LEARNING WITHOUT FORGETTING: TASK AWARE MULTI-TASK LEARNING FOR MULTI-MODALITY TASKS

Anonymous authors
Paper under double-blind review

ABSTRACT

Existing Multi-Task Learning(MTL) strategies like joint or meta-learning focus more on shared learning and have little to no scope for task-specific learning. This creates the need for a distinct shared pretraining phase and a task-specific finetuning phase. The finetuning phase creates separate models for each task, where improving the performance of a particular task necessitates forgetting some of the knowledge garnered in other tasks. Humans, on the other hand, perform task-specific learning in synergy with general domain-based learning. Inspired by these learning patterns in humans, we suggest a simple yet generic task aware framework to incorporate into existing MTL strategies. The proposed framework computes task-specific representations to modulate the model parameters during MTL. Hence, it performs both shared and task-specific learning in a single phase resulting in a single model for all the tasks. The single model itself achieves significant performance gains over the existing MTL strategies. For example, we train a model on Speech Translation (ST), Automatic Speech Recognition (ASR), and Machine Translation (MT) tasks using the proposed task aware multitask learning approach. This single model achieves a performance of 28.64 BLEU score on ST MuST-C English-German, WER of 11.61 on ASR TEDLium v3, and BLEU score of 23.35 on MT WMT14 English-German tasks. This sets a new state-of-the-art performance (SOTA) on the ST task while having close to SOTA performance on ASR and competitive performance on the MT task.

1 INTRODUCTION

The process of Multi-Task Learning (MTL) on a set of related tasks is inspired by the patterns displayed by human learning. It involves a pretraining phase over all the tasks, followed by a finetuning phase. During pretraining, the model tries to grasp the shared knowledge of all the tasks involved, while in the finetuning phase, task-specific learning is performed to improve the performance. However, as a result of the finetuning phase, the model forgets the information about the other tasks that it learnt during pretraining. Humans, on the other hand, are less susceptible to forgetfulness and retain existing knowledge/skills while mastering a new task. For example, a polyglot who masters a new language learns to translate from this language without losing the ability to translate other languages. Moreover, the lack of task-based flexibility and having different finetuning/pretraining phases cause gaps in the learning process due to the following reasons:

Role Mismatch: Consider the MTL system being trained to perform the Speech Translation(ST), Automatic Speech Recognition(ASR) and Machine Translation(MT) tasks. The Encoder block has a very different role in the standalone ASR, MT and ST models and hence we cannot expect a single encoder to perform well on all the tasks without any cues to identify/use task information. Moreover, there is a discrepancy between pretraining and finetuning hampering the MTL objective.

Task Awareness: At each step in the MTL, the model tries to optimize over the task at hand. For tasks like ST and ASR with the same source language, it is impossible for the model to identify the task and alter its parameters accordingly, hence necessitating a finetuning phase. A few such examples have been provided in Table 1. Humans, on the other hand, grasp the task they have to perform by means of context or explicit cues.

Task	Confounding Task	Prediction	Target
En ASR	En-De ST	Mobilität opportunity to..	Mobility opportunity to ..
En-De MT	En-Ro MT	Situația este alarmantă	Die Situation ist alarmierend
En-De ST	En ASR	I don't understand it.	Das verstehe ich nicht

Table 1: Issues due to the lack of task information in MTL strategies. The model, unable to identify the task, produces the output corresponding to another task either completely or partially.(En:English, De:German, Ro:Romanian)

Although MTL strategies help the finetuned models to perform better than the models directly trained on those tasks, their applicability is limited to finding a good initialization point for the finetuning phase. Moreover, having a separate model for each task increases the memory requirements, which is detrimental in low resource settings.

In order to achieve the goal of jointly learning all the tasks, similar to humans, we need to perform shared learning in synergy with task-specific learning. Previous approaches such as Raffel et al. (2019) trained a joint model for a set of related text-to-text tasks by providing the task information along with the inputs during the joint learning phase. However, providing task information is not straightforward for certain modalities such as speech and image. Also, providing explicit task information might not always be desirable, e.g., automatic multilingual speech recognition task. Thus, a holistic joint learning strategy requires a generic framework which learns task-specific information without any explicit supervision.

In this work, we propose a generic framework which can be easily integrated into the MTL strategies which can extract task-based characteristics. The proposed approach helps align existing MTL approaches with human learning processes by incorporating task information into the learning process and getting rid of the issues related to forgetfulness. We design a modulation network for learning the task characteristics and modulating the parameters of the model during MTL. As discussed above, the task information may or may not be explicitly available during the training. Hence, we propose two different designs of task modulation network to learn the task characteristics; one uses explicit task identities while the other uses the examples from the task as input. The model, coupled with the modulation network, jointly learns on all the tasks and at the same time, performs the task-specific learning. The proposed approach tackles issues related to forgetfulness by keeping a single model for all the tasks, and hence avoiding the expensive finetuning phase. Having a single model for all the tasks also reduces memory constraints, improving suitability for low resource devices.

