

Задача 2D сегментации для определения разломов по сейсмическим данным

Сизова Софья

Май 2021

1 Введение

Задача интерпретации сейсмических данных для определения разломов породы занимает важное место в проблемах нефтеразведки. Геологические разломы (или разрывы) дают важную информацию о формировании подземных пластов, а также возможном нахождении нефте- и газоносных резервуаров. Они представляют собой нарушение сплошности горных пород, без смещения (трещина) или со смещением пород по поверхности разрыва. Эти смещения пластов могут быть хорошо заметны на сейсмическом разрезе, например на рисунке 1. Сейсмический метод

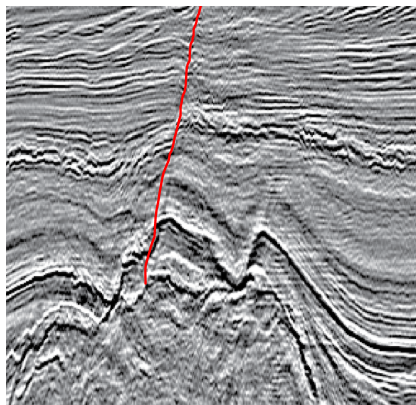


Рис. 1: Пример сейсмического разреза с отмеченным разломом

георазведки основан на разной скорости распространения сейсмических волн в различных средах, эта неоднородность отражается на изображениях и позволяет идентифицировать слои. В настоящее время интерпретация и разметка сейсмических данных чаще всего проводится традиционным способом - вручную. Этот метод имеет ряд недостатков. Во-первых,

даже опытному интерпретатору может понадобиться длительное время (от нескольких недель до месяца) для тщательной обработки данных. Во-вторых, человеческий фактор может привести к субъективности и неоднозначности интерпретации, а также ошибкам. Одним из альтернативных и перспективных методов определения разломов является 2d сегментация с помощью нейронных сетей.

2 Постановка задачи

Задача выделения разломов на сейсмических данных рассматривается как задача 2d сегментации, в которой мы ищем маску изображения, содержащую 1 на месте разломов и 0 в остальной области. В данной работе применяется модель U-Net, на вход которой подаются 2d изображения сейсмических данных в серых тонах. На выходе мы ожидаем получение маски, которая выделяет разломы на изображении.

3 Dataset

В качестве датасета используется открытая и размеченная база данных faultSeg, в которой содержится 200 синтезированных пар (изображение, маска). Они представляют собой кубы размером (128, 128, 128). Однако в данной работе 3D кубы редуцируются до 2D изображений: для каждого 3D изображения выделяются все возможные сечения по осям x и y . Такой подход увеличивает размер обучающей выборки и упрощает слои модели: мы используем 2D свертку, pooling и проч. вместо 3D. Итого мы получаем 51200 пар: изображение + маска. Один из элементов выборки вместе с соответствующей маской приведен на изображении ниже.

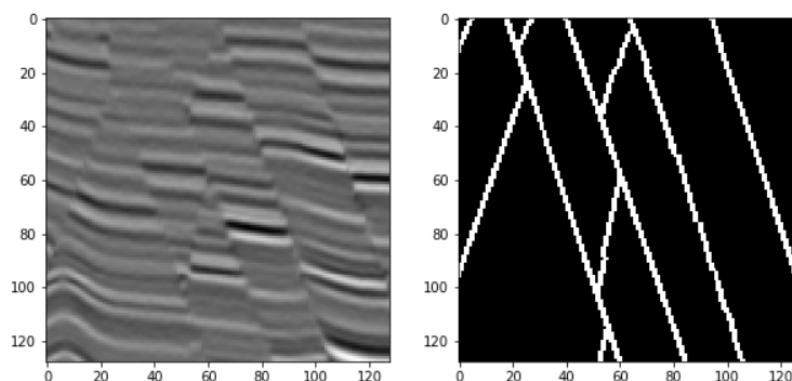


Рис. 2: Сечение сейсмического куба из тренировочной выборки (слева) и соответствующая маска (справа)

В открытой выборке faultSeg содержится также набор 3D сейсмических кубов для валидации. Мы формируем из него валидационную выборку размеченных изображений, используя также сечения исходного куба.

