

3. Техническая документация Raspberry Pi [Электронный ресурс] – 2021. Режим доступа: <https://www.raspberrypi.org/products/raspberry-pi-4-model-b/>. – Дата доступа: 29.03.2021.

**В. А. Рубин**

(ГГУ имени Ф. Скорины, Гомель)

Науч. рук. **С. П. Жогаль**, канд. физ.-мат. наук, доцент

## **СТРУКТУРИРОВАННОЕ НЕЙРОННОЕ ОБУЧЕНИЕ В TFX**

Tensor Flow Extended (далее по тексту TFX) – это бесплатная платформа с открытым исходным кодом, которая предназначена для создания готовых к работе конвейеров машинного обучения. Внутри своего ядра TFX предлагает огромное количество подходов для обработки данных.

Neural Structured Learning (NSL) – это библиотека в TensorFlow, которую возможно использовать при обучении нейронных сетей, сигнал которых структурирован. Она может обрабатывать ввод двумя различными способами: как явный неявный граф и, соответственно, как явный граф, соседи которого динамически генерируются в процессе обучения модели. NSL с явным графом часто используют для предобучения нейронным графам, а NSL с неявным графом используют при состязательном обучении. Эти подходы реализованы в виде формы регуляризации в схеме NSL. Результатом этого является влияние только на процесс обучения и неизменность процесса обслуживания модели.

Вкратце, процесс построения модели с регуляризацией графа можно разбить на три шага:

1. Если граф еще недоступен – построить его.
2. Использовать особенности входного набора данных и граф для увеличения обучающей выборки.
3. Применить расширенные данные из предыдущего шага для регуляризации графа для данной модели.

К сожалению, сразу применить эти шага для TFX-конвейера нельзя, но TFX позволяет настраивать компоненты. И при помощи этой настройки можно обеспечить обработку своих собственных TFX-конвейеров. Для того, чтобы построить ругуляризованную модель графа при помощи TRX, можно воспользоваться настраиваемыми плагинами TFX.

Чтобы проиллюстрировать пример конвейера TFX с NSL, можно рассмотреть задачу классификации тональности текста.

Для построения в TFX регуляризованной модели NSL для этой задачи, необходимо определить три пользовательских компонента. Для краткости были опущены компоненты, которые обычно идут после компонента Trainer, такие как Evaluator, Pusher и т.д.

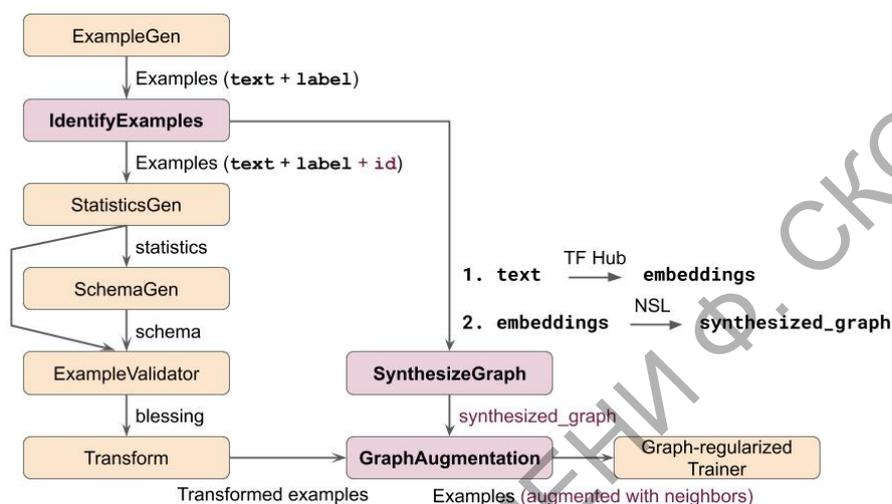


Рисунок 1 – Пример конвейера TFX для классификации текста с использованием регуляризации графа

На этом рисунке только пользовательские компоненты (выделены розовым цветом) и компонент тренера с регуляризацией графиков имеют логику, связанную с NSL. Стоит отметить, что показанные здесь пользовательские компоненты являются только иллюстративными, и можно построить функционально эквивалентный конвейер другими способами.

Как упоминалось в самом начале, еще одним аспектом нейронного структурированного обучения является состязательное обучение, когда вместо использования явных соседей из графа для регуляризации неявные соседи создаются динамически и враждебно, чтобы запутать модель.

В итоге можно заключить, что регуляризация с использованием состязательных примеров – это эффективный способ повысить надежность модели. Состязательное обучение с использованием NSL можно легко интегрировать в конвейер TFX. Для этого не требуются какие-либо пользовательские компоненты, и необходимо обновить только обучающий компонент.