To evaluate the proposed framework, we conduct two sets of experiments. First, we include the task information during MTL on text-to-text tasks to show the effect of task information. Secondly, we train a model on tasks with different modalities and end goals, where providing task information is not straightforward. Our proposed framework allows the model to learn the task characteristics without any explicit supervision, and hence train a single model which performs well on all the tasks. The main contributions of this work are as follows:

- We propose an approach to tackle the issue of forgetfulness which occurs during the finetuning phase of existing MTL strategies.
- Our model, without any finetuning, achieves superior performance on all the tasks which alleviates the need to keep separate task-specific models.
- Our proposed framework is generic enough to be used with any MTL strategy involving tasks with multiple modalities.

2 TASK-AWARE JOINT LEARNING

An overview of our proposed approach is shown in Figure 1.

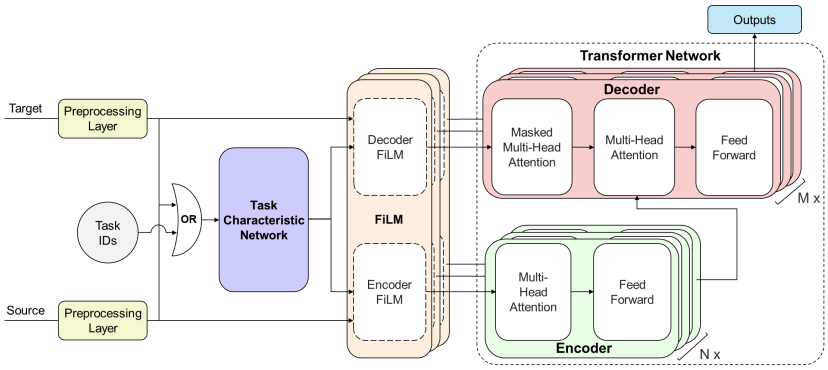


Figure 1: Overview of task aware multi-task learning. Task Characteristics Network (TCN) produces a task embedding to modulate the parameters of base network. Inputs to the TCN are chosen based on type of TCN.

2.1 Base Model

In general, the sequence-to-sequence architecture consists of two components: (1) an encoder which computes a set of representations $\boldsymbol{X} = \{\boldsymbol{x_1}, \cdots, \boldsymbol{x_m}\} \in \mathbb{R}^{m \times d}$ corresponding to $\boldsymbol{x}$, and a decoder coupled with attention mechanism (Bahdanau et al., 2015) dynamically reads encoder's output and predicts target language sequence $\boldsymbol{Y} = \{\boldsymbol{y_1}, \cdots, \boldsymbol{y_n}\} \in \mathbb{R}^{n \times d}$. It is trained on a dataset D to maximize the $p\left(\boldsymbol{Y}|\boldsymbol{X};\theta\right)$, where θ are parameters of the model.

We use the Transformer Vaswani et al. (2017) as our base model. Based on the task modalities, we choose the preprocessing layer in the Transformer, i.e., speech or the text (text-embedding) preprocessing layer. The speech preprocessing layer consists of a stack of k CNN layers with stride 2 for both time and frequency dimensions. This layer compresses the speech sequence and produces the output sequence such that input sequences corresponding to all the tasks have similar dimensions, d. The overview of the base sequence-to-sequence model is shown in the rightmost part of Figure 1.

2.2 Task Modulation Network

The task modulation network performs two operations. In the first step, it computes the task characteristics ($\boldsymbol{t_e}$) using the task characteristics layer. It then modulates the model parameters $\boldsymbol{\theta}$ using $\boldsymbol{t_e}$ in the second step.

2.2.1 Task Characteristics Network:

We propose two types of Task Characteristics Networks(TCN) to learn the task characteristics, where one uses explicit task identities while the other uses source-target sequences as input.

Explicit Task Information: In this approach, the tasks involved are represented using different task identities and fed as input to this TCN as one hot vectors. This network consists of a feed-forward layer which produces the task embedding used for modulating the model parameters.

$$\boldsymbol{t_e} = FFN(\boldsymbol{e}), \tag{1}$$

where $\boldsymbol{e} \in \mathbb{R}^s$ is a on-hot encoding of s tasks used during joint learning.

