Towards Label-Agnostic Emotion Embeddings

Sven Buechel **Luise Modersohn** **Udo Hahn**
Jena University Language & Information Engineering (JULIE) Lab
Friedrich-Schiller-Universität Jena, Fürstengraben 27, 07743 Jena, Germany
firstname.lastname@uni-jena.de
https://julielab.de

Abstract

Research in emotion analysis is scattered across different label formats (e.g., polarity types, basic emotion categories, and affective dimensions), linguistic levels (word vs. sentence vs. discourse), and, of course, (few well-resourced but much more under-resourced) natural languages and text genres (e.g., product reviews, tweets, news). The resulting heterogeneity makes data and software developed under these conflicting constraints hard to compare and challenging to integrate. To resolve this unsatisfactory state of affairs we here propose a training scheme that learns a shared latent representation of emotion independent from different label formats, natural languages, and even disparate model architectures. Experiments on a wide range of datasets indicate that this approach yields the desired interoperability without penalizing prediction quality. Code and data are archived under DOI 10.5281/zenodo.5466068.

1 Introduction

Emotion analysis in the field of NLP[1] has experienced a remarkable evolution of representation schemes. Starting from the early focus on *polarity*, i.e., the main distinction between positive and negative feelings emerging from natural language utterances (Hatzivassiloglou and McKeown, 1997; Turney and Littman, 2003), the number and variety of label formats, i.e., groups of emotional target variables and their associated value ranges, has been growing rapidly (Bostan and Klinger, 2018; De Bruyne et al., 2020). This development is a double-edged sword though.

On the one hand, the wide variety of available label formats allows NLP models to become more informative and richer in expressive power. This gain is because many of the newer representation schemes follow well-researched branches of psychological theory, such as basic emotion categories or affective dimensions (Ekman, 1992; Russell and Mehrabian, 1977), which offer information complementary to each other (Stevenson et al., 2007). Others argue that different emotional nuances turn out to be particularly useful for specific targeted downstream applications (Bollen et al., 2011; Desmet and Hoste, 2013).

On the other hand, this proliferation of label formats has led to a severe loss in cross-data comparability. As Tab. 1 illustrates, the total volume of available gold data is spread not only over distinct languages but also a huge number of emotion annotation schemes. Consequently, comparing or even merging data from different rating studies is often impossible. This, in turn, contributes to the development of an unnecessarily large number of prediction models, each with limited coverage of the full range of human emotion.

To escape from these dilemmata, we propose a method that mediates between such different representation schemes. In contrast to previous work which unified *some* sources of heterogeneity (see §2), to the best of our knowledge, our approach is the first to learn a representation space for emotions that *generalizes* over individual languages, emotion label formats, and distinct model architectures for emotion analysis.

Technically speaking, our approach consists of a set of pre-trained prediction heads that can be easily attached to existing state-of-the-art neural models. Doing so, a model learns to *embed* language items of a particular domain in a shared representation space that resembles an “interlingua for emotion”. These “emotion embeddings” capture a rich array of affective nuances and allow for a direct comparison of emotional load between heterogeneous samples (see Fig. 1). They may thus form a solid basis for a broad range of linguistic, psychological, and cultural follow-up studies.

[1] We use “emotion” as an umbrella term for phenomena such as polarity, sentiment, feelings, or affective states.

Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, pages 9231–9249
November 7–11, 2021. ©2021 Association for Computational Linguistics

Sample	**Val**	**Aro**	**Dom**	**Joy**	**Ang**	**Sad**	**Fea**	**Dis**
rollercoaster	8.0○	8.1○	5.1○	3.4□	1.4□	1.1□	2.8□	1.1□
urine	3.3○	4.2○	5.2○	1.9□	1.4 □	1.2□	1.4□	2.6□
szczęśliwy [(a)]	2.8●	4.0○						
College tution continues climbing				0■	54■	40■	3■	31■
A gentle, compassionate drama about grief and healing	*pos*△							
喇叭這一代還是差勁透了。[(b)]	2.8○	6.1○						

Value Ranges: ○[1, 9] ●[−3, 3] △{*pos*, *neg*} □[1, 5] ■[0, 100]

