An Empirical Study of Scaling Law for Scene Text Recognition

Miao Rang* Zhenni Bi* Chuanjian Liu Yunhe Wang† Kai Han†
Huawei Noah's Ark Lab
{rangmiao1,bizhenni,liuchuanjian,
yunhe.wang,kai.han}@huawei.com

Abstract

The laws of model size, data volume, computation and model performance have been extensively studied in the field of Natural Language Processing (NLP). However, the scaling laws in Scene Text Recognition (STR) have not yet been investigated. To address this, we conducted comprehensive studies that involved examining the correlations between performance and the scale of models, data volume and computation in the field of text recognition. Conclusively, the study demonstrates smooth power laws between performance and model size, as well as training data volume, when other influencing factors are held constant. Additionally, we have constructed a large-scale dataset called ***REBU-Syn****, which comprises 6 M real samples and 18 M synthetic samples. Based on the disclosed scaling law and new dataset, we successfully trained a scene text recognition model, achieving a new state-of-the-art on 6 common test benchmarks with top-1 average accuracy of 97.42%. The models and dataset are publicly available at large-ocr-model.github.io.*

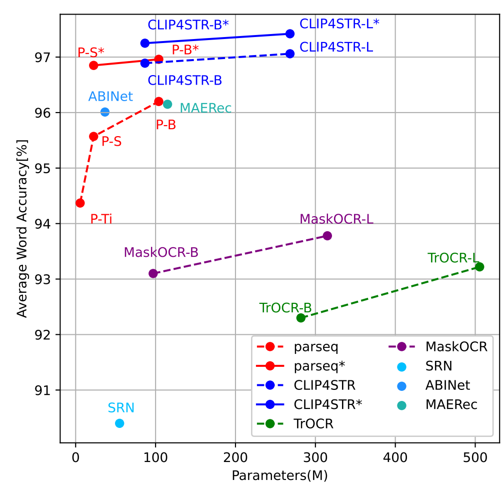


Figure 1. Mean word accuracy vs Parameters on the 6 common test benchmarks. P-Ti, P-S and P-B refer to PARSeq-Ti, PARSeq-S and PARSeq-B, respectively. * indicates training with REBU-Syn.

1. Introduction

Optical Character Recognition (OCR) technologies are pivotal in extracting textual content from images, including scanned documents and photographs. However, this paper narrows its focus to the text recognition phase, specifically to Scene Text Recognition (STR). STR stands out in the OCR field due to its complexity and the unique challenges it presents, such as variable illumination, occlusion, distortion, and viewing angles. These factors pose significant challenges in text identification compared to recognizing text in scanned documents. As a rapidly evolving field of research, STR offers substantial opportunities for technological advancements and innovation. Given this context, our study aims to specifically explore the scaling laws that are applicable to STR. By delving into the study of scaling laws, our primary objective is to gain a deeper understanding of how adjustments in system parameters impact the performance of STR. This investigation holds the potential to unveil new avenues for improvement in this challenging domain.

With the introduction of large-scale models in deep learning, an increasing number of academics are focusing on the potential and growth trends of these models, hoping that they will contribute to the development and design of future models. In the field of NLP [4, 8], numerous experiments have been carried out to investigate scaling model laws [14, 18, 24, 72]. The results show that the larger the volume of data fed into the neural network, the better its performance. Therefore, large language models trained on vast amounts of data have dominated the field of NLP. However, in the STR domain, research predominantly focuses on enhancing model performance using fixed

*These authors contributed equally to this work.
†Corresponding Author.

data volumes and model sizes [3, 31, 53]. Studies specifically addressing the scaling laws in large STR models are noticeably sparse, which casts uncertainty on the potential impact of large-scale models and substantial data volumes in STR. Transformer-based models have achieved state-of-the-art performance in various text recognition tasks and challenges [31, 47, 52, 58].

In this paper, we explore the scaling laws of STR by transformer-based models. Our focus is on unraveling the relationships between model size, data volume and computation with the model performance. Our experimental framework encompasses a wide range of models, with parameter counts ranging from 50 million to 1 billion, and data volumes that vary from 1 million to 1 billion training samples. Additionally, we extend our exploration to computational durations ranging from 100 to 1000 hours. This comprehensive analysis allows us to draw insightful conclusions about the scaling law in text recognition. Furthermore, we introduce a novel dataset called **REBU-Syn**, which combines real-world and synthetic data. This dataset has been meticulously compiled from existing public datasets, providing a valuable resource for further research in this field.

Throughout this research, we develop an advanced method for large-scale training. This method involves a comprehensive examination of various strategies, such as optimizing training hyperparameters, analyzing data distributions and utilizing pre-training techniques. Our objective is to create a model characterized by exceptional precision and accuracy. The culmination of these efforts is the training of CLIP4STR-L using REBU-Syn. This approach results in achieving a groundbreaking state-of-the-art performance of 97.42% on the test benchmark (see Fig 1). The following is a compilation of additional scaling laws for STR observations:

- The scaling law holds in the field of STR. There exist smooth power laws between the size of the model, the volume of data, computation and performance.
- Large-scale models make better use of samples than small-scale models which means that large models achieve lower error rate with fixed amount of data.
- The proportion of training data from different sources is crucial for model training.
- Models pre-trained on STR-related data are more effective in STR tasks than models pretrained on general images like ImageNet.

2. Related Work

Model Scale Recent research has extensively explored the scaling laws for Transformer language models, particularly in the field of NLP [24, 33]. These studies have established a set of universal principles for modeling scale. However, research specifically addressing STR remains scarce. Transformer-based methods, known for their higher tolerance to increased model depth and width, have been applied in various fields [10, 13, 15, 30, 46]. This study leverages these methods, with a specific focus on their application in STR, to provide guidance on making effective adjustments in model size.

Data Scale In the domain of image recognition, the scale of data plays a critical role. The performance of various models is significantly influenced by the size of the datasets used [5, 48, 50]. While different model types require varying data volumes, some previous methods [2] explored the impact of STR recognition tasks on different data scales, but their main focus was on CNN-based or attention-based approaches [7, 9, 29, 34, 53, 55], and they focus solely on reducing the data scale. Furthermore, the availability of public datasets has facilitated extensive research and experimentation in this field [5, 48, 50]. This paper aims to build upon these foundations by conducting a comprehensive investigation into the effects of data scale, both at the lower and upper limits, as well as the distribution of data in STR tasks. Additionally, this study offers new insights into the alignment of real and synthetic data during the training of optimal models, filling a gap in current research.