Implicit Task Information: Providing task information for all the tasks might not be the best option since some of the tasks might have overlapping goals/characteristics. For example, translating written text along with spoken text sentences from English-to-German, or automatic multilingual speech recognition. The Implicit TCN computes the task embeddings using example sequences from the tasks. It consists of four sub-layers: (1) Sequence Representation Layer, (2) Bi-directional Attention Layer, (3) Sequence Summary Layer, and (4) Task Embedding Layer.

The sequence representation sub-layer consists of uni-directional Transformer Encoder (TE) blocks Vaswani et al. (2017). It takes the source and target sequences from the tasks as input and produces

self-attended source and target sequences.

$$\boldsymbol{x}^{sa} = \mathrm{TE}(\boldsymbol{x}), \quad \boldsymbol{y}^{sa} = \mathrm{TE}(\boldsymbol{y}). \tag{2}$$

This sub-layer computes the contextual representation of the sequences.

The Bi-directional attention sub-layer takes the self-attended source and target sequences from the previous layer as input and computes the relation between them using Dot-Product Attention Luong et al. (2015). As a result, we get target aware source ($\boldsymbol{x}^{at}$) and source aware target ($\boldsymbol{y}^{as}$) representations as outputs.

$$\boldsymbol{x}^{at} = \mathrm{Attention}(\boldsymbol{x}^{sa}, \boldsymbol{y}^{sa}), \quad \boldsymbol{y}^{as} = \mathrm{Attention}(\boldsymbol{y}^{sa}, \boldsymbol{x}^{sa}). \tag{3}$$

The sequence summary sub-layer is similar to the sequence representation sub layer and summarizes the sequences. The sequence summaries are given by:

$$\boldsymbol{x}^{s} = \mathrm{TE_u}(\boldsymbol{x}^{at}), \quad \boldsymbol{y}^{s} = \mathrm{TE_u}(\boldsymbol{y}^{as}). \tag{4}$$

Equation 4 summarizes the sequences $\boldsymbol{x}^{at}$ and $\boldsymbol{y}^{as}$ which contain the contextual and attention information. Specifically, we take the last tokens from both $\boldsymbol{x}^{s}$ and $\boldsymbol{y}^{s}$ as they can see the whole sequence; they act as a summary of the sequence.

The task embedding layer computes $\boldsymbol{t_e}$ by taking the output of the Sequence Summary Layer($\boldsymbol{s}$) and applying a feed-forward network:

$$\boldsymbol{s} = \mathrm{concat}(x^s_m; y^s_n), \quad \boldsymbol{t_e} = FFN(\boldsymbol{s}), \tag{5}$$

where m and n are the source and target sequence lengths.

2.2.2 Modulating Model Parameters

We modulate the parameters (θ) of the network (Section 2.1) to account for the task-specific variation during MTL over a set of tasks. We achieve this by scaling ($\boldsymbol{\gamma}$) and shifting ($\boldsymbol{\beta}$) the outputs of each layer (e.g., transformer block) including any preprocessing layers in the model adopted based on the Feature-wise Linear Modulation (FiLM; Perez et al. (2018)). The $\boldsymbol{\gamma}$ and $\boldsymbol{\beta}$ parameters are obtained from the task embedding $\boldsymbol{t_e}$ either by using equation 1 or 5.

$$\boldsymbol{\gamma} = \boldsymbol{t_e}[: d], \quad \boldsymbol{\beta} = \boldsymbol{t_e}[d :], \tag{6}$$

where $\boldsymbol{t_e} \in \mathbb{R}^{2d}$, and d is the hidden dimension of the model.

Once we have $\boldsymbol{\gamma}$ and $\boldsymbol{\beta}$, we apply the feature-wise linear modulation (Perez et al., 2018) to compute the modulated output (O_l) for each block of the model.

$$O_l = \boldsymbol{\gamma} * f_l(v_l; \theta_l) + \boldsymbol{\beta}, \quad l = 1, \cdots, L, \tag{7}$$

where L is the total number of blocks in the model and f_l represents the l_{th} block of the model with parameters $\theta_l \in \theta$ and inputs v_l.

2.3 Training

MTL has been successfully applied across different applications of machine learning such as natural language processing (Hashimoto et al., 2016; Collobert & Weston, 2008), speech recognition (Liu et al., 2019; Deng et al., 2013), computer vision (Zhang et al., 2014; Liu et al., 2015; Girshick, 2015), and drug discovery (Ramsundar et al., 2015). It comes in many forms: joint learning, learning to learn, and learning with auxiliary tasks. We consider two MTL strategies: (1) joint learning and (2) learning to learn to train on set of S tasks, $\mathrm{T} = \{\tau^1, \cdots, \tau^S\}$ with corresponding datasets $D = \{D^1, \cdots, D^S\}$.

