

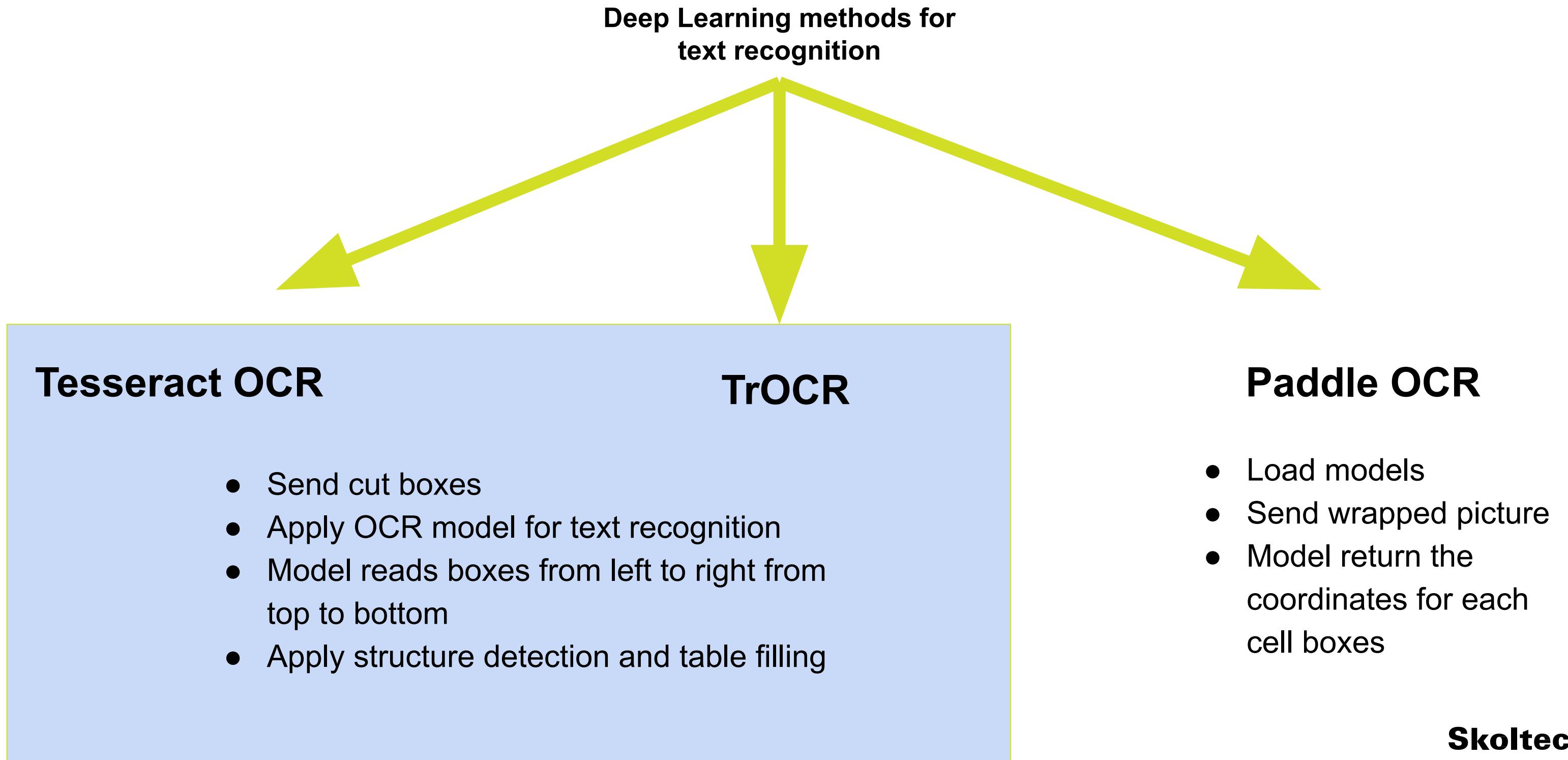
# AI-aided Optical Table Recognition

*Danil Sherki, Tatiana Alekhina, Grigoreva Anastasia*  
Petroleum Engineering

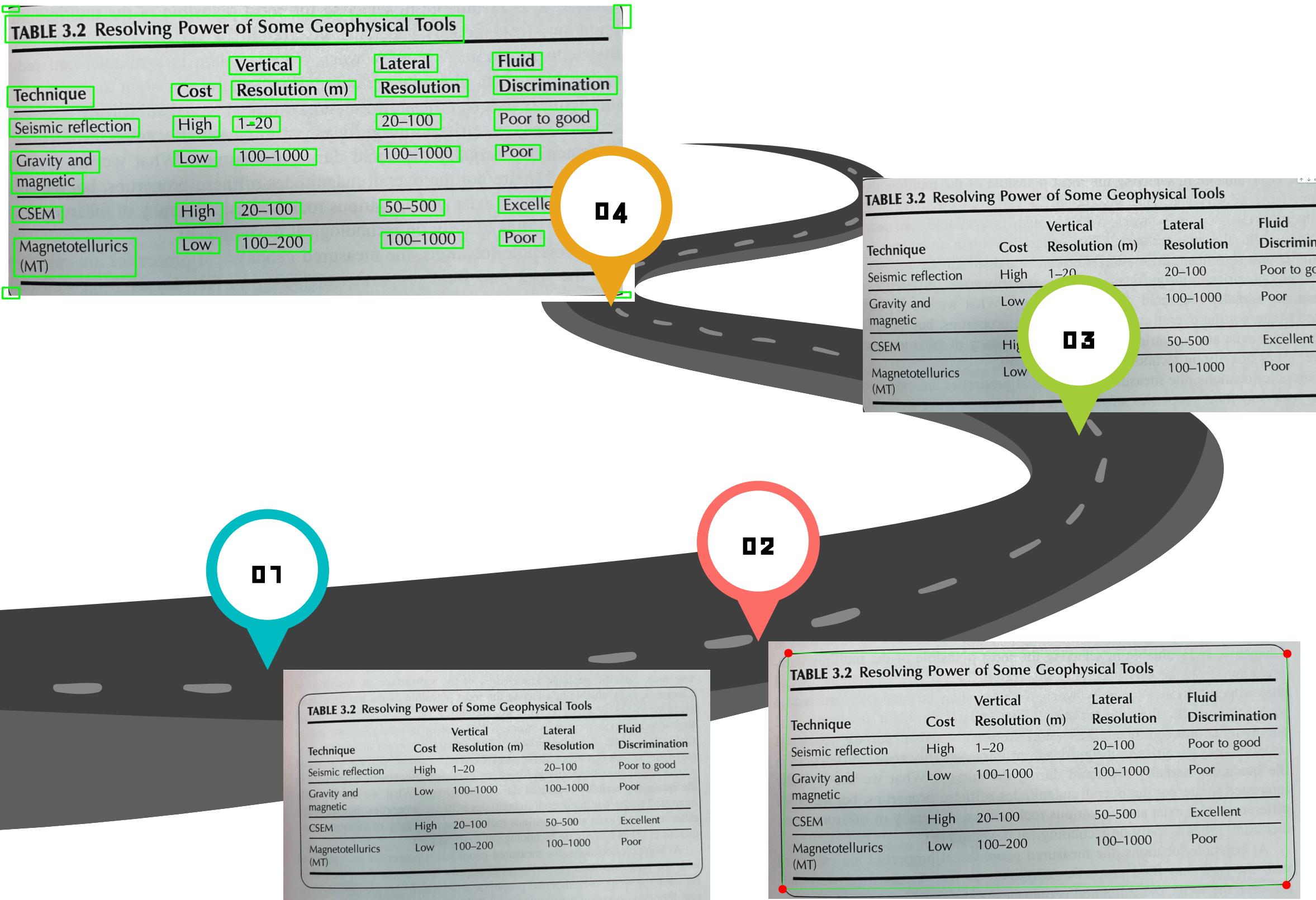
# Problem description

AI-assisted table recognition is needed to automatically extract information from tables, thereby reducing time and errors when processing large amounts of data. Also a lot of old scientific notes from previous century has only physical mastery and has no digital copy.

# Optical Character Recognition (OCR)



# Image preprocessing



- **Binarization, Otsu thresholding, Inverting and dilation**

- **Contour and order pts detection**

- **Warp Perspective**

- **Preprocessing, line erosion and bounding boxes detection**

# Tesseract OCR

Tesseract OCR (Optical Character Recognition) is an open-source software library and command-line tool developed by Google. It is designed to recognize and extract text from images and convert it into machine-readable text format. Tesseract OCR works by following a series of steps to process the input image and extract the text from it.

