Fine-Tuning A Multiplexer Network

Jayanta Banik, Ankit Gupta and **Yue Dong**
Department of Computer Science
University of California, Riverside
{jbani004, agupt135, yue.dong}@ucr.edu

Abstract

This paper presents a simple technique for generalized large language models. We show that collection of small supervised models, instruction-tuned on task-specific data sets-substantially improves performance in generalized tasks. We have taken a 3M parameter supervised model and instruction-tuned it on three different task-specific data sets curated using natural language instruction templates. We evaluate this model, which we call FAMNet, with state-of-the-art models (SOTA) such as FLAN and GPT3.All trained models are available at https://github.com/UCR-CS260-winter23-NLP/FAMNet.

1 Introduction

Generalized Language Models such as FLAN[1] and GPT3[2] have shown remarkable performance when performing generalized tasks such as Natural Language Inference (NLI) and Closed-book QA, among others. They are still not at par when compared to the task-specific instruction-tuned supervised models such as T5. For example, FLAN's performance on NLI task with CB data set, though surpasses GPT3, still lacks by a huge difference with T5 NLI instruction-tuned supervised model. The primary reason, could be that generalized models are trained on all task data set and task-specific supervised models train on one particular task, outperforming the SOTA models.

In this paper, we introduce a simple technique to create a generalized model that runs the background of several small task-specific models that could appeal to a greater audience. We make use of the fact that task-specific supervised models perform relatively well on different sets of tasks, such as NLI or Closed-Book QA. We have taken three different task-specific data sets (Translation: EN-DE, TriviaQA, and BoolQA), and ran them through promptsource to generate the prompts-based data set. We divide the architecture components into three smaller groups — Mux_selector, Mux_input_parser, and HOSOTAs. We leverage the fact that the SOTA models, such as BERT and T5, can be finetuned for a particular task to achieve higher performance, this was the motivation for this paper. We divide the architecture components into three smaller groups — Mux_selector, Mux_input_parser, and HOSOTAs. We refer to this pipeline of models as FAMNet, for Finetuning a Multiplexer Network.

The Input prompt flows from Mux_selector; for our case, we took BERT and finetuned it to recognize different tasks. The prompt and the task selected then flows to Mux_input_parser; for our experiment, we took T5-small and finetuned it for summarizing inputs from the given prompted input. The selected task and the summarized input then flow into the HOSOTAs(Hashmap Of State Of The Arts); for our experiment, we use the BERTS and T5 models, fine-tuned on various tasks, to perform the task specified on the summarized input based on the task selected. For example, Given the input "Translate et tu brute from Latin to English", We want Mux_selector(BERT) to recognize this as a translation task from Latin to English and Mux_input_parser(T5) to summarize the input to "et tu brute." Now the summarized input will be passed on to the Latin-to-English SOTA translation model, selected by Mux_selector, giving the desired output. The resulting model is named as FAMNet, for Fine-Tuning A Multiplexer Network.

To evaluate the performance of FAMNet, we compare our results with the FLAN models available. We perform experiments on the three

[1](https://doi.org/10.48550/arXiv.2109.01652
[2]https://doi.org/10.48550/arXiv.2005.14165

tasks, Translation, BoolQ, NLI, and Closed-book QA. The setup ensures these tasks are trained for T5.

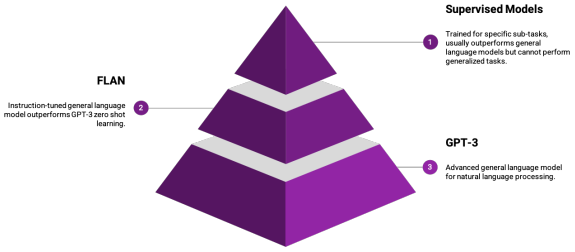

Figure 1: Overview of performance supported by FLAN[1] and GPT3 vs supervised models

Creating a layer for all the instruction-tuned supervised models is a simple method that can yield promising results. With the inclusion of pre-trained BERT and BART, we ensure that we are taking the best of both worlds and implementing it in FAMNet. Our preliminary results show promising abilities, and with the availability of different task data sets, FAMNet can be modified to incorporate different tasks and outperform future models as well. The multiplexer ideology can incorporate any future SOTA models in the Hasmap of SOTA and thus provide higher performance than generalized LLMs.

