_D}{\|y_Q\| \|y_D\|}$$

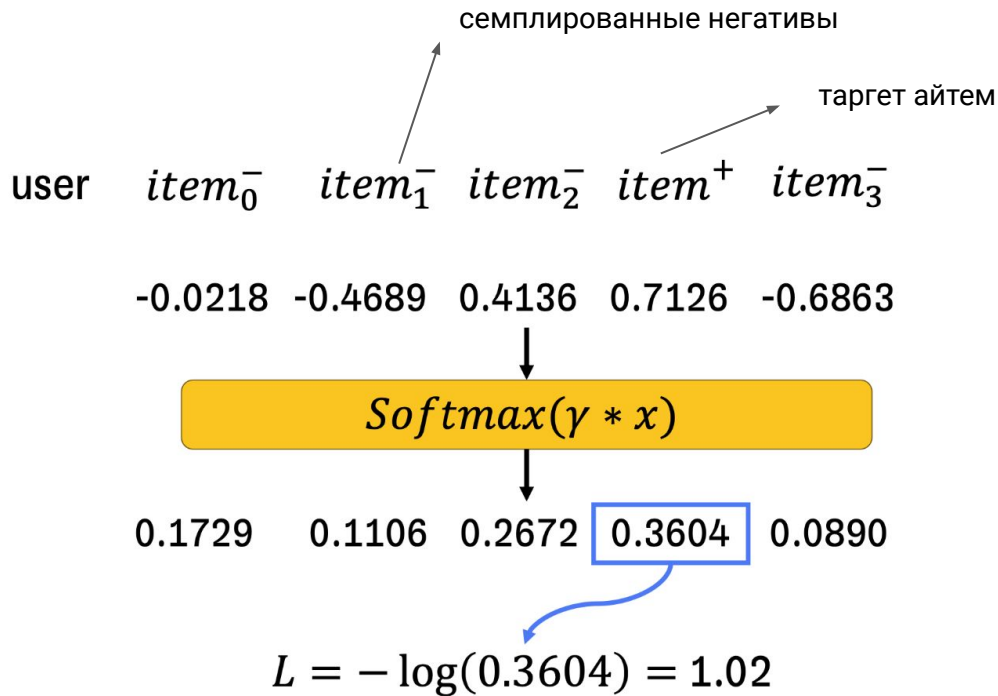
$$P(D|Q) = \frac{\exp(\gamma R(Q, D))}{\sum_{D' \in \mathcal{D}} \exp(\gamma R(Q, D'))} \quad \gamma - \text{smoothing factor}$$

$$L(\Lambda) = -\log \prod_{(Q, D^+)} P(D^+|Q) \quad \begin{array}{l} D^+ - \text{позитивный пример} \\ D^- - \text{негативные примеры} \\ \mathcal{D} = D^+ \cup D^- \end{array}$$

Deep Similarity Similarity Model (DSSM)

