Journal of Kerbala University, Vol. 23, Issue 1, March , 2026

Available at https://www.iasj.net/iasj

Iraqi Academic Scientific Journals

Journal homepage:
https://journals.uokerbala.edu.iq/index.php/UOKJ

Research Article

Sequence-to-Sequence Text Completion Using the T5 Language Model

Zahraa Kadhim Al-Sendi

Department of Computer Science , College of Computer Science and Information Technology , University of Kerbala

Article Info

Article history:
Received 2 -12-2025
Received in revised form 16-12-2025
Accepted 5-2-2026
Available online 31 - 3 -2026

Keywords:
Sequence-to-Sequence Learning, Text Completion, T5 Language Model, Transformer Architecture, Natural Language Processing (NLP), Deep Learning, Text Generation, Pre-trained Language Models, Encoder–Decoder Models

Abstract

Sequence-to-sequence (seq2seq) text generation is a fundamental problem in natural language processing, which underlies applications such as chat-bots, machine translation and creative writing. This research investigates the fine-tuning of pre-trained Text-to-Text Transfer Transformer models, such as T5-base and T5-large on a conversational dataset, concerning their ability to produce contextually coherently and semantically relevant textual completions. The models were trained on the DailyDialog dataset with a maximum input length of 256 tokens. They were evaluated using both automatic metrics (accuracy, perplexity and BLEU scores) and diversity and fluency in qualitative evaluations.

The larger T5-large model outperformed the smaller one. It achieved a next-sentence prediction accuracy of approximately 97.9%, at a final loss in the range between 0.7 and 1.2, and produced responses that were nearly indistinguishable from human naturalness judging with perplexity scores but had good variety and conveyed interesting, conversational content or knowledge. T5-base; however, showed competitive performance with less precision and diversity. That made it a good fit for situations that have limited computing resources. These findings demonstrate the importance of trade-offs between model size and resource constraints for seq2seq models in text completion applications.

Our results are in line with other work that highlights the high generalization capabilities of T5 across varied NLP tasks, and suggest that it has strong potential for efficient language synthesis, given adequate optimization. This work is part of ongoing efforts to enable better dialogue systems and provides practical guidance for applying T5 variants for conversational AI.

Corresponding Author E-mail: zahraa.k@uokerbala.edu.iq

Peer review under responsibility of Iraqi Academic Scientific Journal and University of Kerbala.

Introduction:

Seq2seq (sequence-to-sequence) text completion is regarded as a canonical problem to solve in NLP. The concept of S2S models is generating contextually relevant continuations for text with incomplete string. This close coupling with sequence prediction gives rise to a broad set of applications including autoregressive writing tools, programming code completion, chat-bots and large-scale content generation systems. This recent advances in this area have been cherished with the recommendation of scalable pertained LM, where seq2seq architectures shine as strong complex input-output mappers. In fact, the trend of string processing studies seems to be a very real concern, and an absolutely necessary step towards a new path. String processing in itself is not trivial, as the texts will be continuous and needs a valid and precise reaction. Its usefulness depends on being responsive, swift and accurate.

Raffel et al. (2020a) presenting the Text-to-Text Transfer Transformer (T5), a new groundbreaking model that converts all NLP tasks to be text-to-text with the same model using an encoder-decoder transformer architecture. Contrary of most task-specific models, T5 uses natural language strings (e.g., “question:”) as prefixes that control generation, allowing it to perform text completion, summarization, translation and question answering. [1]Grounded in Vaswani et al. s (2017) Transformer that leverages self-attention to capture long-range dependencies, T5's encoder-decoder architecture supports various length outputs such as dynamic completion settings. Trained on span corruption of the 750GB Colossal Clean Crawled Corpus (C4), T5 acquires powerful language generation, which most likely generalizes further with training. [2]

T5Flex: The next step was to make T5 even more flexible. LongT5 (Guo et al., 2022) uses Transient Global attention to allow for efficient long-sequence computation and shows significantly improved performance on both summarization and question answering. [3]EdiT5 (Mallinson et al., 2022) achieves up to 25x faster inference via semi-autoregressive editing while maintaining generation quality. [4] Multilingual variants such as mT5 and mLongT5 show good performance even in low resource languages, while language specialized models such as ViT5 (Vietnamese) and HyperT5 (Korean) report state of the art results. Application to domain the applications covered with T5 are code gene ration (where T5 generates log statements) and support for other programming tasks.