2 Literature Review

2.1 BERT

BERT [2](Bidirectional Encoder Representations from Transformers) is one of a kind, state-of-the-art natural language processing (NLP) model that was first introduced by researchers at Google in 2018[3]. BERT is used as a classifier to identify which task to pick from a large corpus of tasks.

Since its introduction, BERT has been adopted and used for various NLP tasks, including language modeling, classifications, and question-answering. So far, as per research, BERT outperforms all the previous state-of-the-art models on these tasks.

Though BERT performs quite well and can perform well on a wide range of tasks with the proven effectiveness of capturing relationships between words, it requires a huge amount of data to be trained and a lot of computational resources.

2.2 T5

T5 (Text-to-Text Transfer Transformer)[**?**] is another state-of-the-art sequence-to-sequence model which was again developed by Google AI[4] for natural language processing (NLP) tasks just like BERT. With the inclusion of all the tasks BERT can perform, T5 can also perform machine translation, beating the scores of BART[3].

T5 is widely used for machine translation tasks. Studies show that T5 achieves state-of-the-art performance on several benchmarks of machine translation tasks, and fine-tuning the model on specific language pairs further improves the performance.

Furthermore, T5 has also been evaluated for closed-book question-answering (QA) tasks on the SQuAD dataset. Research shows that T5 performs competitively with the other state-of-the-art models for closed-book QAs, showing that it has the ability to understand effectively and generate natural-language responses.

Additionally, T5 also performs well on Boolean QA. Though due to limited data set availability, commenting on the performance of T5 for such a task is hard. It still performs well in comparison to the current state-of-the-art models.

Overall, it can be said that T5 is an effective model for general-purpose tasks, with a strong performance in text classification and machine translation tasks. It can perform well from a wide range of tasks, and fine-tuning it on specific tasks makes it even more accurate and reliable. Thus, it becomes a promising model for further research.

2.3 FLAN

FLAN (Few-shot Language Adaptation Network)[5] is a recently proposed approach in natural language processing (NLP) for few-shot learning. The literature on FLAN focuses its effectiveness for various NLP tasks, such as natural language understanding (NLU) and dialogue systems. FLAN performs well with limited training data.

Experimentations have been performed for FLAN on complex NLP tasks, such as sentiment analysis and named entity recognition. Studies have found that FLAN can adapt to these tasks with limited training data, demonstrating its potential for addressing the challenge of data scarcity in NLP.

Overall, the literature on FLAN in NLP highlights its effectiveness as a few-shot learning approach, particularly for NLU and dialogue systems. Future research may explore its performance in

[3] https://doi.org/10.48550/arXiv.1810.04805

[4] https://arxiv.org/abs/1910.10683

[5] https://doi.org/10.48550/arXiv.2109.01652

other NLP tasks and potential applications in the industry. FLAN has a higher performance on Zero short task generalizations than GPT3, however in their report, Supervised learners have shown prominent performance over the generalized models.

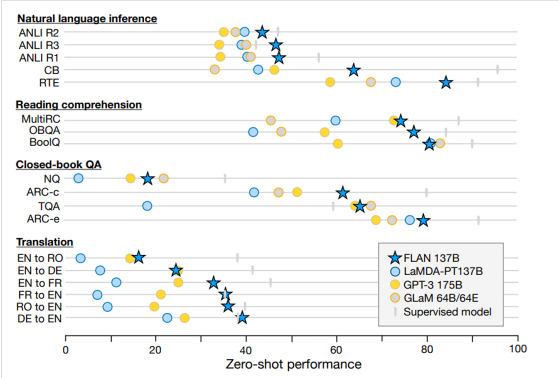

Figure 2: FLAN[1] report of it's performance on zero short learning vs supervised models

2.4 BoolQ

BoolQ is a recently proposed natural language processing (NLP) dataset[6] that focuses on boolean question answering.