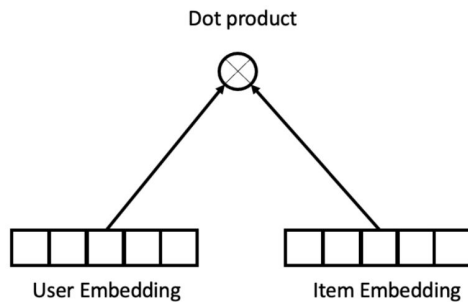


могут иметь общие веса, а могут
быть и разными нейросетями

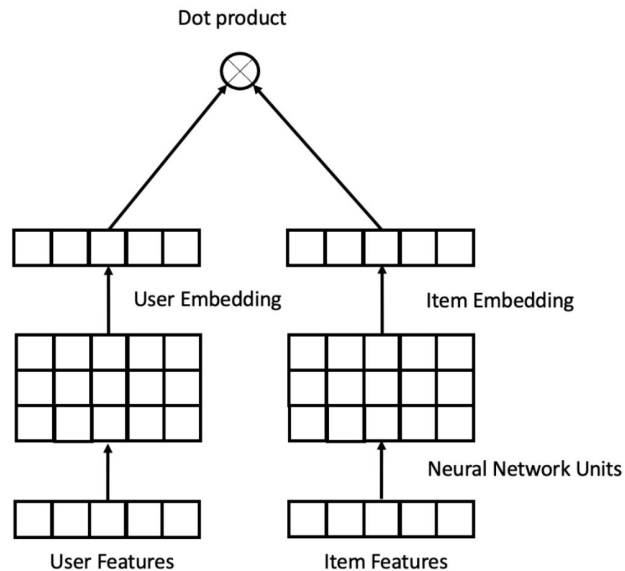


DSSM vs MF

a) Matrix Factorization task



b) DSSM task



Deep & Cross Network

СТАТЬЯ

2.2 Cross Network

The key idea of our novel cross network is to apply explicit feature crossing in an efficient way. The cross network is composed of cross layers, with each layer having the following formula:

$$\mathbf{x}_{l+1} = \mathbf{x}_0 \mathbf{x}_l^T \mathbf{w}_l + \mathbf{b}_l + \mathbf{x}_l = f(\mathbf{x}_l, \mathbf{w}_l, \mathbf{b}_l) + \mathbf{x}_l, \quad (3)$$

THEOREM 3.1. Consider an l -layer cross network with the $i+1$ -th layer defined as $\mathbf{x}_{i+1} = \mathbf{x}_0 \mathbf{x}_i^T \mathbf{w}_i + \mathbf{x}_i$. Let the input to the network be $\mathbf{x}_0 = [x_1, x_2, \dots, x_d]^T$, the output be $g_l(\mathbf{x}_0) = \mathbf{x}_l^T \mathbf{w}_l$, and the parameters be $\mathbf{w}_i, \mathbf{b}_i \in \mathbb{R}^d$. Then, the multivariate polynomial $g_l(\mathbf{x}_0)$ reproduces polynomials in the following class:

$$\left\{ \sum_{\alpha} c_{\alpha}(\mathbf{w}_0, \dots, \mathbf{w}_l) x_1^{\alpha_1} x_2^{\alpha_2} \dots x_d^{\alpha_d} \mid 0 \leq |\alpha| \leq l+1, \alpha \in \mathbb{N}^d \right\},$$

слой, описывающий сумму всевозможных полиномов до некоторой степени (кол-во слоев) между входными фичами

