UTILIZING CROSS-VERSION CONSISTENCY FOR DOMAIN ADAPTATION: A CASE STUDY ON MUSIC AUDIO

Lele Liu, Christof Weiß
Center for Artificial Intelligence and Data Science, Universität Würzburg, Germany
{lele.liu,christof.weiss}@uni-wuerzburg.de

ABSTRACT

Deep-learning models are commonly trained on large annotated corpora, often in a specific domain. Generalization to another domain without annotated data is usually challenging. In this paper, we address such unsupervised domain adaptation based on the teacher–student learning paradigm. For improved efficacy in the target domain, we propose to exploit cross-version scenarios, i.e., corresponding data pairs assumed to obtain the same yet unknown labels. More specifically, our idea is to compare teacher annotations across versions and use only consistent annotations as labels to train the student model. Examples of cross-version data include the same text by different speakers (in speech recognition) or the same character by different writers (in handwritten text recognition). In our case study on music audio, versions are different recorded performances of the same composition, aligned with music synchronization techniques. Taking multi-pitch estimation (a multi-label classification task) as an example, we show that exploiting cross-version information in student training helps to improve the transfer from a source domain (piano) to unseen and more complex target domains (singing/orchestra).

1 INTRODUCTION

Annotating large amounts of real-world data is usually tedious. To alleviate neural network training from extensive annotation works, unsupervised domain adaptation (UDA) tries to adapt models trained on source domains (e. g., synthetic data) to non-annotated target domains (e. g., complex real-world data), see Liu et al. (2022). Over the years, various methods have been proposed to tackle the UDA problem including encoder–decoder techniques (Wang & Deng, 2018), adversarial learning (Sankaranarayanan et al., 2018; He et al., 2020), and self-training (Liu et al., 2021). A possible method for UDA is teacher–student learning (Hu et al., 2022) where pseudo-labels predicted by the teacher model are used for training the student model in the target domain. For example, Amosy & Chechik (2020) proposed a regularization step to enforce consistency between teacher and student representations. Koh & Fernando (2022) and Scherer et al. (2022) explore consistency regularization by modelling inter-pixel relationships between model outputs under different perturbations.

2 PROPOSED STRATEGY

Our work is based on this teacher–student learning paradigm. Rather than exploiting cross-domain consistency, we explore the use of *cross-version consistency* for improving domain adaptation. We consider the case of having access to pairs of versions in the target domain. Examples of versions are different handwritings of the same text, different photographs of the same scene, or different performances of the same composition, which we need to align to obtain locally corresponding pairs. Given labelled data from the teacher (source) domain $\{\boldsymbol{X}_\mathrm{T}, \boldsymbol{y}_\mathrm{T}\}$ and unlabelled data from the student (target) domain $\{\boldsymbol{X}_\mathrm{S}^a, \boldsymbol{X}_\mathrm{S}^b\}$, where a and b are versions, our task is to adapt a teacher model $f_\mathrm{T} : \boldsymbol{X}_\mathrm{T} \rightarrow \boldsymbol{y}_\mathrm{T}$ trained on the teacher domain to a student model $f_\mathrm{S} : \boldsymbol{X}_\mathrm{S} \rightarrow \boldsymbol{y}_\mathrm{S}$ for the student domain. We estimate pseudo-labels for the student domain examples using the teacher model by $\boldsymbol{y}_\mathrm{S} = f_\mathrm{T}(\boldsymbol{X}_\mathrm{S})$. Since different versions should have the same predictions at semantically corresponding positions, we only use consistent pseudo-labels for training the student network and discard inconsistent pairs. With this strategy (Figure 1), we aim for obtaining more reliable labels to train the student model, which we hypothesize to improve model efficacy in the student domain.

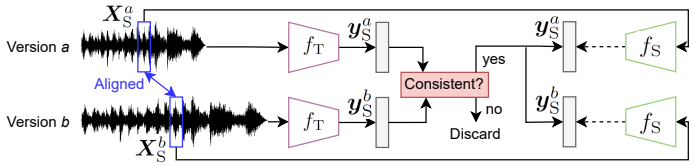

Figure 1: Domain adaptation pipeline using cross-version consistency.

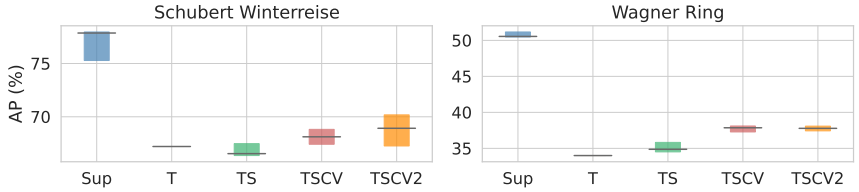

Figure 2: Average-Precision (AP) scores on the two student domain test sets for different strategies.