BoolQ poses several challenges for NLP models, such as understanding complex sentence structures and reasoning about logical relationships. Thus, BoolQ is a useful benchmark for evaluating the performance of NLP models in boolean question-answering tasks.

Studies have found that NLP models that are trained on BoolQ achieve high accuracy on boolean question-answering tasks, and fine-tuning the models on such tasks can further improve the performance of models.

Overall, the literature on BoolQ in NLP highlights its usefulness as a benchmark dataset for boolean question-answering tasks and its potential for evaluating and improving the performance of NLP models.

2.5 Trivia_QA

Almost 650,000 question-answer pairs comprise the large-scale, open-domain dataset known as Trivia QA. The dataset was created by the University of Washington[7] academics to evaluate how well question-answering models handle a variety of general knowledge queries.

Several research studies have used Trivia QA to assess how well question-answering models function. According to one study, a neural network model that had been trained on the dataset outperformed multiple baseline models and attained an accuracy of 54.4% on a test set of questions. In a further study, the performance of various question-answering models on Trivia QA was compared. The model with the highest accuracy was discovered using external information sources, such as Wikipedia and knowledge graphs.

Trivia QA has become a well-liked benchmark dataset for assessing how well question-answering models perform on open-domain general knowledge questions. Its availability has sparked the development of novel ways for managing open-domain question answering. Its vastness and variety of questions make it a difficult test for models.

2.6 WMT16

WMT16 [4], or the Fourth Conference on Machine Translation, was a shared task and conference for machine translation that took place in 2016. As part of the collaborative work, machine translation models were trained and tested on various language pairs, including English-German, German-English, English-Romanian, and Romanian-English.

The WMT16 dataset has been used in numerous studies to assess how well machine translation algorithms perform. According to one study, a neural machine translation model performed better on the English-German and German-English language pairs than numerous conventional statistical machine translation models. In a further study, the neural machine translation models outperformed other machine translation models when tested on English-Romanian and Romanian-English language pairings.

In general, the WMT16 dataset has gained popularity as a benchmark for assessing how well machine translation models perform over a range of language pairs. Its accessibility has encouraged the creation of new machine translation methods, particularly those that use neural networks.

3 Methodology

The FAMNet architecture is a novel idea which has not been implemented so far. It is based on three state-of-the-art (SOTA) models, fine-tuned as needed. All the methods have been converted into a library for ease of access for this project and

[6] https://doi.org/10.48550/arXiv.1905.10044
[7] https://doi.org/10.48550/arXiv.1705.03551

further studies.

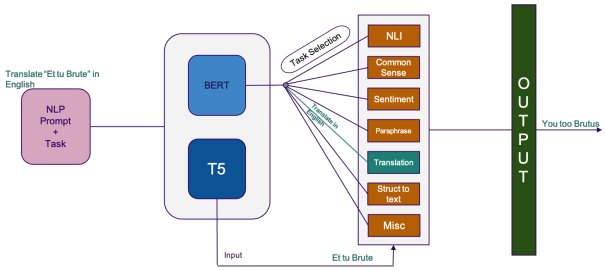

Figure 3: Model Architecture

3.1 BERT - Mux_selector

We have selected BERT because it is small, fast and reliable. Also, BERT attains a surprisingly good accuracy when trained for our multiplexer task selection which is discussed in detail in the results section. When compared with T5, BERT still stands strong; losing just 0.034% in the accuracy score but since it is such a small model compared to the T5, the selection of mux_selector was apparent.

We are using the BERT model to select which type of task our prompt has. BERT has been fine-tuned to select the type of task from a corpus of trivia_qa, Closed Book QA, Translation (en-de), and Translation (de-en). We have also trained BERT on different tasks apart from the specified tasks keeping a future outlook in mind. Thus, the multiplexer selector model is one of a kind, expandable selector and can be further fine-tuned to include more tasks in the future. This selector model will pick the model from the corpus of supervised fine-tuned models.

