Analysing Softmax Entropy Minimization for Adaptating Multitask Models at Test-time

Soumyajit Chatterjee[1], **Abhirup Ghosh**[2], **Fahim Kawsar**[1,3], **Mohammad Malekzadeh**[1]
[1]Nokia Bell Labs, Cambridge, UK, [2]University of Birmingham, UK
[3]University of Glasgow, UK
soumyajit.chatterjee@nokia-bell-labs.com, a.ghosh.1@bham.ac.uk
{fahim.kawsar, mohammad.malekzadeh}@nokia-bell-labs.com

Abstract

Multitask models have been the key to the most AI-driven applications on smart devices like phones. Such applications often infer on-devices using a pre-trained model. However, pre-trained multitask models fail when in-the-wild data distribution differs from the training data. While adapting to the target test data is a natural solution, conventional algorithms from transfer learning and unsupervised domain adaptation are impractical in the above on-device adaptation requirement due to the unavailability of labeled runtime data and limited resources at the devices. Recent methods in *Test-time Adaptation* (TTA) are deemed suitable as they neither require access to labels for test data nor the training data. However, the current state-of-the-art (SOTA) TTA approaches only consider models with single-task objectives and thus may fail to capture the nuances of multitask modeling.

For the first time in literature, we systematically explore this novel but practical problem regime. Firstly, we investigate the impact of different tasks on the entropy of class probability distribution, a key optimization criterion for many TTA approaches. Next, we extend a popular SOTA TTA approach and systematically investigate its performance on a benchmark multitask image dataset under various domain shifts. With different experiments, we observe that the current TTA approaches fail to capture the intricacies of the different tasks. We envision this study will pave the way for further investigation and development of TTA approaches designed explicitly for multitask architectures.

1 Introduction

An emerging paradigm in edge computing is to utilize the same pre-trained machine learning (ML) model for multiple purposes. For instance, a ML model for vision deployed on a smartphone may need to process the camera data to identify faces, segment the background, classify the scenery, and other tasks simultaneously [1, 2]. Such a *multitask model* offers a much better usage of smartphones' memory, battery, and other compute resources. Existing multitask models typically consists of a shared *encoder* followed by a task-specific *classifiers* [3]. In practice, a common strategy is to train the multitask model at server side, and then deploy the model on the edge devices for *on-device inference*, which improves the users' privacy, since personal data does not leave users' devices.

The key problem with pre-trained multitask models is that they often experience a significant drop in performance when the distribution of test data (*i.e.,* target data) shifts from the data used during training (*i.e.,* source data). In the most real-world scenarios, the target data is unlabeled, due to the cost and complexities of labelling or the lack of expertise. Therefore, classical transfer learning methods [4] are not suitable, as they typically require a minimum amount of labeled data. Moreover, while methods of unsupervised domain adaptation (UDA) [5, 6, 7, 8, 9] can adapt to unlabeled target

38th Workshop on Fine-Tuning in Machine Learning (NeurIPS 2024).

data, they all require access to source data at adaptation time. Storing training data on each user's device is highly impractical due to storage limitations, and the source data may also be private. To adapt to unlabeled target data without needing to store source training data, the community has developed *test-time adaptation* (TTA) solutions [10, 11, 12, 13, 14, 15, 16], which do not assume any modification of the training pipeline, *e.g.,* employing a self-supervised task.

In TTA, the most common approach is to minimize the entropy of the softmax outputs, and based on such an unsupervised objective function, to fine-tune some specific layers of the model, like the batch normalization [10, 14], at test time. Entropy-based TTA, due to empirical success, is also combined with other state-of-the-art (SOTA) approaches, such as psuedo-labeling [13, 17], sharpness-aware optimization [14, 18], and contrastive learning [16]. Overall, *softmax-entropy minimization* is still one of the key component of many objective functions for TTA. The main limitation of current TTA methods is that they only focus on a single classification task, and rely on softmax-entropy minimization as a proxy to improve the performance of this single task [19]. However, in multitask models, the softmax outputs of different tasks may not be aligned or correlated, even though they share the same input features. The extension of TTA to multitask models, and the challenges of standard softmax-entropy minimization as the main optimization criterion, has not yet been explored.