Scaling Laws The rapid advancement of Large Language Models (LLMs) like ChatGPT [41] and GPT-4 [42] has sparked research into universal scaling laws [24, 30] in deep learning. These studies explore the relationship between model size, data volume, computation and performance, providing training principles for large models in NLP. [20] describes laws for autoregressive generative modeling. Similar scaling theories have also emerged in the field of computer vision [72], as demonstrated by the training of ViT-G with 2B parameters [72]. Furthermore, recent work has been done on the scaling law of CLIP [45] has revealed task- and dataset-dependent scaling behaviors [32]. Building upon these foundational insights, this study represents a unique exploration of scaling laws within the context of STR. Specifically, it explores the allocation of parameters and the internal structure of the transformer model, with the aim of optimizing performance for text recognition. This investigation makes a unique contribution to the expanding body of research on scaling laws, particularly in the underexplored domain of STR.

3. Method Details

In this paper, our primary focus is to explore the scaling laws for the transfer performance of Transformer-based models in text recognition tasks, with TrOCR as a pioneering pure Transformer model, and PARSeq excelling in accuracy, aligning perfectly with our research scope. Concurrently, we have amalgamated all publicly available datasets to construct the REBU-Syn dataset. This paper also includes a thorough analysis of the data proportions obtained from various sources. Finally, we will provide a detailed overview of the training parameter settings used in our study.

3.1. Model Scaling

TrOCR TrOCR [31] is a text recognition model that utilizes pure Transformer architecture. It integrates pre-trained Computer Vision (CV) and NLP models. And it is the first work that jointly leverages image Transformer and text Transformer for text recognition tasks. The scaling laws of Transformer language models [24] and Vision Transformers [72] have been studied, the scaling laws for models in the text recognition have not yet been explored. Based on this, we scaled the TrOCR model sizes and attempted to analyze the accuracy change curves of models with varying sizes.

In TrOCR, the encoder and decoder parts utilize pre-trained image Transformer and text Transformer models, respectively. These pre-trained models utilize large-scale unlabeled data for image understanding and language modeling. As a result, TrOCR does not require additional language models for post-processing, and the model outperforms current state-of-the-art models in tasks related to printing and handwriting recognition. In order to continue benefiting from pre-training for related tasks, we select the most suitable combination of encoder and decoder in TrOCR for scaling.

For TrOCR-S, we use DeiT$_{SMALL}$ [59] to initialize the encoder and MiniLM [64] to initialize the decoder. TrOCR-B uses BEIT$_{BASE}$ [4] to initialize the encoder and RoBERTa$_{LARGE}$ [36] to initialize the decoder. TrOCR-L and TrOCR-H utilize BEIT$_{LARGE}$ to initialize the encoder and RoBERTa$_{LARGE}$ to initialize the decoder. The model's parameters range from 43.09 million to 1 billion, and the details of the parameters are shown in Table 1.

Model	Encoder			FLOPs (G)	Params (M)
	layers	hidden sizes	heads		
TROCR-S	12	384	6	13.31	43.09
TROCR-B	12	768	12	62.01	281.87
TROCR-L	24	1024	16	191.00	505.50
TROCR-H	48	1200	16	497.91	1037.61

Table 1. Architecture specifications of TrOCR variants.

PARSeq PARSeq [6] follows an encoder-decoder architecture. PARSeq is also based on the Transformer framework with excellent accuracy, which perfectly fits the scope of our research. The encoder part utilizes the Vision Transformer (ViT) model to extract image features, while the decoder follows the same architecture as the pre-LayerNorm [63]. Transformer decoder in this study utilizes twice the number of attention heads, where nhead = dmodel/32. In contrast to the standard ViT, the encoder removes the $[class]$ token and inputs all the output tokens into the decoder.

PARSeq has two models in the original paper, PARSeq-Ti and PARSeq-S. In order to investigate the law of large models in the field of text recognition, the scaling law of the ViT model was demonstrated. [72]. Based on this, we scaled PARSeq to 4 different sizes. On the basis of the original paper PARSeq-S, the model was expanded to 3 sizes: PARSeq-B, PARSeq-L, and PARSeq-H. The scale of the model was also expanded from 22 million to 0.6 billion. The configurations with different scale PARSeq models can be seen in Table 2.

Model	Encoder			FLOPs (G)	Params (M)
	layers	hidden sizes	heads		
PARSeq-S	12	384	6	2.76	22.51
PARSeq-B	12	768	12	17.20	104.01
PARSeq-L	24	1024	16	49.90	335.92
PARSeq-H	32	1280	16	98.10	682.14

Table 2. Architecture specifications of PARSeq variants.

3.2. Dataset

Training Dataset The training datasets for text recognition are typically categorized into synthetic and real data. Historically, scene text recognition models primarily relied on synthetic data due to the scarcity of real-world data. However, the recent increase in the availability of real data has shifted this trend. It has been observed that models trained on real data tend to be more sample-efficient compared to those trained on synthetic data. In light of this, we meticulously collected both synthetic and real data, employing various strategies to construct the REBU-Syn dataset. This dataset comprises approximately 6M real data samples and 18M public synthetic data samples, as detailed in Table 3. The ratio of synthetic to real data in REBU-Syn is 3:1. Furthermore, we utilized synthesis technology to generate an additional 60M data samples, similar to MJST, termed $MJST^{+}$.

Real Dataset We gather real images from 4 widely-accessible datasets to assemble the **REBU**. The **R** component consists of commonly used real data [6], including COCO-Text (COCO) [60], RCTW17 [54], Uber-Text (Uber) [67], ArT [12], LSVT [57], MLT19 [40], ReCTS [35], TextOCR [56] and OpenVINO [27]. A detailed analysis of these datasets is presented in [6]. **U**, another segment of REBU, includes 4 million labeled images across 14 datasets, collectively referred to as Union14M-L [6]. **B** represents the training data from benchmark sources, encompassing datasets such as IIIT 5k-word (IIIT5k) [38], Street View Text (SVT) [62], ICDAR13 [25] and ICDAR15 [26]. Furthermore, **E** is composed of images from two commonly used real datasets in text detection tasks, namely Total Text [11] and CTW1500 [71].This inclusion significantly expands the range of real data in our collection.

Public Synthetic Dataset MJSynth (MJ) [22] and SynthText (ST) [19] are two widely-utilized synthetic datasets in the field of scene text recognition, containing 8.9M million and 5.5M million data samples respectively. Addtionly, we incorporated two other composite datasets into our study. Curved SyntheText (CST) and SyntheAdd (SA) [21]. CST is specifically designed for text detection tasks, primarily comprising curved text data. SA, generated with the SynthText engine, is aimed at synthesizing less common characters, including punctuation marks.

Generate Synthetic Dataset To closely align with the MJ and ST datasets, we created $MJST^+$ using two data generation tools: TextRecognitionDataGenerator[1] and SynthText[2]. The TextRecognitionDataGenerator is adept at producing data that mimics complex scenes, encompassing effects such as blurring, tilting and distortion. SynthText, on the other hand, specializes in synthesizing data akin to ST, resulting in samples that blend more seamlessly with natural scenes.