3.2 T5 - Mux_input_parser

T5 has been fine-tuned to generate inputs for the supervised models from the prompt. We have trained T5 after modifying a large corpus of data, which tells the T5 model to generate the input from the summarized output given by mux_input_parser. By fine-tuning T5 in this way, it acts as the text summarization model, comparable to BART model but performs even better as T5 super-seeds BART. The summarizing task is considered for the prompt given to extract the subject of the task for the next model.

3.3 HoSoTAs

Once the task is selected and the[4] input has been generated from both the models described above. It is then passed to our final layer, which is a multiplexer model of three tasks currently consisting of Closed Book QA, Boolean QA and Translation (en-de). The task is selected from the multiplexer selector based on the prompt asked, and the input is provided by the T5-Input Generator. The input passed on the model the generator selects gives an output based on the task.

4 Experimental Setup

The Models used in the Experiment are taken from the Huggingface collection of SOTA. We fine-tune different models on different tasks. The Dataset to train Mux_selector was created by combining parts of Boolq[5], TriviaQA[6], WMT16[4], and Cola datasets available on hugging face Datasets. Mux_selector was then evaluated for categorical Accuracy, and the loss function Categorical-Crossentropy was used. Fig (a) (b) shows us training accuracy and loss. The Mux_input_parser was based on t5-small , which showed promising results on sentence extractor and a few shot learning abilities on new tasks.

The mux_selector was trained for multiple variations of hyperparameters and the optimised hyperparameter was found to be as follows: epochs = 5 batch_size = 64 init_lr = 1e-5; Please mind that the grid search approach was only able to search through the parameters that was able to run on NVIDIA P100_1GPU. with cuda_11.2.r11.2.

The Mux_input_parser was implemented from the scratch T5-small model, with functions clearly outlining training, dataset_parsers, validations, among others. It is then fine-tuned on a large corpus of data in order to generate input for the multiplexer model. Fine-tuning is done on the model parameters as model = t5-small, train batch size = 64, valid batch size = 64, train epochs = 4, val epochs = 5, learning rate = 1e-4, seed = 42. Due to limited computational power; batch size, epochs and learning rate were kept at a minimum. NVIDIA V100_8gpu was used for this task which showed relatively better performance by NVIDIA P100_1GPU. on cuda_11.2.r11.2.

5 Results

The mux_selector with 98.8% validation accuracy clearly distinguished between different types of tasks. The model is trained with the intuition of classifying the latent prompt in the input string. Below we show the evaluation of the test data set

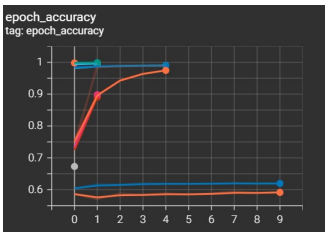

Figure 4: Mux selector accuracy

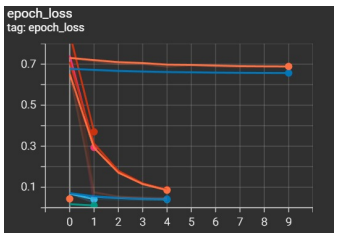

Figure 5: Mux selector loss

and Fig 6. shows manual prompting.

```
Mux_selector.predict('True or False? Sun rises from east?')
'BoolQ'

Mux_selector.predict('Translate "du es" in English')
'translate_de_en'

Mux_selector.predict('Translate "Good night" to German')
'translate_en_de'

Mux_selector.predict('Who is the president of USA?')
'TriviaQA'

Mux_selector.predict('Is this statement gramatically correct: i no no vegies i only icecream')
'cola'
```