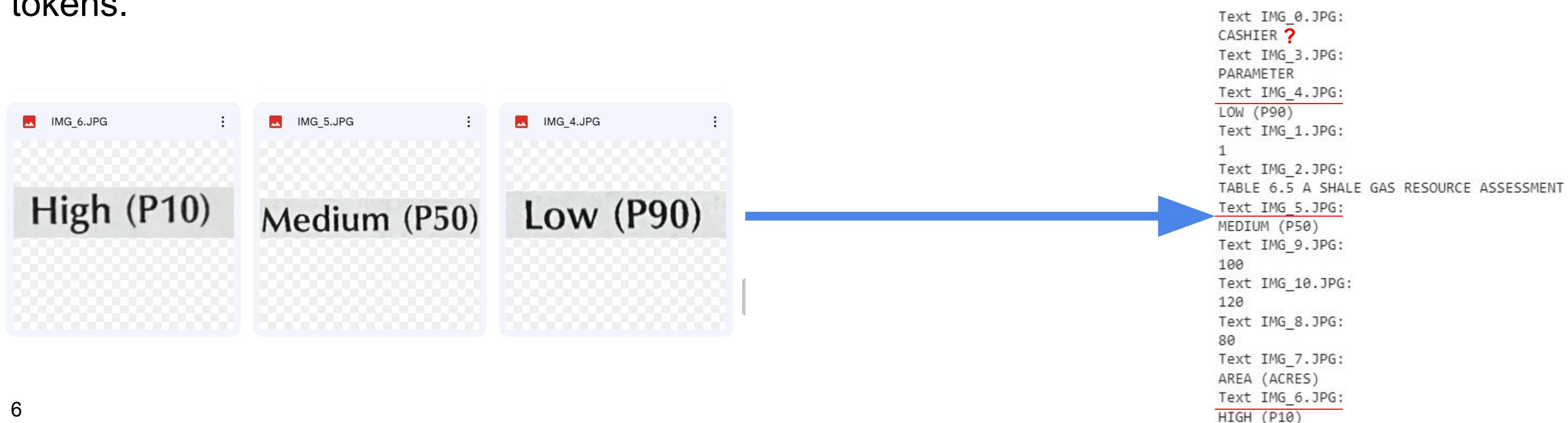


0	
1	120\n
2	Thickness (ft)\n
3	300\n
4	600\n
5	900\n
6	Porosity (%)\n
7	
8	
9	
10	Recovery factor (%)\n
11	TABLE 6.5 A Shale Gas Resource Assessment\n
12	10\n
13	20\n

# Transformers OCR

**Model Architecture:** TrOCR consists of an image Transformer encoder and an autoregressive text Transformer decoder. The model leverages the Transformer architecture for both image understanding and wordpiece-level text recognition

Images are presented to the model as a sequence of fixed-size patches (resolution 16x16), which are linearly embedded. One also adds absolute position embeddings before feeding the sequence to the layers of the Transformer encoder. Next, the Transformer text decoder autoregressively generates tokens.



# 飞桨 PaddleOCR

Method	BSD68		Urban100	
	10	70	10	70
CBM3D [14]	35.91	26.00	36.00	26.31
TNRD [12]	33.36	23.83	33.60	22.63
DnCNN [69]	36.31	26.56	36.21	26.17
MemNet [51]	N/A	25.08	N/A	24.96
IRCNN [70]	36.06	N/A	35.81	N/A
FFDNet [71]	36.14	26.53	35.77	26.39
RDN [77]	36.47	26.85	36.69	27.63
IPT (ours)	<b>38.30</b>	<b>28.21</b>	<b>39.07</b>	<b>28.80</b>

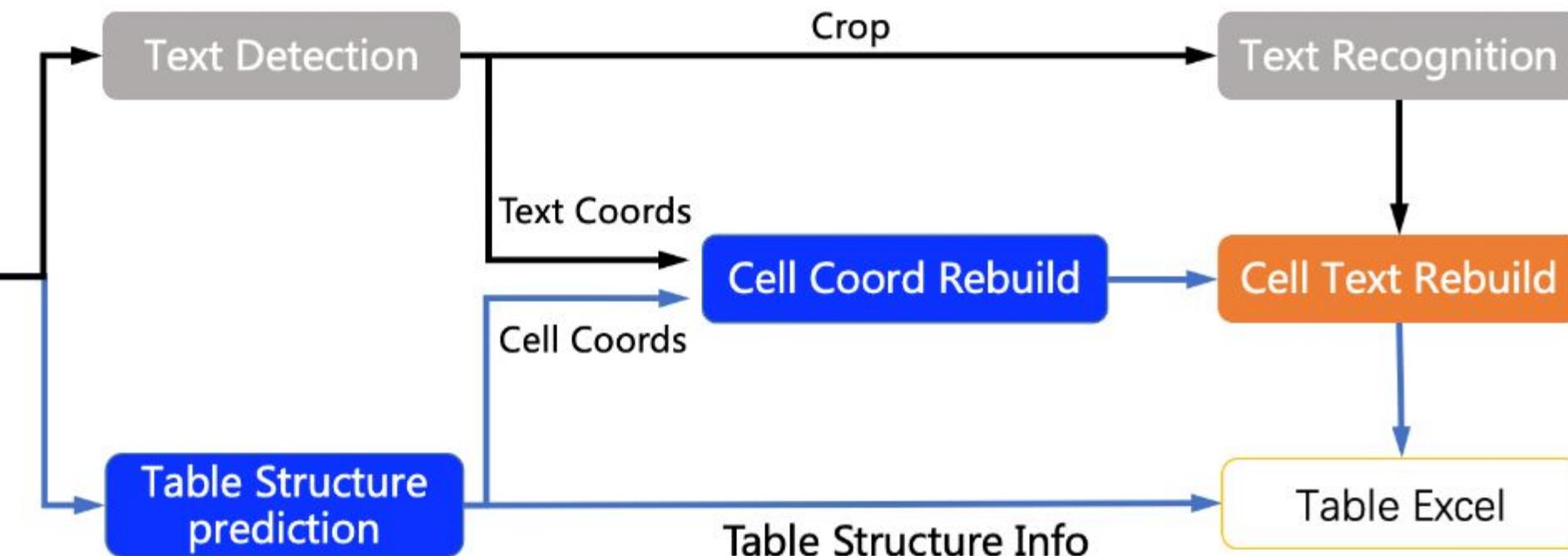


TABLE 3.2 Resolving Power of Some Geophysical Tools

Technique	Cost	Vertical Resolution (m)	Lateral Resolution	Fluid Discrimination
Seismic reflection	High	1–20	20–100	Poor to good
Gravity and magnetic	Low	100–1000	100–1000	Poor
CSEM	High	20–100	50–500	Excellent
Magnetotellurics (MT)	Low	100–200	100–1000	Poor

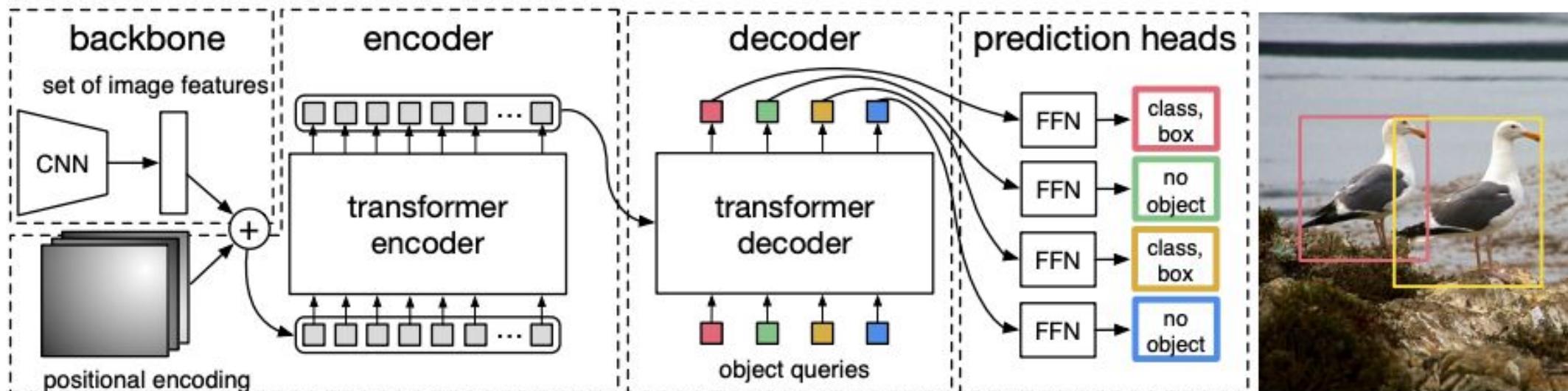


TABLE 3.2 Resolving Power of Some Geophysical Tools

Lateral	Fluid	Vertical	Discrimination	Resolution (m)	Resolution	Technique	Cost
Poor to good	High	20–100	1–20	Seismic reflection			
Poor	100–1000	Low	100–1000	Gravity and magnetic			
Excellent	50–500	High	20–100	CSEM			
Poor	100–1000	Low	100–200	Magnetotellurics (MT)			

# TableTransformer(TaTr)

## End-to-End Object Detection with Transformers



TaTr is DeTr trained on the table data

- Trained on 1 million samples
- Can recognize complicated elements of the table
- Have customizing CNN backbones for feature extracting

"PubTables-1M: Towards comprehensive table extraction from unstructured documents"

Sample Group	Some Year 1 Head Start Participation	No Year 1 Head Start Participation	Total
All Randomly Assigned (N=4,667):			
3-Year-Old Cohort			
Head Start Group	85.1%	14.9%	100%
Control Group	17.3%	82.7%	100%
4-Year-Old Cohort			
Head Start Group	79.8%	20.2%	100%
Control Group	13.9%	86.1%	100%

(b) PubTables-1M

Sample Group	Some Year 1 Head Start Participation	No Year 1 Head Start Participation	Total
All Randomly Assigned (N=4,667):			
3-Year-Old Cohort			
Head Start Group	85.1%	14.9%	100%
Control Group	17.3%	82.7%	100%
4-Year-Old Cohort			
Head Start Group	79.8%	20.2%	100%
Control Group	13.9%	86.1%	100%