To augment the diversity of the generated corpus, we sourced 700,000 corpus entries from the most extensively utilized English corpus website globally[3]. For the background selection in our synthesized images, we employed natural scene pictures provided by SynthText as the backdrop. Utilizing these two synthesis methods, we successfully synthesized a total of 60M data samples. Code for data synthesis is available[4].

Source	Dataset	Instances
Public Real	Real (R)	3.3M
Public Real	Extra Real Data (E)	15k
Public Real	BenchMark (B)	7.5K
Public Real	Union14M (U)	3.1M
Public Synthetic	MJ	5.5M
Public Synthetic	ST	8.9M
Public Synthetic	CST	1.8M
Public Synthetic	SA	1.2M
Generate Synthetic	$MJST^+$	60M

Table 3. Statistics of REBU-Syn datasets, including Public Real and Public Synthetic. Generate Synthetic can be used additionally.

Test Dataset To assess the performance of our model, we utilized 6 publicly available real scene text datasets: IIIT5k-Words (IIIT5k) [38], Street View Text (SVT) [62], ICDAR 2013 (IC13) [25], ICDAR 2015 (IC15) [26] , SVT-Perspective (SVTP) [43] and CUTE80 (CUTE) [49]. Both the IC13 and IC15 test sets have various subdivisions. We follow the division proposed by Yu et al [70], using a version of the IC15 test set containing 1,811 images, and the IC13 test set comprising 857 images.

However, to address challenges posed by differing annotation formats and the presence of duplicate, non-Latin, and damaged samples, we employed the following data fusion strategy:

- **Polygonal Text** We sourced synthesized data from datasets used in text detection tasks with polygonal annotation boxes, such as Curved SyntheText, SyntheAdd and STR Benchmark.To adapt these polygonal texts for use, we improved upon the method proposed in [6]. Our approach involves identifying the minimum bounding box of the polygon and applying a perspective transformation, avoiding direct clipping using maximum and minimum coordinates. This method retains challenging samples, as suggested in [6], while minimizing background interference, thus enabling the recognizer to focus on pertinent areas.
- **Remove invalid chars and samples** Focusing on Latin characters, which have extensive data availability, we retained samples composed only of letters and symbols. Samples not in our predefined dictionary were discarded.
- **Remove duplicate data** As we integrated multiple datasets, some of which overlapped, we meticulously removed any duplicate entries.

3.3. Experiment Settings

We took use of the publicly available implementations of TrOCR and PARSeq as baseline models. To achieve optimal performance, we tailored the number of training epochs and adjusted the learning rates. The specific implementation details are as follows:

Hyper-Parameters For our experiments, we use V100 GPUs equipped with 32GB of memory to train all models. The learning rates are set differently for various models. Specifically, TrOCR-S is trained with a batch size of 1024 and a learning rate of 4e$-$4. TrOCR-B employs a batch size of 256 with a learning rate of 1e$-$4, and TrOCR-L operates with a batch size of 128 and a learning rate of 4e-5. We use BPE [51] of Fairseq and SentencePiece [28] for tokenizing text lines into word pieces. For PARSeq models, a consistent learning rate of 7e$-$4 is used, with the batch size adjusted to be as close to 1024 as possible.

Evaluation Metrics Word accuracy was the primary metric for evaluating the datasets of scene text. In this work, we standardized the final output string to match the commonly used 36-character set (lowercase alphanumeric) to ensure a fair comparison across different models and datasets.

4. Results and Analysis

4.1. Smooth Power Laws

Model performance is primarily influenced by three variables: the number of model parameters N, the volume of the training data D, and the computation of the model C. In this section, we explore the power laws among these

[1] https://github.com/Belval/TextRecognitionDataGenerator
[2] https://github.com/ankush-me/SynthText
[3] https://www.english-corpora.org/corpora.asp
[4] https://github.com/large-ocr-model/large-ocr-model.github.io

influential factors with model performance E. To effectively characterize the scaling of models, we have conducted training with a variety of models, including TrOCR and PARSeq.

Model	Avg	Regular Text			Irregular Text		
		IC13 857	IIIT5k 3,000	SVT 647	CUTE80 288	IC15 1,811	SVTP 645
TrOCR-S	81.93	90.65	85.60	85.94	74.31	72.73	78.44
TrOCR-B	88.56	96.14	92.00	91.56	80.56	81.14	83.91
TrOCR-L	89.84	96.50	92.90	92.81	84.38	82.52	86.72
TrOCR-H	**90.94**	**97.31**	**93.57**	**94.22**	**87.50**	**83.79**	**88.59**

Table 4. Word accuracy with different TrOCR model sizes. Train data: Synthetic datasets with MJ and ST.

4.1.1 The power law of model when data is fixed.

- **Scaling TrOCR Models** We trained 4 different scales (ranging in size from 43.09M to 1B) of TrOCR models. In order to maintain fairness and consistency with the experimental setting in the original TrOCR paper, we use MJ and ST to train TrOCR models with different model sizes. The experimental results on 6 common test benchmarks are shown in Table 4. As shown in Fig 2a, our analysis reveals a linear relationship on the log-log plot between the parameter count N and modeling performance. This relationship can be described by a power-law equation ($E = aC^b$). Employing Algorithm 1 in the appendix, we utilized the first three models (TrOCR-S, TrOCR-B and TrOCR-L) to obtain the power function equation $E(\cdot)$. The last model (TrOCR-H) accurately aligns with the fitted straight line, demonstrating the effectiveness of the power law. The power law of the TrOCR model is as follows.
$$E(N) = \left(1.97 * 10^4/N\right)^{0.223} \quad (1)$$
- **Scaling PARSeq Models** To further validate the power law in relation to model parameters, we trained PARSeq models across 4 different scales with sizes ranging from 22M to 0.6B parameters, using the **REBU-Syn** dataset. The results of these experiments on 6 common test benchmarks are detailed in Table 5. As shown in Fig 3, the PARSeq demonstrates a similar trend to that observed with TrOCR, reinforcing the existence of the power law on model size. The power law of the PARSeq model is as follows.

$$E(N) = \left(6.316 * 10^{-74}/N\right)^{0.018} \quad (2)$$

Model	Avg	Regular Text			Irregular Text		
		IC13 857	IIIT5k 3,000	SVT 647	CUTE80 288	IC15 1,811	SVTP 645
PARSeq-S	96.85	98.72	98.63	98.45	99.65	92.27	95.97
PARSeq-B	96.96	99.07	98.53	**98.76**	99.31	**92.44**	96.74
PARSeq-L	97.03	**99.30**	98.63	98.61	99.31	92.32	97.21
PARSeq-H	**97.06**	99.11	**98.93**	98.45	**99.65**	91.66	**97.67**

Table 5. Word accuracy with different model size of PARSeq. Train data: REBU-Syn.