Figure 6: Output from Mux_selector

```
sentence: [b'How many flakes of snow have to
    fall in the 24 hours of the 25th December
    anywhere in the UK to be classified as a
    White christmas?']
BERT results: TriviaQA
BERT raw results: tf.Tensor([-0.89830846 4.1511
    -1.0675467 -1.904335  -1.8493086 ],
    shape=(5,), dtype=float32)

sentence: [b'John Sentamu was given which
    English title in May 2005?']
BERT results: TriviaQA
BERT raw results: tf.Tensor([-1.1317111
    4.2693243 -0.48774132 -1.8697778 -2.231168
    ], shape=(5,), dtype=float32)

sentence: [b"Translate 'Sie tauschen dies ein
    gegen die Position eines Regierungschefs
    und die Beteiligung an einer Regierung.'
    from german to english"]
BERT results: translate_de_en
BERT raw results: tf.Tensor([-3.50506
    -0.8466267 -1.0336276 4.3787556
    -1.1171744], shape=(5,), dtype=float32)

sentence: [b"Translate 'Probleme ergeben sich
    aber insbesondere aus der
    J\xc3\xa4hrlichkeit und der
    projektbezogenen Mittelzuweisung
    beispielsweise von PHARE im Vergleich zur
    Mehrj\xc3\xa4hrigkeit von INTERREG-Mitteln
    und der ma\xc3\x9fnahmenbezogenen
    Mittelzuweisung.' from german to english"]
BERT results: translate_de_en
BERT raw results: tf.Tensor([-3.0211427
    -1.1711048 -1.2709228 4.5655065
    -1.8191719], shape=(5,), dtype=float32)

sentence: [b"Translate 'Haben Sie einen
    besseren Vorschlag?' from german to
    english"]
BERT results: translate_de_en
BERT raw results: tf.Tensor([-3.881825
    0.51207757 -1.5129592 3.507092 -1.4322205
    ], shape=(5,), dtype=float32)

sentence: [b'will there be a fourth series of
    the tunnel Yes or No?']
BERT results: BoolQ
BERT raw results: tf.Tensor([ 4.3716226
    -0.48491254 -0.8415204 -3.0352914
    -0.6093738 ], shape=(5,), dtype=float32)

sentence: [b"Translate 'Das ist ein
    \xc3\xb6kologisch sehr sinnvoller Ansatz.'
    from german to english"]
BERT results: translate_de_en
BERT raw results: tf.Tensor([-3.4431727
    -0.67424685 -0.95403993 4.2911263
    -0.8401095 ], shape=(5,), dtype=float32)

sentence: [b"Translate 'In conclusion, while
    key infrastructure projects have been
    supported by the European Regional
    Development Fund and the Cohesion Fund, we
    should remember that the European Social
    Fund has played a very important role in
    helping the less well-off in our society.'
    from english to german"]
BERT results: translate_en_de
BERT raw results: tf.Tensor([-2.0162787
    -1.0498806 -1.245141 -0.9883183 4.3060446],
    shape=(5,), dtype=float32)

sentence: [b"Translate 'There is an error which
    has still not been corrected and that is on
    Amendment No 4.' from english to german"]
BERT results: translate_en_de
BERT raw results: tf.Tensor([-0.75302047
    -2.4693284 0.48138073 -2.4750087 2.9476888
    ], shape=(5,), dtype=float32)

sentence: [b'Who starred alongside Polly James
    in the first series of The Liver Birds?']
BERT results: TriviaQA
BERT raw results: tf.Tensor([-0.5286281
    4.390407 -1.3263522 -1.280483 -2.5498195],
    shape=(5,), dtype=float32)

sentence: [b"Translate 'Secondly, it is also
    clear that reform will require some new
    investment in training for skills and in
    technology.' from english to german"]
BERT results: translate_en_de
BERT raw results: tf.Tensor([-1.6232833
    -1.7147628 -1.2703862 -1.388701 4.251369 ],
    shape=(5,), dtype=float32)

sentence: [b"Translate 'Ich m\xc3\xb6chte mich
```

```
    bei Herrn Jonckheer bedanken, der die damit
    verbundenen Probleme in seinem Bericht sehr
    gr\xc3\xbcndlich behandelt. Er untersucht,
    wie wir sicherstellen k\xc3\xb6nnen,
    da\xc3\x9f diese L\xc3\xa4nder unseren
    Anforderungen gerecht werden, aber auch,
    wie gleiche Wettbewerbsbedingungen
    geschaffen werden.' from german to english"]
BERT results: translate_de_en
BERT raw results: tf.Tensor([-2.7949142
    -0.62656355 -1.2358865 4.6114492 -1.676721
    ], shape=(5,), dtype=float32)

sentence: [b"Translate 'In dem Vorschlag der
    Kommission werden jedoch nicht alle
    relevanten Fragen ber\xc3\xbccksichtigt,
    wie beispielsweise die kalten klimatischen
    Bedingungen in den n\xc3\xb6rdlichen
    Regionen.' from german to english"]
BERT results: translate_de_en
BERT raw results: tf.Tensor([-3.0029683
    -0.93357337 -1.0703613 4.63887 -1.2463793
    ], shape=(5,), dtype=float32)
```