(d) FinTabNet.a6

Sample Group	Some Year 1 Head Start Participation	No Year 1 Head Start Participation	Total
All Randomly Assigned (N=4,667):			
3-Year-Old Cohort			
Head Start Group	85.1%	14.9%	100%
Control Group	17.3%	82.7%	100%
4-Year-Old Cohort			
Head Start Group	79.8%	20.2%	100%
Control Group	13.9%	86.1%	100%

(f) FinTabNet.a6 + PubTables-1M

# TableTransformer(TaTr)

Model Name	$n_{\text{params}}$	$n_{\text{layers}}$	$d_{\text{model}}$	$n_{\text{heads}}$	$d_{\text{head}}$	Batch Size	Learning Rate
GPT-3 Small	125M	12	768	12	64	0.5M	$6.0 \times 10^{-4}$
GPT-3 Medium	350M	24	1024	16	64	0.5M	$3.0 \times 10^{-4}$
GPT-3 Large	760M	24	1536	16	96	0.5M	$2.5 \times 10^{-4}$
GPT-3 XL	1.3B	24	2048	24	128	1M	$2.0 \times 10^{-4}$
GPT-3 2.7B	2.7B	32	2560	32	80	1M	$1.6 \times 10^{-4}$
GPT-3 6.7B	6.7B	32	4096	32	128	2M	$1.2 \times 10^{-4}$
GPT-3 13B	13.0B	40	5140	40	128	2M	$1.0 \times 10^{-4}$
GPT-3 175B or "GPT-3"	175.0B	96	12288	96	128	3.2M	$0.6 \times 10^{-4}$

**Table 2.1:** Sizes, architectures, and learning hyper-parameters (batch size in tokens and learning rate) of the models which we trained. All models were trained for a total of 300 billion tokens.

## 2.1 Model and Architectures

We use the same model and architecture as GPT-2 [RWC<sup>+</sup>19], including the modified initialization, pre-normalization, and reversible tokenization described therein, with the exception that we use alternating dense and locally banded sparse attention patterns in the layers of the transformer, similar to the Sparse Transformer [CGRS19]. To study the dependence of ML performance on model size, we train 8 different sizes of model, ranging over three orders of magnitude from 125 million parameters to 175 billion parameters, with the last being the model we call GPT-3. Previous work [KMH<sup>+</sup>20] suggests that with enough training data, scaling of validation loss should be approximately a smooth power law as a function of size; training models of many different sizes allows us to test this hypothesis both for validation loss and for downstream language tasks.

Table 2.1 shows the sizes and architectures of our 8 models. Here  $n_{\text{params}}$  is the total number of trainable parameters,  $n_{\text{layers}}$  is the total number of layers,  $d_{\text{model}}$  is the number of units in each bottleneck layer (we always have the feedforward layer four times the size of the bottleneck layer,  $d_{\text{ff}} = 4 * d_{\text{model}}$ ), and  $d_{\text{head}}$  is the dimension of each attention head. All models use a context window of  $n_{\text{ctx}} = 2048$  tokens. We partition the model across GPUs along both the depth and width dimension in order to minimize data-transfer between nodes. The precise architectural parameters for each model are chosen based on computational efficiency and load-balancing in the layout of models across GPU's. Previous work [KMH<sup>+</sup>20] suggests that validation loss is not strongly sensitive to these parameters within a reasonably broad range.

## 2.2 Training Dataset

## Table-Transformer Detection

Model Name	$n_{\text{params}}$	$n_{\text{layers}}$	$d_{\text{model}}$	$n_{\text{heads}}$	$d_{\text{head}}$	Batch Size	Learning Rate
GPT-3 Small	125M	12	768	12	64	0.5M	$6.0 \times 10^{-4}$
GPT-3 Medium	350M	24	1024	16	64	0.5M	$3.0 \times 10^{-4}$
GPT-3 Large	760M	24	1536	16	96	0.5M	$2.5 \times 10^{-4}$
GPT-3 XL	1.3B	24	2048	24	128	1M	$2.0 \times 10^{-4}$
GPT-3 2.7B	2.7B	32	2560	32	80	1M	$1.6 \times 10^{-4}$
GPT-3 6.7B	6.7B	32	4096	32	128	2M	$1.2 \times 10^{-4}$
GPT-3 13B	13.0B	40	5140	40	128	2M	$1.0 \times 10^{-4}$
GPT-3 175B or "GPT-3"	175.0B	96	12288	96	128	3.2M	$0.6 \times 10^{-4}$

**Table 2.1:** Sizes, architectures, and learning hyper-parameters (batch size in tokens and learning rate) of the models which we trained. All models were trained for a total of 300 billion tokens.

## 2.1 Model and Architectures

We use the same model and architecture as GPT-2 [RWC<sup>+</sup>19], including the modified initialization, pre-normalization, and reversible tokenization described therein, with the exception that we use alternating dense and locally banded sparse attention patterns in the layers of the transformer, similar to the Sparse Transformer [CGRS19]. To study the dependence of ML performance on model size, we train 8 different sizes of model, ranging over three orders of magnitude from 125 million parameters to 175 billion parameters, with the last being the model we call GPT-3. Previous work [KMH<sup>+</sup>20] suggests that with enough training data, scaling of validation loss should be approximately a smooth power law as a function of size; training models of many different sizes allows us to test this hypothesis both for validation loss and for downstream language tasks.

Table 2.1 shows the sizes and architectures of our 8 models. Here  $n_{\text{params}}$  is the total number of trainable parameters,  $n_{\text{layers}}$  is the total number of layers,  $d_{\text{model}}$  is the number of units in each bottleneck layer (we always have the feedforward layer four times the size of the bottleneck layer,  $d_{\text{ff}} = 4 * d_{\text{model}}$ ), and  $d_{\text{head}}$  is the dimension of each attention head. All models use a context window of  $n_{\text{ctx}} = 2048$  tokens. We partition the model across GPUs along both the depth and width dimension in order to minimize data-transfer between nodes. The precise architectural parameters for each model are chosen based on computational efficiency and load-balancing in the layout of models across GPU's. Previous work [KMH<sup>+</sup>20] suggests that validation loss is not strongly sensitive to these parameters within a reasonably broad range.

## 2.2 Training Dataset

Table Table (rotated)

Class + Score + bboxes

# TableTransformer(TaTr)

Model Name	$n_{\text{params}}$	$n_{\text{layers}}$	$d_{\text{model}}$	$n_{\text{heads}}$	$d_{\text{head}}$	Batch Size	Learning Rate
GPT-3 Small	125M	12	768	12	64	0.5M	$6.0 \times 10^{-4}$
GPT-3 Medium	350M	24	1024	16	64	0.5M	$3.0 \times 10^{-4}$
GPT-3 Large	760M	24	1536	16	96	0.5M	$2.5 \times 10^{-4}$
GPT-3 XL	1.3B	24	2048	24	128	1M	$2.0 \times 10^{-4}$
GPT-3 2.7B	2.7B	32	2560	32	80	1M	$1.6 \times 10^{-4}$
GPT-3 6.7B	6.7B	32	4096	32	128	2M	$1.2 \times 10^{-4}$
GPT-3 13B	13.0B	40	5140	40	128	2M	$1.0 \times 10^{-4}$
GPT-3 175B or "GPT-3"	175.0B	96	12288	96	128	3.2M	$0.6 \times 10^{-4}$

Model Name	$n_{\text{params}}$	$n_{\text{layers}}$	$d_{\text{model}}$	$n_{\text{heads}}$	$d_{\text{head}}$	Batch Size	Learning Rate
GPT-3 Small	125M	12	768	12	64	0.5M	$6.0 \times 10^{-4}$
GPT-3 Medium	350M	24	1024	16	64	0.5M	$3.0 \times 10^{-4}$
GPT-3 Large	760M	24	1536	16	96	0.5M	$2.5 \times 10^{-4}$
GPT-3 XL	1.3B	24	2048	24	128	1M	$2.0 \times 10^{-4}$
GPT-3 2.7B	2.7B	32	2560	32	80	1M	$1.6 \times 10^{-4}$
GPT-3 6.7B	6.7B	32	4096	32	128	2M	$1.2 \times 10^{-4}$
GPT-3 13B	13.0B	40	5140	40	128	2M	$1.0 \times 10^{-4}$
GPT-3 175B or "GPT-3"	175.0B	96	12288	96	128	3.2M	$0.6 \times 10^{-4}$

Table-Transformer Structure Recognition

OCR technology

Model Name	$n_{\text{params}}$	$n_{\text{layers}}$	$d_{\text{model}}$	$n_{\text{heads}}$	$d_{\text{head}}$	Batch Size	Learning Rate
GPT-3 Small	125M	12	768	12	64	0.5M	$6.0 \times 10^{-4}$
GPT-3 Medium	350M	24	1024	16	64	0.5M	$3.0 \times 10^{-4}$
GPT-3 Large	760M	24	1536	16	96	0.5M	$2.5 \times 10^{-4}$
GPT-3 XL	1.3B	24	2048	24	128	1M	$2.0 \times 10^{-4}$
GPT-3 2.7B	2.7B	32	2560	32	80	1M	$1.6 \times 10^{-4}$
GPT-3 6.7B	6.7B	32	4096	32	128	2M	$1.2 \times 10^{-4}$
GPT-3 13B	13.0B	40	5140	40	128	2M	$1.0 \times 10^{-4}$
GPT-3 175B or "GPT-3"	175.0B	96	12288	96	128	3.2M	$0.6 \times 10^{-4}$