In this paper, we present the first comprehensive analysis of SOTA TTA methods in multitask scenarios, motivated by the complex interaction between heterogeneous task objectives and the diverse forms of domain shifts that can occurs at test time. We investigate (a) how the distribution of softmax entropy changes for each task across different domain shifts, and (b) how the correlation between tasks impacts the adaptation of the shared encoder. In particular, we extend one of the most practical TTA methods, test entropy minimization (TENT) [10], to adapt multitask models while analyzing the uncertainties associated with different task objectives and their impact on each other.

In summary, the followings are the key contributions of this paper:

1. To the best of our knowledge, we introduce the first analysis of TTA for multitask models.
2. On top of CelebA dataset [20], and using the well-know domain shift [21], we create and publish a benchmark dataset for multitask domain adaptation.
3. We extend TENT [10] to multitask architectures and study its performance for adaptation across different task objectives.
4. We present the key insights gained from our analysis and provide directions for developing TTA algorithms suitable for multitasking architecture.

2 Related Work

To address domain shift at test time, one solution is transfer learning [4], although challenges arise due to the lack of labeled target data [10]. To overcome this, unsupervised domain adaptation (UDA) leverages unlabeled target data alongside labeled source data to fine tune a model [5, 6, 7, 8, 9]. Current domain adaptation strategies for multitask models like [22, 23, 24] mostly follow UDA approach, in conjunction with other strategies like cross-task distillation [23]. However, these UDA approaches often rely on resource-intensive adversarial training with cross-domain loss. Moreover, they needed access to the labeled source training data, which can become impractical with large-scale models and limited device resources.

Test-time training (TTT) [25, 26, 27] is an alternative to UDA, by using a multitask architecture and a self-supervised auxiliary task, such as image rotation for re-training the model at the test time. However, such a specific model design requirement may need control of the source data and training pipeline, which can be an impractical assumption [16].

In contrast, test-time adaptation (TTA) emerges as a more robust solution, requiring no specific alterations to the model architecture while optimizing the entropy of the model's outputs at test time. TTA can effectively simplify the adaptation process, avoiding the need for auxiliary tasks and model re-engineering [10, 11, 12, 13, 14, 15, 16]. The problem is that TTA is often only suitable for single-task models [19]. Notably, to the best of our knowledge, there is no previous work that explore the extension of TTA for multitask models.

3 Methodology

We discuss the two major ingredients of our exploration: i) the *dataset* for benchmarking multitask TTA, and ii) the *policies* to extend a STOA TTA method, called TENT, for multi-task settings.

3.1 CelebA-C: a New Benchmark for Multitask TTA

Building on top of the well-known multi-task image dataset, CelebA [20], we use a benchmark image corruptions algorithm [21]. CelebA contains 200K facial images, each annotated with 40 binary attributes, but it has highly imbalanced labels [28]. Considering this, (1) we select the four most balanced attributes as *tasks* for training a multi-task model (see Table 1), and (2) we choose weighted F$_1$-score as the metric for evaluations. The four tasks are T_A:'Attractive', T_M:'Male', T_S:'Smiling', and T_L:'Wearing Lipstick', with the distribution of each label shown in Figure 2. These tasks have varying degree of correlation as depicted in Figure 1 measured with Pearson correlation coefficient. For example, T_A and T_M show a negative correlation of -0.39, whereas T_A and T_S show a slight positive correlation of 0.15.

In [21], 15 corruption to images are introduced, to mimic real-world domain shift in images. We select 6 corruptions to explore: *Gaussian Noise*, *Shot Noise*, *Impulse Noise*, *Defocus Blur*, *Brightness*, and *Contrast*. As an example, Gaussian Noise adds noise drawn from a Gaussian distribution to each pixel of the original image. The standard deviation of the Gaussian distribution depends on the severity of the corruption. We follow the definition provided by [21] and divide the severity into a scale of 1 to 5. To make our analyses more practical, we only evaluate the highest possible corruption with a severity of 5.