The T5-Small model was trained to extract the sentences from the given prompts to feed into the supervised model. Fig. 6 shows the run history for the T5[7] model with summarizing task.

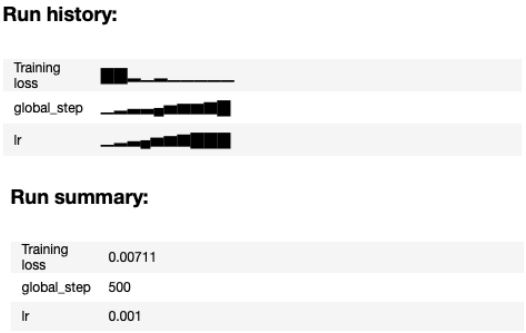

Figure 7: Run history of Mux_input_parser

6 Related Work

Even after a lot of research, we were unable to find any such model or project which is using a multiplexer in order to identify tasks and generate outputs.

7 Discussions Conclusions

In this project, we have created a novel architecture for selecting the task given a prompt (via BERT)(mux_selector) and then passing it through the input-decoder (T5-fine-tuned)(mux_input_parser) to generate the input for the multiplexer network of supervised models. The FAMNet model shows promising results with the scope of future progress. The whole architecture is designed in such a way as to accommodate any new NLP task into the model. Comparing the results with current SOTA models is beyond the scope of this project, but we aim to build a much more powerful FAMNet model in the future and compare it with the resulting metric of FLAN and GPT-3.

References

[1] Jason Wei, Maarten Bosma, Vincent Y. Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M. Dai, and Quoc V. Le. Finetuned language models are zero-shot learners. *CoRR*, abs/2109.01652, 2021.

[2] Iulia Turc, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Well-read students learn better: On the importance of pre-training compact models. *arXiv preprint arXiv:1908.08962v2*, 2019.

[3] Ala Alam Falaki. Fine-tune bart for translation on wmt16 dataset (and train new tokenizer).

[4] Ond rej Bojar, Rajen Chatterjee, Christian Federmann, Yvette Graham, Barry Haddow, Matthias Huck, Antonio Jimeno Yepes, Philipp Koehn, Varvara Logacheva, Christof Monz, Matteo Negri, Aurelie Neveol, Mariana Neves, Martin Popel, Matt Post, Raphael Rubino, Carolina Scarton, Lucia Specia, Marco Turchi, Karin Verspoor, and Marcos Zampieri. Findings of the 2016 conference on machine translation. In *Proceedings of the First Conference on Machine Translation*, pages 131–198, Berlin, Germany, August 2016. Association for Computational Linguistics.

[5] Christopher Clark, Kenton Lee, Ming-Wei Chang, Tom Kwiatkowski, Michael Collins, and Kristina Toutanova. Boolq: Exploring the surprising difficulty of natural yes/no questions. In *NAACL*, 2019.

[6] Mandar Joshi, Eunsol Choi, Daniel Weld, and Luke Zettlemoyer. triviaqa: A Large Scale Distantly Supervised Challenge Dataset for Reading Comprehension. *arXiv e-prints*, page arXiv:1705.03551, 2017.

[7] HuggingFace. T5 library.