Table 1: Sample entries from various sources described along eight emotional variables: [VAD]—**Val**ence (≈ **Pol**arity), **Aro**usal, **Dom**inance, and [BE5]—**Joy**, **Ang**er, **Sad**ness, **Fea**r, and **Dis**gust. Samples differ in languages addressed (English, Polish, Mandarin), linguistic domain (word vs. text, register) and label format (covered variables and their value ranges).
Translations: [(a)] "happy" (from Polish); [(b)] "This product generation still has terrible speakers." (from Mandarin)

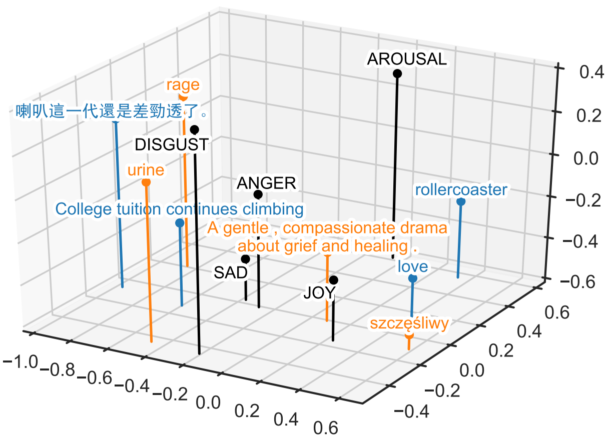


Figure 1: Emotional loading of heterogenous samples in common representation space with selected emotion variables (in capitals); first three principal components. Color only used as visual aid. Translations for non-English items are given in Tab. 1.

In terms of practical benefits, our method allows models to predict label formats unseen during training and lowers space requirements by reducing a large number of format-specific models to a small number of format-agnostic ones. Although not in the center of interest of this study, our approach also often leads to small improvements in prediction quality, as experiments on 13 datasets for 6 natural languages reveal.

2 Related Work

Representing Emotion. At the heart of computational emotion representation lies a set of *emotion variables* ("classes", "constructs") used to capture different facets of affective meaning. Researchers may choose from a multitude of approaches designed in the long and controversial history of the psychology of emotion (Scherer, 2000; Hofmann et al., 2020). A popular choice are so-called *basic emotions* (Alm et al., 2005; Aman and Szpakowicz, 2007; Strapparava and Mihalcea, 2007), such as the six categories identified by Ekman (1992): *Joy*, *Anger*, *Sadness*, *Fear*, *Disgust*, and *Surprise* (**BE6**, for short). A subset of these excluding *Surprise* (**BE5**) is often used for emotional word datasets in psychology ("affective norms") which are available for a wide range of languages.

Affective dimensions constitute a popular alternative to basic emotions (Yu et al., 2016; Sedoc et al., 2017; Buechel and Hahn, 2017; Li et al., 2017; Mohammad, 2018). The most important ones are *Valence* (negative vs. positive, thus corresponding to the notion of *polarity*; Turney and Littman, 2003) and *Arousal* (calm vs. excited) (**VA**). These two dimensions are sometimes extended by *Dominance* (feeling powerless vs. empowered; **VAD**).

Other theories influential for NLP include Plutchik's (2001) *Wheel of Emotion* (Mohammad and Turney, 2013; Abdul-Mageed and Ungar, 2017; Tafreshi and Diab, 2018; Bostan et al., 2020) and appraisal dimensions (Balahur et al., 2012; Troiano et al., 2019; Hofmann et al., 2020). Yet frequently, studies do not follow any of these established approaches but rather design a customized set of variables in an ad-hoc fashion, often driven by the availability of user-labeled data in social media, or the specifics of an application or domain which requires attention to particular emotional nuances (Bollen et al., 2011; Desmet and Hoste, 2013; Staiano and Guerini, 2014; Qadir and Riloff, 2014; Li et al., 2016; Demszky et al., 2020).

This proliferating diversity of emotion label formats is the reason for the lack of comparability outlined in §1. Our work aims to unify these heterogeneous labels by learning to translate them into a shared distributional representation (see Fig. 1).