Task	Description
T_A	if person is *attractive*
T_M	if person is a *male*
T_S	if person is *smiling*
T_L	if person is *wearing lipstick*

Table 1: Task descriptions

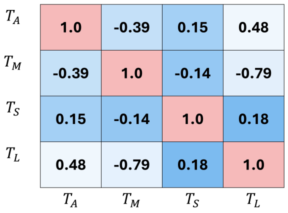


Figure 1: Task correlation

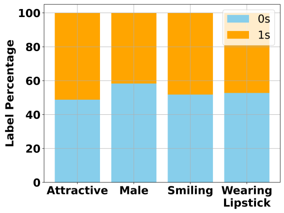


Figure 2: Label Imbalance

3.2 M-TENT: An extension of TENT to Multitask Models

To adapt multitask models, we extend one of the most popular TTA algorithms, test entropy minimization (TENT) [10], initially designed for single-task models only. TENT uses softmax-entropy minimization to fine-tune the affine parameters of the batch normalization (BN) layers of the model using the unlabeled target data at test time.

In Figure 3, we visualize the idea of **M-TENT**: our extension of TENT for multitask models. M-TENT allows us to study the effectiveness of softmax-entropy minimization in TTA for multitask models. In particular, in M-TENT, we compute the average of softmax entropies across all the task heads and utilize it to fine-tune the statistics of BN layers in the shared encoder. Formally, the entropy loss for the shared encoder $\mathcal{L}_{se}$ can be represented as

$$\mathcal{L}_{se} = \frac{1}{M} \sum_{m=1}^{M} H(\hat{y}^m)$$

where M is the number of tasks, $H(\hat{y}^m)$ represents Shannon entropy, and $\hat{y}^m$ denotes the model's softmax outputs for the m^{th} task. This specific design of $\mathcal{L}_{se}$ allows us to setup different ***policies*** in M-TENT for selecting one or more tasks that would contribute to the adaptation of the final model under domain shift. In particular, we utilize a subset of tasks for adaptation to better understand **the marginal gain** of using different tasks and **impact of their correlations with other tasks**.

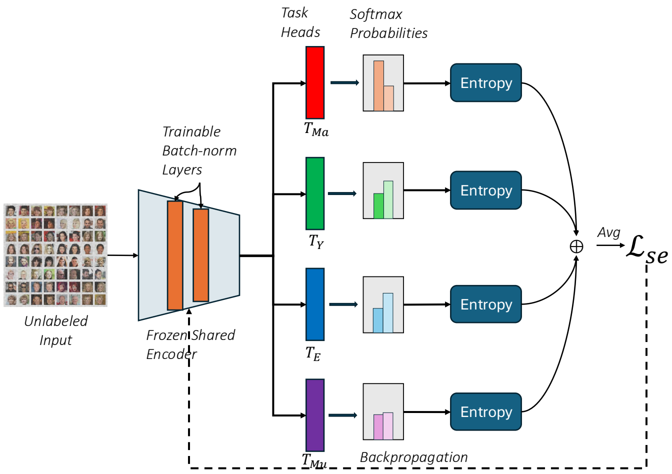


Figure 3: Overall setup of adapting the multitask model using softmax entropy minimization. We use a ResNet18 as the shared encoder and update only the affine parameters of the batch norm layers at the test-time for adaptation.

4 Evaluations: Performance under Domain Shift

Experimental Setup. We use ResNet18 [29] as the shared encoder, and to design a multitask model we add one fully-connected layer as the head per each task. We train the model (ResNet18 and all the heads) with 70% of the CelebA dataset (as the source dataset) using cross-entropy loss function. To create the CelebA-C dataset (see § 3.1), we put the remaining 30% of the CelebA dataset as the held-out target dataset. For the target dataset, we do not use the labels to be aligned with our application constraint. All the reported results are based on this setup.