3 A CASE STUDY ON MUSIC AUDIO TRANSCRIPTION

Our case study on music audio is based on a music transcription sub-task denoted as framewise transcription or *multi-pitch estimation* (Benetos et al., 2019)—a multi-label classification task, which predicts active pitches (quantized, like "piano keys") for a given time frame of a polyphonic music recording. We consider a domain shift scenario from one specific instrumentation in the teacher domain to other instrumentations in the student domain. Concretely, our teacher domain are piano performances, where target labels can be easily obtained using pianos with key sensors. We use such data from the MAESTRO dataset (Hawthorne et al., 2019) and test transfer to two student domains: 19th century Lieder (art songs, i. e., piano with expressive singing) using the Schubert Winterreise dataset (SWD) (Weiß et al., 2021), and late Romantic opera (large orchestra with highly expressive singing) using the Wagner Ring dataset (WRD) (Weiß et al., 2023). Compared to the subtle domain shift between performances, these shifts are substantially more challenging due to the different acoustic properties (timbre, vibrato, etc.) of voices and instruments. Both SWD and WRD have pitch labels—allowing us to compare with supervised training—and different versions of the same compositions (9 versions in SWD, 16 in WRD), from which we sample version pairs.

We compare five different configurations on the student domain: `Sup`: Supervised training using ground-truth labels of SWD or WRD; `T`: Teacher model without domain adaptation; `TS`: Vanilla teacher–student learning, where the student model is trained using pseudo-labels generated by the teacher model; `TSCV`: Cross-version teacher–student learning, where the student model is trained using only those cross-version consistent pseudo-labels generated by the teacher model; and `TSCV2`: Cross-version TS, using all version pairs by interpolating inconsistent pseudo-labels.

We use the convolutional ResNet by Weiß & Peeters (2022) both as teacher and student model (same size, 4.8M parameters, achieving an average precision (AP) score of 89.9% in the teacher domain). We split the student datasets into training, validation, and test sets in a way that they neither overlap regarding work parts (song or act) nor versions. Figure 2 shows the student domain results (AP). Compared to supervised training (`Sup`), the teacher's efficacy on the student domain is substantially worse without adaptation (`T`), which we expect due to the different input signals. Vanilla teacher–student learning (`TS`) only sometimes has a positive effect when adapting to the student domain. In contrast, adding the consistency constraint (`TSCV`) leads to a stable improvement over both (`T`) and (`TS`). We observe comparable efficacy for `TSCV2` where we replace consistency filtering by pseudo-label interpolation. The expressive operas in WRD lead to worse results, especially for the domain transfer. In such complex scenarios, where high-quality annotations are hard to obtain, cross-version information has a stronger positive effect, which is encouraging for applying this strategy in the wild.

4 CONCLUSION

This study explores cross-version scenarios for domain adaptation in the teacher–student learning paradigm. We verify the usefulness of cross-version information in a case study on framewise music transcription where we adapt a model trained on solo piano music to more complex instrumentations.

REPRODUCIBILITY

For reproducibility purposes, we release our experimental code at: https://github.com/cheriell/Cross-Version-MPE. The datasets used for our study are publicly available (Hawthorne et al., 2019; Weiß et al., 2021; 2023). The pre-trained models' weights are available at https://zenodo.org/records/10936492.

URM STATEMENT

We acknowledge that both authors of this work meet the URM criteria of ICLR 2024 Tiny Papers Track.

ACKNOWLEDGEMENTS

This work was funded by the German Research Foundation (Deutsche Forschungsgemeinschaft, DFG) within the Emmy Noether Junior Research Group on *Computational Analysis of Music Audio Recordings: A Cross-Version Approach* (DFG WE 6611/3-1, Grant No. 531250483).

REFERENCES

Ohad Amosy and Gal Chechik. Teacher-student consistency for multi-source domain adaptation. *CoRR*, abs/2010.10054, 2020.

Emmanouil Benetos, Simon Dixon, Zhiyao Duan, and Sebastian Ewert. Automatic music transcription: An overview. *IEEE Signal Processing Magazine*, 36(1):20–30, 2019. doi: 10.1109/MSP.2018.2869928.

Curtis Hawthorne, Andriy Stasyuk, Adam Roberts, Ian Simon, Cheng-Zhi Anna Huang, Sander Dieleman, Erich Elsen, Jesse H. Engel, and Douglas Eck. Enabling factorized piano music modeling and generation with the MAESTRO dataset. In *Proceedings of the International Conference on Learning Representations (ICLR)*, New Orleans, Louisiana, USA, 2019.