Analyzing Emotion. There are several subtasks in emotion analysis that require distinct model types. Word-level prediction (or "emotion lexicon induction") is concerned with the emotion associated with an individual word out of context. Early work exploited primarily surface patterns of word usage (Hatzivassiloglou and McKeown, 1997; Turney and Littman, 2003) whereas more recent activities rely on more sophisticated statistical signals encoded in word embeddings (Amir et al., 2015; Rothe et al., 2016; Li et al., 2017). Combinations of high-quality embeddings with feed-forward nets have proven to be very successful, rivaling human annotation capabilities (Buechel and Hahn, 2018b).

In contrast, modeling emotion of sentences or short texts (jointly referred to as "text") was traditionally based largely on lexical resources (Taboada et al., 2011). Later, those were combined with conventional machine learning techniques (Mohammad et al., 2013) before being widely replaced by neural end-to-end approaches (Socher et al., 2013; Kim, 2014; Abdul-Mageed and Ungar, 2017). Current state-of-the-art results are achieved by transfer learning with transformer models (Devlin et al., 2019; Zhong et al., 2019; Delbrouck et al., 2020).

Our work complements these lines of research by providing a method that allows existing models to embed the emotional loading of some unit of language in a common emotion embedding space. This broadens the range of emotional nuances said models can capture. Importantly, our method learns a representation not for a specific unit of language itself but the emotion attached to it. This differs from previous work aiming to increase the affective load of, e.g., word embeddings (see below).

Emotion Embeddings. Several existing studies have used the term "emotion embeddings" (or similar phrasing) to characterize their work, yet either use the term in a different way or tackle a different problem compared to our study.

In more detail, Wang et al. (2020) present a method for increasing the emotional content of word embeddings based on re-ordering vectors according to the similarity in their emotion values, referring to the result as "emotional embeddings". Similarly, Xu et al. (2018) learn word embeddings that are particularly rich in affective information by sharing an embedding layer between models for different emotion-related tasks. They refer to these embeddings as "generalized emotion representation". Different from our work, these two studies primarily learn to represent *words* (with a focus on their affective meaning though), not emotions themselves. They are thus in line with previous research aiming to increase the affective load of word embeddings (Faruqui et al., 2015; Yu et al., 2017; Khosla et al., 2018).

Shantala et al. (2018) improve a dialogue system by augmenting their training data with emotion predictions from a separate system. Predicted emotion labels are fed into the dialogue model using a representation ("emotion embeddings") learned in a supervised fashion with the remainder of the model parameters. These embeddings are specific to their architecture and training dataset, they do not generalize to other label formats. Gaonkar et al. (2020) as well as Wang and Zong (2021) learn vector representations for emotion classes from annotated text datasets to explicitly model their semantics and inter-relatedness. Yet again, these emotion embeddings (the class representations) do not generalize to other datasets and label formats. Han et al. (2021) propose a framework for learning a common embedding space as a means of joining information from different modalities in multimodal emotion data. While these embeddings generalize over different modalities (audio and video), they do not generalize across languages and label formats. In summary, different from these studies, our emotion embeddings are not bound to any particular model architecture or dataset but instead generalize across domains and label formats, thus allowing to directly compare, say, English language items with BE5 ratings to Mandarin ones with VA ratings (see Tab. 1 vs. Fig. 1).

Coping with Incompatibility. In face of the variety of emotion formats, Felbo et al. (2017) present a transfer learning approach in which they pre-train a model with self-supervision to predict emojis in a large Twitter dataset, thus learning a representation that captures even subtle emotional nuances. Similarly, multi-task learning can be used to fit a model on multiple datasets potentially having different label formats, thus resulting in shared hidden representations (Tafreshi and Diab, 2018; Augenstein et al., 2018). While representations learned with these approaches generalize across different label formats, they do not generalize across model architectures or language domains.

Cross-lingual approaches learn a common latent representation for different languages but these representations are often specific to only one pair of languages and do not generalize to other label formats (Gao et al., 2015; Abdalla and Hirst, 2017; Barnes et al., 2018). Similarly, recent work with Multilingual BERT (Devlin et al., 2019) shows strong performance in cross-lingual zero-shot transfer (Lamprinidis et al., 2021), but samples from different languages still end up in different regions of the embedding space (Pires et al., 2019). These approaches are also specific to a particular model architecture so that they do not naturally carry over to, e.g., single-word emotion prediction. Multimodal approaches to emotion analysis show some similarity to our work, as they learn a common latent representation for several modalities which can be seen as separate domains (Zadeh et al., 2017; Han et al., 2021; Poria et al., 2019). However, these representations are typically specific to a single dataset and are not meant to generalize further.