Для увеличения обучающей выборки мы также используем аугментацию данных. В данной работе применяется размытие со случайным ядром, отражение и поворот. В статье [1] рекомендуется выбирать поворот только на 90, 270 или 180 градусов, чтобы не испортить данные.

Исходная выборка разбивается на train и test множества, в соотношении 80% и 20%.

4 Model U-Net

Модель U-Net считается стандартной архитектурой свёрточных нейронных сетей, которая широко применяется для задач сегментации. Согласно [2] U-Net превосходит многие существующие архитектуры в сегментации изображений, так как требует меньший размер обучающей выборки и показывает более точные результаты.

В данной работе используется упрощенная модификация U-Net, аналогичная той, которую применили в статье [1]. Архитектура симметрична и состоит из сужающегося и расширяющего пути.

Сужающий путь состоит из 3 групп слоёв: два свёрточных слоя с функцией активации "ReLU" + слой "пулинга" с сигмоидом в качестве

функции активации. Слой "MaxPooling2D" с ядром (2,2) уменьшает изображение вдвое. В то же время на каждом слое количество признаков увеличивается (для первого слоя используется 16). Затем применяется еще одна пара сверточных слоёв с 128-ю свойствами. При переходе на расширяющий слой применяется функция UpSampling, являющаяся обратной функции "MaxPooling".

На расширяющем слое мы наоборот увеличиваем размер изображения до исходного, применяя функцию "UpSampling" (являющуюся обратной функции "MaxPooling"), а также вдвое уменьшаем количество признаков для пар сверточных слоёв. Слои модели приведены на изображении 4.

```
inputs = Input(input_size)
conv1 = Conv2D(16, 3, activation='relu', padding='same')(inputs)
conv1 = Conv2D(16, 3, activation='relu', padding='same')(conv1)
pool1 = MaxPooling2D(pool_size=(2,2))(conv1)

conv2 = Conv2D(32, 3, activation='relu', padding='same')(pool1)
conv2 = Conv2D(32, 3, activation='relu', padding='same')(conv2)
pool2 = MaxPooling2D(pool_size=(2,2))(conv2)

conv3 = Conv2D(64, 3, activation='relu', padding='same')(pool2)
conv3 = Conv2D(64, 3, activation='relu', padding='same')(conv3)
pool3 = MaxPooling2D(pool_size=(2,2))(conv3)

conv4 = Conv2D(128, 3, activation='relu', padding='same')(pool3)
conv4 = Conv2D(128, 3, activation='relu', padding='same')(conv4)

up5 = concatenate([UpSampling2D(size=(2,2))(conv4), conv3], axis=-1)
conv5 = Conv2D(64, 3, activation='relu', padding='same')(up5)
conv5 = Conv2D(64, 3, activation='relu', padding='same')(conv5)

up6 = concatenate([UpSampling2D(size=(2,2))(conv5), conv2], axis=-1)
conv6 = Conv2D(32, 3, activation='relu', padding='same')(up6)
conv6 = Conv2D(32, 3, activation='relu', padding='same')(conv6)

up7 = concatenate([UpSampling2D(size=(2,2))(conv6), conv1], axis=-1)
conv7 = Conv2D(16, 3, activation='relu', padding='same')(up7)
conv7 = Conv2D(16, 3, activation='relu', padding='same')(conv7)

conv8 = Conv2D(1, 1, activation='sigmoid')(conv7)
```

Рис. 3: Слои модели U-Net

Задачу сегментации можно интерпретировать как бинарную классификацию, потому что на выходе мы получаем маску с единицами на месте разрезов и 0 в остальной области. Поэтому логично использовать бинарную кросс-энтропию (binary cross-entropy) в качестве функции потерь. Она определяется уравнением (1), где N - это количество пикселей

