



BLURM

Dating-app



**Необходимость
знакомства**



**Нестабильность
западных решений**



**Низкая эффективность
традиционных шаблонов**

Решение



**Неопределенность
последующего свайпа
увеличивает количество
совпадений**

**Приложение для
знакомств, основанное
на соревновательной
системе**



Функция просмотра ленты мероприятий

*Увеличивает аудиторию за счет охвата аудитории афиш-приложений

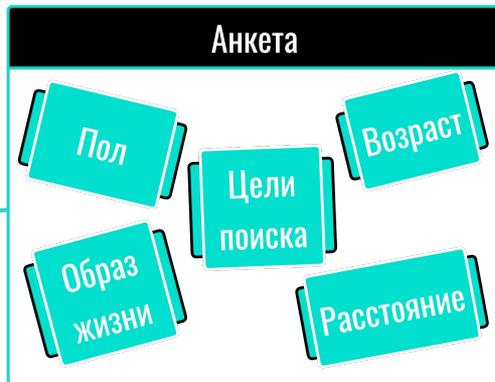
Решение



Прототип интерфейса



Находим
рекомендации



Код на Python, модель Рекомендательной системы

```
class BERTRecommender(SequentialRecommender):  
    def __init__(self, config, training_dataset, test_dataset):  
        super(BERTRecommender, self).__init__(config, training_dataset, test_dataset)  
        block_num = int(config['-n_blocks'])  
        drop_rate = float(config['-drop_rate'])  
        head_num = int(config['-n_heads'])  
        self.aug_rate = float(config['-mask_rate'])  
        self.model = BERT(self.data, self.emb_size, self.max_len, block_num, head_num, drop_rate)  
  
    def train(self, dataset):  
        model = self.model.cuda()  
        optimizer = torch.optim.Adam(model.parameters(), lr=self.learning_rate)  
        for epoch in range(self.maxEpoch):  
            model.train()  
            for n, batch in enumerate(dataset):  
                seq, pos, y, neg_idx, seq_len = batch  
                aug_seq, masked, labels = self.item_mask_for_bert(seq, seq_len, self.aug_rate, self.data.item_num + 1)  
                seq_emb = model.forward(aug_seq, pos)  
                rec_loss = self.calculate_loss(seq_emb, masked, labels)  
                batch_loss = rec_loss + l2_reg_loss(self.reg, model.item_emb)  
                optimizer.zero_grad()  
                batch_loss.backward()  
                optimizer.step()
```

Технология



Ранжируем





TAM ▶

24 млрд. Р

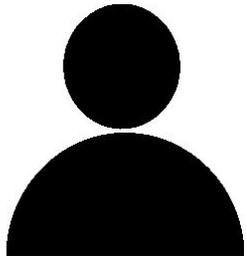
SAM ▶

6 млрд. Р

SOM ▶

1 млрд. Р

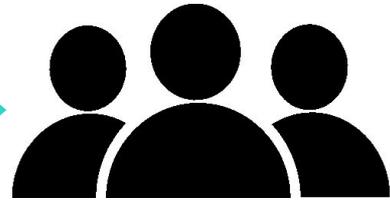
Бизнес-модель



B2C



B2B



**Платная подписка для
аккаунта-пользователя**

**Платная подписка для
аккаунта-мероприятия**

**Функционал платной подписки
для одного мероприятия**

**Платная подписка для
аккаунта-мероприятия**

Road map



Разработана концепция, макет
интерфейса приложения

05.2023

Привлечение специалистов в отдел
разработки, Создание приложения с
полным функционалом

11.2024

Разработан сайт, создан
прототип приложения

05.2024

Маркетинг приложения,
улучшение и подбора

01.2025



Планируемое разделение бюджета

- ❑ 350 000 - Конфигурация Сервера для ИИ
- ❑ 30 000 - Конфигурация Сервера платформы
- ❑ 300 000 - Зарплата разработчиков ИИ и Mobile-разработчиков

*Остальные расходы в заявке

Доработать текущее приложение и платформу для работы на территории России

Разработать собственный ИИ для реализации ранжирования пользователей

Реализовывать рекламную кампанию, привлекая афиши в приложение



Шелепов Павел
Основатель, CEO

Team

Контакты:
pshelepov@yandex.ru
Tg: @stonexix



Санников Вадим
Разработчик, CTO