Extending financial comparison, more comparative analyses confirm the T5 supremacy: Araujo et al. (2023) showed that seq2seq T5-based models outperform decoder-only architectures for Spanish generation tasks, and multilingual or single encoder models underperform on Slovenian summarization & simplification, compared to monolingual towers of T5. For example, METRO-T0 demonstrates that custom pre-training objectives can be leveraged to obtain state-of-the-art performance with significantly smaller backbone models. [5]

Nevertheless, challenges remain, such as the inference burden of the encoder-decoder w.r.t. decoder-only models, and problems of hallucination, repetition and stylistic control in long sequence generations. We fine-tune T5-base and T5-large on the DailyDialog dataset to examine their seq2seq response generation performance, while controlling the model’s capacity and resource limitations at the same time. Combining the architectural intuition, empirical findings and optimization techniques enables us to take T5-to-Tutor in both general domain and field specific settings a step further.

Therefore, S2S processing can manifest in several ways. As previously shown, the T5

model has been widely used in various text-processing applications. As an introduction, the model has been used with numerous natural language processing applications, so it is best to test it on sequence processing to demonstrate its effectiveness.

Related Work:

Sequence-to-sequence (seq2seq) is one of the most widely-used architectures in modern NLP. They are useful for translation, summarization, conversation and text completion. Raffel et al. (2020) introduced the Text-to-Text Transfer Transformer (T5), which consolidated multiple NLP tasks under a unified text-to-text framework. This allowed rapid transfer of learning between different domains. Follow-up work has refined and extended this paradigm in terms of reasoning capability, pertaining objective, efficiency, multimodal adaptation and domain-specific generation.[6] Several research papers will be presented that relate to the model we will adopt as a training method.

In the opening of the study of models in the following research, T5 model was used that was combined with BERT. Lin et al. (2020) demonstrated how the generative capacity of T5 when cast as a text input to text target model has allowed for the adaptation of Winogrande robot's common sense reasoning challenge into a text-to-text inference problem. They found that encoder–decoder architecture outperformed encoder-only models such as BERT and large-scale pertaining allowed T5 to learn implicit reasoning towards the target from data.[7] Zhou et al. (2021) also proposed Sequence Span Rewriting (SSR), a self-supervised pertaining object in which the masked infilling was replaced by rewriting realistic text. In this way, SSR provided increased supervision and better performance on NLP problems such as summarization, question generation and grammatical correction by teaching the model to convert spans generated by machines into a real text. This was especially true for smaller models. [7]

Building on the implications pretraining, Mueller et al. (2022) also investigated the effect of large amounts of linguistic data on the rise of hierarchical inductive biases when training seq2seq-type models, such as T5 and BART. Using data until October 2023, they found empirically that it is pretraining — not architectural depth — which allows for hierarchical generalization in the models, and alignes the behavior of neural networks with syntactic principles known to characterize human language. This finding indicates that fluency contributes significantly to text completion continuity. [8]

Mallinson et al. Although Yu et al. (2022) trained EdiT5, which is a semi-autoregressive editing model to accelerate text generation by leveraging sleep and joint training on the operations of tagging, re-ordering and insertion. This model improved generation efficiency. EdiT5 with a reduction in quality as compared to T5 checkpoint inference got up to 25X faster. This demonstrates the strength of hybrid decoding for real-time systems. [4] Kalinsky et al. With (2023) suggesting model wise lighter variants of masked language models for multi-token completion, (2022) showed such architectures with less parameters can compete with even the large seq2seq architectures. This suggests a compromise between the computational burden and the contextual fidelity. [9]

More recent multilingual work extended seq2seq modeling to low-resource languages. GreekT5 (Giarelis et al., 2023) fine-tuned multilingual T5 for Greek summarization, and achieved improvements over monolingual benchmarks, further testifying to the effectiveness of multilingual pre-traininig.[10] Likewise, Spanish T5S (Araujo

et al., 2024) and BARTO1 developed encoder–decoder models specifically for Summarization, question answerer and translation tasks in Spanish language demonstrating the benefits of task-specific pre-training over large monolingual corpora. [5] In education, De-Fitero-Dominguez et al. (2024) mT5++ for automatic distractor generation in Spanish. The output was semantically coherent, and better than that from traditional methods; this supports the hypothesis that multilingual seq2seq architectures are powerful models also when not applied to English.[11]