In a recent survey on text emotion datasets, Bostan and Klinger (2018) point out naming inconsistencies between label formats. They build a joint resource that unifies twelve datasets under a common file format and annotation scheme. Annotations were unified based on the semantic closeness of their class names (e.g., merging *"happy"* and "*Joy*"). This approach is limited by its reliance on *manually* crafted rules which are difficult to formulate, especially for numerical label formats.

In contrast, emotion representation mapping (or "label mapping") aims at *automatically* learning such conversion schemes between formats from data (especially from "double-annotated" samples, such as the first two rows in Tab. 1; Stevenson et al., 2007; Calvo and Mac Kim, 2013; Buechel and Hahn, 2018a). As the name suggests, label mapping operates exclusively on the gold ratings, without actually deriving representations for language items. It can, however, be used as a post-processor, converting the prediction of another model to an alternative label format (used as a baseline in §4). Label mapping learns to transform *one* format *into another*, yet without establishing a more general representation. In a related study, De Bruyne et al. (2022) indeed do learn a common representation for different label formats by applying variational autoencoders to multiple emotion lexicons. However, their method still only operates exclusively on the gold ratings without actually predicting labels based on words or texts.

In summary, while there are methods to learn common emotion representations across *either* languages, linguistic domains, label formats, or model architectures, to the best of our knowledge, our proposal is the first to achieve all this simultaneously.

3 Methods

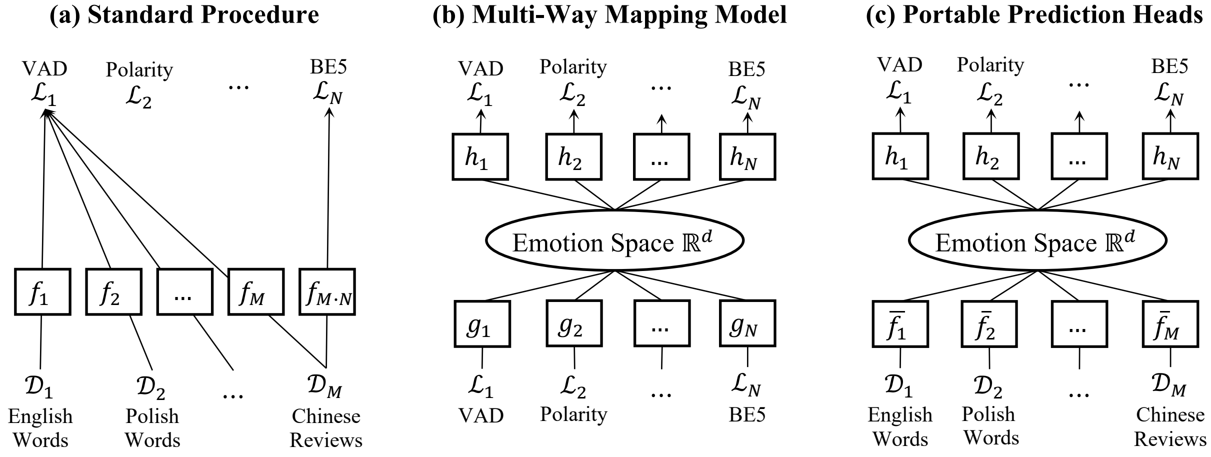
Let (X, Y) be a dataset with samples $X{:=}\{x_1, \dots x_n\}$ and labels $Y{:=}\{y_1, \dots, y_n\}$. The aim of emotion analysis is to find a model f that best predicts Y given X. Let us assume that the samples X are drawn from one of M domains $\mathcal{D}_1, \dots, \mathcal{D}_M$ and the labels are drawn from one of N label formats $\mathcal{L}_1, \dots, \mathcal{L}_N$. A domain refers to the vocabulary or a particular register of a given language (word- and text-level prediction). A label format is a set of valid labels with reference to particular emotion constructs. For instance, the VAD format consists of vectors (v, a, d) where the components v, a, d refer to *Valence*, *Arousal*, and *Dominance*, respectively, and are bound within a specified interval, e.g., $[1, 9]$.