4.1.2 The power-law of data when model size is fixed.

- **Scaling Data Volume on TrOCR** In order to explore the impact of data volume on model performance. We use MJ+ST and $MJST^+$ for training TrOCR-B. We randomly sampled data at various scales, with sizes ranging from 0.75M to 75M. The experimental results of TrOCR-B based on data of different scales are compiled in Table 6. We used various levels of data volume (solid blue line) to fit the power function (Eq. 3) as shown by the solid grey line in Fig. 2b. The remaining portion of the data size (represented by the dashed blue line) still closely follows the power function, providing further evidence that the data volume adheres to the power function.

$$E(D) = \left(1.84 * 10^5/D\right)^{-0.327} \quad (3)$$

Data Volume	Volume	Avg	Regular Text			Irregular Text		
			IC13 857	IIIT5k 3,000	SVT 647	CUTE80 288	IC15 1,811	SVTP 645
5%	0.75M	50.35	64.53	57.77	47.60	29.51	39.98	38.14
10%	1.50M	52.61	64.18	59.43	55.63	33.68	42.13	42.33
25%	3.75M	74.86	86.11	79.30	80.22	60.07	63.83	71.47
50%	7.50M	76.47	86.35	79.73	80.83	60.07	64.88	72.40
100%	15.00M	88.56	96.14	92.00	91.56	80.56	81.14	83.91
500%	75.00M	**93.09**	**97.32**	**93.51**	**96.33**	**89.93**	**86.47**	**91.47**

Table 6. TrOCR-B average accuracy in different percent of training data.

- **Scaling Data Volume on PARSeq** Based on the power law of data volume, we utilize REBU-Syn in PARSeq-S training. By gradually expanding the data samples, the accuracy of PARSeq-S has been significantly improved in the Table 7.

Data	Volume	Avg	Regular Text			Irregular Text		
			IC13 857	IIIT5k 3,000	SVT 647	CUTE80 288	IC15 1,811	SVTP 645
R	3.30M	95.57	97.32	97.87	97.37	97.22	90.34	94.73
R+B+E	3.32M	95.63	97.43	97.97	97.84	98.96	90.28	93.64
R+B+E+U	6.42M	96.12	**99.53**	97.93	97.53	98.96	91.39	95.66
R+B+E+U+MJST	20.82M	96.45	98.48	98.40	97.84	98.61	91.44	96.43
R+B+E+U+MJST+Syn	23.82M	**96.85**	98.93	**98.63**	**98.61**	**99.31**	**92.32**	**97.21**

Table 7. PARSeq-S average accuracy in different percent of training data.

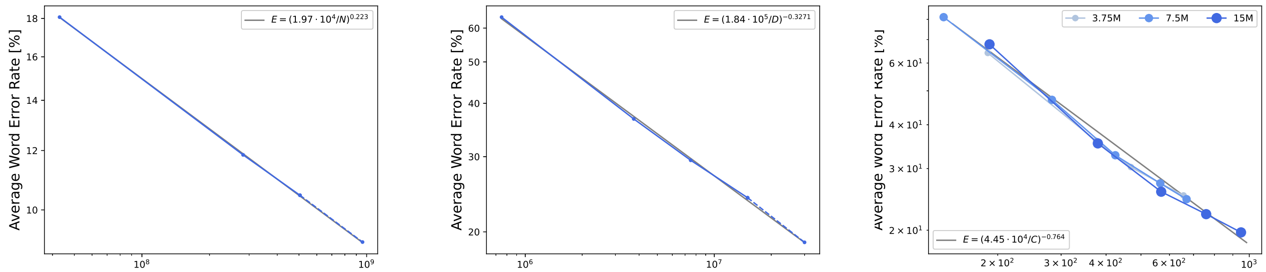
4.1.3 The power law of computation

With the power laws of model size and data volume separately, we infer that the error rate and the compute budget can also be fit with the power law. We perform the study on TrOCR model. The outcome is depicted as the gray line on the plot on the right-hand side of Fig. 2c. It can be fitted with the power formula as in Eq. 4.

$$E(C) = \left(4.45 * 10^4/C\right)^{-0.3271} \quad (4)$$

4.2. Other Observations

Large-scale models make better use of samples. As we continue to enlarge the model size, the model accuracy improves consistently. The phenomenon can be observed in



(a) Model sizes (Params)

(b) Data volumes (M)

(c) Computation (training hours)

Figure 2. Improvement in TrOCR model performance with increasing model sizes, data volumes, and training computation. Model performance is measured by calculating the average word error rate on 6 common test benchmarks **Left**: Evaluation of model performance with changing model sizes. **Center**: Evaluation of model performance with varying data volumes. **Right**: Performance variations with different data sizes under varying computational resources. The x-axis represents the model's training time, measured in 8 GPU hours. For optimal performance, all three factors must be scaled up in tandem. Empirical performance exhibits a power-law relationship with each individual factor when it is not constrained by the other two factors.

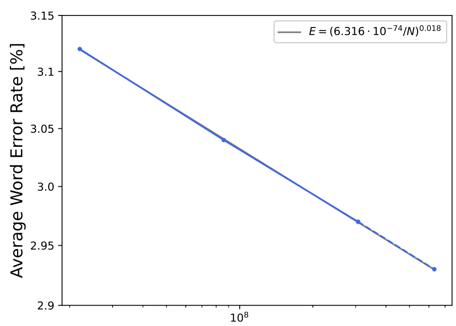


Figure 3. The average word error rate on 6 common test benchmarks was calculated using the PARSeq model size. The solid line represents the fitted power law $E(\cdot)$, and the points on the dotted line correspond to the power law equation.

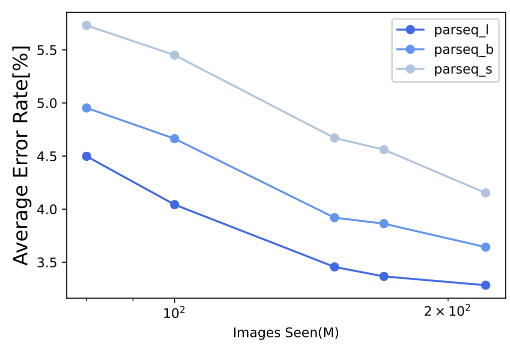


Figure 4. Average word error rate on 6 common test benchmarks, with respect to images seen (batch size times number of steps) during PARSeq training stage of different sizes.