	Model Name	$n_{\text{params}}$	$n_{\text{layers}}$	$d_{\text{model}}$	$n_{\text{heads}}$	$d_{\text{head}}$	Batch Size	Learning Rate
0	GPT-3 Small	125M	12	768	12	64	0.5M	$6.0 \times 10^{-4}$
1	GPT-3 Medium	350M	24	1024	16	64	0.5M	$3.0 \times 10^{-4}$
2	GPT-3 Large	760M	24	1536	16	96	0.5M	$2.5 \times 10^{-4}$
3	GPT-3 XL	1.3B	24	2048	24	128	1M	$2.0 \times 10^{-4}$
4	GPT-3 2.7B	2.7B	32	2560	32	80	1M	$1.6 \times 10^{-4}$
5	GPT-3 6.7B	6.7B	32	4096	32	128	2M	$1.2 \times 10^{-4}$
6	GPT-3 13B	13.0B	40	5140	40	128	2M	$1.0 \times 10^{-4}$
7	GPT-3 175B or "GPT-3"	175.0B	96	12288	96	128	3.2M	$0.6 \times 10^{-4}$

# TableTransformer(TaTr)

Model Name	$n_{\text{params}}$	$n_{\text{layers}}$	$d_{\text{model}}$	$n_{\text{heads}}$	$d_{\text{head}}$	Batch Size	Learning Rate
GPT-3 Small	125M	12	768	12	64	0.5M	$6.0 \times 10^{-4}$
GPT-3 Medium	350M	24	1024	16	64	0.5M	$3.0 \times 10^{-4}$
GPT-3 Large	760M	24	1536	16	96	0.5M	$2.5 \times 10^{-4}$
GPT-3 XL	1.3B	24	2048	24	128	1M	$2.0 \times 10^{-4}$
GPT-3 2.7B	2.7B	32	2560	32	80	1M	$1.6 \times 10^{-4}$
GPT-3 6.7B	6.7B	32	4096	32	128	2M	$1.2 \times 10^{-4}$
GPT-3 13B	13.0B	40	5140	40	128	2M	$1.0 \times 10^{-4}$
GPT-3 175B or "GPT-3"	175.0B	96	12288	96	128	3.2M	$0.6 \times 10^{-4}$

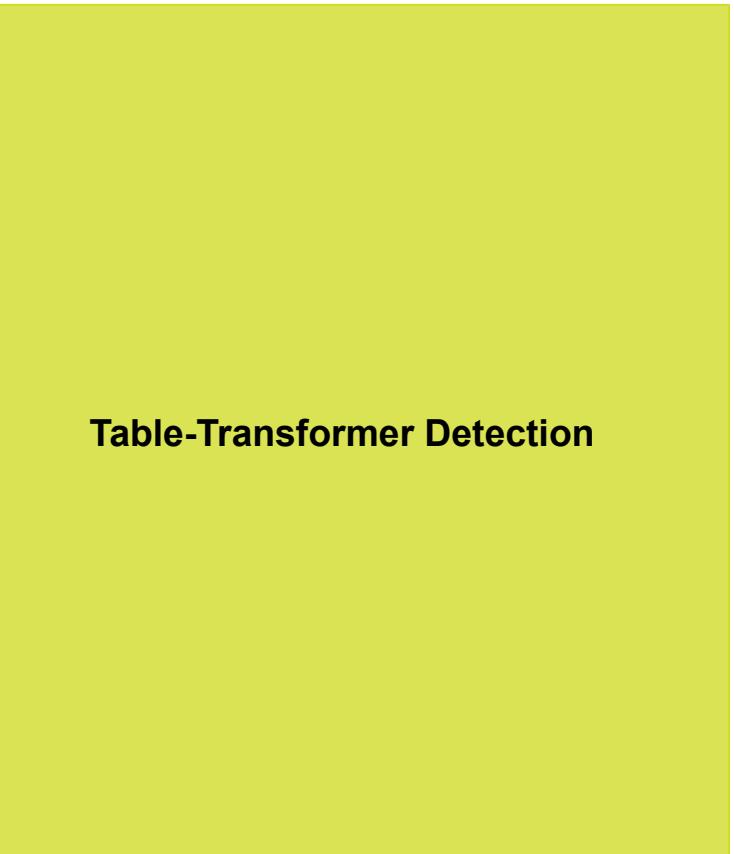
**Table 2.1:** Sizes, architectures, and learning hyper-parameters (batch size in tokens and learning rate) of the models which we trained. All models were trained for a total of 300 billion tokens.

## 2.1 Model and Architectures

We use the same model and architecture as GPT-2 [RWC<sup>+</sup>19], including the modified initialization, pre-normalization, and reversible tokenization described therein, with the exception that we use alternating dense and locally banded sparse attention patterns in the layers of the transformer, similar to the Sparse Transformer [CGRS19]. To study the dependence of ML performance on model size, we train 8 different sizes of model, ranging over three orders of magnitude from 125 million parameters to 175 billion parameters, with the last being the model we call GPT-3. Previous work [KMH<sup>+</sup>20] suggests that with enough training data, scaling of validation loss should be approximately a smooth power law as a function of size; training models of many different sizes allows us to test this hypothesis both for validation loss and for downstream language tasks.

Table 2.1 shows the sizes and architectures of our 8 models. Here  $n_{\text{params}}$  is the total number of trainable parameters,  $n_{\text{layers}}$  is the total number of layers,  $d_{\text{model}}$  is the number of units in each bottleneck layer (we always have the feedforward layer four times the size of the bottleneck layer,  $d_{\text{ff}} = 4 * d_{\text{model}}$ ), and  $d_{\text{head}}$  is the dimension of each attention head. All models use a context window of  $n_{\text{ctx}} = 2048$  tokens. We partition the model across GPUs along both the depth and width dimension in order to minimize data-transfer between nodes. The precise architectural parameters for each model are chosen based on computational efficiency and load-balancing in the layout of models across GPU's. Previous work [KMH<sup>+</sup>20] suggests that validation loss is not strongly sensitive to these parameters within a reasonably broad range.

## 2.2 Training Dataset



## Table-Transformer Detection

Model Name	$n_{\text{params}}$	$n_{\text{layers}}$	$d_{\text{model}}$	$n_{\text{heads}}$	$d_{\text{head}}$	Batch Size	Learning Rate
GPT-3 Small	125M	12	768	12	64	0.5M	$6.0 \times 10^{-4}$
GPT-3 Medium	350M	24	1024	16	64	0.5M	$3.0 \times 10^{-4}$
GPT-3 Large	760M	24	1536	16	96	0.5M	$2.5 \times 10^{-4}$
GPT-3 XL	1.3B	24	2048	24	128	1M	$2.0 \times 10^{-4}$
GPT-3 2.7B	2.7B	32	2560	32	80	1M	$1.6 \times 10^{-4}$
GPT-3 6.7B	6.7B	32	4096	32	128	2M	$1.2 \times 10^{-4}$
GPT-3 13B	13.0B	40	5140	40	128	2M	$1.0 \times 10^{-4}$
GPT-3 175B or "GPT-3"	175.0B	96	12288	96	128	3.2M	$0.6 \times 10^{-4}$

**Table 2.1:** Sizes, architectures, and learning hyper-parameters (batch size in tokens and learning rate) of the models which we trained. All models were trained for a total of 300 billion tokens.

## 2.1 Model and Architectures

We use the same model and architecture as GPT-2 [RWC<sup>+</sup>19], including the modified initialization, pre-normalization, and reversible tokenization described therein, with the exception that we use alternating dense and locally banded sparse attention patterns in the layers of the transformer, similar to the Sparse Transformer [CGRS19]. To study the dependence of ML performance on model size, we train 8 different sizes of model, ranging over three orders of magnitude from 125 million parameters to 175 billion parameters, with the last being the model we call GPT-3. Previous work [KMH<sup>+</sup>20] suggests that with enough training data, scaling of validation loss should be approximately a smooth power law as a function of size; training models of many different sizes allows us to test this hypothesis both for validation loss and for downstream language tasks.

Table 2.1 shows the sizes and architectures of our 8 models. Here  $n_{\text{params}}$  is the total number of trainable parameters,  $n_{\text{layers}}$  is the total number of layers,  $d_{\text{model}}$  is the number of units in each bottleneck layer (we always have the feedforward layer four times the size of the bottleneck layer,  $d_{\text{ff}} = 4 * d_{\text{model}}$ ), and  $d_{\text{head}}$  is the dimension of each attention head. All models use a context window of  $n_{\text{ctx}} = 2048$  tokens. We partition the model across GPUs along both the depth and width dimension in order to minimize data-transfer between nodes. The precise architectural parameters for each model are chosen based on computational efficiency and load-balancing in the layout of models across GPU's. Previous work [KMH<sup>+</sup>20] suggests that validation loss is not strongly sensitive to these parameters within a reasonably broad range.