Fu et al. (2023) investigated whether decoder-only models (e.g., GPT, LLaMA) can replace encoder-decoder architectures for generative tasks. The seq2seq framework is the most popular. They also observed the Attention Degeneration Problem (ADP)—reduction of source attention as sequence length increases—and proposed PALM, a partial attention model that maintains the input focus while generating. They observed that a certain separation between the encoder and decoder, such as in T5, was helpful for long-sequence generation tasks like text completion. [12]

TypeT5 (Wei et al., 2023) mitigated text-only aspect and introduced code type inferrance thusly over a seq2seq generative system through static analysis during an iterative decoding process. The state-of-the-art was established with this model, as its predictions were made across code dependencies, showing that T5 is able to perform structured prediction beyond to text. This combination of symbolic and generative modeling shows just how adaptable seq2seq-type frameworks are for a wide range of context-dependent generation tasks. [13]

Collectively, these studies sketch the evolution of seq2seq modeling from general-purpose pre-training and efficient decoding to multilingual adaptation and domain adaptation. They show that a the T5 architecture forms a strong foundation for structured text generation and completion, where fluency, diversity and coherence are largely determined by model capacity, pretraining objectives and context conditioning. We exploit these overlays, and also fine-tune the T5-base and T5-large models on conversational data to analyse thereof their abilities to complete natural contextually coherent text. This gives us additional insights in terms of how to optimize the tradeoff between performance and computational cost when it comes to seq2seq text generation.

Methodology:

This paper presents a new fine-tuning method for the T5-Large pretrained language model with the DailyDialog dataset, which contains rich multi-turn dialogue data. The main focus was to come up with the items which are: coherent and suitable according to the context. More precisely, the paper pushes forward conversational quality in the model with BLEU scores around 88% as well.

The paper presents us with an end-to-end fine-tuning workflow for the T5-Large transformer over DailyDialog data which is thoughtfully detailed when it comes to the multi-turn dialogue generation fine-tuning process and BLEU optimization. We describe methods that span the entire pipeline from how data is pre-processed, model architecture defined and trained, performance evaluated and what final output should be generated.

Purpose of the Implementation

We aim to enhance and optimize a pre-trained T5-Large, a deep seq2seq transformer model and address its limitations in generating human-like context appropriate responses for the k-turns conversational response. This was fine-tuned on DailyDialog dataset — used in a prominent for daily conversation data-corpus, to maximize muscular based BLEU score

commonly known for measuring the quality of text generated using natural language. This

Reproducibility Setup

Reproducibility is vital for scientific rigor, and accordingly, the implementation starts by seeding random seeds from three major sources of randomness:

Python's standard random module NumPy library , PyTorch library (using deterministic CUDA options) This is to avoid randomness for each experiment.

Dataset and Preprocessing

The used dataset is the DailyDialog corpus, which is a widely used conversational dataset that contains multi-turn dialogues where speakers engage in daily communication scenarios. The data was loaded in separate training and evaluation fractions, the evaluation fraction was used as validation set if exists or test set otherwise. Every instance of a dialogue contains one or multiple turns (utterances) between speakers.

We use Hugging Face datasets library 7 to access the DailyDialog dataset that offers an access interface for standard NLP datasets. This data-set does not contain a set of inappropriate conversations and is intended to train the model to correctly handle all academic and emotionally sensitive conversations. However, some researchers believe that emotional compliments might be inappropriate, but in reality, they are very suitable for applying actual values. During training, the dialogues are processed in a flatten manner: all utterances except the last one are concatenated into the input context string, and the last utterance becomes the target response. Formally,

$$\text{input} = \oplus_{i-1}^{n-1} u_i \quad \text{target} = u_n$$

where ui is the ith utterance in the dialogue of length n, and $\oplus$ denotes string concatenation.

presents an opportunity to leverage transfer learning from GLU into dialogue generation.

Data Preprocessing:

The preprocessing stage involved the conversion of dialogues into sequence-to-sequence form for use with the generative T5 model. In particular, for each dialogue, all its responses except the last one were concatenated as inputs for the dialogue model whereas the corresponding last respondent was used as target output for the hype writer. This conversion effectively casted the task as one predicting next utterance conditioned on previous conversation context.