As our first training strategy, we use Joint Learning (JL) (Caruana, 1997), which is the most commonly used training strategy for MTL. In JL, the model parameters, including the output layer, are shared across all the tasks involved in the training. For the second training strategy under the learning-to-learn approach, we use a variant of meta-learning, Modality Agnostic Meta Learning (MAML) (Finn

S. No.	MTL Strategy	Training Approach	En-De (↑)	En-Tr((↑)	En-Ro((↑)
1	Joint Learning	Vanilla	7.65	3.59	5.55
2		OHV(Raffel et al., 2019)*	16.77	6.10	11.79
3	Meta Learning	MAML (Finn et al., 2017a)*	7.74	2.93	5.78
4		OHV(Raffel et al., 2019)*	16.08	5.93	12.43
This Work					
5	Joint Learning	Implicit TCN	14.18	4.38	13.47
6	Meta Learning	Implicit TCN	14.50	4.43	14.32

Table 2: Performance(BLEU) of the models trained on the Single Modality for the En-De, En-Ro and En-Tr Machine Translation task. *Models based on OHV and MAML are inspired from Raffel et al. (2019) and Finn et al. (2017a).

et al., 2017a). Even though MAML is mostly used in few-shot learning settings, we use it since it allows for task-specific learning during the meta-train step and it has also been shown to provide improvements in the field of speech translation(Indurthi et al., 2020).

We resolve the source-target vocabulary mismatch across different tasks in MTL by using a vocabulary of subwords (Sennrich et al., 2016) computed from all the tasks. We sample a batch of examples from D^s and use this as input to the TCN and the Transformer model. To ensure that each training example uses the task embedding computed using another example, we randomly shuffle this batch while using them as input to the TCN. This random shuffling improves the generalization performance by forcing the network to learn task-specific characteristics ($\boldsymbol{t_e}$) in Equation 1 or 5. We compute the task embedding in the meta-train step as well; however, the parameters of the TCN are updated only during the meta-test step. During inference time, we use the precomputed task embeddings using a batch of examples randomly sampled from the training set.

3 EXPERIMENTS

3.1 TASKS AND DATASETS

We conduct two sets of experiments, one with the tasks having the same input modality, i.e., text and another over tasks having different input modalities, i.e., speech and text. The main motivation behind the text-based experiments is to establish the importance of providing task information in MTL. The second set of experiments help us demonstrate the effectiveness of our approach in a more generic setup. We incorporate the proposed task modulation framework into joint and meta-learning strategies and analyze its effects.

3.1.1 SINGLE MODALITY EXPERIMENTS

We perform the small scale text-to-text machine translation task over three language pairs English-German/Romanian/Turkish (En-De/Ro/Tr). We keep English as the source language, which makes it crucial to use task information and produce different outputs from the same input. Since it is easier to provide task identity through one-hot vectors in text, we provide the task information by simply prepending the task identity to the source sequence of each task, e.g., "*translate from English to German*", "*translate from English to Turkish*" similar to Raffel et al. (2019). We also train models using our proposed framework to learn the task information and shared knowledge jointly.

For En-De, we use 1.9M training examples from the `Europarl v7` dataset. `Europarl dev2006` and `News Commentary nc-dev2007` are used as the dev and `Europarl devtest2006`, `Europarl test2006` and `News Commentary nc-devtest2007` as the test sets. For En-Tr we train using 200k training examples from the `setimes2` dataset. We use `newsdev2016` as the dev and `newstest2017` as the test set. For En-Ro, we use 600k training examples from `Europarl v8` and `setimes2` datasets. We use `newsdev2016` as dev and `newstest2016` as the test set.

S. No.	MTL Strategy	Training	ST (↑)	ASR(↓)	MT (↑)
1	N/A	Direct Learning	22.14	25.25	27.93
2	Joint Learning	Pretraining	0.81	14.83	23.62
3		+ Finetuning	25.99	12.95	22.63
4	Meta Learning (Indurthi et al., 2020)	Pretraining	0.51	32.93	11.40
5		+ Finetuning	26.03	12.83	22.15
		This Work			
6	Joint Learning	Explicit TCN	28.34	11.99	23.15
7		Implicit TCN	**28.64**	**11.61**	23.35
8		Taskwise Best Models	28.88	11.36	23.71
9	Meta Learning	Explicit TCN	28.16	11.68	23.13
10		Implicit TCN	**28.55**	**11.65**	23.41
11		Taskwise Best Models	28.59	11.28	23.57

Table 3: Performance of the models trained on the Multiple Modality tasks, i.e., ST (MuST-C En-De), ASR (TED-LIUM 3), and MT (WMT'15 En-De).