во входном изображении, y_i - бинарный лэйбл и $p(y_i)$ обозначает соответствующую вероятность предсказания.

$$\mathbb{L} = -\frac{1}{N} \sum_{i=1}^N \left(y_i \cdot \log(p(y_i)) + (1 - y_i) \cdot \log(1 - p(y_i)) \right). \quad (1)$$

Для компиляции модели используется оптимайзер "Adam". Этот метод реализует стохастический метод градиентного спуска и согласно [3] хорошо подходит для больших обучающих выборок, как в нашем случае. Некоторые параметры модели приведены в таблице ниже. Выбор размера батчей и маленького количества эпох обусловлен ограничениями в памяти личного компьютера и значительным вычислительным временем. Тем не менее, точность предсказаний модели даже после 5 эпох достигает значения 94%, что является приемлемым результатом.

Таблица 1: Параметры запуска модели U-Net

| Learning rate | Batch size | Epochs number |
|---------------|------------|---------------|
| 0.001 | 64 | 5 |

5 Результаты

С помощью TensorBoard - удобного инструмента для визуализации процесса обучения, мы оцениваем изменение точности (accuracy) и функции потерь (loss) для тренировочного и валидационного множеств. На рис. 5 изображена эволюция метрики точности для первых 5 эпох для тренировочного (оранжевый) и валидационного (синий) множеств. Начиная с третьей эпохи точность заметно увеличивается и достигает значения 94%.

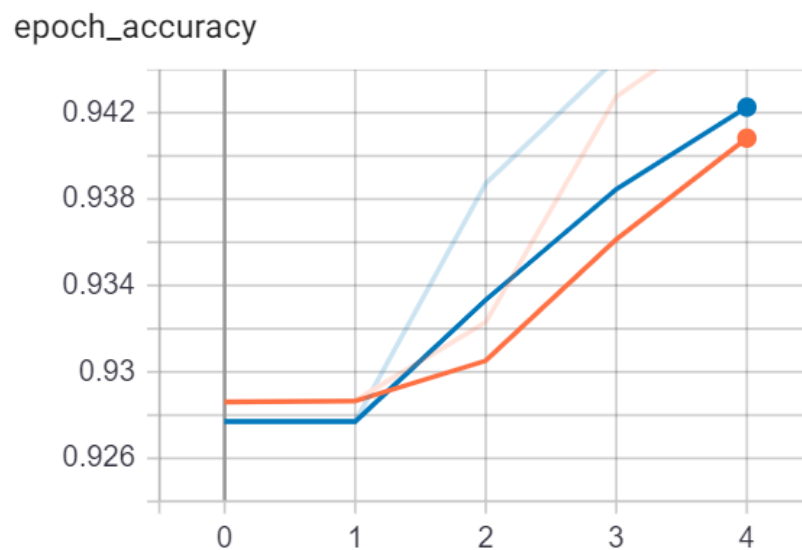


Рис. 4: Эволюция метрики точности для тренировочного (оранжевый) и валидационного (синий) датасетов

Значение функции потерь также убывает с каждой эпохой. К пятой эпохе значение loss-функции для валидационного и тренировочного датасетов близко к 0.17 (рис. 5).