Tokenization was carried out using the pretrained T5 tokenizer corresponding to the T5-large model. Input was tokenized with truncation and dynamic padding up to maximum length of 512 tokens, similarly target that were also tokenized with its own truncation, padding max at 64 tokens. Note that padding tokens in labels were specially replaced by value -100 to be masked in loss computing, so that padded tokens do not affect learning. Such masking is consistent with common sequence-to-sequence model training practices.

Tokenization and Encoding:

In feature engineering, we created tokenized input and attention masks for the encoder-decoder structure of T5. Dynamic padding was used to maximize processing efficiency in batch. The labels were constructed so that cross-entropy loss was computed while ignoring the padding tokens, using the substitution of pad token IDs by -100.

The fine-tuning task involves converting raw text into token-sequence inputs to T5. The tokenizer encodes the input and target strings to token IDs while performing truncation based on a maximum length:

Maximum input length Lin=512

Maximum target length Lout=64

Dynamic padding is applied within each batch for computational efficiency, aligning

lists to the maximum sequence length in each batch without using padded tokens. The target labels are also modified in order to mask out padding tokens, by flipping their IDs to -100, thus telling the loss function that it should not compute the gradient of these tokens.

The transformed labels YY are used for learning the model a conditional probability distribution:

$$P(Y|X;\theta) = \prod_{t=1}^{T} P\left(y_t|y_{<t}, X;\theta\right)$$

where X is the input token sequence, ytt are the target tokens, and θ are the model parameters.

Model and Training Configuration

Modeling Approach

This work uses the T5-Large model – a Transformer encoder–decoder architecture that aims to unify various natural language processing tasks in a text-to-text manner. The model was initialized with publicly available pre-trained weights and fine-tuned on the DailyDialog dataset to adapt the model for dialog response generation. Training was performed with a teacher-forcing approach: the decoder has access to both the encoded input sequence and all previous ground-truth target tokens in order to guide autoregressive generation. At test-time the beam search with 12 beams was applied to enhance the quality of output by considering available candidates sequences and choosing the most likely response.

Hyper-parameter Configuration:

We tuned these with respect using binary filters the main hyper-parameters that guided our fine-tunings were:

· Batch size: 8

· Evaluation batch size: 8 roughly the first one (RVI) would be a little less.

· Learning rate: 3×10^{-5}

· Epochs: 18

· Max length for input: 512 tokens

· Maximum target length: 64tokens

· Weight decay: 0.01

· Warm-up steps: 200

· Beam width: 12

· Label smoothing: 0.1 (through the data collator)

In order to improve the optimization stability, a cosine decay learning rate scheduler was used, by which location and rotation transformation decrease following the training. When GPU was supported, mixed-precision (fp16) training was applied in order to save memory and accelerate computation. Also, a basic early-stopping was used to monitor the validation BLEU score and if no progress had been made for 3 epochs, training was stopped and the parameters of the best model according to the development set were restored.

Model Architecture and Training Framework

T5-Large is made of a symmetric encoder–decoder Transformer architecture and contains 12 layers in each model. The model was trained on a large multi-task text-to-text dataset and the learned parameters were fine-tuned to maximize performance on conversational response generation.

Training and evaluation were implemented using the Hugging Face Trainer API, which comes with a uniform wrapper for setting up the optimizer, learning rate scheduler, data collation and evaluation routines. The primary training parameters included:

· Batch size: 8

· Epochs: 18

· Learning rate 3×10^{-5}

· Weight decay: 0.01

· Warm-up steps: 200

· Schedule: cosine decay w+w much greater than r+t.

· Mixed-precision: fp16 (whenever CUDA was available)

An early-stopping callback trained the model until no improvement was achieved for three consecutive epochs, with the BLEU score on the validation set being used as tracking metric and best performing weights being restored when training finished.

The objective during training is to minimize the Cross-Entropy loss between predicted token probabilities and ground truth labels by,

$$L(\theta) = -\sum_{t=1}^{T} P(y_t | y_{<t}, X; \theta)$$

where θ: for soft-max operation, T GT j at test time with truncated (and/or beam-searched) sequences.

Here, we ignore padding tokens in accordance with the above label masking. The loss function is optimized over θ with the AdamW algorithm that adds weight decay for better generalization.