3.1.2 Multiple Modality Experiments

To alleviate the data scarcity issue in Speech Translation (ST), several MTL strategies have been proposed to jointly train the ST task with Automatic Speech Recognition (ASR) and Machine Translation (MT) tasks. These MTL approaches lead to significant performance gains on both ST and ASR tasks after the finetuning phase. We evaluate our proposed framework based on this multimodal MTL setting since passing the task information explicitly via prepending labels(like the text-to-text case) in the source sequence is not possible. We use the following datasets for ST English-German, ASR English, MT English-German tasks:

MT En-De: We use the Open Subtitles (Lison et al., 2019) and WMT 19 corpora. WMT 19 consists of Common Crawl, Europarl v9, and News Commentary v14 datasets(22M training examples).

ASR English: We used five different datasets namely LibriSpeech (Panayotov et al., 2015), MuST-C (Di Gangi et al., 2019), TED-LIUM (Hernandez et al., 2018), Common Voice (Ardila et al., 2020) and filtered IWSLT 19 (nie, 2019) to train the English ASR task.

ST Task: We use the Europarl ST (Iranzo-Sánchez et al., 2019), IWSLT 2019 (nie, 2019) and MuST-C (Di Gangi et al., 2019) datasets. Since ST task has lesser training examples, we use data augmentation techniques (Lakumarapu et al., 2020) to increase the number of training examples.

Please refer to the appendix for more details about the data statistics and data augmentation techniques used. All the models reported in this work use the same data settings for training and evaluation.

3.2 Implementation Details and Metrics

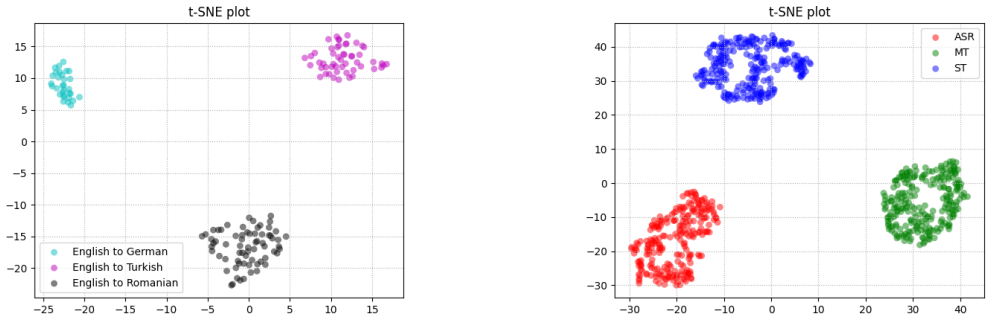
We implemented all the models using Tensorflow 2.2 framework. For all our experiments, we use the Transformer(Vaswani et al., 2017) as our base model. The hyperparameter settings such as learning rate, scheduler, optimization algorithm, and dropout have been kept similar to the Transformer, other than the ones explicitly stated to be different. The ASR performance is measured using Word Error Rate (WER) while ST and MT performances are calculated using the detokenized cased BLEU score (Post, 2018). We generate word-piece based universal vocabulary(Gu et al., 2018b) of size 32k using source and target text sequences of all the tasks. For the task aware MTL strategies, we choose a single model to report the results rather than finding the best model for each task separately.

We train the text-to-text translation models using 6 Encoder and Decoder layers with a batch size of 2048 text tokens. The training is performed using NVIDIA P40 GPU for 400k steps.

In multi-modality experiments, the speech signals are represented using 80-dimensional log-Mel features and use 3 CNN layers in the preprocessing layer described in Section 2.1. We use 12 Encoder and Decoder layers and train for 600k steps using 8 NVIDIA V100 GPUs. For the systems without TCN, we perform finetuning for 10k steps on each task.

S. No.	Task	Approach	(BLEU(↑)/WER(↓))
1	MT (WMT'15 En-De)	Liu et al. (2020)(60 Encoders)	30.10
2		Our Approach(12 Encoders)	23.71
3	ST (MuSTC En-De)	Indurthi et al. (2020)	22.11
4		Pino et al. (2020)	25.99
5		Lakumarapu et al. (2020)	27.51
6		Our Approach	**28.88**
7	ASR (TED-LIUM 3)	Pham et al. (2019)(36 Encoders)	10.20
8		Pham et al. (2019)(12 Encoders)	12.40
9		Our Approach(12 Encoders)	11.28

Table 4: Performance of the models trained on the Multiple Modality tasks, i.e., ASR, ST and MT.



(a) Single Modality Experiments

(b) Multiple Modality Experiments

Figure 2: t-SNE plot of the task modulation output shows a clear demarcation between various tasks.