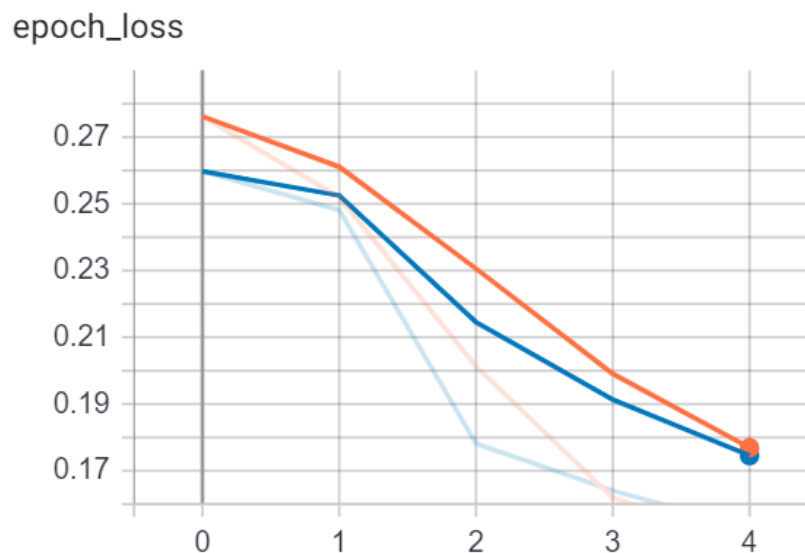


Рис. 5: Эволюция функции потерь для тренировочного (оранжевый) и валидационного (синий) датасетов

Перед отрисовкой предсказанной маски и её сравнением с исходной разметкой мы применяем некоторые преобразования изображения:

- Дилатация с ядром (5,5) для расширения и связывания полос, соответствующих разломам
- Эрозия "erode" для уменьшения полос маски, соответствующих наличию разломов
- Применение бинарного тресхолда для более наглядного отображения разломов

При последовательном применении дилатации и эрозии можно исправить "неточности" предсказаний, а именно убрать некоторые "разрывы" в предсказанных линиях разломов.

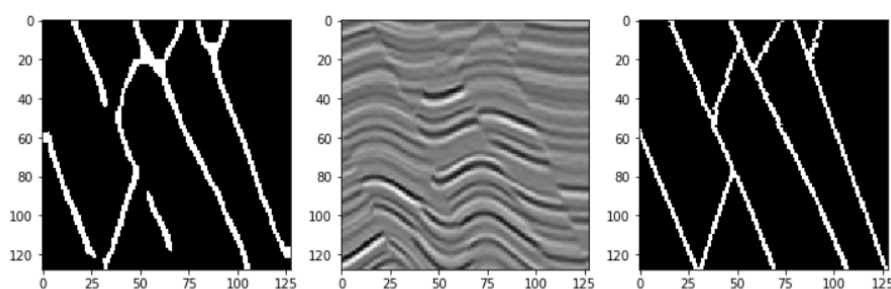


Рис. 6: Сечение сейсмического куба из тестовой выборки (посередине), предсказанная маска (слева), исходная разметка (справа)

На рисунке 6 приведено сравнение предсказанной маски (слева) с исходной разметкой (справа) для одного из сечений сейсмического куба (посередине). Выделенные разломы действительно заметны на сейсмических данных, и никакие кроме задетектированных на изображении посередине не обнаруживаются. Также, зрительно заметна схожесть предсказанной маски с исходной разметкой, однако есть некоторые неточности: разрывы в линиях и более широкие линии разломов.

Для более точного сравнения предсказанной маски с имеющейся разметкой в данной работе была опробована метрика "intersection over union". Она часто применяется для оценки точности предсказаний в задачах детекции и сегментации. На рис.7 приведен код для подсчёта этой метрики.

```
intersection = np.logical_and(target, prediction)
union = np.logical_or(target, prediction)
iou_score = np.sum(intersection) / np.sum(union)
```

Рис. 7: Код для вычисления метрики Intersection over union

Для рис. 6 мы получили $IOU = 0.57$. Однако это значение является приемлемым. Дело в том, что метрика "intersection over unit" хорошо работает для сегментации "больших" объектов на картинке: если нейронная сеть детектирует такой объект со слегка сдвинутыми границами, то значение IOU все равно будет большим. В нашем же случае линии разрезов очень тонкие, и даже незначительное отклонение на пару пикселей от исходной разметки вызывает сильное уменьшение IOU. Более того, в реальных данных толщина разломов обычно незначительна и не представляет интереса для геологов, которых интересует только местоположение этих линий.

В качестве эксперимента я добавила в модель два слоя дропаута, чтобы избежать переобучения, и сравнила полученные результаты с моделью U-Net без дропаута. Эволюция "ассигасу" приведена на изображении ниже. По сравнению с изображением 5 (для модели без дропаута) значение ассигасу ниже, чем при использовании дропаута. Функция потерь (рис.9) также снизилась по сравнению с моделью без дропаута (0.17 и 0.13).