## 2.2 Training Dataset

Table    Table (rotated)

Class + Score + bboxes

GPT-3 Small	125M	12	768	12	64	0.5M	$6.0 \times 10^{-4}$
GPT-3 Medium	350M	24	1024	16	64	0.5M	$3.0 \times 10^{-4}$
GPT-3 Large	760M	24	1536	16	96	0.5M	$2.5 \times 10^{-4}$
GPT-3 XL	1.3B	24	2048	24	128	1M	$2.0 \times 10^{-4}$
GPT-3 2.7B	2.7B	32	2560	32	80	1M	$1.6 \times 10^{-4}$
GPT-3 6.7B	6.7B	32	4096	32	128	2M	$1.2 \times 10^{-4}$
GPT-3 13B	13.0B	40	5140	40	128	2M	$1.0 \times 10^{-4}$
GPT-3 175B or "GPT-3"	175.0B	96	12288	96	128	3.2M	$0.6 \times 10^{-4}$

**Table 2.1:** Sizes, architectures, and learning hyper-parameters (batch size in tokens and learning rate) of the models which we trained. All models were trained for a total of 300 billion tokens.

## 2.1 Model and Architectures

We use the same model and architecture as GPT-2 [RWC<sup>+</sup>19], including the modified initialization, pre-normalization, and reversible tokenization described therein, with the exception that we use alternating dense and locally banded sparse attention patterns in the layers of the transformer, similar to the Sparse Transformer [CGRS19]. To study the dependence of ML performance on model size, we train 8 different sizes of model, ranging over three orders of magnitude from 125 million parameters to 175 billion parameters, with the last being the model we call GPT-3. Previous work [KMH<sup>+</sup>20] suggests that with enough training data, scaling of validation loss should be approximately a smooth power law as a function of size; training models of many different sizes allows us to test this hypothesis both for validation loss and for downstream language tasks.

Table 2.1 shows the sizes and architectures of our 8 models. Here  $n_{\text{params}}$  is the total number of trainable parameters,  $n_{\text{layers}}$  is the total number of layers,  $d_{\text{model}}$  is the number of units in each bottleneck layer (we always have the feedforward layer four times the size of the bottleneck layer,  $d_{\text{ff}} = 4 * d_{\text{model}}$ ), and  $d_{\text{head}}$  is the dimension of each attention head. All models use a context window of  $n_{\text{ctx}} = 2048$  tokens. We partition the model across GPUs along both the depth and width dimension in order to minimize data-transfer between nodes. The precise architectural parameters for each model are chosen based on computational efficiency and load-balancing in the layout of models across GPU's. Previous work [KMH<sup>+</sup>20] suggests that validation loss is not strongly sensitive to these parameters within a reasonably broad range.

Can't recognize the table from the page

Can't recognize the contour from the table picture

Model Name	$n_{\text{params}}$	$n_{\text{layers}}$	$d_{\text{model}}$	$n_{\text{heads}}$	$d_{\text{head}}$	Batch Size	Learning Rate
GPT-3 Small	125M	12	768	12	64	0.5M	$6.0 \times 10^{-4}$
GPT-3 Medium	350M	24	1024	16	64	0.5M	$3.0 \times 10^{-4}$
GPT-3 Large	760M	24	1536	16	96	0.5M	$2.5 \times 10^{-4}$
GPT-3 XL	1.3B	24	2048	24	128	1M	$2.0 \times 10^{-4}$
GPT-3 2.7B	2.7B	32	2560	32	80	1M	$1.6 \times 10^{-4}$
GPT-3 6.7B	6.7B	32	4096	32	128	2M	$1.2 \times 10^{-4}$
GPT-3 13B	13.0B	40	5140	40	128	2M	$1.0 \times 10^{-4}$
GPT-3 175B or "GPT-3"	175.0B	96	12288	96	128	3.2M	$0.6 \times 10^{-4}$

**Table 2.1:** Sizes, architectures, and learning hyper-parameters (batch size in tokens and learning rate) of the models which we trained. All models were trained for a total of 300 billion tokens.

# What is the solution?

**Table 1: Descriptive characteristics of participating 4–7 year old children and households.**

	Total (N = 80)	TV in Bedroom (N = 19)	No TV in Bedroom (N = 61)
Variable			
Gender (boys/girls)	42/38	10/9	32/29
Demographics	Mean ± SD	Mean ± SD	Mean ± SD
Age (years)	6.0 ± 1.2	6.4 ± 1.0	5.9 ± 1.3
BMI (kg/m <sup>2</sup> )	19.2 ± 3.0	20.1 ± 4.4	19.0 ± 2.4
BMI percentile	90.8 ± 6.8	91.1 ± 6.7	90.7 ± 6.9
zBMI (standardized measure of BMI)	1.6 ± 0.6	1.6 ± 0.6	1.6 ± 0.6
Environment			
Televisions in the home	3.0 ± 1.3	4.1 ± 1.6	2.6 ± 1.0†
Computers in the home	1.0 ± 0.6	1.0 ± 0.6	1.1 ± 0.6
People residing in the home	4.3 ± 0.9	4.4 ± 1.0	4.2 ± 0.9
Parental estimates of television (hours/week)	25.6 ± 13.0	26.5 ± 17.2	25.4 ± 11.5
Actual television (hours/week)	23.4 ± 11.1	29.8 ± 14.4	21.4 ± 9.1‡
Ethnicity	N (%)		
White	62 (77.5)		
African American	4 (5.0)		
Hispanic	7 (8.8)		
More than one race	7 (8.8)		

Note: zBMI was calculated using the National Center for Health

(CDC) [17]

† P &lt; 0.001

‡ P &lt; 0.005

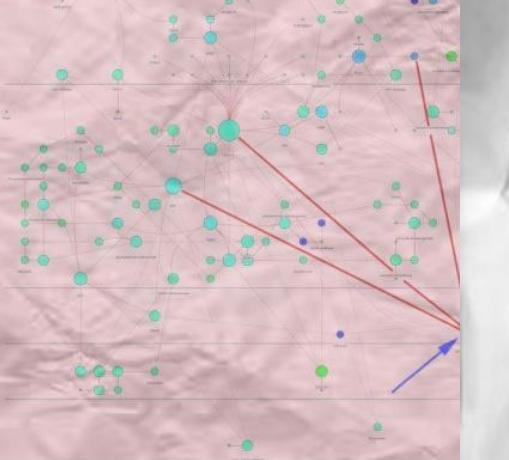


Figure 1 Diagram showing the structure of the boar spermatozoa capacitation network and the distribution of actin polymerization nodes. The network is a complex graph where node size is proportional to connection number and color gradient represents path length. A red arrow points to an actin polymerization node.

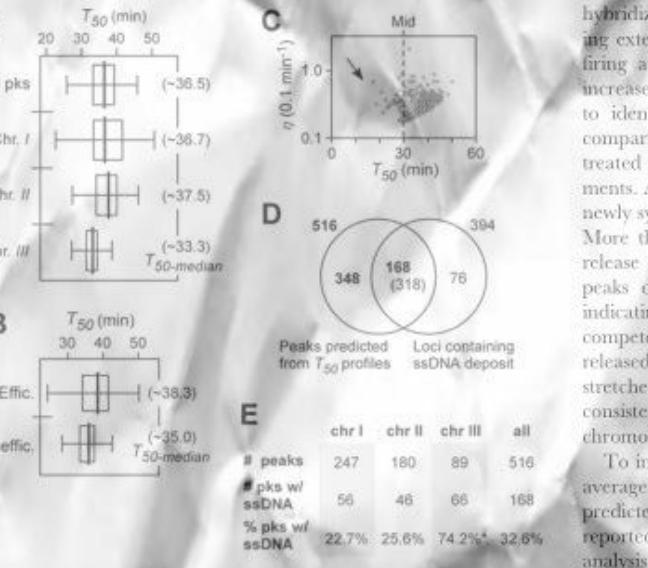
**Table 2: Most connected nodes (the hubs) of capacitation network**

Node	Number of links
[Ca <sup>2+</sup> ] <sub>i</sub>	28
ATP	15
Tyr phosphorylation	13
PKA	9
ADP	8
PLDI	8
NADH	8
Actin polymerization	8

The number of nodes represent number of edges represents the clustering coefficient is calculated by dividing the number of edges connecting the  $\bar{N}$  neighbor nodes by the number of possible edges between them. The diameter is the largest distance between any two nodes. The average path length is the mean of the shortest paths between all pairs of nodes.

**Table 3: Main topological network after "actin polymerization"**

Parameter	
N° nodes	
N° edges	
Clustering coefficient	
Diameter	
Averaged N° neighbours	
Char. path length	
Degree distribution	

The number of nodes represent number of edges represents the clustering coefficient is calculated by dividing the number of edges connecting the  $\bar{N}$  neighbor nodes by the number of possible edges between them. The diameter is the largest distance between any two nodes. The average path length is the mean of the shortest paths between all pairs of nodes.Figure 5. Characteristics of average firing timing  $T_{50}$  and (maximum)**Table 4: Analysis of prognostic factors for progression-free survival (Cox proportional hazard with frailty)**

Factor	Univariate		Multivariate*			
	HR	(95% CI)	p-value	HR	(95% CI)	p-value
Radiation therapy	0.19	(0.11–0.31)	< 0.001	0.21*	(0.13–0.34)	< 0.001
Radiotherapy dose (>50 Gy)	0.60	(0.38–0.97)	0.028			
Fraction size ( $\geq 2$ Gy)	0.59	(0.37–0.95)	0.036			
Resection margins <sup>a</sup>	1.07	(0.72–1.58)	n.s.			
Indication radiotherapy						
Adjuvant radiotherapy	0.42	(0.25–0.72)	0.002			
Radiotherapy at recurrence	2.69	(1.63–4.41)	< 0.001			
Primary radiotherapy	0.36	(0.11–1.15)	n.s.			
Tumour localization						
Head-neck	0.96	(0.46–2.00)	n.s.			
Trunk	0.67	(0.37–1.18)	n.s.			
Abdominal wall	0.42	(0.21–0.85)	0.017	0.28*	(0.15–0.53)	< 0.001
Extremities <sup>b</sup>	2.5	(1.68–3.62)	< 0.001			
Potential etiological factors						
Gender (male)	0.90	(0.61–1.32)	n.s.			
Age (years)	0.95	(0.62–1.45)	n.s.			
	0.99	(0.98–1.00)	n.s.			