3.3 Results

3.3.1 Single Modality Experiments

The results for the text-to-text translation models trained with different MTL strategies have been provided in Table 2. The MTL models with prepended task label (Raffel et al., 2019) are referred to as OHV (One Hot Vector). Unlike T5, we don't initialize the models with the text embeddings from large pretrained language model (Devlin et al., 2018). Instead, we focus on establishing the importance of task information during MTL and having a single model for all the tasks. As we can see from the results, providing the task information via text labels or implicitly using the proposed task aware MTL leads to significant performance improvements compared to the MTL without the task information. The models trained using OHV have better performance than those trained using implicit TCN. However, providing OHV via text labels is not always possible for tasks involving non-text modalities such as speech and images.

3.3.2 Multi Modality Experiments

We evaluate the proposed two TCNs and compare them with the vanilla MTL strategies. The performance of all the models is reported in Table 3.

Effect of Task Information: The models trained using task aware MTL achieve significant performance gains over the models trained using vanilla MTL approach. Our single model achieves superior performance compared to the vanilla MTL models even after the finetuning. This shows that not only is the task information essential to identify the task, but also helps to extract the shared knowledge better. Our JL and MAML models trained with task aware MTL achieve improvements of (+2.65, +2.52) for ST, (-1.34, -1.18) for ASR, and (+0.72, +1.26) for the MT task. MAML has some scope for task-specific learning during its meta train step, which explains why the improvements for MAML are slightly lesser than JL for ST and ASR tasks.

We also report results using Direct Learning (DL) approach, where separate models are trained for each task, to compare with MTL models. All the MTL models outperform the DL models on ST and ASR tasks have comparable performance on MT task.

Explicit v/s Implicit TCN: Our proposed implicit TCN learns the task characteristics directly from the examples of each task and achieves slightly better performance than the models trained using explicit TCN. This indicates that it is better to learn the task information implicitly, specifically for tasks having overlapping characteristics. Figure 2 contains the tSNE plots for task embeddings obtained from the implicit TCN for single and multi-modality experiments. We can observe that the implicit TCN is also able to separate all the three tasks effectively without any external supervision.

Single model for all tasks: We select one single model for reporting the results for our approach, since, having a single model for multiple tasks is favourable in low resource settings. However, we also report the best models corresponding to each task (row 8 and 11 of Table 3). We observe that choosing a single model over task-specific models did not result in any significant performance loss.

Comparison with existing works: The design of our system, i.e., the parameters and the related tasks were fixed keeping the ST task in mind. We compare the results of our system with the recent works in Table 4. We set a new state-of-the-art (SOTA) on the ST En-De MuST-C task. Although we did not achieve a SOTA on the ASR task, we achieve a 9.03% improvement in the WER as compared to the Transformer based model (Pham et al., 2019) with the same number of Encoder and Decoder blocks. The best transformer-based MT model achieves a BLEU score of 30.10, however, it uses 60 Encoder blocks. The performance drop on the MT task is attributed to simply training a bigger model without using any additional initialization techniques proposed in Liu et al. (2015); Wu et al. (2019). However, the MT task helps the other tasks and improves the overall performance of the system.

4 Related Work

Various MTL techniques have been widely used to improve the performance of end-to-end neural networks. These techniques are known to solve issues like overfitting and data scarcity. Joint learning (Caruana, 1997) improves the generalization by leveraging the shared information contained in the training signals of related tasks. MAML (Finn et al., 2017b) was proposed for training a joint model on a variety of tasks, such that it can quickly adapt to new tasks. Both the learning approaches require a finetuning phase resulting in different models for each task. Moreover, during finetuning phase the model substantially forgets the knowledge acquired during the large-scale pretraining.

One of the original solutions to this problem is pseudo-rehearsal, which involves learning the new task while rehearsing generated items representative of the previous task. This has been investigated and addressed to a certain extent in Atkinson et al. (2018) and Li & Hoiem (2018). He et al. (2020) address this by using a mix-review finetuning strategy, where they include the pretraining objective during the finetuning phase. Raffel et al. (2019) take a different approach by providing the task information to the model and achieve performance improvements on different text-to-text tasks. Although this alleviates the need for finetuning, it cannot be extended to the tasks involving complex modalities. In our work, we propose a generic framework on top of MTL to provide task information to the model which can be applied irrespective of the task modalities. It also removes the need for finetuning, tackling the issue of forgetfulness at its root cause.

In the domain of neural machine translation, several MTL approaches have been proposed (Gu et al., 2018a;c). Similarly, recent works have shown that jointly training ST, ASR, and MT tasks improved the overall performance (Liu et al., 2019; Indurthi et al., 2020). However, all these require a separate finetuning phase and hence suffer from the issues mentioned above.