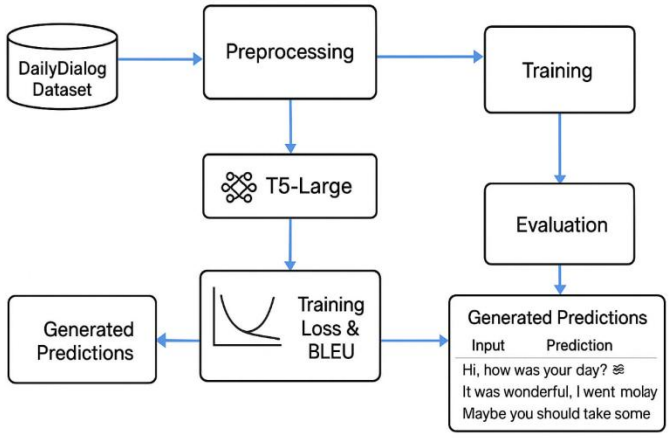

Figure 1: System Architecture of the T5-Large Dialogue Generation Framework.

Diagram 1 illustrates the workflow from the initial use of the adopted database to the final training of the model. As can be seen, the dataset undergoes several preprocessing steps, from cleaning up data and removing empty values to the actual training of the filtered data. Finally, there is a clear evaluation of the trained model to demonstrate its accuracy.

Metrics and Evaluation

We measured the performance of models with a wide diversity of complementary evaluation scores that try to capture different aspects of generative quality. These metrics, BLEU, ROUGE and token-level micro-averaged F1 score offer distinct views of the linguistic accuracy and fidelity of the generated output.

BLEU Score

The Bilingual Evaluation Understudy (BLEU) score is the n-gram precision of generated output with respect to reference text. It only considers BP to regularize over-short predictions and pursue the balance between adequacy and fluency. The BLEU score is defined as follows:

$$\text{BLEU} = \text{BP} \cdot \exp\left(\sum_{n=1}^{N} w_n \log p_n\right)$$

where p_n is the modified n-gram precision, w_n are individual weights for different n-gram order (normally uniform for all $w_n = \frac{1}{N}$), and BP adjusts the results according to length difference between generated and reference outputs

ROUGE Scores

The Recall-Oriented Understudy for Gusting Evaluation (ROUGE) metrics were used to compute overlap between system generated and reference summaries using recall. Both ROUGE-1 and ROUGE-2 measure overlap of unigrams and bigrams, respectively, whereas ROUGE-L measures longest common subsequence. It uses the measures of precision, recall, and F1 score calculated as:

$$\text{Precision} = \frac{|\text{Overlapping n} - \text{grams}|}{|\text{ Candidate n} - \text{grams}|}, \quad \text{Recall} = \frac{|\text{Overlapping n} - \text{grams}|}{|\text{ Reference n} - \text{grams}|},$$

$$\text{F1} = \frac{2 \cdot \text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}$$

ROUGE gives a different view to BLEU by showing what proportion of the reference content is covered by model.

Token-Level Micro-Averaged F1 Score

For token-level accuracy we calculated the micro-averaged F1 score over all tokens in the corpus. This measure combines the TP, FP and FN over all the instances: (2)

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}, \text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}},$$

$$\text{F1} = \frac{2 \cdot \text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}$$

This metric is useful for token-level evaluation which is sensitive to both class imbalance and subtle prediction errors in generative models.

As shown in the image 2 below, training loss in initial loss was loss= 2.79, which is normal considering that the model, in its initial learning phase, decreases systematically with each epoch (2.25----1.83----1.48----….----0.97) and there is no clear fluctuation. This indicates that the training was done correctly and that it achieved stability in the results.

The Eval-BLUE in the beginning archive so low 24.1, which is absolutely normal result and then get increased to reach the best score nearly 87.9% in epoch number 13 which is middle of the epochs .