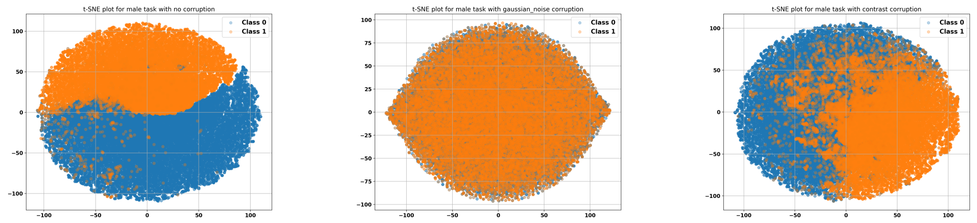
We first examine the accuracy of the pre-trained multi-task model on the in-distribution test data (*i.e.,* CelebA) and the domain-shifted test data (*i.e.,* CelebA-C). In Table 2, we present the accuracy of the CelebA trained model **without adaptation**.

Table 2: Test binary-classification accuracy of a CelebA trained model on CelebA-C. A particular shift have varying impacts across different tasks. Bold font shows the lowest percentage weighted F_1-Score for a task across shifts.

	Source Distribution	**Gaussian Noise**	**Shot Noise**	**Impulse Noise**	**Defocus Blur**	**Brightness**	**Contrast**
Attractive	79.56	35.08	50.74	35.08	55.52	35.08	32.95
Male	97.91	43.30	44.44	43.30	44.15	43.30	87.89
Smiling	91.07	35.86	35.86	35.86	35.86	35.86	54.13
Wearing Lipstick	92.45	30.69	30.74	30.69	37.01	30.69	37.14

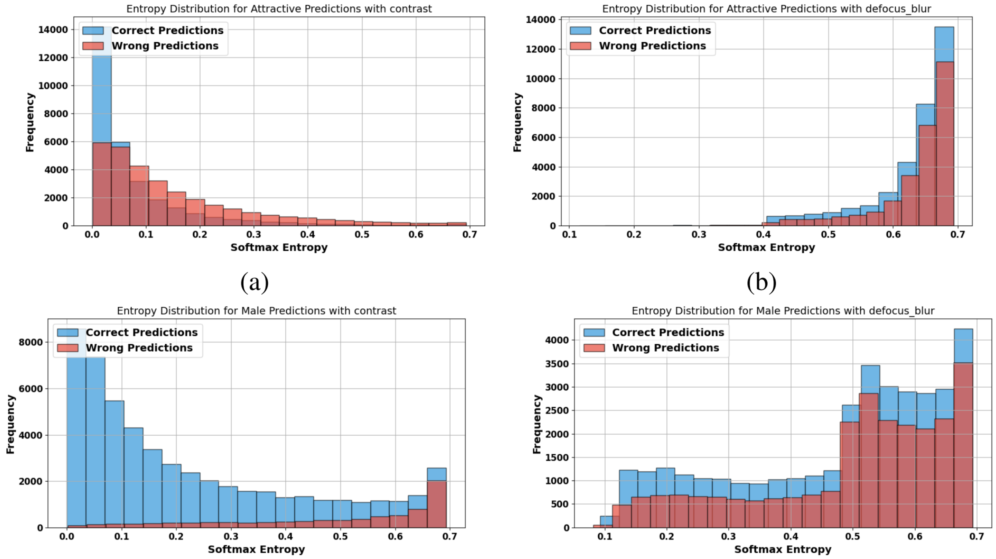
In-distribution and Domain-shifted Accuracy. We observe that each type of domain shifts impact each tasks in a different way, which causes significant accuracy degradation for some tasks. Notably, the results show that with distribution shift the model is failing not only in identifying the correct features but also **learning incorrect mapping with the class labels**. For example, in most cases the weighted F_1-score highlights that the model is mapping a significant number of features to completely opposite labels. This also highlights how the distribution shift impacts the general prediction for multitask models across different task objectives.

However, certain domain shifts like 'contrast', have lower impact on performance on some of the tasks like T_M. Figure 4 shows the impact of shifts on the feature embeddings extracted by the unadapted shared encoder for the task T_M. We observe that the *model fails to segregate the classes in the feature space*, causing the significant *degradation of model performance* for domain shifts like the addition of Gaussian noise.



(a) (b) (c)

Figure 4: Feature embeddings from the CelebA pre-trained model for T_M with – (a) No domain shift (b) Gaussian noise shift, and (c) Contrast shift.