5 Conclusion

This work proposes a task-aware framework which helps to improve the learning ability of the existing multitask learning strategies. It addresses the issues faced during vanilla multitask learning, which includes forgetfulness during finetuning and the problems associated with having separate models for each task. The proposed approach helps to align better the existing multitask learning strategies with human learning. It achieves significant performance improvements with a single model on a variety of tasks which is favourable in low resource settings.

REFERENCES

The IWSLT 2019 Evaluation Campaign, November 2019. Zenodo. doi: 10.5281/zenodo.3525578. URL https://doi.org/10.5281/zenodo.3525578.

Rosana Ardila, Megan Branson, Kelly Davis, Michael Henretty, Michael Kohler, Josh Meyer, Reuben Morais, Lindsay Saunders, Francis M. Tyers, and Gregor Weber. Common voice: A massively-multilingual speech corpus, 2020.

Craig Atkinson, Brendan McCane, Lech Szymanski, and Anthony V. Robins. Pseudo-recursal: Solving the catastrophic forgetting problem in deep neural networks. *CoRR*, abs/1802.03875, 2018. URL http://arxiv.org/abs/1802.03875.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. In *Computer Science Mathematics CoRR*, 2015.

Rich Caruana. Multitask learning. *Machine learning*, 28(1):41–75, 1997.

Ronan Collobert and Jason Weston. A unified architecture for natural language processing: Deep neural networks with multitask learning. In *Proceedings of the 25th International Conference on Machine Learning*, ICML '08, pp. 160–167, New York, NY, USA, 2008. Association for Computing Machinery. ISBN 9781605582054. doi: 10.1145/1390156.1390177. URL https://doi.org/10.1145/1390156.1390177.

L. Deng, G. Hinton, and B. Kingsbury. New types of deep neural network learning for speech recognition and related applications: an overview. In *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 8599–8603, 2013.

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.

Mattia A. Di Gangi, Roldano Cattoni, Luisa Bentivogli, Matteo Negri, and Marco Turchi. MuST-C: a Multilingual Speech Translation Corpus. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pp. 2012–2017, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics. doi: 10.18653/v1/N19-1202. URL https://www.aclweb.org/anthology/N19-1202.

Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pp. 1126–1135. JMLR. org, 2017a.

Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pp. 1126–1135. JMLR. org, 2017b.

R. Girshick. Fast r-cnn. In *2015 IEEE International Conference on Computer Vision (ICCV)*, pp. 1440–1448, 2015.

Jiatao Gu, Hany Hassan, Jacob Devlin, and Victor O.K. Li. Universal neural machine translation for extremely low resource languages. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pp. 344–354, New Orleans, Louisiana, June 2018a. Association for Computational Linguistics. doi: 10.18653/v1/N18-1032. URL https://www.aclweb.org/anthology/N18-1032.

Jiatao Gu, Hany Hassan, Jacob Devlin, and Victor O.K. Li. Universal neural machine translation for extremely low resource languages. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pp. 344–354, New Orleans, Louisiana, June 2018b. Association for Computational Linguistics. doi: 10.18653/v1/N18-1032. URL https://www.aclweb.org/anthology/N18-1032.

Jiatao Gu, Yong Wang, Yun Chen, Victor O. K. Li, and Kyunghyun Cho. Meta-learning for low-resource neural machine translation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pp. 3622–3631, Brussels, Belgium, October-November 2018c. Association for Computational Linguistics. doi: 10.18653/v1/D18-1398. URL https://www.aclweb.org/anthology/D18-1398.

Kazuma Hashimoto, Caiming Xiong, Yoshimasa Tsuruoka, and Richard Socher. A joint many-task model: Growing a neural network for multiple NLP tasks. *CoRR*, abs/1611.01587, 2016. URL http://arxiv.org/abs/1611.01587.

Tianxing He, Jun Liu, Kyunghyun Cho, Myle Ott, Bing Liu, James Glass, and Fuchun Peng. Analyzing the forgetting problem in the pretrain-finetuning of dialogue response models, 2020.

François Hernandez, Vincent Nguyen, Sahar Ghannay, Natalia Tomashenko, and Yannick Estève. Ted-lium 3: Twice as much data and corpus repartition for experiments on speaker adaptation. *Lecture Notes in Computer Science*, pp. 198–208, 2018. ISSN 1611-3349. doi: 10.1007/978-3-319-99579-3_21. URL http://dx.doi.org/10.1007/978-3-319-99579-3_21.

S. Indurthi, H. Han, N. K. Lakumarapu, B. Lee, I. Chung, S. Kim, and C. Kim. End-end speech-to-text translation with modality agnostic meta-learning. In *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 7904–7908, 2020.