Table 4 and 5. To improve training results, we can modify recognition models built on the Transformer architecture by utilizing the scaling laws of the vision Transformer. As shown in Figure 4 with respect to the total number of images "seen" during PARSeq training stage of different sizes, it is clear that larger models utilize samples more effectively than their smaller models. When PARSeq models of different sizes are trained with an equal number of samples, smaller models exhibit a higher error rate compared to larger models. Furthermore, we observed that larger models tend to require fewer epochs to converge. For instance, PARSeq-S reached optimal accuracy in 32 epochs, whereas PARSeq-B needed only 14 epochs, and PARSeq-L just 5 epochs. These findings suggest that with adequate training resources, it is more beneficial to train a larger model for fewer steps. This mirrors similar findings in language modelling [24] and machine translation [17]. However, when training time is a limiting factor, opting for a smaller model may be more practical.

The proportion of training data from various sources is crucial for model training. The REBU-Syn comprises both real and synthetic data. According to prior studies [6, 73], real data typically outperforms synthetic data in training efficiency, though synthetic data still plays a valuable role. Due to the high costs associated with obtaining and labeling real data, which often do not meet the volume requirements for model training, reliance on synthetic data is necessary. However, the effectiveness of synthetic data raises a question: Does more synthetic data always equate to better performance? Our findings suggest that an optimal ratio between real and synthetic data is crucial for enhanced model performance.

To achieve this objective, we conducted an experiment to investigate the proportional relationship between data obtained from various sources and determine the most efficient utilization of synthetic data. Synthetic data can be primarily categorized into two types: MJ+ST and Syn (CST+SA). MJ+ST, characterized by its large volume but homogeneous nature (consisting mostly of straight and clear samples), contrasts with SynText, which has a smaller volume (only one-fifth of MJ+ST) and primarily consists of curved texts. To evaluate the impact of different synthetic data sources on model accuracy, we trained PARSeq using a combination of real data and these synthetic datasets. The results, as presented in Table 8, are revealing. The accuracy achieved using real data combined with MJ+ST is 96.24%, only marginally higher by 0.05% compared to using real data with Syn. Given that the volume of MJ+ST is five times that of Syn, this implies that complex synthetic data is more sample-efficient. By simultaneously utilizing synthetic data from both MJ+ST and SynText along with real data, we observed a substantial enhancement in the accuracy of PARSeq, elevating it to state-of-the-art levels. This combination of diverse synthetic data styles, when integrated with real data, expands the range of the training data distribution. Such comprehensive coverage effectively enhances the overall quality and performance of the model.

Real DataSet	Syn DataSet	Data Ratio	Word Acc
R+E+B+U	Syn	1:0.5	96.19
R+E+B+U	MJ+ST	1:2.5	96.24
R+E+B+U	MJ+ST+Syn	1:3	**96.85**

Table 8. PARSeq-S average accuracy of integrating diverse synthetic and real data types.

Additionally, we investigated the effect of different synthetic-to-real data ratios on the accuracy of PARSeq-S. We maintained a constant amount of real data and progressively increased the volume of synthetic data. The ratio of synthetic to real data varied from 0.5 to 5 times. These varying proportions were achieved through random sampling. To augment the total volume of synthetic data, we randomly selected 18M samples from $MJST^{+}$ and combined them with the synthetic data in REBU-Syn, culminating in a total of 36M synthetic data samples.

Data Ratio	Word Acc
Real:Syn=1:0.5	96.32
Real:Syn=1:1	96.50
Real:Syn=1:2	96.59
Real:Syn=1:3	**96.85**
Real:Syn=1:4	96.76
Real:Syn=1:5	95.70

Table 9. PARSeq-S average accuracy on 6 common test benchmarks with varying ratios of synthetic and real data.

While synthetic data proves effective, it requires careful balancing with real data. As shown in Table 9, a gradual increase in synthetic data led to some improvement in accuracy. Notably, the highest accuracy of 96.85% was achieved with a synthetic-to-real data ratio of 1:3. Beyond this ratio, the accuracy began to decline, likely due to the data distribution becoming overly skewed towards synthetic data, which can adversely affect model performance. <u>Therefore, we recommend a real-to-synthetic data ratio of 1:3.</u> This balance offers the most significant improvement in accuracy without incurring excessive training costs.

Pre-training Data	Training Data	Backbone	Word Acc
Scratch	R+E+B+U	ViT-S	96.12
Scratch	R+E+B+U+Syn	ViT-S	96.85
R+E+B+U+Syn	R+E+B+U	ViT-S	**96.96**
Scratch	R+E+B+U+Syn	ViT-L	97.03
ImageNet-21k	R+E+B+U+Syn	ViT-L	96.74

Table 10. Average accuracy achieved by using visual task pre-training and STR task pre-training on 6 common test benchmarks.

Task-related pre-trained models are more effective. The utility of pretrained models in low-level vision tasks is well-known, but their applicability in STR tasks warrants investigation. To address this, we experimented with various pretrained models, some trained on ImageNet and others specifically for text recognition tasks. In the last two rows of Table 10, we maintained consistent training schedules, learning rates and epochs as used for PARSeq. Intriguingly, the ImageNet-21k pre-trained models underperformed compared to those trained from scratch, a trend observed in both PARSeq and CLIP4STR models. This suggests that pretraining on non-STR-specific tasks might not be beneficial, and can even be detrimental to STR performance. The STR task necessitates a connection between visual and textual elements, akin to the CLIP experiment's original purpose, whereas purely visual tasks focus more on high-level semantics and lack the textual nuances critical for STR.

Additionally, when we trained PARSeq-S using the REBU-Syn dataset, it achieved a higher accuracy of 96.85% compared to training solely on the real data REBU. Further fine-tuning the 96.85% model with REBU led to an increased accuracy of 97.01%, indicating an improvement. This demonstrates the efficacy of task-related pretrained models in STR tasks. To attain higher accuracy, a recommended approach is <u>training on all data first and then fine-tune on real data.</u>

4.3. Comparison with SOTA Methods

Recently, the remarkable performance of CLIP4STR across multiple benchmarks prompted us to conduct further experiments, guided by our scaling law. Initially, we focused on data composition, employing a 3:1 ratio of synthetic to real data for training the model, in conjunction with fine-tuning using a pre-trained model for related tasks. Our reproducible results led to a notable improvement in CLIP4STR-B, enhancing its accuracy from 96.54% to 97.25%, an in-

