

ТЕХНОЛОГИЧЕСКАЯ КАРТА ЗАНЯТИЯ

Тема занятия: Transfer learning.

Аннотация к занятию: обучающиеся познакомятся с понятием transfer learning, научатся применять transfer learning для обучения нейросетей, изучат подход дообучения нейросетей как реализацию переноса знаний.

Цель занятия: формирование у обучающихся представления о переносе знаний из одной нейросети в другую — transfer learning.

Задачи занятия:

- познакомить обучающихся с понятием transfer learning;
- научиться применять transfer learning для обучения нейросетей;
- изучить один из подходов к реализации переноса знаний — fine-tuning;
- изучить алгоритм дообучения нейросетей.

Ход занятия

Этап занятия	Время	Деятельность педагога	Комментарии, рекомендации для педагогов
Организационный этап	5 мин.	<p>Добрый день! Давайте вспомним, что делают нейросети, когда в обучающем датасете мало данных?</p> <p>Возможный ответ обучающихся: нейросети часто переобучаются.</p> <p>Особенно это верно, когда задача сложная. Пример такой задачи — классификация изображений опухолей. В медицине данных для обучения нейросетей всегда очень мало: их сложно собрать. Задача при этом тоже довольно сложная: опухоли бывают совершенно разных форм и размеров.</p> <p>Другой пример сложной задачи, для которой очень непросто собрать достаточно данных — машинный перевод с малораспространённых языков. Чтобы обучить модель для машинного перевода с одного языка на другой, нужно огромное количество параллельных предложений. Параллельные предложения — это пары: предложение на одном языке и его точный перевод на второй язык. Чаще всего такие предложения собираются из интернета. Например, из</p>	<p>Приветствие. Создание в классе атмосферы психологического комфорта</p>

		<p>«Википедии», в которой одна и та же статья часто переведена на много разных языков. Но если язык не сильно распространён, то и текстов в интернете на нем будет немного. И достаточное количество предложений на этом языке и их переводов на другой язык набрать не получится.</p> <p>Вот такая частая проблема. Она возникает повсеместно, не только в медицинском домене и машинном переводе. Задачи машинного обучения становятся всё сложнее, нейросети для их решения становятся всё глубже, и это требует всё большего числа данных для обучения. В общем, в машинном обучении данных всегда мало.</p> <p>Что же делать? Один из способов бороться с этой проблемой — это transfer learning, перенос знаний от одной модели машинного обучения к другой.</p>	
<p>Постановка цели и задач занятия. Мотивация учебной деятельности обучающихся</p>	<p>7 мин.</p>	<p>Какая сегодня тема урока?</p> <p>Возможные ответы обучающихся:</p> <ul style="list-style-type: none"> • обучение нейросетей; • transfer learning. <p>Тема урока — «Transfer learning».</p>	<p>Способствовать обсуждению мотивационных вопросов</p>

		<p>Как вы думаете, какие задачи мы сможем сегодня решить?</p> <p>Возможные ответы обучающихся:</p> <ul style="list-style-type: none"> • узнаем, что такое transfer learning; • увидим, как transfer learning применяется для обучения нейросетей. <p>Мы узнаем:</p> <ul style="list-style-type: none"> • что такое transfer learning и как он применяется для обучения нейросетей; • какой алгоритм применяется для обучения нейросетей. 	
<p>Изучение нового материала</p>	<p>50 мин.</p>	<p>Вопрос для обсуждения В чём заключается идея переноса знаний?</p> <p>Ответы обучающихся: Идея переноса знаний следующая: пусть у нас есть задача машинного обучения, для которой мало данных. Например, задача классификации опухолей. И пусть у нас есть другая, но похожая задача, для которой у нас есть большее количество данных. Например, задача классификации опухолей, но другого типа или в другом органе. Так как для второй задачи у нас есть большое количество данных, то мы можем успешно обучить нейросеть решать эту задачу. Затем мы можем неким образом использовать знания, полученные этой обученной нейросетью, чтобы обучить другую</p>	

нейросеть, которая будет решать первую задачу, для которой у нас мало данных. В процессе обучения под вторую задачу нейросеть выучила полезные паттерны, полезные фильтры свёрточных слоёв, которые выделяют полезную информацию для классификации опухолей. Так как первая задача идейно похожа на вторую — это тоже классификация опухолей — мы можем использовать знания, выученные нейросетью для второй задачи, чтобы обучить нейросеть лучше решать первую задачу. Знания, выученные нейросетью при решении второй задачи, помогут компенсировать отсутствие большого числа данных для первой задачи и помочь нейросети не переобучиться под первую задачу.