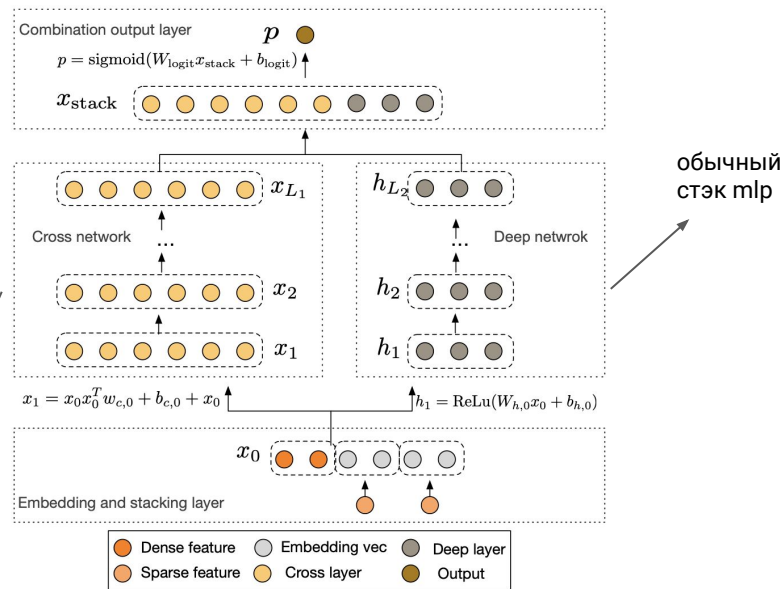


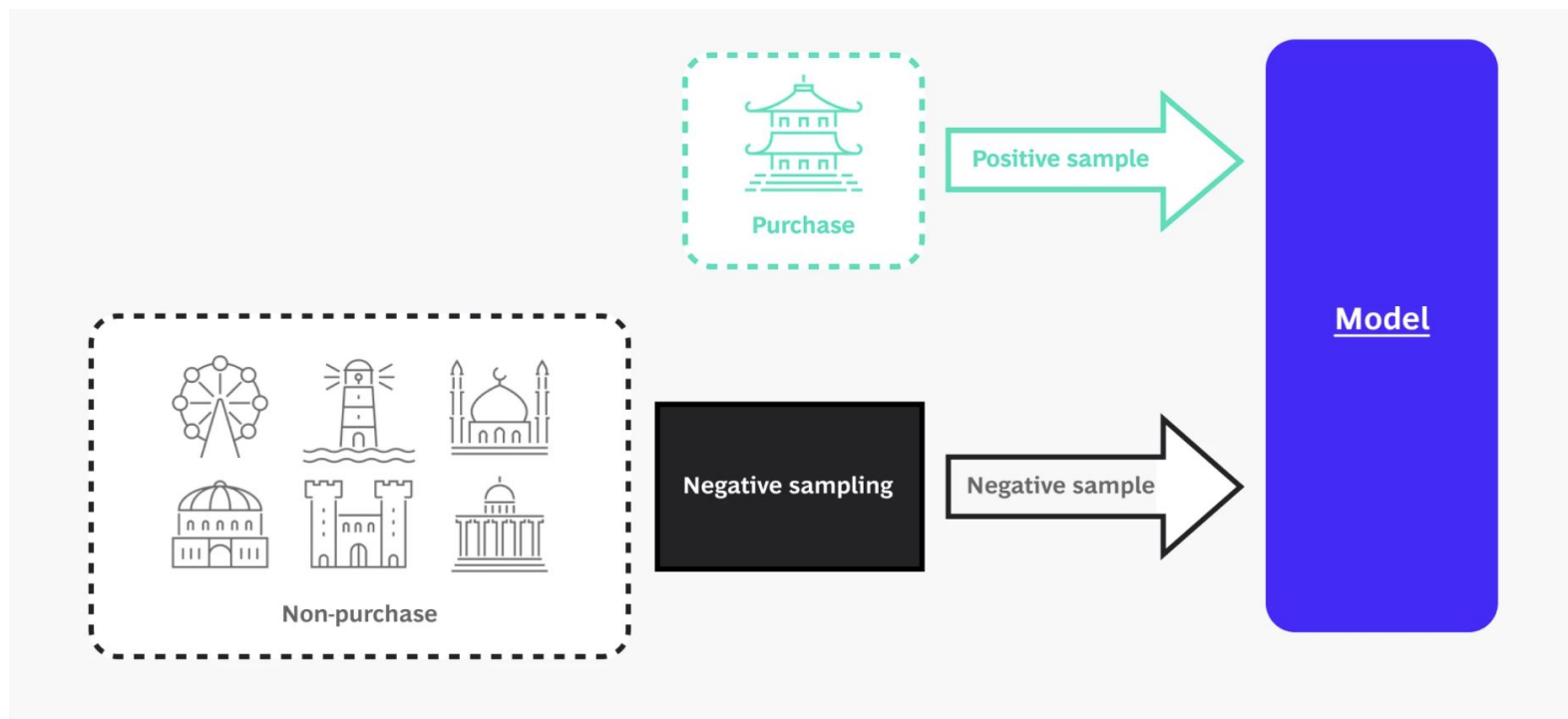
Figure 1: The Deep & Cross Network

Обучение нейросети

Процесс обучения модели для рекомендаций



Процесс обучения модели для рекомендаций



Процесс обучения модели для рекомендаций

Есть несколько подходов к семплированию негативов:

1. **Равномерное семплирование** - каждый объект имеет одинаковую вероятность попасть в качестве негатива.
2. **Частотное семплирование** - семплируем объекты исходя из их встречаемости в датасете. Частным случаем является **Popularity sampling**.
3. **InBatch sampling** - не делается никакого семплирования, а все остальные строки (позитивы) в батче расцениваем как негативы для текущей строки.
4. **Hard negative mining** - учимся только на негативах, с которыми наша модель плохо справляется.