Method	Training data	Regular Text			Irregular Text			Avg
		IC13 857	IIIT5k 3,000	SVT 647	CUTE80 288	IC15 1,811	SVTP 645	
ViTSTR-B [1]	MJ+ST	92.40	88.40	87.70	81.30	72.60	81.80	85.46
SE-ASTER [44]	MJ+ST	92.80	93.80	89.60	83.60	80.0	81.40	88.35
TRBA [3]	MJ+ST	93.4	92.10	88.90	89.20	77.4	89.30	90.07
SRN [69]	MJ+ST	95.5	94.80	91.50	87.80	82.70	85.10	90.42
TextScanner [61]	MJ+ST	94.90	95.70	92.70	91.60	83.50	84.80	91.15
VisionLAN [65]	MJ+ST	95.7	95.80	91.70	88.50	83.7	86.00	91.23
PETR [66]	MJ+ST	97.00	95.80	92.40	89.90	83.30	86.20	91.42
ABINet [16]	MJ+ST	97.4	96.20	93.50	89.20	86.00	89.30	92.66
MaskOCR(ViT-B) [37]	MJ+ST	98.1	95.8	94.7	89.2	87.3	89.9	93.1
PARSeq$_A$ [6]	MJ+ST	96.20	97.00	93.60	92.20	82.90	88.90	93.16
MATRN [39]	MJ+ST	95.80	96.60	95.00	93.50	82.80	90.60	93.20
TrOCR$_{Large}$ [31]	MJ+ST+B	97.30	94.10	96.10	95.10	84.10	93.00	93.23
DiG-ViT-B [68]	MJ+ST	96.90	96.70	94.60	91.30	87.10	91.00	93.41
MaskOCR(ViT-L) [37]	MJ+ST	98.2	98.0	96.9	95.8	90.1	94.6	93.78
ViTSTR-S [1]	Real	97.80	97.90	96.00	96.20	89.00	91.50	94.85
DiG-ViT-B [68]	Real	97.60	97.60	96.50	96.50	88.90	92.90	94.86
PARSeq$_A$$^\sharp$ [6]	Real	97.32	97.87	97.37	97.22	90.34	94.73	95.57
ABINet [16]	Real	98.00	98.60	98.20	97.20	90.50	94.10	96.01
MAERec-B [23]	Union14M-L	98.10	98.50	97.80	98.60	90.7	94.40	96.15
CLIP4STR-B [73]	Real	98.36	98.73	97.68	98.96	91.39	96.74	96.89
CLIP4STR-L [73]	Real	98.48	**99.43**	98.15	98.96	91.66	97.36	97.06
CLIP4STR-B*	REBU-Syn	99.29	98.96	**98.76**	99.65	92.27	97.83	97.25
CLIP4STR-L*	REBU-Syn	**99.42**	99.13	98.61	**99.65**	**92.6**	**98.13**	**97.42**

Table 11. Word accuracy on 6 common test benchmarks, * indicates training with REBU-Syn, Avg is the weighted average result on 6 common test benchmarks. $\sharp$ indicates reproduced by us.

crease of 0.65%. This achievement represents the best result to date in the text recognition task.

To further delve into the impact of larger models, we replicated this experiment on CLIP4STR-L. This model achieved a new state-of-the-art, recording a top-1 average accuracy of 97.42% on 6 common test benchmarks, as detailed in Table 11. These findings underscore the significant role of large models in advancing the field of STR.

5. Discussion and Conclusion

In this paper, we demonstrate the existence of smooth power laws in the field of STR. Specifically, we show that increasing the model size, data volume, and computational resources leads to predictable improvements in model performance. Additionally, we identified several key principles for effective model training in STR: 1) Large-scale models utilize samples more efficiently. 2) The proportion of training data from various sources is crucial for model training; 3) Task-related pre-trained models enhance effectiveness. Beyond identifying these guiding principles, we compiled a large-scale dataset to improve the performance of the STR model. Leveraging these rules, we successfully trained a model that achieved a new state-of-the-art average accuracy of 97.42% on the test benchmark.

We conduct extensive experiments on both model scaling and data scaling, successfully demonstrating the existence of scaling laws in STR. Furthermore, we observe that data scaling is a particularly advantageous approach as it enhances model accuracy without incurring additional costs during training or inference. However, challenges persist in the realm of model scaling. While large-scale models exhibit superior performance with substantial data, their training is considerably more costly. Adjusting each parameter can be extremely expensive, with each training iteration potentially costing millions of dollars. To optimize the performance of these models, careful selection of the best hyperparameters during training is essential. We hope that our work can attract more researchers' attention to reduce the training cost of large-scale models.

Our experiments are based on large-scale natural scene text data sets. In the future, we will consider exploring the scaling law in more challenging text recognition data sets such as handwriting and historical texts.

References

[1] Rowel Atienza. Vision transformer for fast and efficient scene text recognition. 2021. 8

[2] Jeonghun Baek, Geewook Kim, Junyeop Lee, Sungrae Park, Dongyoon Han, Sangdoo Yun, Seong Joon Oh, and Hwalsuk Lee. What is wrong with scene text recognition model comparisons? dataset and model analysis. In Proceedings of the IEEE/CVF international conference on computer vision, pages 4715–4723, 2019. 2

[3] Jeonghun Baek, Yusuke Matsui, and Kiyoharu Aizawa. What if we only use real datasets for scene text recognition? toward scene text recognition with fewer labels. 2021. 2, 8

[4] Hangbo Bao, Li Dong, Songhao Piao, and Furu Wei. Beit: Bert pre-training of image transformers. arXiv preprint arXiv:2106.08254, 2021. 1, 3

[5] Andrei Barbu, David Mayo, Julian Alverio, William Luo, Christopher Wang, Dan Gutfreund, Josh Tenenbaum, and Boris Katz. Objectnet: A large-scale bias-controlled dataset for pushing the limits of object recognition models. Advances in neural information processing systems, 32, 2019. 2

[6] Darwin Bautista and Rowel Atienza. Scene text recognition with permuted autoregressive sequence models. 2022. 3, 4, 6, 8

[7] Fedor Borisyuk, Albert Gordo, and Viswanath Sivakumar. Rosetta: Large scale system for text detection and recognition in images. In Proceedings of the 24th ACM SIGKDD international conference on knowledge discovery & data mining, pages 71–79, 2018. 2

[8] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. Advances in neural information processing systems, 33:1877–1901, 2020. 1

[9] Zhanzhan Cheng, Fan Bai, Yunlu Xu, Gang Zheng, Shiliang Pu, and Shuigeng Zhou. Focusing attention: Towards accurate text recognition in natural images. In Proceedings of the IEEE international conference on computer vision, pages 5076–5084, 2017. 2

[10] Rewon Child, Scott Gray, Alec Radford, and Ilya Sutskever. Generating long sequences with sparse transformers. arXiv preprint arXiv:1904.10509, 2019. 2

[11] Chee Kheng Ch'ng and Chee Seng Chan. Total-text: A comprehensive dataset for scene text detection and recognition. In 2017 14th IAPR international conference on document analysis and recognition (ICDAR), pages 935–942. IEEE, 2017. 3