(a) (b)

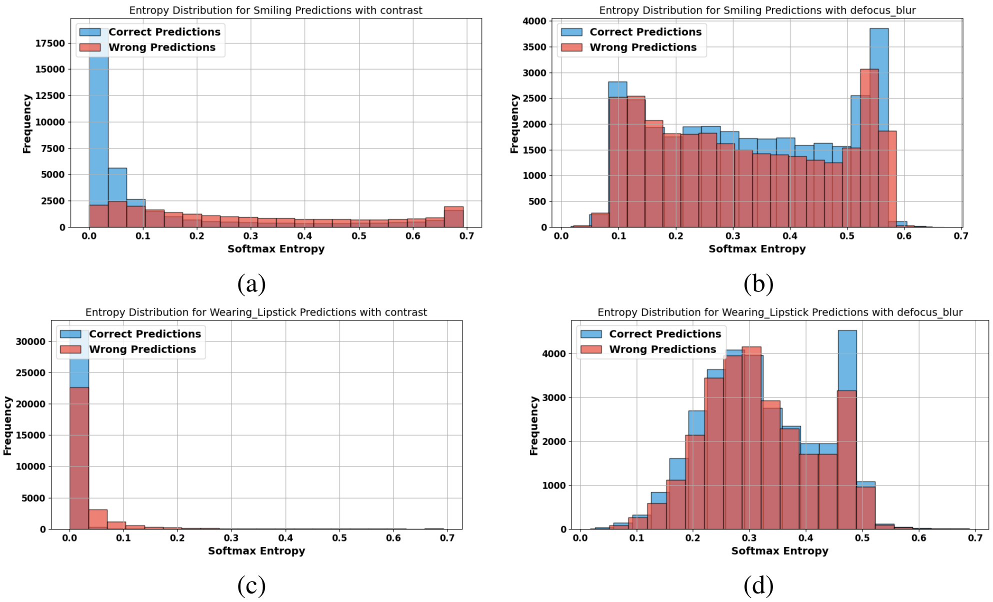
(c) (d)

Figure 5: Distribution of softmax entropy across different tasks with various domain shifts – (a) Noise Contrast on 'Attractive', (b) Defocus blur on 'Attractive', (c) Contrast on 'Male', and (d) Defocus blur on 'Male'.

Key Observations from class distribution entropy. As indicated in works like [10], softmax-entropy can be a good measure of the certainty of a model. The example figures shown in Figure 5 and Figure 6 show that different domain shifts impact the certainty of the model differently. In most cases, the model tends to become more uncertain (higher entropy) under domain shift. However, for some instances like Figure 5 (a) and Figure 6 (b)-(d), the model tends to provide wrong predictions with higher certainty (lower entropy) under the impact of noises like defocus blur and contrast shift. This highlights that the same domain shift can also cause a model to make confident yet wrong predictions for some tasks while being more uncertain on others.

5 Analysing M-TENT on CelebA-C

Evaluating M-TENT with Policies In this section, we evaluate four policies for M-TENT. These policies are designed to based on the observed correlation between the tasks (as shown in Figure 1). The primary goal of using these policies is to see how the correlation impact the uncertainty between the tasks under the distribution shift. More specifically, we design the following four policies for a principled evaluation. i) using all four tasks for adaptation, ii) using only T_A for adaptation, iii) using T_A and T_L for adaption, and iv) using only T_M for adaptation. For each policies with more than one tasks involved, the loss is averaged across the tasks used for adaptation. The performance



(a)

(b)

(c)

(d)

Figure 6: Distribution of softmax entropy across different tasks with various domain shifts – (a) Noise Contrast on ‘Smiling’, (b) Defocus blur on ‘Smiling’, (c) Contrast on ‘Wearing Lipstick’, and (d) Defocus blur on ‘Wearing Lipstick’.

accuracy of M-TENT is reported in Table 3, and the critical observations are summarized as follows.

Table 3: Test accuracy after adapting using M-TENT. The drop in performance is reported by a negative change highlighted in bold and red with a down-arrow. Increase in performance is reported in green and an uparrow with certain entries in bold highlighting those conditions where even after adaptation the performance was less than 60%.