<sup>a</sup>Without frailty calculated because Cox proportional hazard did not converge.

Abbreviations: HR: hazard ratio; CI: Confidence interval; n.s.: not significant.

better for patients who had received radiotherapy ( $p < 0.001$ ) (Fig. 2).**Survival analysis – Prognostic factors**

The univariate analysis of possible prognostic factors revealed a significantly lower risk of recurrence related to the following factors: additional irradiation, a fraction size of  $\geq 2$  Gy with a hazard rate of 60%, a total dose  $> 50$  Gy with a hazard rate of 59% ( $p = 0.028$ , Table 4). In multivariate analysis radiotherapy treatment and tumour localization in the abdominal wall were independent positive prognostic factors (Table 4). The comparison of adjuvant post-operative radiotherapy versus radiotherapy at recurrence found adjuvant radiotherapy to be significantly better ( $p < 0.001$ ). Age was not a prognosticator. A more advanced age does not reduce the risk for a desmoid tumour. No age relation was found.

**Surgery**

Wide surgical excision is considered to be the standard treatment and can result in a cure. Cure is defined as no tumour progression or relapse. Published data indicate that the likelihood of local recurrence after surgery alone is high with reported recurrence rates ranging from 20% to 90% [3,14–20]. The local recurrence rate of 32% observed in this study was therefore low. Recurrence rates of up to 68% after resection have been described [14].

**Discussion**  
The optimal treatment for patients with aggressive fibromatosis remains unclear. Desmoid tumours are slowly proliferating tumours. The ultimate treatment goal is tumour control as the probability of dying from aggressive fibromatosis is relatively low. Patients with an intra-abdominal desmoid tumour are at a higher risk of local tumour progression or of local or combined treatment. This may be a challenge in the presence of multiple primary tumours. Recurrence and treatment modalities will evaluate the value of a

could, for example, be given post-operatively or after a recurrence. The same patient could have had several resections and radiotherapy at some point in time. For this reason we have performed, in addition to the classic actuarial analysis (with the last follow-up as endpoint) a Cox Hazard Frailty Analysis. The classic analysis does not take into account a possible tumour and patient related risk, where some tumours keep recurring after the same primary treatment, whereas the Hazard Frailty Analysis takes into account the time-related probability of occurrence of failure and considers as such each patient individually.

**Surgery**  
Wide surgical excision is considered to be the standard treatment and can result in a cure. Cure is defined as no tumour progression or relapse. Published data indicate that the likelihood of local recurrence after surgery alone is high with reported recurrence rates ranging from 20% to 90% [3,14–20]. The local recurrence rate of 32% observed in this study was therefore low. Recurrence rates of up to 68% after resection have been described [14].

**Conclusion**  
The optimal treatment for patients with aggressive fibromatosis remains unclear. Desmoid tumours are slowly proliferating tumours. The ultimate treatment goal is tumour control as the probability of dying from aggressive fibromatosis is relatively low. Patients with an intra-abdominal desmoid tumour are at a higher risk of local tumour progression or of local or combined treatment.

This may be a challenge in the presence of multiple primary tumours. Recurrence and treatment modalities will evaluate the value of a

overnight (15 hr) culture other prior to filter sterilization and concentrated 15-fold. Bedford, MA, 15  $\mu$ l of the DNase test agar activity was HCl to precipitate clearance around it.

**Production of the PIA**  
To assess the normalized with centrifugation. C (pH 8.0) and b centrifugation K (20 mg/ml) of Tris-buffer,

[pH 7.4], each extract membrane using a BIO-Dot microfiltration Laboratories, Inc., Hercules, CA). After drying, the presence amount of PIA was assessed using anti-PIA antiserum and the WesternBreeze chemiluminescence immunodetection kit (Invitrogen Corp., Carlsbad, CA).

**Statistical analysis**  
Statistical analysis of results comparing wild-type strains was done using the Student's t-test. Statistical analysis of results comparing different strains with their isogenic *sadA* and *agg* mutants and their *sadA*-complemented derivatives was done by ANOVA based on all pairwise comparisons. In both cases  $p$  values  $< 0.05$  were considered significant.
**Acknowledgments**

The authors thank Drs. Kim Jefferson (Virginia Commonwealth University, Richmond, VA) and Gerald Pier (Channing Laboratory, Harvard Medical School) for the generous gift of anti-PIA antibody.

**Author Contributions**

Conceived and designed the experiments: KEB LNM AKZ MSS. Performed the experiments: KEB LNM LMG AKZ. Analyzed the data: KEB LNM AKZ LNS MSS. Contributed reagents/materials/analysis: JCH KWB. Wrote the paper: MSS.

**Table 1: Knowledge, attitudes and practices regarding A(H1N1) pandemic influenza and A(H1N1) pandemic influenza vaccine**

	Strongly Disagree/ Disagree	Somewhat Disagree	Somewhat Agree	Agree/Strongly Agree
<b>Family Physicians (n = 207)*</b>				
A(H1N1) pandemic influenza ...				
Is a serious disease	4%	10%	32%	49%
Would occur frequently in Canada without vaccination	8%	10%	26%	52%
Generate a significant economic burden in Canada	4%	5%	17%	69%
A(H1N1) pandemic influenza vaccine will be...				
Safe	5%	10%	35%	38%
Effective	4%	9%	46%	26%
Well accepted by the public	6%	17%	35%	38%
Well accepted by vaccinators	3%	8%	26%	58%
I will recommend A(H1N1) pandemic vaccine to my patients	1%	6%	20%	68%
<b>Paediatricians (n = 714)*</b>				
A(H1N1) pandemic influenza ...				
Is a serious disease	3%	6%	27%	59%
Would occur frequently in Canada without vaccination	6%	6%	24%	59%
Generate a significant economic burden in Canada	2%	3%	15%	77%
A(H1N1) pandemic influenza vaccine will be...				
Safe	2%	6%	36%	43%
Effective	3%	6%	45%	28%
Well accepted by the public	5%	13%	34%	42%
Well accepted by vaccinators	2%	4%	21%	67%
I will recommend A(H1N1) pandemic vaccine to my patients	2%	3%	15%	75%

\*Due to missing responses, row percentages may not add up to 100%.

all 1852 Canadian paediatricians. The Canadian Medical Directory [12] was used to identify paediatricians and to obtain a random sample of family physicians. This database contains more than 58 000 listings of medical contact information and is updated each year. Based on the analytical

# TableTransformer(TaTr)

Model Name	$n_{\text{params}}$	$n_{\text{layers}}$	$d_{\text{model}}$	$n_{\text{heads}}$	$d_{\text{head}}$	Batch Size	Learning Rate
GPT-3 Small	125M	12	768	12	64	0.5M	$6.0 \times 10^{-4}$
GPT-3 Medium	350M	24	1024	16	64	0.5M	$3.0 \times 10^{-4}$
GPT-3 Large	760M	24	1536	16	96	0.5M	$2.5 \times 10^{-4}$
GPT-3 XL	1.3B	24	2048	24	128	1M	$2.0 \times 10^{-4}$
GPT-3 2.7B	2.7B	32	2560	32	80	1M	$1.6 \times 10^{-4}$
GPT-3 6.7B	6.7B	32	4096	32	128	2M	$1.2 \times 10^{-4}$
GPT-3 13B	13.0B	40	5140	40	128	2M	$1.0 \times 10^{-4}$
GPT-3 175B or "GPT-3"	175.0B	96	12288	96	128	3.2M	$0.6 \times 10^{-4}$

**Table 2.1:** Sizes, architectures, and learning hyper-parameters (batch size in tokens and learning rate) of the models which we trained. All models were trained for a total of 300 billion tokens.

## 2.1 Model and Architectures

We use the same model and architecture as GPT-2 [RWC<sup>+</sup>19], including the modified initialization, pre-normalization, and reversible tokenization described therein, with the exception that we use alternating dense and locally banded sparse attention patterns in the layers of the transformer, similar to the Sparse Transformer [CGRS19]. To study the dependence of ML performance on model size, we train 8 different sizes of model, ranging over three orders of magnitude from 125 million parameters to 175 billion parameters, with the last being the model we call GPT-3. Previous work [KMH<sup>+</sup>20] suggests that with enough training data, scaling of validation loss should be approximately a smooth power law as a function of size; training models of many different sizes allows us to test this hypothesis both for validation loss and for downstream language tasks.