[12] Chee-Kheng Chng, Yuliang Liu, Yipeng Sun, Chun Chet Ng, Canjie Luo, Zihan Ni, ChuanMing Fang, Shuaitao Zhang, Junyu Han, Errui Ding, Jingtuo Liu, Dimosthenis Karatzas, Chee Seng Chan, and Lianwen Jin. Icdar2019 robust reading challenge on arbitrary-shaped text (rrc-art). 2019. 3

[13] Mostafa Dehghani, Stephan Gouws, Oriol Vinyals, Jakob Uszkoreit, and Łukasz Kaiser. Universal transformers. arXiv preprint arXiv:1807.03819, 2018. 2

[14] Mostafa Dehghani, Josip Djolonga, Basil Mustafa, Piotr Padlewski, Jonathan Heek, Justin Gilmer, Andreas Steiner, Mathilde Caron, Robert Geirhos, Ibrahim Alabdulmohsin, et al. Scaling vision transformers to 22 billion parameters. arXiv preprint arXiv:2302.05442, 2023. 1

[15] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805, 2018. 2

[16] Shancheng Fang, Hongtao Xie, Yuxin Wang, Zhendong Mao, and Yongdong Zhang. Read like humans: Autonomous, bidirectional and iterative language modeling for scene text recognition. 2021. 8

[17] Patrick Fernandes, Behrooz Ghorbani, Xavier Garcia, Markus Freitag, and Orhan Firat. Scaling laws for multilingual neural machine translation. 2023. 6

[18] Behrooz Ghorbani, Orhan Firat, Markus Freitag, Ankur Bapna, Maxim Krikun, Xavier Garcia, Ciprian Chelba, and Colin Cherry. Scaling laws for neural machine translation. arXiv preprint arXiv:2109.07740, 2021. 1

[19] Ankush Gupta, Andrea Vedaldi, and Andrew Zisserman. Synthetic data for text localisation in natural images. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 2315–2324, 2016. 4

[20] Tom Henighan, Jared Kaplan, Mor Katz, Mark Chen, Christopher Hesse, Jacob Jackson, Heewoo Jun, Tom B Brown, Prafulla Dhariwal, Scott Gray, et al. Scaling laws for autoregressive generative modeling. arXiv preprint arXiv:2010.14701, 2020. 2

[21] Mingxin Huang, Yuliang Liu, Zhenghao Peng, Chongyu Liu, Dahua Lin, Shenggao Zhu, Nicholas Yuan, Kai Ding, and Lianwen Jin. Swintextspotter: Scene text spotting via better synergy between text detection and text recognition. In proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pages 4593–4603, 2022. 4

[22] Max Jaderberg, Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. Synthetic data and artificial neural networks for natural scene text recognition. arXiv preprint arXiv:1406.2227, 2014. 4

[23] Qing Jiang, Jiapeng Wang, Dezhi Peng, Chongyu Liu, and Lianwen Jin. Revisiting scene text recognition: A data perspective. 2023. 8

[24] Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. Scaling laws for neural language models. arXiv preprint arXiv:2001.08361, 2020. 1, 2, 3, 6

[25] Dimosthenis Karatzas, Faisal Shafait, Seiichi Uchida, Masakazu Iwamura, Lluis Gomez i Bigorda, Sergi Robles Mestre, Joan Mas, David Fernandez Mota, Jon Almazan Almazan, and Lluis Pere De Las Heras. Icdar 2013 robust reading competition. pages 1484–1493, 2013. 3, 4

[26] Dimosthenis Karatzas, Lluis Gomez-Bigorda, Anguelos Nicolaou, Suman Ghosh, Andrew Bagdanov, Masakazu Iwamura, Jiri Matas, Lukas Neumann, Vijay Ramaseshan Chandrasekhar, Shijian Lu, et al. Icdar 2015 competition on robust reading. pages 1156–1160, 2015. 3, 4

[27] Ilya Krylov, Sergei Nosov, and Vladislav Sovrasov. Open images v5 text annotation and yet another mask text spotter.

In Asian Conference on Machine Learning, pages 379–389. PMLR, 2021. 3

[28] Taku Kudo and John Richardson. Sentencepiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. 2018. 4

[29] Chen-Yu Lee and Simon Osindero. Recursive recurrent nets with attention modeling for ocr in the wild. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 2231–2239, 2016. 2

[30] Dmitry Lepikhin, HyoukJoong Lee, Yuanzhong Xu, Dehao Chen, Orhan Firat, Yanping Huang, Maxim Krikun, Noam Shazeer, and Zhifeng Chen. Gshard: Scaling giant models with conditional computation and automatic sharding. 2020. 2

[31] Minghao Li, Tengchao Lv, Jingye Chen, Lei Cui, Yijuan Lu, Dinei Florencio, Cha Zhang, Zhoujun Li, and Furu Wei. Trocr: Transformer-based optical character recognition with pre-trained models. arXiv preprint arXiv:2109.10282, 2021. 2, 3, 8

[32] Xianhang Li, Zeyu Wang, and Cihang Xie. An inverse scaling law for clip training. arXiv preprint arXiv:2305.07017, 2023. 2

[33] Peter J Liu, Mohammad Saleh, Etienne Pot, Ben Goodrich, Ryan Sepassi, Lukasz Kaiser, and Noam Shazeer. Generating wikipedia by summarizing long sequences. arXiv preprint arXiv:1801.10198, 2018. 2

[34] Wei Liu, Chaofeng Chen, and Kwan-Yee Wong. Char-net: A character-aware neural network for distorted scene text recognition. In Proceedings of the AAAI conference on artificial intelligence, 2018. 2

[35] Xi Liu, Rui Zhang, Yongsheng Zhou, Qianyi Jiang, Qi Song, Nan Li, Kai Zhou, Lei Wang, Dong Wang, Minghui Liao, Mingkun Yang, Xiang Bai, Baoguang Shi, Dimosthenis Karatzas, Shijian Lu, and C. V. Jawahar. Icdar 2019 robust reading challenge on reading chinese text on signboard. 2019. 3

[36] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized bert pretraining approach. arXiv preprint arXiv:1907.11692, 2019. 3

[37] Pengyuan Lyu, Chengquan Zhang, Shanshan Liu, Meina Qiao, Yangliu Xu, Liang Wu, Kun Yao, Junyu Han, Errui Ding, and Jingdong Wang. Maskocr: text recognition with masked encoder-decoder pretraining. arXiv preprint arXiv:2206.00311, 2022. 8

[38] Anand Mishra, Karteek Alahari, and CV Jawahar. Scene text recognition using higher order language priors. BMVC-British machine vision conference, 2012. 3, 4

[39] Byeonghu Na, Yoonsik Kim, and Sungrae Park. Multi-modal text recognition networks: Interactive enhancements between visual and semantic features. 2022. 8