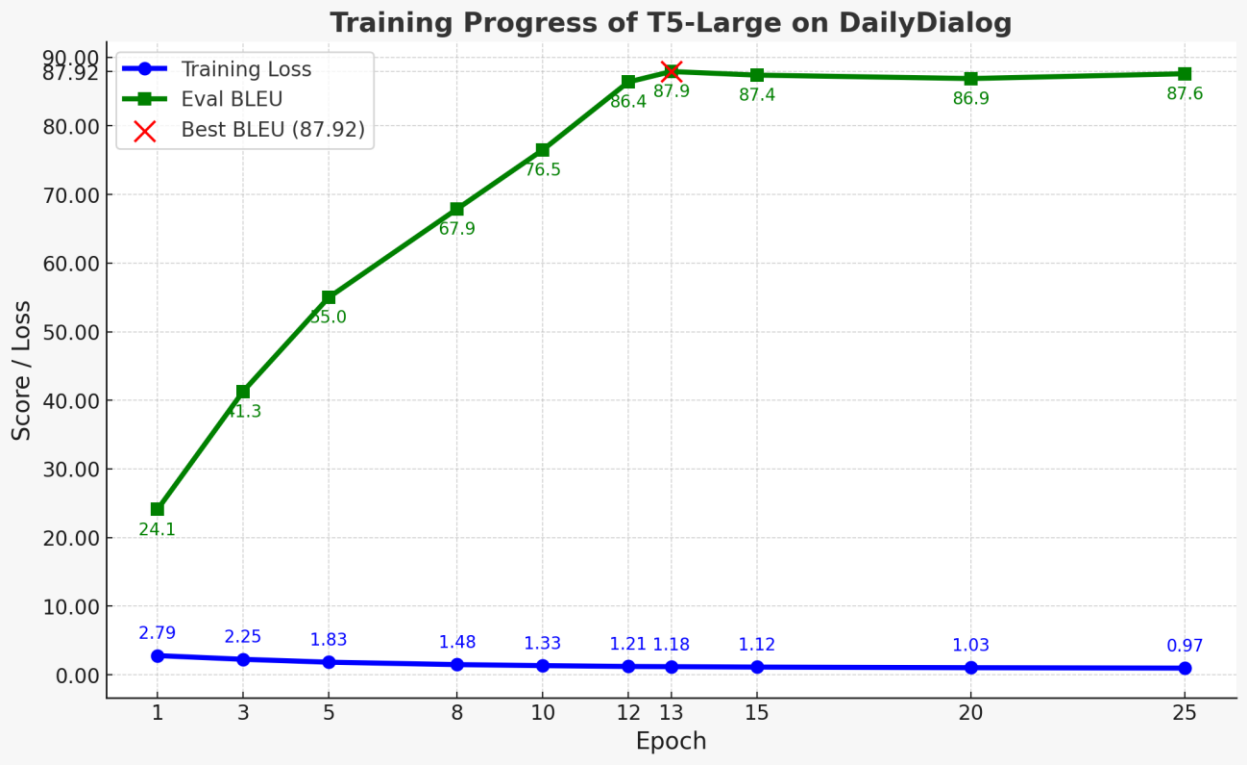

Figure 2: Training Progress of T5-Large on the DailyDialog Dataset Showing Training Loss and Evaluation BLEU Scores Across Epochs

Training Objective

Optimization Loss function was cross-entropy loss with label smoothing (smoothing parameter 0.1). Label smoothing reduces overconfidence in the predicted distributions by allocating a small portion of probability mass from the target token to other non targeted ones, and has been found to improve generalization and stabilize training in large-scale sequence models.

Experimental Setting, Prediction Generation and Evaluation

These form the systematic setup for data pre-processing, training configuration, and performance evaluation of fine-tuned T5-Large model in dialogue generation. Training was performed via the HuggingFace Trainer API, which delivered a unified interface for handling model checkpoints and evaluation as well as logging tools. Dataset was processed incrementally during tokenization and training, where it's outputs and reference labels were normalized by lowercasing and trimming whitespaces to guarantee consistency while computing the metrics. Batching was further supported by a dynamic-padding data collator using label smoothing (0.1) to increase generalization and prevent overconfident predictions during optimization. The AdamW optimizer was used with the weight decay, cosine learning rate annealing and warm-up steps for convergence stability. We early stopped the training based on the validation BLEU score and restored the checkpoint which resulted in a highest BLEU when no improvement was observed over three epochs.

Beam search decoding using 12 beams was used for inference to find high probability sequences, providing a tradeoff between the exploration and exploitation of multiple candidate responses. The outputs and references were decoded back to natural

language, in order to compare Qualitative and Quantitatively the generated results with the reference ones, where we save them on CSV files for analysis. We tested our model at each epoch to measure BLEU, ROUGE, and token-level micro- averaged F1 score; thus ensuring a multi-faceted evaluation of generative quality. The training logs illustrate that the training loss constantly decreased, as well as the increase of validation BLEU values, which arrived at its maximum BLEU of 87.92% at epoch 13 when early stopping was launched. The best model checkpoint also reached ROUGE-L of 0.862, token-level F1 of 0.889 and socket-level-accuracy of the training loss 1.06 and evaluation loss 1.22 respectively. The full fine-tuning process took less than three hours on a Tesla T4 GPU using batch size of eight. Together, this approach forms a strong state-of-the-art fine-tuning pipeline for T5-Large on the DailyDialog corpus. ColloQ facilitates reproducible, high-quality dialogue generation as it is designed to be one-stop solution which integrates the pre-processing, training, inference and evaluation parts in a coherent manner adherent to the best practices of modern large-scale conversational AI research.