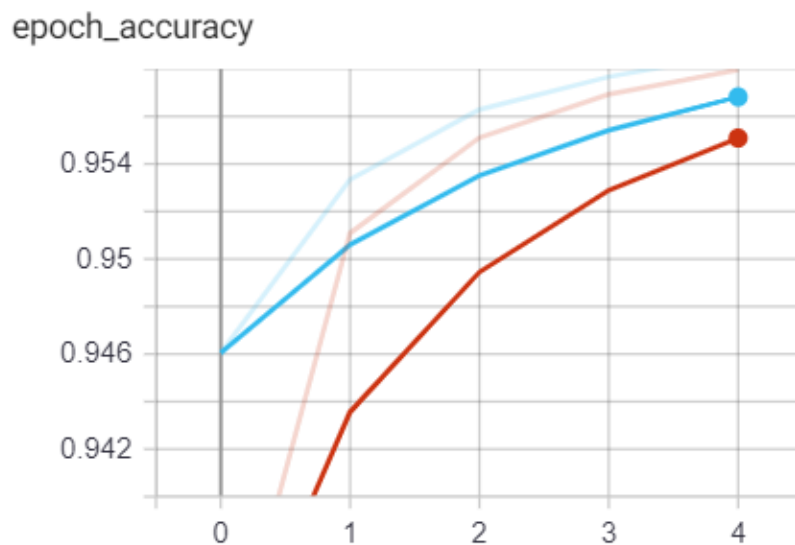


Рис. 8: Эволюция точности для тренировочного (оранжевый) и валидационного (синий) датасетов с использованием Dropout слоёв

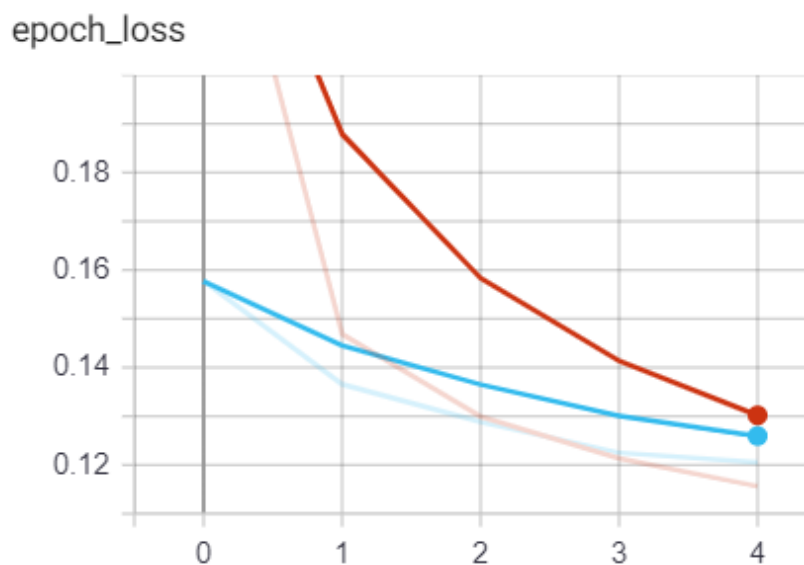


Рис. 9: Эволюция точности потерь для тренировочного (оранжевый) и валидационного (синий) датасетов с использованием Dropout слоёв

Если мы рассмотрим предсказания, полученные с помощью U-Net с дропаутами, (рис. 10), то заметим, что они также оказались точнее. В

этом случае нет разрывов в линиях, предсказанная маска больше похожа на исходную разметку. Даже значение IOU увеличилось: сейчас мы получаем 0.68.