Task	All Tasks						Only Attractive					
	Gaussian	Shot	Impulse	Defocus	Brightness	Contrast	Gaussian	Shot	Impulse	Defocus	Brightness	Contrast
Attractive	↓ **32.54**	↓ **32.54**	↓ **32.55**	↓ **33.01**	↓ **32.85**	↓ **32.94**	↓ **32.54**	↓ **32.54**	↓ **32.54**	↓ **32.87**	↓ **32.69**	↓ **32.73**
Male	↓ **25.45**	↓ **25.59**	↓ **25.45**	↑ **46.13**	↑ 93.13	↑ 96.01	↓ **34.96**	↓ **35.4**	↓ **35.53**	↑ **48.67**	↑ 86.73	↑ 89.48
Smiling	↑ **35.89**	↑ **35.9**	↑ **35.88**	↑ **36.29**	↑ 85.27	↑ 88.17	↑ **37.03**	↑ **37.61**	↑ **36.96**	↑ **43.93**	↑ 79.11	↑ 81.92
Wearing Lipstick	↑ **37.0**	↑ **37.01**	↑ **37.01**	↑ **37.27**	↑ 88.1	↑ **38.71**	↑ **37.01**	↑ **37.02**	↑ **37.02**	↑ **38.73**	↑ **51.81**	↑ **45.26**
Task	Attractive and Wearing Lipstick						Only Male					
	Gaussian	Shot	Impulse	Defocus	Brightness	Contrast	Gaussian	Shot	Impulse	Defocus	Brightness	Contrast
Attractive	↓ **32.54**	↓ **32.54**	↓ **32.55**	↓ **32.89**	↓ **32.77**	↓ **32.82**	↓ **32.72**	↓ **32.6**	↓ **32.85**	↓ **47.96**	↑ 66.57	↑ 69.36
Male	↓ **28.44**	↓ **29.0**	↓ **28.83**	↓ **41.95**	↑ **45.49**	↓ **58.24**	↓ **25.54**	↓ **25.74**	↓ **25.76**	↓ **42.69**	↑ 92.78	↑ 96.49
Smiling	↑ **36.77**	↑ **37.31**	↑ **36.84**	↑ **42.52**	↑ 75.63	↑ 78.96	↑ **37.64**	↑ **39.04**	↑ **38.04**	↑ **50.09**	↑ 86.66	↑ 87.83
Wearing Lipstick	↑ **37.0**	↑ **37.01**	↑ **37.0**	↑ **37.2**	↑ **45.25**	↑ **38.25**	↑ **37.03**	↑ **37.03**	↑ **37.06**	↑ **48.89**	↑ 86.87	↑ 87.91

Key Observations

Firstly, the straightforward policy of **averaging the softmax entropy across all tasks might not be the best policy** for adapting a multitask model. For example, considering the performance of the model across all tasks under brightness and contrast, as shown in Table 3, adaptation using only T_M provides significantly higher weighted F_1-score for all tasks as compared to when adapting using all the tasks.

Secondly, we observe that the straightforward **inter-task correlation may not be a strong indicator of the best policy for adaptation**. For example, the task T_A is positively correlated with T_S and T_L. However, as shown in Table 3, adapting only T_A does not improve the model's performance for the tasks T_S and T_L. On the other hand, adapting T_M, which is negatively correlated with all other tasks, improves the model performance significantly for all the tasks under the domain shifts caused by shifts in brightness and contrast.

Finally, we can conclude that **the effect of domain shift** is **heterogeneous across the tasks and the adaptation policies** we tried. For example, adaptation to brightness and contrast noise is comparably more successful across all policies. In contrast, adaptation to shot, impulse, and Gaussian noise is more challenging as the model mostly makes insignificant gains or no gains while adapting for any tasks under these domain shifts.

6 Discussion and Future Work

Works like TENT have been developed in the past few years, and since then, there have been many developments in TTA with works like [14, 15] even refining entropy minimization with sharpness-aware strategies. Similarly, works like SHOT [13] and AdaContrast [16] consider vanilla softmax entropy as a part of the overall optimization function. Alternatively, a few approaches like [30, 31] allow an optimization-free TTA for fine-tuning the models at runtime. The optimization-free approaches often rely on the class prototypes to fine-tune the model at runtime, albeit still depend on entropy-based sample selection at runtime [31]. Such an approach with class prototype-based feature matching and classifier fine-tuning may need additional investigation considering a multitask model where a particular shift has varying impacts across the different tasks.