Table 2.1 shows the sizes and architectures of our 8 models. Here  $n_{\text{params}}$  is the total number of trainable parameters,  $n_{\text{layers}}$  is the total number of layers,  $d_{\text{model}}$  is the number of units in each bottleneck layer (we always have the feedforward layer four times the size of the bottleneck layer,  $d_{\text{ff}} = 4 * d_{\text{model}}$ ), and  $d_{\text{head}}$  is the dimension of each attention head. All models use a context window of  $n_{\text{ctx}} = 2048$  tokens. We partition the model across GPUs along both the depth and width dimension in order to minimize data-transfer between nodes. The precise architectural parameters for each model are chosen based on computational efficiency and load-balancing in the layout of models across GPU's. Previous work [KMH<sup>+</sup>20] suggests that validation loss is not strongly sensitive to these parameters within a reasonably broad range.

## 2.2 Training Dataset

## Fine-Tuned Table-Transformer Detection

Model Name	$n_{\text{params}}$	$n_{\text{layers}}$	$d_{\text{model}}$	$n_{\text{heads}}$	$d_{\text{head}}$	Batch Size	Learning Rate
GPT-3 Small	125M	12	768	12	64	0.5M	$6.0 \times 10^{-4}$
GPT-3 Medium	350M	24	1024	16	64	0.5M	$3.0 \times 10^{-4}$
GPT-3 Large	760M	24	1536	16	96	0.5M	$2.5 \times 10^{-4}$
GPT-3 XL	1.3B	24	2048	24	128	1M	$2.0 \times 10^{-4}$
GPT-3 2.7B	2.7B	32	2560	32	80	1M	$1.6 \times 10^{-4}$
GPT-3 6.7B	6.7B	32	4096	32	128	2M	$1.2 \times 10^{-4}$
GPT-3 13B	13.0B	40	5140	40	128	2M	$1.0 \times 10^{-4}$
GPT-3 175B or "GPT-3"	175.0B	96	12288	96	128	3.2M	$0.6 \times 10^{-4}$

**Table 2.1:** Sizes, architectures, and learning hyper-parameters (batch size in tokens and learning rate) of the models which we trained. All models were trained for a total of 300 billion tokens.

## 2.1 Model and Architectures

We use the same model and architecture as GPT-2 [RWC<sup>+</sup>19], including the modified initialization, pre-normalization, and reversible tokenization described therein, with the exception that we use alternating dense and locally banded sparse attention patterns in the layers of the transformer, similar to the Sparse Transformer [CGRS19]. To study the dependence of ML performance on model size, we train 8 different sizes of model, ranging over three orders of magnitude from 125 million parameters to 175 billion parameters, with the last being the model we call GPT-3. Previous work [KMH<sup>+</sup>20] suggests that with enough training data, scaling of validation loss should be approximately a smooth power law as a function of size; training models of many different sizes allows us to test this hypothesis both for validation loss and for downstream language tasks.

Table 2.1 shows the sizes and architectures of our 8 models. Here  $n_{\text{params}}$  is the total number of trainable parameters,  $n_{\text{layers}}$  is the total number of layers,  $d_{\text{model}}$  is the number of units in each bottleneck layer (we always have the feedforward layer four times the size of the bottleneck layer,  $d_{\text{ff}} = 4 * d_{\text{model}}$ ), and  $d_{\text{head}}$  is the dimension of each attention head. All models use a context window of  $n_{\text{ctx}} = 2048$  tokens. We partition the model across GPUs along both the depth and width dimension in order to minimize data-transfer between nodes. The precise architectural parameters for each model are chosen based on computational efficiency and load-balancing in the layout of models across GPU's. Previous work [KMH<sup>+</sup>20] suggests that validation loss is not strongly sensitive to these parameters within a reasonably broad range.

## 2.2 Training Dataset

 Table  Table (rotated)

Class + Score + bboxes

# TableTransformer(TaTr)

Model Name	$n_{\text{params}}$	$n_{\text{layers}}$	$d_{\text{model}}$	$n_{\text{heads}}$	$d_{\text{head}}$	Batch Size	Learning Rate
GPT-3 Small	125M	12	768	12	64	0.5M	$6.0 \times 10^{-4}$
GPT-3 Medium	350M	24	1024	16	64	0.5M	$3.0 \times 10^{-4}$
GPT-3 Large	760M	24	1536	16	96	0.5M	$2.5 \times 10^{-4}$
GPT-3 XL	1.3B	24	2048	24	128	1M	$2.0 \times 10^{-4}$
GPT-3 2.7B	2.7B	32	2560	32	80	1M	$1.6 \times 10^{-4}$
GPT-3 6.7B	6.7B	32	4096	32	128	2M	$1.2 \times 10^{-4}$
GPT-3 13B	13.0B	40	5140	40	128	2M	$1.0 \times 10^{-4}$
GPT-3 175B or “GPT-3”	175.0B	96	12288	96	128	3.2M	$0.6 \times 10^{-4}$

**Table 2.1:** Sizes, architectures, and learning hyper-parameters (batch size in tokens and learning rate) of the models which we trained. All models were trained for a total of 300 billion tokens.

## 2.1 Model and Architectures

We use the same model and architecture as GPT-2 [RWC<sup>+</sup>19], including the modified initialization, pre-normalization, and reversible tokenization described therein, with the exception that we use alternating dense and locally banded sparse attention patterns in the layers of the transformer, similar to the Sparse Transformer [CGRS19]. To study the dependence of ML performance on model size, we train 8 different sizes of model, ranging over three orders of magnitude from 125 million parameters to 175 billion parameters, with the last being the model we call GPT-3. Previous work [KMH<sup>+</sup>20] suggests that with enough training data, scaling of validation loss should be approximately a smooth power law as a function of size; training models of many different sizes allows us to test this hypothesis both for validation loss and for downstream language tasks.

Table 2.1 shows the sizes and architectures of our 8 models. Here  $n_{\text{params}}$  is the total number of trainable parameters,  $n_{\text{layers}}$  is the total number of layers,  $d_{\text{model}}$  is the number of units in each bottleneck layer (we always have the same bottleneck layer,  $d_{\text{ff}} = 4 * d_{\text{model}}$ ), and  $d_{\text{head}}$  is the dimension of each attention head. All models use a context window of  $n_{\text{ctx}} = 2048$  tokens. We partition the model across GPUs along both the depth and width dimension in order to minimize data transfer between nodes. The precise architectural choices for each model are chosen based on computational efficiency and load-balancing in the layout of models across GPU’s. Previous work [KMH<sup>+</sup>20] suggests that validation loss is not strongly sensitive to these parameters within a reasonably broad range.

## 2.2 Training Dataset

## Fine-Tuned Table-Transformer Detection

## Fine-Tuned Table-Transformer Structure Recognition

Model Name	$n_{\text{params}}$	$n_{\text{layers}}$	$d_{\text{model}}$	$n_{\text{heads}}$	$d_{\text{head}}$	Batch Size	Learning Rate
GPT-3 Small	125M	12	768	12	64	0.5M	$6.0 \times 10^{-4}$
GPT-3 Medium	350M	24	1024	16	64	0.5M	$3.0 \times 10^{-4}$
GPT-3 Large	760M	24	1536	16	96	0.5M	$2.5 \times 10^{-4}$
GPT-3 XL	1.3B	24	2048	24	128	1M	$2.0 \times 10^{-4}$
GPT-3 2.7B	2.7B	32	2560	32	80	1M	$1.6 \times 10^{-4}$
GPT-3 6.7B	6.7B	32	4096	32	128	2M	$1.2 \times 10^{-4}$
GPT-3 13B	13.0B	40	5140	40	128	2M	$1.0 \times 10^{-4}$
GPT-3 175B or “GPT-3”	175.0B	96	12288	96	128	3.2M	$0.6 \times 10^{-4}$

**Table 2.1:** Sizes, architectures, and learning hyper-parameters (batch size in tokens and learning rate) of the models which we trained. All models were trained for a total of 300 billion tokens.

## 2.1 Model and Architectures

We use the same model and architecture as GPT-2 [RWC<sup>+</sup>19], including the modified initialization, pre-normalization, and reversible tokenization described therein, with the exception that we use alternating dense and locally banded sparse attention patterns in the layers of the transformer, similar to the Sparse Transformer [CGRS19]. To study the dependence of ML performance on model size, we train 8 different sizes of model, ranging over three orders of magnitude from 125 million parameters to 175 billion parameters, with the last being the model we call GPT-3. Previous work [KMH<sup>+</sup>20] suggests that with enough training data, scaling of validation loss should be approximately a smooth power law as a function of size; training models of many different sizes allows us to test this hypothesis both for validation loss and for downstream language tasks.

of our 8 models. Here  $n_{\text{params}}$  is the total number of trainable parameters,  $n_{\text{layers}}$  is the number of units in each bottleneck layer (we always have the same bottleneck layer,  $d_{\text{ff}} = 4 * d_{\text{model}}$ ), and  $d_{\text{head}}$  is the dimension of each attention head. All models use a context window of  $n_{\text{ctx}} = 2048$  tokens. We partition the model across GPUs along both the depth and width dimension in order to minimize data-transfer between nodes. The precise architectural choices for each model are chosen based on computational efficiency and load-balancing in the layout of models across GPU’s. Previous work [KMH<sup>+</sup>20] suggests that validation loss is not strongly sensitive to these parameters within a reasonably broad range.