Gewen He, Xiaofeng Liu, Fangfang Fan, and Jane You. Classification-aware semi-supervised domain adaptation. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR Workshops 2020, Seattle, WA, USA, June 14-19, 2020*, pp. 4147–4156. Computer Vision Foundation / IEEE, 2020. doi: 10.1109/CVPRW50498.2020.00490.

Chengming Hu, Xuan Li, Dan Liu, Xi Chen, Ju Wang, and Xue Liu. Teacher-student architecture for knowledge learning: A survey, 2022. arXiv preprint arXiv:2210.17332 [cs.LG].

Kian Boon Koh and Basura Fernando. Consistency regularization for domain adaptation. In *European Conference on Computer Vision*, pp. 347–359, 2022.

Xiaofeng Liu, Bo Hu, Xiongchang Liu, Jun Lu, Jane You, and Lingsheng Kong. Energy-constrained self-training for unsupervised domain adaptation. In *2020 25th International Conference on Pattern Recognition (ICPR)*, pp. 7515–7520, 2021. doi: 10.1109/ICPR48806.2021.9413284.

Xiaofeng Liu, Chaehwa Yoo, Fangxu Xing, Hyejin Oh, Georges El Fakhri, Je-Won Kang, Jonghye Woo, et al. Deep unsupervised domain adaptation: A review of recent advances and perspectives. *APSIPA Transactions on Signal and Information Processing*, 11(1), 2022.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2019.

Meinard Müller, Yigitcan Özer, Michael Krause, Thomas Prätzlich, and Jonathan Driedger. Sync Toolbox: A Python package for efficient, robust, and accurate music synchronization. *Journal of Open Source Software (JOSS)*, 6(64):3434:1–4, 2021. doi: 10.21105/joss.03434.

Swami Sankaranarayanan, Yogesh Balaji, Carlos D Castillo, and Rama Chellappa. Generate to adapt: Aligning domains using generative adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 8503–8512, 2018.

Sebastian Scherer, Stephan Brehm, and Rainer Lienhart. Consistency regularization for unsupervised domain adaptation in semantic segmentation. In *Image Analysis and Processing – ICIAP 2022*, pp. 500–511, 2022. doi: 10.1007/978-3-031-06427-2_42.

John Thickstun, Zaïd Harchaoui, Dean P. Foster, and Sham M. Kakade. Invariances and data augmentation for supervised music transcription. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 2241–2245, Calgary, Canada, 2018.

Mei Wang and Weihong Deng. Deep visual domain adaptation: A survey. *Neurocomputing*, 312: 135–153, 2018. doi: 10.1016/J.NEUCOM.2018.05.083.

Christof Weiß, Frank Zalkow, Vlora Arifi-Müller, Meinard Müller, Hendrik Vincent Koops, Anja Volk, and Harald Grohganz. Schubert Winterreise dataset: A multimodal scenario for music analysis. *ACM Journal on Computing and Cultural Heritage (JOCCH)*, 14(2):25:1–18, 2021. doi: 10.1145/3429743.

Christof Weiß, Vlora Arifi-Müller, Michael Krause, Frank Zalkow, Stephanie Klauk, Rainer Kleinertz, and Meinard Müller. Wagner Ring dataset: A complex opera scenario for music processing and computational musicology. *Transactions of the International Society for Music Information Retrieval (TISMIR)*, 6(1):135–149, 2023. doi: 10.5334/TISMIR.161.

Christof Weiß and Geoffroy Peeters. Comparing deep models and evaluation strategies for multi-pitch estimation in music recordings. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 30:2814–2827, 2022. doi: 10.1109/TASLP.2022.3200547.

A Appendix

A.1 Cross-version consistency calculation

We use a music synchronization algorithm (Müller et al., 2021) based on dynamic time warping to align two music performances of the same composition and use the aligned time frames as a data pair from the student domain. We then calculate and compare the teacher annotations for the aligned data pair.