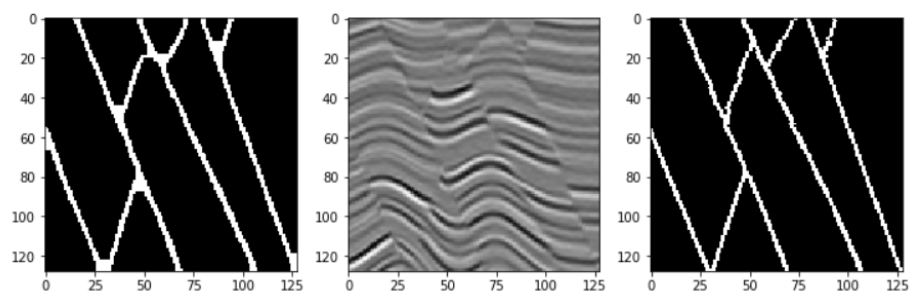


Рис. 10: Сечение сейсмического куба из тестовой выборки (посередине), предсказанная маска (слева) с использованием дропаута, исходная разметка (справа)

Ниже представлены предсказанные маски для другого сечения без дропаута и с его использованием. Сверху подписано значение метрики "intersection over union": 0.54 без дропаута и 0.66 с дропаутом. Предсказанная маска с использованием дропаута заметно лучше: разрывов нет и линии более ровные, что соответствует разломам.

0.5450704225352113

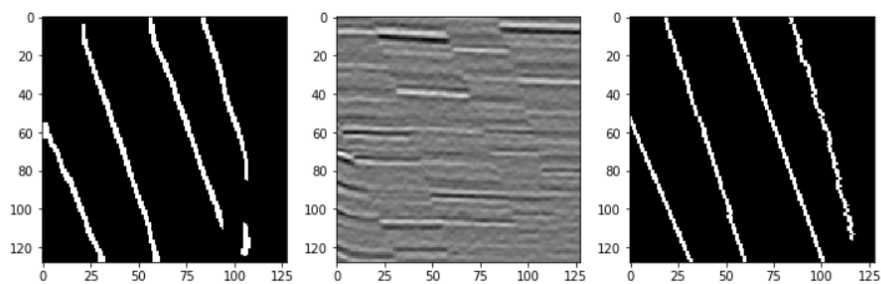


Рис. 11: Сечение сейсмического куба из тестовой выборки (посередине), предсказанная маска (слева) без использования дропаута, исходная разметка (справа)

0.6597774244833068

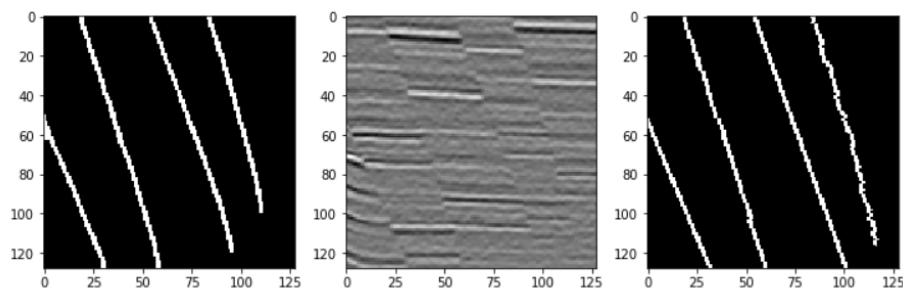


Рис. 12: Сечение сейсмического куба из тестовой выборки (посередине), предсказанная маска (слева) с использованием дропаута, исходная разметка (справа)

6 Сравнение со статьёй

Давайте сравним результаты, полученные для задачи 2d-сегментации, реализованной в данной работе, с результатами задачи 3d-сегментации в статье [1], в которой была также использована архитектура U-Net и в качестве функции потерь использована бинарная кросс-энтропия. На рис. 13 приведена эволюция точности с ростом эпох. На пятой эпохе для тренировочной выборки точность составляла лишь 0.85, тогда как в модели для 2d сегментации (рис.4 и 9) точность уже выше (0.942 и 0.954).