BCE (Pointwise loss example)

вероятность того, что пользователь провзаимодействует с айтемом

$$\mathcal{L}_{\text{BCE}} = -\frac{1}{|I_k^-| + 1} \left(\log(\sigma(s_{i^+})) + \sum_{i \in I_k^-} \log(1 - \sigma(s_i)) \right)$$

кол-во семплированных негативов + таргет, с которым взаимодействовал пользователь

$$\hat{p}_i = \sigma(s_i) = \frac{1}{1 + e^{-s_i}}$$

предсказание модели (dot score, etc)

BPR (Pairwise loss example)

In formal, the training data is constructed by tuples in the form of (u, i, j) , which represents that the user u prefers the item i over the item j . The Bayesian formulation of BPR which aims to maximize the posterior probability is given below:

$$p(\Theta | >_u) \propto p(>_u | \Theta)p(\Theta) \quad (21.5.1)$$

Where Θ represents the parameters of an arbitrary recommendation model, $>_u$ represents the desired personalized total ranking of all items for user u . We can formulate the maximum posterior estimator to derive the generic optimization criterion for the personalized ranking task.

$$\begin{aligned} \text{BPR-OPT} &:= \ln p(\Theta | >_u) \\ &\propto \ln p(>_u | \Theta)p(\Theta) \\ &= \ln \prod_{(u,i,j \in D)} \sigma(\hat{y}_{ui} - \hat{y}_{uj})p(\Theta) \\ &= \sum_{(u,i,j \in D)} \ln \sigma(\hat{y}_{ui} - \hat{y}_{uj}) + \ln p(\Theta) \\ &= \sum_{(u,i,j \in D)} \ln \sigma(\hat{y}_{ui} - \hat{y}_{uj}) - \lambda_{\Theta} \|\Theta\|^2 \end{aligned} \quad (21.5.2)$$

where $D \stackrel{\text{def}}{=} \{(u, i, j) \mid i \in I_u^+ \wedge j \in I \setminus I_u^+\}$ is the training set, with I_u^+ denoting the items the user u liked, I denoting all items, and $I \setminus I_u^+$ indicating all other items excluding items the user liked. \hat{y}_{ui} and \hat{y}_{uj} are the predicted scores of the user u to item i and j , respectively. The prior $p(\Theta)$ is a normal distribution with zero mean and variance-covariance matrix Σ_{Θ} . Here, we let $\Sigma_{\Theta} = \lambda_{\Theta} I$.

InfoNCE / XentLoss / SSM (Sampled Softmax Loss) (Listwise loss example)

$$\hat{p}_i = \text{SSM}(s_i, I_k^-) = \frac{e^{s_i}}{e^{s_{i^+}} + \sum_{j \in I_k^-} e^{s_j}}$$
$$\mathcal{L}_{SSM} = - \sum_{i \in \{I_k^- \cup i^+\}} y(i) \log(\hat{p}_i) = -\log(\text{SSM}(s_{i^+}))$$

скор взаимодействия пользователя с таргет айтемом

множество негативов == айтемов, с которыми пользователь не взаимодействовал

множество семплированных негативов + таргет айтем

Вопросы?



Ставим “+”,
если вопросы есть



Ставим “-”,
если вопросов нет

Практика

Вопросы?



Ставим “+”,
если вопросы есть



Ставим “-”,
если вопросов нет

Рефлексия

Рефлексия



С какими впечатлениями уходите с вебинара?



Как будете применять на практике то, что узнали на вебинаре?

Цели вебинара

К концу занятия вы сможете

1. Изучить какие нейросетевые подходы применяют на практике
2. Понимать основные тренды развития нейросетевых моделей для рекомендаций

**Заполните, пожалуйста,
опрос о занятии
по ссылке в чате**