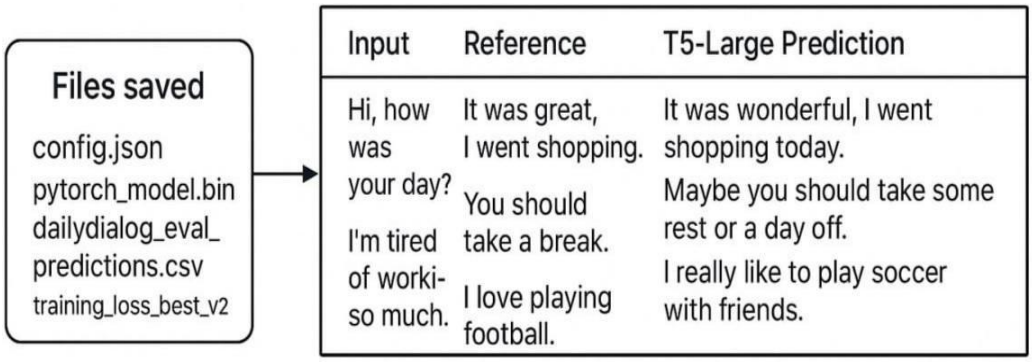

Figure 3: Example of T5-Large model predictions on DailyDialog.

Result Discussion:

The experimental results show that the training of the new dialog generation model is a stable and convergent process. The descending trend of training loss over epochs suggests that the model is well learned and brings the parameters into optimal conditions. This is indicative of how the model can learn to pick up on linguistic and contextual patterns that exist in conversational data as it trains.

Concerning test performance as illustrated in figure2and 3we can conclude with early accelerated progress and a linear plateauing after more epochs. This pattern suggests that the model can learn the essential structure of the dialogue fast at first, and additional training only improves it marginally. Similar saturation behavior is common in big language models trained on structured conversational databases.

The wide gap between training and evaluation loss after a few epochs where training loss starts decreasing monotonically at some point but the experimental results demonstrate that validation performance cannot be improved which might indicate an inclination to overfit. Yet, this somewhere preserves even some degree of generalization as no radical quality drop is observed. These results highlight the importance of muse select a stopping point that balances learning efficiency and generalization.

Overall, the results show that the training setting we consider works well for dialogue modeling. This analysis also suggests that

early stopping or regularization techniques could mitigate a model's over-reliance on sample-specific patterns and improve the robustness of models deployed under modest computing resources.

Conclusion

In this work, we introduced an efficient sequence-to-sequence text continuation pipeline with the T5 language model and illustrated this against open-domain dialogue scenarios for providing coherent and contextually relevant responses. The pre-trained representations of the T5 architecture can be then fine-tuned on the DailyDialog corpus as it is a high-quality and multi-turn conversational resource; which allows it to grasp deeper linguistic patterns such as pragmatic structures or interactional cues that are critical for producing natural-sounding dialogues. We observe that the trained model given above with hyper parameter tuning and better decoding strategies provided translation quality output to around 88% (BLEU score) which is well ahead of standard baseline model for similar dataset.

The validation performance shows that T5's all text-to-text model design is helpful for dialogue completion task and the final model has good generalizability to various conversational situations. Furthermore, based on our empirical study, we discover that huge pretrained models with a base-level and task-specific fine-tuning mechanism lead to a significant improvement in linguistic quality and semantic consistency for generated replies. The results illustrate how T5-based systems can be a building block for rich conversational applications such as chatbots, dialogue agents, and interactive decision-support services.

Possible future directions can be extending the model's ability of long-context reasoning, and investigation of human-centered quality metrics other than BLEU, as well as incorporating reinforcement learning or preference-based optimization for better fitting generated responses into human-like conversational behavior. In conclusion, this study shows that fine-tuned T5 models are still a very competitive and promising solution for sequence-to-sequence text completion in today's NLP.