The key objective of this paper is to take the first step in analyzing and discussing the impact of domain shift on multitask models, considering TTA as a solution mechanism. In the future, we aim to integrate and analyze the specific impact of domain shift on all these approaches in a multitask setting as part of a more extensive future study. We also envision that the lessons learned from the study would allow us to develop a more robust TTA framework catering to a multitask model's needs.

Notably, the above problem appears beyond vision applications; for example, a healthcare application on a smartwatch might utilize a single model to detect arrhythmia and atrial fibrillation from PPG sensor data [3]. While we limit our focus to vision applications in this exploration, other modalities are equally important to be explored in the future. Also, we believe, our way to create CelebA-C dataset will be useful in benchmarking similar methods in the future.

7 Conclusions

In this paper, we explore the idea of performing TTA on multitask models using a softmax entropy minimization-based approach. To perform this, we first introduce the benchmark CelebA-C dataset and extend the SOTA TTA algorithm to develop M-TENT, a framework for performing TTA on multitask models. Systematic analysis shows that the effect of domain shift is heterogeneous across different tasks and highlights the necessity of designing TTA approaches specific to the needs of the multitask model.

References

[1] Rajeev Ranjan, Vishal M Patel, and Rama Chellappa. Hyperface: A deep multi-task learning framework for face detection, landmark localization, pose estimation, and gender recognition. *IEEE transactions on pattern analysis and machine intelligence*, 41(1):121–135, 2017.

[2] Apple. A multi-task neural architecture for on-device scene analysis. `https://machinelearning.apple.com/research/on-device-scene-analysis`, published June 2022. [Accessed 18-08-2024].

[3] Jessica Torres-Soto and Euan A Ashley. Multi-task deep learning for cardiac rhythm detection in wearable devices. *NPJ digital medicine*, 3(1):116, 2020.

[4] Jeff Donahue, Yangqing Jia, Oriol Vinyals, Judy Hoffman, Ning Zhang, Eric Tzeng, and Trevor Darrell. Decaf: A deep convolutional activation feature for generic visual recognition. In Eric P. Xing and Tony Jebara, editors, *Proceedings of the 31st International Conference on Machine Learning*, volume 32 of *Proceedings of Machine Learning Research*, pages 647–655, Bejing, China, 22–24 Jun 2014. PMLR.

[5] Yu Sun, Eric Tzeng, Trevor Darrell, and Alexei A Efros. Unsupervised domain adaptation through self-supervision. *arXiv preprint arXiv:1909.11825*, 2019.

[6] Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario March, and Victor Lempitsky. Domain-adversarial training of neural networks. *Journal of machine learning research*, 17(59):1–35, 2016.

[7] Garrett Wilson and Diane J Cook. A survey of unsupervised deep domain adaptation. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 11(5):1–46, 2020.

[8] Mahsa Baktashmotlagh, Mehrtash T Harandi, Brian C Lovell, and Mathieu Salzmann. Unsupervised domain adaptation by domain invariant projection. In *Proceedings of the IEEE international conference on computer vision*, pages 769–776, 2013.

[9] Youngjae Chang, Akhil Mathur, Anton Isopoussu, Junehwa Song, and Fahim Kawsar. A systematic study of unsupervised domain adaptation for robust human-activity recognition. 4(1), mar 2020.

[10] Dequan Wang, Evan Shelhamer, Shaoteng Liu, Bruno Olshausen, and Trevor Darrell. Tent: Fully test-time adaptation by entropy minimization. *arXiv preprint arXiv:2006.10726*, 2020.

[11] Shuaicheng Niu, Jiaxiang Wu, Yifan Zhang, Zhiquan Wen, Yaofo Chen, Peilin Zhao, and Mingkui Tan. Towards stable test-time adaptation in dynamic wild world. *arXiv preprint arXiv:2302.12400*, 2023.

[12] Qin Wang, Olga Fink, Luc Van Gool, and Dengxin Dai. Continual test-time domain adaptation. In *Proceedings of Conference on Computer Vision and Pattern Recognition*, 2022.