В реальной жизни transfer learning тоже работает. Например, человеку, который знает английский язык, будет проще выучить другой, похожий язык, например, испанский. Ему понадобится меньше времени и обучающих примеров. А всё потому, что структуры английского и испанского языков во многом похожи: в них есть похожие слова, грамматические и лексические структуры.

Это общая идея переноса знаний: использовать знания, полученные одной нейросетью, для обучения нейросети под другую задачу. По сути, это и есть

перенос знаний: мы передаём знания одной сети другой.

Подходов к реализации переноса знаний множество. Мы обсудим один из них. Он называется fine-tuning, или, по-русски, дообучение нейросети.

Давайте разбираться, что же это такое. Пусть у нас есть задача классификации картинок диких животных на десять классов. И пусть у нас для этой задачи очень мало данных. Конечно, в реальности картинок диких животных в интернете можно найти сколько угодно, но давайте представим, что мы живем без интернета, и чтобы собрать датасет, нам нужно самим фотографировать диких животных. А сделать это не так-то просто.

Если мы будем обучать нейросеть на нашем небольшом датасете с нуля, она, скорее всего, переобучится. Поэтому мы сделаем вот что.

Возьмём нейросеть и обучим её на датасете ImageNet. Можно сказать, что мы предобучили нейросеть на датасете ImageNet.

Потому что теперь мы берём нашу предобученную нейросеть и дообучаем её на датасете диких животных. То есть берём нейросеть, которая в

процессе обучения на ImageNet получила полезные знания о том, как устроены объекты окружающего мира, какие формы у них есть, и затем обучаем её под нашу задачу классификации диких животных. Мы надеемся, что благодаря знаниям, полученным из ImageNet, нейросеть сможет лучше решить задачу классификации диких животных даже в условиях небольшого количества данных.

Правда, вот так просто ничего работать не будет. Тут есть два нюанса.

Первый — это то, что в датасете диких животных десять классов. А в ImageNet была тысяча. Так как изначально мы обучали нашу нейросеть на ImageNet, то в последнем слое нашей сети была тысяча нейронов. А если мы хотим дообучить эту же нейросеть на датасете диких животных, то в её последнем слое должно быть уже десять нейронов.

Решение здесь простое. Перед дообучением сети удалим последний слой сети и заменим его на новый, в котором будет десять нейронов на выходе. Изначально перед дообучением веса этого слоя будут случайными, но в процессе дообучения они подстроятся под остальные веса сети.

Это был первый нюанс, мы успешно с ним справились.

Второй нюанс состоит вот в чём: так как мы дообучаем всю нейросеть целиком, то она всё равно переобучится. Какая разница, какие полезные паттерны научились выделять фильтры слоёв нейросети на ImageNet, если затем мы всё равно дообучаем сеть только на датасете диких животных? Все веса всех фильтров забудут, что они выучили на ImageNet, и перестроятся под датасет диких животных. И переобучатся, так как в датасете мало данных.

Решение здесь такое. Давайте перед дообучением заморозим несколько первых слоёв нейросети. Заморозим — это значит, что фильтры первых свёрточных слоёв нейросети не будут обновляться, когда мы будем дообучать сеть на диких животных. Они останутся такими, какими были после обучения на ImageNet.

Какой смысл в заморозке слоёв? Мы помним, что при обучении свёрточной нейросети начальные слои реагируют на более простые, низкоуровневые паттерны на картинках, а дальние слои — на более высокоуровневые, составные паттерны. Простые паттерны присутствуют практически на всех картинках, какую бы задачу мы ни решали. Поэтому та информация, которую первые слои нейросети научились выделять в ходе обучения на ImageNet, будет полезна и тогда, когда мы нейросеть будем дообучать на диких животных. А вот последние

свёрточные слои учились выделять информацию о наличии сложных паттернов, которые были на картинках ImageNet. И совсем не все из них будут актуальны для классификации диких животных. Поэтому дальнейшие сверточные слои логично дообучить на новом датасете, чтобы они научились выделять те сложные паттерны, которые помогут классифицировать именно диких животных.

Можно сказать, что первые свёрточные слои учатся понимать основу, базовую структуру объектов на картинках. И эти знания о базовой структуре объектов мы потом переносим на другую задачу. Нейросети теперь нужно обучаться под задачу классификации диких животных не с нуля: ей только нужно научиться собирать паттерны десяти диких животных из известных ей форм: тех, которые выделяют первые предобученные свёрточные слои. Это похоже на пример с английским и испанским языками: человеку проще учить испанский, если он знает английский, потому что он переносит знания о базовой структуре английского языка на испанский язык. Ему остается только выучить дополнительные, высокоуровневые особенности испанского.