Table  Table (rotated)

Model Name	$n_{\text{params}}$	$n_{\text{layers}}$	$d_{\text{model}}$	$n_{\text{heads}}$	$d_{\text{head}}$	Batch Size	Learning Rate
GPT-3 Small	125M	12	768	12	64	0.5M	$6.0 \times 10^{-4}$
GPT-3 Medium	350M	24	1024	16	64	0.5M	$3.0 \times 10^{-4}$
GPT-3 Large	760M	24	1536	16	96	0.5M	$2.5 \times 10^{-4}$
GPT-3 XL	1.3B	24	2048	24	128	1M	$2.0 \times 10^{-4}$
GPT-3 2.7B	2.7B	32	2560	32	80	1M	$1.6 \times 10^{-4}$
GPT-3 6.7B	6.7B	32	4096	32	128	2M	$1.2 \times 10^{-4}$
GPT-3 13B	13.0B	40	5140	40	128	2M	$1.0 \times 10^{-4}$
GPT-3 175B or “GPT-3”	175.0B	96	12288	96	128	3.2M	$0.6 \times 10^{-4}$

# Other examples

	Group	p	
	Brain infections (N = 145, %)	Controls (N = 292, %)	
Mean follow-up years to epilepsy (SD)*	7.7 (3.6)	7.9 (3.5)	>0.99
Mean follow-up years to ADHD (SD)*	7.8 (3.8)	7.8 (3.6)	>0.99
Mean follow-up years to ASD(SD)*	7.7 (3.8)	7.9 (3.7)	>0.99
Gender			0.28
Male	98 (67)	182 (62)	
Female	47 (33)	110 (38)	
Infection mean age (months) (SD)	41.2 (60.70)	44.8 (41.24)	0.46
Stratified by age [yr]			>0.99
0-2	92 (63.4)	184 (63)	
2-5	15 (10.3)	31 (10)	
5-10	16 (11.0)	33 (11)	
>10	22 (15.1)	44 (15)	
Associated pathogens			N/A
Enterovirus	78 (53.7)		
Herpes simplex virus	8 (5.5)		
Group B streptococcus	28 (19.3)		
S. pneumoniae	10 (6.8)		
Others	21 (14.4)		<0.01
Neurodevelopmental outcomes			
ID or CP	48 (33.1)	9 (3.0)	
Epilepsy	11 (7.5)	1 (0.3)	
ADHD	4 (2.7)	6 (2.0)	
ASD	1 (0.7)	0 (0)	
Epilepsy+ADHD	1 (0.7)	1 (0.3)	
ADHD+ASD	1 (0.7)	1 (0.3)	

	Group	p	
	Brain infections (N = 145, %)	Controls (N = 292, %)	
Mean follow-up years to epilepsy (SD)*	7.7 (3.6)	7.9 (3.5)	>0.99
Mean follow-up years to ADHD (SD)*	7.8 (3.8)	7.8 (3.6)	>0.99
Mean follow-up years to ASD(SD)*	7.7 (3.8)	7.9 (3.7)	>0.99
Gender			0.28
Male	98 (67)	182 (62)	
Female	47 (33)	110 (38)	
Infection mean age (months) (SD)	41.2 (60.70)	44.8 (41.24)	0.46
Stratified by age [yr]			>0.99
0-2	92 (63.4)	184 (63)	
2-5	15 (10.3)	31 (10)	
5-10	16 (11.0)	33 (11)	
>10	22 (15.1)	44 (15)	
Associated pathogens			N/A
Enterovirus	78 (53.7)		
Herpes simplex virus	8 (5.5)		
Group B streptococcus	28 (19.3)		
S. pneumoniae	10 (6.8)		
Others	21 (14.4)		
Neurodevelopmental outcomes			
ID or CP	48 (33.1)	9 (3.0)	
Epilepsy	11 (7.5)	1 (0.3)	
ADHD	4 (2.7)	6 (2.0)	
ASD	1 (0.7)	0 (0)	
Epilepsy+ADHD	1 (0.7)	1 (0.3)	
ADHD+ASD	1 (0.7)	1 (0.3)	

Characteristic	Patients with Maspin HSCORE above cutpoint (N = 13)	Patients with Maspin HSCORE below cutpoint (N = 9)	P-value
Age at surgery	69.3 + 7	67.3 + 10.3	
Gender			0.647
Male	5	2	
Female	8	7	
Tumor greatest dimension			0.011
<4 cm	7	1	
4-6 cm	5	2	
>6 cm	1	6	
Histologic grade			0.658
Well/Moderate	9	7	
Poor	4	2	
Percentage of tumor necrosis <sup>18</sup>			0.550
<50	2	3	
51-75	5	2	
>75	6	4	
Penitumoral border liver <sup>19</sup>			0.103
Infiltrated	8	2	
Pushed	5	6	
Lymphoid infiltration <sup>1</sup>			0.342
Absent	7	3	
Present	6	6	
Stage			0.148
I	4	0	
II	2	1	
III	7	8	

Characteristic	Patients with Maspin HSCORE above cutpoint (N = 13)	Patients with Maspin HSCORE below cutpoint (N = 9)	P-value
Age at surgery	69.3 + 7	67.3 + 10.3	0.647
Gender			
Male	5	2	
Female	8	7	
Tumor greatest dimension			0.011
<4 cm	7	1	
4-6 cm	5	2	
>6 cm	1	6	
Histologic grade			0.658
Well/Moderate	9	7	
Poor	4	2	
Percentage of tumor necrosis <sup>18</sup>			0.550
<50	2	3	
51-75	5	2	
>75	6	4	
Penitumoral border liver <sup>19</sup>			0.103
Infiltrated	8	2	
Pushed	5	6	
Lymphoid infiltration <sup>1</sup>			0.342
Absent	7	3	
Present	6	6	
Stage			0.148
I	4	0	
II	2	1	
III	7	8	

Data cell Column header cell Projected row header cell

Inh	C (M)	E <sub>corr</sub> (mV/SCE)	I <sub>corr</sub> (μA/cm <sup>2</sup> )	β <sub>c</sub> (mV/dec)	-β <sub>c</sub> (mV/dec)	η%
Blank	—	-473.80	916.6 (±1.78)	163.6 (±1.10)	155.0 (±1.33)	—
DAEP1	10 <sup>-1</sup>	-489.32	075.40 (±0.36)	149.7 (±0.87)	151.3 (±1.79)	91.7
DAEP1	10 <sup>-4</sup>	-473.58	133.67 (±0.57)	139.8 (±0.95)	168.7 (±1.68)	85.4
DAEP1	10 <sup>-3</sup>	-429.96	172.98 (±0.85)	133.9 (±0.91)	168.2 (±1.67)	81.2
DAEP1	10 <sup>-2</sup>	-516.72	250.88 (±0.93)	172.7 (±1.02)	177.6 (±1.71)	72.6
DAEP2	10 <sup>-1</sup>	-489.28	070.52 (±0.23)	084.9 (±0.65)	158.0 (±1.24)	92.3
DAEP2	10 <sup>-4</sup>	-474.57	089.45 (±0.28)	067.7 (±0.42)	106.1 (±1.02)	90.0
DAEP2	10 <sup>-3</sup>	-476.71	145.78 (±0.45)	136.8 (±0.89)	165.7 (±1.56)	84.0
DAEP2	10 <sup>-2</sup>	-516.57	212.64 (±0.87)	111.4 (±0.77)	134.7 (±1.45)	76.8

Inh	C (M)	E <sub>corr</sub> (mV/SCE)	I <sub>corr</sub> (μA/cm <sup>2</sup> )	β <sub>c</sub> (mV/dec)	-β <sub>c</sub> (mV/dec)	η%
Blank	—	-473.80	916.6 (±1.78)	163.6 (±1.10)	155.0 (±1.33)	—
DAEP1	10 <sup>-1</sup>	-489.32	075.40 (±0.36)	149.7 (±0.87)	151.3 (±1.79)	91.7
DAEP1	10 <sup>-4</sup>	-473.58	133.67 (±0.57)	139.8 (±0.95)	168.7 (±1.68)	85.4
DAEP1	10 <sup>-3</sup>	-429.96	172.98 (±0.85)	133.9 (±0.91)	168.2 (±1.67)	81.2
DAEP1	10 <sup>-2</sup>	-516.72	250.88 (±0.93)	172.7 (±1.02)	177.6 (±1.71)	72.6
DAEP2	10 <sup>-1</sup>	-489.28	070.52 (±0.23)	084.9 (±0.65)	158.0 (±1.24)	92.3
DAEP2	10 <sup>-4</sup>	-474.57	089.45 (±0.28)	067.7 (±0.42)	106.1 (±1.02)	90.0
DAEP2	10 <sup>-3</sup>	-476.71	145.78 (±0.45)	136.8 (±0.89)	165.7 (±1.56)	84.0
DAEP2	10 <sup>-2</sup>	-516.57	212.64 (±0.87)	111.4 (±0.77)	134.7 (±1.45)	76.8

Data cell Column header cell Projected row header cell

# Conclusion

- Existing methods require image preprocessing, and still can't rebuilt tables
- From distorted, crumpled pictures tables hardly can be identified with traditional OCR methods
- There are some pretty NN good working solution, but they trained with “ideal” data
- Fine-tuning is good solution for increasing quality on two task
  - Table Detection on the photo
  - Table Structure recognition with complex elements
- Fine-Tuning of two SOTA NN in this task had been done of two from three pipeline steps
- Average Precision @ $[IoU=0.50:0.95]$  was increased from 0.745 to 0.929
- Further research will be for OCR technology (NN) fine-tuning

**Thank you for your attention!**