Cross-Version Strategy 1. As our first strategy, denoted as `TSCV` in the main text, we only consider consistent version pairs. We then include both $\boldsymbol{X}_{\mathrm{S}}^{a}$ and $\boldsymbol{X}_{\mathrm{S}}^{b}$ for training with pseudo-labels

$$\boldsymbol{y}_{\mathrm{S}} = \boldsymbol{y}_{\mathrm{S}}^{a} = \boldsymbol{y}_{\mathrm{S}}^{b} := f_{\mathrm{T}}(\boldsymbol{X}_{\mathrm{S}}^{a}) = f_{\mathrm{T}}(\boldsymbol{X}_{\mathrm{S}}^{b}) \tag{1}$$

and simply discard other version pairs (see Figure 1). Since this strategy may substantially reduce the amount of training data, we also test relaxed variants where we (1) alleviate the consistency calculation from minor differences between the pseudo-labels and (2) allow for a certain tolerance in the alignment. In our reported results, we allow for a maximum of 2 pitch differences between version pairs and ± 2 time frames in the alignment error.

Cross-Version Strategy 2. As an alternative strategy, denoted as `TSCV2`, we aim for resolving the conflicting pseudo-labels in inconsistent version pairs. To this end, we update the pseudo-labels by interpolating them across versions

$$\boldsymbol{y}_{\mathrm{S}} = \boldsymbol{y}_{\mathrm{S}}^{a} = \boldsymbol{y}_{\mathrm{S}}^{b} := \frac{1}{2} \cdot \left(f_{\mathrm{T}}(\boldsymbol{X}_{\mathrm{S}}^{a}) + f_{\mathrm{T}}(\boldsymbol{X}_{\mathrm{S}}^{b}) \right) . \tag{2}$$

Please note that the output of the teacher model f_{T} is binarized. As a result, the resulting activation of a pitch is 0 or 1 (for consistent pairs regarding this pitch) or 0.5 (in case of inconsistence).

A.2 Training configurations

We use the same training configurations across teacher and student models (also for the supervised learning model). During training, we use a learning rate of 0.0002, a batch size of 25, and the AdamW optimizer (Loshchilov & Hutter, 2019) We use the binary cross-entropy loss for the labelling task. We perform early stopping if the validation loss does not decrease over twelve epochs.

Table 1: Detailed results on the student domains.

	Schubert Winterreise Dataset					Wagner Ring Dataset				
Model	Run	P	R	F	AP	Run	P	R	F	AP
`Sup`	1	54.7	85.8	66.5	75.3	1	53.6	46.7	49.8	50.5
	2	62.3	81.4	70.2	78.0	2	52.1	51.5	51.6	51.2
	3	65.9	78.4	71.2	78.0	3	55.0	43.7	48.6	50.6
`T`	–	72.3	56.0	62.0	67.2	–	45.2	29.6	35.3	34.0
`TS`	1	70.5	57.2	62.2	66.4	1	44.0	36.0	39.1	34.9
	2	66.9	61.5	63.1	66.5	2	46.6	32.2	37.5	35.8
	3	67.5	63.1	64.4	67.5	3	44.5	35.7	39.2	34.5
`TSCV`	1	71.9	58.3	63.5	68.1	1	48.3	32.1	38.1	37.3
	2	71.9	58.6	63.8	68.8	2	46.1	36.9	40.7	38.1
	3	73.1	54.9	61.8	67.4	3	48.2	33.1	38.9	37.9
`TSCV2`	1	72.5	61.5	65.9	70.2	1	51.2	27.3	35.2	38.1
	2	70.0	63.6	66.0	68.9	2	50.8	26.4	34.3	37.4
	3	65.5	65.9	65.1	67.3	3	49.1	30.4	37.2	37.8

For each training configuration (except for the teacher model `T`), we repeat the experiment three times (runs) to account for randomization effects in our evaluation.

When training the teacher model, we use the original train/validation/test split provided with the MAESTRO dataset (v3.0.0). For the training on the student domain, we use a split where neither versions nor work parts overlap between subsets (*neither-split*). For the SWD, we use five versions for training, two for validation and two for testing; for the WRD, we use nine versions for training, three versions for validation and the remaining three for testing.[1]

A.3 Detailed results

More detailed results on the repeating runs and different evaluation metrics can be found in Table 1. For evaluation, we use a threshold of 0.4 as in Thickstun et al. (2018) to calculate the precision (P), recall (R) and F-score (F). We first average measures over time frames for each test piece and secondly over pieces in the test set. This prevents longer work parts from being weighted more in the evaluation. Since P, R, and F are directly influenced by the choice of the threshold, we report the average precision score (AP , the area under the Precision–Recall curve), which is invariant to the threshold. We also reported this metric in the main text. We used the AP scores of the repeating runs to plot Figure 2; the ordered results of the repeating runs correspond to the upper bound, middle line, and lower bound of the box plots.

[1]For practical reasons (re-using an internal prior version of the WRD), the set of versions slightly deviates from the ones published in Weiß et al. (2023) with two additional and three missing versions.