Javier Iranzo-Sánchez, Joan Albert Silvestre-Cerdà, Javier Jorge, Nahuel Roselló, Adrià Giménez, Albert Sanchis, Jorge Civera, and Alfons Juan. Europarl-st: A multilingual corpus for speech translation of parliamentary debates. *arXiv preprint arXiv:1911.03167*, 2019.

Nikhil Kumar Lakumarapu, Beomseok Lee, Sathish Reddy Indurthi, Hou Jeung Han, Mohd Abbas Zaidi, and Sangha Kim. End-to-end offline speech translation system for IWSLT 2020 using modality agnostic meta-learning. In *Proceedings of the 17th International Conference on Spoken Language Translation*, pp. 73–79, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.iwslt-1.7. URL https://www.aclweb.org/anthology/2020.iwslt-1.7.

Zhizhong Li and Derek Hoiem. Learning without forgetting. *IEEE Trans. Pattern Anal. Mach. Intell.*, 40(12):2935–2947, 2018. doi: 10.1109/TPAMI.2017.2773081. URL https://doi.org/10.1109/TPAMI.2017.2773081.

Pierre Lison, Jörg Tiedemann, Milen Kouylekov, et al. Open subtitles 2018: Statistical rescoring of sentence alignments in large, noisy parallel corpora. In *LREC 2018, Eleventh International Conference on Language Resources and Evaluation*. European Language Resources Association (ELRA), 2019.

Xiaodong Liu, Jianfeng Gao, Xiaodong He, Li Deng, Kevin Duh, and Ye-yi Wang. Representation learning using multi-task deep neural networks for semantic classification and information retrieval. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 912–921, Denver, Colorado, May–June 2015. Association for Computational Linguistics. doi: 10.3115/v1/N15-1092. URL https://www.aclweb.org/anthology/N15-1092.

Xiaodong Liu, Kevin Duh, Liyuan Liu, and Jianfeng Gao. Very deep transformers for neural machine translation. *arXiv preprint arXiv:2008.07772*, 2020.

Yuchen Liu, Hao Xiong, Zhongjun He, Jiajun Zhang, Hua Wu, Haifeng Wang, and Chengqing Zong. End-to-end speech translation with knowledge distillation. *CoRR*, abs/1904.08075, 2019. URL http://arxiv.org/abs/1904.08075.

Thang Luong, Hieu Pham, and Christopher D. Manning. Effective approaches to attention-based neural machine translation. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pp. 1412–1421, Lisbon, Portugal, September 2015. Association for Computational Linguistics. doi: 10.18653/v1/D15-1166. URL https://www.aclweb.org/anthology/D15-1166.

V. Panayotov, G. Chen, D. Povey, and S. Khudanpur. Librispeech: An asr corpus based on public domain audio books. In *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 5206–5210, 2015.

Ethan Perez, Florian Strub, Harm de Vries, Vincent Dumoulin, and Aaron C. Courville. Film: Visual reasoning with a general conditioning layer. In *AAAI*, 2018.

Ngoc-Quan Pham, Thai-Son Nguyen, Jan Niehues, Markus Müller, and Alex Waibel. Very deep self-attention networks for end-to-end speech recognition. *CoRR*, abs/1904.13377, 2019. URL http://arxiv.org/abs/1904.13377.

Juan Pino, Qiantong Xu, Xutai Ma, Mohammad Javad Dousti, and Yun Tang. Self-training for end-to-end speech translation. *arXiv preprint arXiv:2006.02490*, 2020.

Matt Post. A call for clarity in reporting BLEU scores. In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pp. 186–191, Belgium, Brussels, October 2018. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/W18-6319.

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. *arXiv preprint arXiv:1910.10683*, 2019.

Bharath Ramsundar, Steven Kearnes, Patrick Riley, Dale Webster, David Konerding, and Vijay Pande. Massively multitask networks for drug discovery, 2015.

Rico Sennrich, Barry Haddow, and Alexandra Birch. Neural machine translation of rare words with subword units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 1715–1725, Berlin, Germany, August 2016. Association for Computational Linguistics. doi: 10.18653/v1/P16-1162. URL https://www.aclweb.org/anthology/P16-1162.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, pp. 5998–6008, 2017.

Lijun Wu, Yiren Wang, Yingce Xia, Fei Tian, Fei Gao, Tao Qin, Jianhuang Lai, and Tie-Yan Liu. Depth growing for neural machine translation. *arXiv preprint arXiv:1907.01968*, 2019.

Zhanpeng Zhang, Ping Luo, Chen Change Loy, and Xiaoou Tang. Facial landmark detection by deep multi-task learning. In David Fleet, Tomas Pajdla, Bernt Schiele, and Tinne Tuytelaars (eds.), *Computer Vision – ECCV 2014*, pp. 94–108, Cham, 2014. Springer International Publishing. ISBN 978-3-319-10599-4.