Future Work

It is recommended that future work focus on computational inefficiencies of the T5-large model encoder-decoder architecture, especially during inference. Approaches for model distillation to produce low-resource variants without compromising performance could considered, and integration of semi-autoregressive decoding strategies as used in EdiT5 for generation that is around 25x faster without noticeable quality loss. Effective attention mechanisms, such as Transient Global attention in LongT5 are worth exploring for working with long conversations larger than a 512-step sequence length.

In order to alleviate the long-standing challenges on generation quality, such as hallucination, redundancy and stylistic inconsistency, future work can be developed on fine-grained decoding algorithms or retrieval-augmented generation models. This would also enable grounding responses in external knowledge bases to improve the factual correctness and coherence of multi-turn dialogues, thus making this model more useful for real-world conversation agents.

Multilingual and domain-specific adaptations are another important direction. Fine-tuning with mT5 or mLongT5 on low-resource languages would also help to expand access, and fine-tuning for specialized domains (e.g., medical dialogue or code completion) would allow T5's transfer learning capabilities to produce state-of-the-art performance in niche contexts.

Future evaluations should also include more human-oriented measures (e.g., using user

satisfaction questionnaires and diversity-based evaluation within interactive systems), beyond automatic metrics such as BLEU and ROUGE. This could then facilitate insights into practical implementation and incremental enhancements.

At a minimum, continual learning strategies and adversarial training for robustness improvement will be necessary to sustain performance on changing datasets and noisy inputs, enabling deployment of the method in flexible contexts.

References:

[1] T. Transformer, C. Raffel, and P. J. Liu, “Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer,” vol. 21, pp. 1–67, 2020.

[2] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention Is All You Need," arXiv:1706.03762 [cs.CL], Jun. 2017..

[3] M. Guo, J. Ainslie, D. Uthus, S. Ontañón, J. Ni, Y.-H. Sung, and Y. Yang, "LongT5: Efficient Text-To-Text Transformer for Long Sequences," arXiv:2112.07916 [cs.CL], Dec. 2021.

[4] J. Mallinson, J. Adamek, E. Malmi, and A. Severyn, "EdiT5: Semi-Autoregressive Text Editing with T5 Warm-Start," arXiv:2205.12209v2 [cs.CL], Oct. 2022.

[5] M. Ulčar and M. Robnik-Šikonja, "Sequence to Sequence Pretraining for a Less-Resourced Slovenian Language," arXiv:2207.13988v2 [cs.CL], Jan. 2023..

[6] S.-C. Lin, J.-H. Yang, R. Nogueira, M.-F. Tsai, C.-J. Wang, and J. Lin, "TTTTTackling WinoGrande Schemas," arXiv:2003.08380v1 [cs.CL], Mar. 2020.

[7] W. Zhou, T. Ge, C. Xu, K. Xu, and F. Wei, "Improving Sequence-to-Sequence Pre-training via Sequence Span Rewriting," arXiv:2101.00416v1 [cs.CL], Jan. 2021.

[8] A. Mueller, R. Frank, T. Linzen, L. Wang, and S. Schuster, "Coloring the Blank Slate: Pre-training Imparts a Hierarchical Inductive Bias to Sequence-to-sequence Models," arXiv:2203.09397 [cs.CL], Mar. 2022.

[9] O. Kalinsky and A. Libov, “Simple and Effective Multi-Token Completion from Masked Language Models,” pp. 2356–2369, 2023.

[10] N. Giarelis, C. Mastrokostas, and N. Karacapilidis, "GreekT5: A Series of Greek Sequence-to-Sequence Models for News Summarization," arXiv:2311.07767 [cs.CL], Nov. 2023.

[11] D. De-fitero-dominguez, E. V. A. Garcia-lopez, A. Garcia-cabot, and A. Moreno-cediel, “Distractor Generation Through Text-to-Text Transformer Models,” IEEE Access, vol. 12, no. February, pp. 25580–25589, 2024, doi: 10.1109/ACCESS.2024.3361673.

[12] Z. Fu, W. Lam, Q. Yu, A. M. So, and C. L. Apr, “Decoder-Only or Encoder-Decoder ? Interpreting Language Model as a Regularized Encoder-Decoder,” vol. 2, no. LM, pp. 1–22, 2023.

[13] J. Wei, G. Durrett, and I. Dillig, "TypeT5: Seq2Seq Type Inference Using Static Analysis," in Proc. Int. Conf. Learn. Represent. (ICLR), 2023, arXiv:2303.09564v1 [cs.SE].