[13] Jian Liang, Dapeng Hu, and Jiashi Feng. Do we really need to access the source data? source hypothesis transfer for unsupervised domain adaptation. In *International Conference on Machine Learning (ICML)*, pages 6028–6039, 2020.

[14] Taesik Gong, Jongheon Jeong, Taewon Kim, Yewon Kim, Jinwoo Shin, and Sung-Ju Lee. Note: Robust continual test-time adaptation against temporal correlation. *Advances in Neural Information Processing Systems*, 35:27253–27266, 2022.

[15] Taesik Gong, Yewon Kim, Taeckyung Lee, Sorn Chottananurak, and Sung-Ju Lee. Sotta: Robust test-time adaptation on noisy data streams. *arXiv preprint arXiv:2310.10074*, 2023.

[16] Dian Chen, Dequan Wang, Trevor Darrell, and Sayna Ebrahimi. Contrastive test-time adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 295–305, 2022.

[17] Jian Liang, Dapeng Hu, Yunbo Wang, Ran He, and Jiashi Feng. Source data-absent unsupervised domain adaptation through hypothesis transfer and labeling transfer. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 2021. In Press.

[18] Taesik Gong, Jongheon Jeong, Taewon Kim, Yewon Kim, Jinwoo Shin, and Sung-Ju Lee. Note: Robust continual test-time adaptation against temporal correlation, 2023.

[19] Hao Zhao, Yuejiang Liu, Alexandre Alahi, and Tao Lin. On pitfalls of test-time adaptation. In *International Conference on Machine Learning*, pages 42058–42080. PMLR, 2023.

[20] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *Proceedings of International Conference on Computer Vision (ICCV)*, December 2015.

[21] Dan Hendrycks and Thomas Dietterich. Benchmarking neural network robustness to common corruptions and perturbations. *Proceedings of the International Conference on Learning Representations*, 2019.

[22] Timnit Gebru, Judy Hoffman, and Li Fei-Fei. Fine-grained recognition in the wild: A multitask domain adaptation approach. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, Oct 2017.

[23] Jogendra Nath Kundu, Nishank Lakkakula, and R. Venkatesh Babu. Um-adapt: Unsupervised multi-task adaptation using adversarial cross-task distillation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2019.

[24] Tao Sun, Mattia Segu, Janis Postels, Yuxuan Wang, Luc Van Gool, Bernt Schiele, Federico Tombari, and Fisher Yu. Shift: A synthetic driving dataset for continuous multi-task domain adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 21371–21382, June 2022.

[25] Yu Sun, Xiaolong Wang, Liu Zhuang, John Miller, Moritz Hardt, and Alexei A. Efros. Test-time training with self-supervision for generalization under distribution shifts. In *ICML*, 2020.

[26] Yossi Gandelsman, Yu Sun, Xinlei Chen, and Alexei A Efros. Test-time training with masked autoencoders. In Alice H. Oh, Alekh Agarwal, Danielle Belgrave, and Kyunghyun Cho, editors, *Advances in Neural Information Processing Systems*, 2022.

[27] Soumyajit Chatterjee, Fahim Kawsar, and Mohammad Malekzadeh. Poster: Test-time training for sensor data classification via time-series change identification. In *Proceedings of the 25th International Workshop on Mobile Computing Systems and Applications*, HOTMOBILE '24, page 139, New York, NY, USA, 2024. Association for Computing Machinery.

[28] Ethan M Rudd, Manuel Günther, and Terrance E Boult. Moon: A mixed objective optimization network for the recognition of facial attributes. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part V 14*, pages 19–35. Springer, 2016.

[29] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[30] Yusuke Iwasawa and Yutaka Matsuo. Test-time classifier adjustment module for model-agnostic domain generalization. *Advances in Neural Information Processing Systems*, 34:2427–2440, 2021.

[31] Shuoyuan Wang, Jindong Wang, Huajun Xi, Bob Zhang, Lei Zhang, and Hongxin Wei. Optimization-free test-time adaptation for cross-person activity recognition. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.*, 7(4), jan 2024.