Кроме всего этого, заморозка слоёв уменьшает количество обучаемых параметров сети. А это

уменьшает риск переобучения сети на маленьком датасете.

Итак, давайте повторим алгоритм дообучения нейросети.

Пусть у нас есть задача классификации на N классов, и данных для этой задачи мало. Тогда мы делаем следующее.

Первое. Берём датасет с достаточным количеством данных и обучаем на нём нейросеть. В нашем примере это ImageNet.

Второе. Удаляем последний слой нейросети, добавляем новый с N нейронами.

Третье. Замораживаем несколько первых свёрточных слоёв.

И четвёртое: дообучаем нейросеть с замороженными слоями на нашем маленьком датасете.

Вопрос для обсуждения

Как понять, сколько слоёв замораживать?

Ответы обучающихся:

Ответ такой: зависит от того, насколько мало у вас данных для обучения и насколько похожи две задачи: та, на которой вы предобучаете сеть, и та, на которой вы её дообучаете.

Чем менее похожи две задачи, тем меньше слоев нужно замораживать. Потому что если две задачи не похожи, то есть формы объектов на картинках разные, то при предобучении фильтры большого числа слоёв научатся выделять информацию, которая не поможет на новой задаче. И тогда стоит заморозить только первые несколько слоёв: те, которые научились выделять низкоуровневую информацию, которая есть на картинках обоих датасетов.

Чем меньше у вас данных для дообучения, тем больше слоёв нужно замораживать. Потому что если у нейросети останется много слоёв, которые будут обучаться, она переобучится.

Точно сказать в каждом из случаев, сколько слоёв замораживать, а сколько оставить, нельзя. Нужно экспериментировать. Заморозить три слоя, дообучить, посмотреть на результат. Если вышло плохо, попробовать заморозить другое количество слоёв. Например, если сеть переобучилась, попробовать заморозить больше слоёв. И так далее. Со временем, с

		<p>практикой вам будет проще ориентироваться, сколько примерно слоёв нужно замораживать для разных задач.</p> <p>Ну и последнее про дообучение. Если у вас есть задача с маленьким количеством данных и вы хотите использовать дообучение, то встаёт вопрос, на каком датасете сеть предобучать. Не всегда получается найти очень похожий датасет с большим количеством данных. В этом случае ImageNet — отличный вариант. В ImageNet огромное количество классов и различных объектов, а слои нейросети при обучении на ImageNet выучивают огромное количество различных паттернов. Скорее всего, среди них будут те паттерны, которые будут полезны вам в вашей задаче, для которой у вас мало данных.</p> <p>Использование предобученных на ImageNet нейросетей — это очень частая практика. Поэтому во фреймворках, таких как PyTorch, есть возможность скачать готовые нейросети, предобученные на ImageNet.</p>	
<p>Закрепление изученного материала</p>	<p>15 мин.</p>	<p>Вопросы для обсуждения</p> <ul style="list-style-type: none"> • Какая проблема возникает, если обучающих данных мало? • Что такое transfer learning? • В чём заключается один из подходов к transfer Learning: дообучение сети? 	<p>Педагог организует беседу по вопросам</p>

<p>Этап подведения итогов занятия (рефлексия)</p>	<p>8 мин.</p>	<p>Вопросы для обсуждения</p> <ul style="list-style-type: none"> • Чему я научился? • С какими трудностями я столкнулся? • Каких знаний мне не хватает для более глубокого понимания изученного материала? • Достиг ли я поставленных целей и задач? 	<p>Педагог способствует размышлению обучающихся над вопросами</p>
<p>Информация о домашнем задании, инструктаж по его применению</p>	<p>5 мин.</p>	<p>-</p>	

Рекомендуемые ресурсы для дополнительного изучения:

1. Что такое Transfer learning. [Электронный ресурс] – Режим доступа: <https://python-school.ru/wiki/transfer-learning/>.
2. Transfer Learning: как быстро обучить нейросеть на своих данных. [Электронный ресурс] – Режим доступа: <https://habr.com/ru/company/binarydistrict/blog/428255/>.
3. Как transfer learning работает для задач с медицинскими снимками. [Электронный ресурс] – Режим доступа: <https://neurohive.io/ru/novosti/kak-transfer-learning-ispolzujut-dlya-zadach-s-medicinskimi-snimkami/>.