[40] Nibal Nayef, Yash Patel, Michal Busta, Pinaki Nath Chowdhury, Dimosthenis Karatzas, Wafa Khlif, Jiri Matas, Umapada Pal, Jean-Christophe Burie, Cheng lin Liu, and Jean-Marc Ogier. Icdar2019 robust reading challenge on multi-lingual scene text detection and recognition – rrc-mlt-2019. 2019. 3

[41] OpenAI. Chatgpt. https://openai.com/blog/chatgpt/, 2023. 2

[42] OpenAI. Gpt-4 technical report. 2023. 2

[43] Trung Quy Phan, Palaiahnakote Shivakumara, Shangxuan Tian, and Chew Lim Tan. Recognizing text with perspective distortion in natural scenes. pages 569–576, 2013. 4

[44] Zhi Qiao, Yu Zhou, Dongbao Yang, Yucan Zhou, and Weiping Wang. Seed: Semantics enhanced encoder-decoder framework for scene text recognition. 2020. 8

[45] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In International conference on machine learning, pages 8748–8763. PMLR, 2021. 2

[46] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. The Journal of Machine Learning Research, 21(1):5485–5551, 2020. 2

[47] Zobeir Raisi, Mohamed A Naiel, Georges Younes, Steven Wardell, and John S Zelek. Transformer-based text detection in the wild. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 3162–3171, 2021. 2

[48] Benjamin Recht, Rebecca Roelofs, Ludwig Schmidt, and Vaishaal Shankar. Do imagenet classifiers generalize to imagenet? In International conference on machine learning, pages 5389–5400. PMLR, 2019. 2

[49] Anhar Risnumawan, Palaiahankote Shivakumara, Chee Seng Chan, and Chew Lim Tan. A robust arbitrary text detection system for natural scene images. Expert Systems with Applications, 41(18):8027–8048, 2014. 4

[50] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. International journal of computer vision, 115:211–252, 2015. 2

[51] Rico Sennrich, Barry Haddow, and Alexandra Birch. Neural machine translation of rare words with subword units. 2016. 4

[52] Fenfen Sheng, Zhineng Chen, and Bo Xu. Nrtr: A no-recurrence sequence-to-sequence model for scene text recognition. In 2019 International conference on document analysis and recognition (ICDAR), pages 781–786. IEEE, 2019. 2

[53] Baoguang Shi, Xiang Bai, and Cong Yao. An end-to-end trainable neural network for image-based sequence recognition and its application to scene text recognition. IEEE transactions on pattern analysis and machine intelligence, 39 (11):2298–2304, 2016. 2

[54] Baoguang Shi, Cong Yao, Minghui Liao, Mingkun Yang, Pei Xu, Linyan Cui, Serge Belongie, Shijian Lu, and Xiang Bai. Icdar2017 competition on reading chinese text in the wild (rctw-17). 2018. 3

[55] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556, 2014. 2

[56] Amanpreet Singh, Guan Pang, Mandy Toh, Jing Huang, Wojciech Galuba, and Tal Hassner. Textocr: Towards large-scale end-to-end reasoning for arbitrary-shaped scene text. 2021. 3

[57] Yipeng Sun, Jiaming Liu, Wei Liu, Junyu Han, Errui Ding, and Jingtuo Liu. Chinese street view text: Large-scale chinese text reading with partially supervised learning. 2020. 3

[58] Yew Lee Tan, Adams Wai-Kin Kong, and Jung-Jae Kim. Pure transformer with integrated experts for scene text recognition. In Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXVIII, pages 481–497. Springer, 2022. 2

[59] Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Hervé Jégou. Training data-efficient image transformers & distillation through attention. pages 10347–10357, 2021. 3

[60] Andreas Veit, Tomas Matera, Lukas Neumann, Jiri Matas, and Serge Belongie. Coco-text: Dataset and benchmark for text detection and recognition in natural images. 2016. 3

[61] Zhaoyi Wan, Minghang He, Haoran Chen, Xiang Bai, and Cong Yao. Textscanner: Reading characters in order for robust scene text recognition. CoRR, abs/1912.12422, 2019. 8

[62] Kai Wang, Boris Babenko, and Serge Belongie. End-to-end scene text recognition. pages 1457–1464, 2011. 3, 4

[63] Qiang Wang, Bei Li, Tong Xiao, Jingbo Zhu, Changliang Li, Derek F Wong, and Lidia S Chao. Learning deep transformer models for machine translation. arXiv preprint arXiv:1906.01787, 2019. 3

[64] Wenhui Wang, Furu Wei, Li Dong, Hangbo Bao, Nan Yang, and Ming Zhou. Minilm: Deep self-attention distillation for task-agnostic compression of pre-trained transformers. Advances in Neural Information Processing Systems, 33: 5776–5788, 2020. 3

[65] Yuxin Wang, Hongtao Xie, Shancheng Fang, Jing Wang, Shenggao Zhu, and Yongdong Zhang. From two to one: A new scene text recognizer with visual language modeling network. 2021. 8

[66] Yuxin Wang, Hongtao Xie, Shancheng Fang, Mengting Xing, Jing Wang, Shenggao Zhu, and Yongdong Zhang. Petr: Rethinking the capability of transformer-based language model in scene text recognition. IEEE Transactions on Image Processing, 31:5585–5598, 2022. 8

[67] I. Zharkov P. Zhang K. Seifert Y. Zhang, L. Gueguen and B. Kadlec. Uber-text: A large-scale dataset for optical character recognition from street-level imagery. Scene Understanding Workshop-CVPR, 2017. 3

[68] Mingkun Yang, Minghui Liao, Pu Lu, Jing Wang, Shenggao Zhu, Hualin Luo, Qi Tian, and Xiang Bai. Reading and writing: Discriminative and generative modeling for self-supervised text recognition. 2023. 8

[69] Deli Yu, Xuan Li, Chengquan Zhang, Junyu Han, Jingtuo Liu, and Errui Ding. Towards accurate scene text recognition with semantic reasoning networks. CoRR, abs/2003.12294, 2020. 8

[70] Deli Yu, Xuan Li, Chengquan Zhang, Tao Liu, Junyu Han, Jingtuo Liu, and Errui Ding. Towards accurate scene text recognition with semantic reasoning networks. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 12113–12122, 2020. 4

[71] Liu Yuliang, Jin Lianwen, Zhang Shuaitao, and Zhang Sheng. Detecting curve text in the wild: New dataset and new solution. arXiv preprint arXiv:1712.02170, 2017. 3

[72] Xiaohua Zhai, Alexander Kolesnikov, Neil Houlsby, and Lucas Beyer. Scaling vision transformers. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 12104–12113, 2022. 1, 2, 3

[73] Shuai Zhao, Xiaohan Wang, Linchao Zhu, Ruijie Quan, and Yi Yang. Clip4str: A simple baseline for scene text recognition with pre-trained vision-language model. 2023. 6, 8