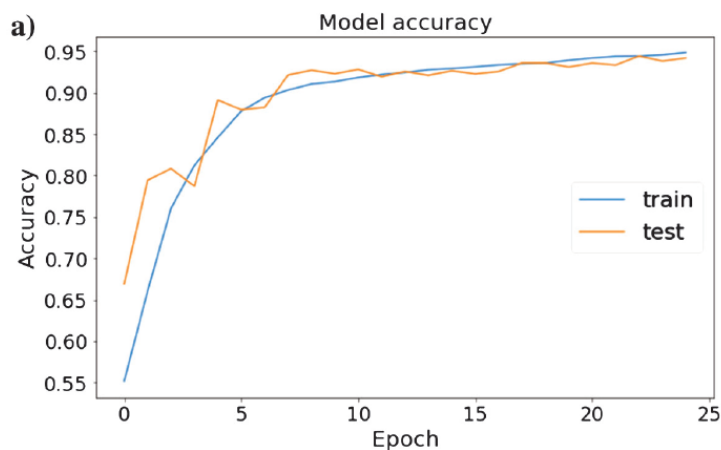


Рис. 13: Аккурасу в статье [1]

7 Возможные пути развития

В исследуемой задаче более 90% изображения не содержит разломы, следовательно соответствует нулевым пикселям на предсказанной маске. Такая несбалансированность весов может быть препятствием для использования обычной бинарной кросс энтропии в качестве функции потерь. В некоторых работах, например в [4] советуют применять сбалансированную функцию бинарной кросс-энтропии:

$$\mathbb{L} = -\frac{1}{N} \sum_{i=1}^N \left(\beta y_i \cdot \log(p(y_i)) + (1 - \beta)(1 - y_i) \cdot \log(1 - p(y_i)) \right), \quad (2)$$

где $\beta = \frac{1}{N} \sum_{i=1}^N (1 - y_i)$ представляет собой соотношение между пикселями, соответствующих пласту без разреза к пикселям всего пласта.

Кроме того, как мне кажется, в данной задаче можно подобрать loss-функцию с несколько другой логикой. Так как нас интересует только местоположение разреза, а не его толщина, логично сравнивать прямые, полученные при изначальной разметки и с помощью предсказаний, пренебрегая их толщиной. То есть в будущем, возможно, имеет смысл находить уравнение кривой разреза, интерполируя ее линейной функцией $y = ax + b$, и сравнивать эти коэффициенты a , b с a_{true} , b_{true} – коэффициентами исходной прямой, а также с некоторым весом учитывать положение этой прямой на изображении.

8 Выводы

В данной работе с применением архитектуры U-Net была решена задача 2d-сегментации для детектирования разломов на сейсмических данных. Предсказания, полученные, в двумерном пространстве имеют более высокую точность по сравнению с трёхмерным случаем реализованным в статье. К тому же, редукция до 2d экономит время работы и вычислительные ресурсы.

Применение бинарной кросс энтропии в качестве функции потерь в целом показало неплохие результаты, однако можно подобрать более подходящую функцию для данной конкретной задачи.

Метрика Intersection over union является, возможно, не самой подходящей для данной задачи, однако она может быть использована для сравнения различных конфигураций модели между собой. Так было установ-

лено, что добавление двух слоев дропаута в архитектуру U-Net существенно увеличивает точность предсказаний.

Список литературы

- [1] X. Wu, L. Liang, Y. Shi and S. Fomel. FaultSeg3D: using synthetic datasets to train an end-to-end convolutional neural network for 3D seismic fault segmentation. Geophysics, 2019
- [2] Ronneberger, O., Fischer, P. and Brox, T., 2015, October. U-net: Convolutional networks for biomedical image segmentation. In International Conference on Medical image computing and computer-assisted intervention (pp. 234–241). Springer, Cham
- [3] Diederik P Kingma M.Sc, Adam: A Method for Stochastic Optimization, 2015, San Diego.
- [4] Xie, S., and Z. Tu, 2015, Holistically-nested edge detection: Proceedings of the IEEE International Conference on Computer Vision, 1395–1403.