3.1 Towards a Common Emotion Space

Fig. 2 provides an overview of our methodology. The naïve approach to emotion analysis is to learn separate models for each language domain, $\mathcal{D}_1, \dots, \mathcal{D}_M$, and label format, $\mathcal{L}_1, \dots, \mathcal{L}_N$, resulting in a potentially very high number of relatively weak models in terms of the emotional nuances they can capture *(a)*. The alternative we propose consists of two steps. First, we train a multi-way mapping that can translate between every pair of label formats $(\mathcal{L}_i, \mathcal{L}_j),\ i, j \in [1, N]$ via a shared intermediate representation layer, the common emotion space *(b)*. In a second step, we adopt existing model architectures to embed samples from a given domain in the emotion space, while the format-specific top layers of said mapping model are now utilized as portable prediction heads. The emotion space then acts as a mediating "interlingua" which connects each language domain, $\mathcal{D}_1, \dots, \mathcal{D}_M$, with each label format, $\mathcal{L}_1, \dots, \mathcal{L}_N$ *(c)*.

3.2 Prediction Head Training

A prediction head here refers to a function h that maps from a Euclidean input space $\mathbb{R}^d$ (the "emotion space") to a label format $\mathcal{L}_j$. We give prediction heads a purposefully minimalist design that consists only of a single linear layer without bias term. Thus, a head h predicts ratings $\hat{y}$ for an emotion embedding $x \in \mathbb{R}^d$ as $h(x) := Wx$, where W is a weight matrix. The reason for this simple head


(a) Standard Procedure

(b) Multi-Way Mapping Model

(c) Portable Prediction Heads

Figure 2: Overview of our methodology, illustrated by several choices of language domains and label formats.

design is to ensure that the affective information is more readily available in the emotion space. Alternatively, we can describe the weight matrix W as a concatenation of row vectors W_i, where each emotion variable corresponds to exactly one row. Thus, as a positive side effect of the lightweight design, we can directly locate emotion variables within the emotion space by interpreting their respective coefficients W_i as position vector (see Fig. 1).

Our challenge is to train a collection of heads $h_1, \ldots h_N$ such that all heads produce *consistent* label outputs for a given emotion embedding from $\mathbb{R}^d$. For example, if the VAD head predicts a *joyful* VAD label, then the BE5 head should also produce a congruent *joyful* BE5 rating. In this sense, the prediction heads are "the heart and soul" of the emotion space: they define which affective state a region of the space corresponds to.

To devise a suitable training scheme for the heads, we first need to elaborate on our understanding of "consistency" between differently formatted emotion labels. We argue that an obvious case of such consistency is found in datasets for emotion label mapping (see §2). A label mapping dataset consists of two sets of labels following different formats $Y_1 {:=} \{y_{1,1}, y_{1,2}, \ldots y_{1,n}\}$ and $Y_2 {:=} \{y_{2,1}, y_{2,2}, \ldots y_{2,n}\}$, respectively. Typically, they are constructed by matching instances from independent annotation studies (e.g., the first two rows in Tab. 1). Thus, we can think of the two sets of labels as "translational equivalents", i.e., differently formatted emotion ratings, possibly capturing different affective nuances, yet still describing the same underlying expression of emotion in humans.

The intuition behind our training scheme is to "fuse" multiple mapping models by forcing them to produce the same intermediate representation for both mapping directions. This results in a multi-way mapping model with a shared representation layer in the middle (the common emotion space) followed by the prediction heads on top (Fig. 2*b*).

In more detail (see also Fig. 3 for an illustration of the following training procedure), let (Y_1, Y_2) be a mapping dataset with a sample (y_1, y_2). We introduce two new, auxiliary models g_1, g_2 that we call *label encoders*. Label encoders embed input ratings in the emotion space $\mathbb{R}^d$ and can be combined with the complementary prediction heads h_2, h_1 to form a mapping model (the subscript here refers to the label format). That is $h_2(g_1(y_1))$ yields predictions for y_2 and $h_1(g_2(y_2))$ for y_1.

Our goal is to align both the intermediate representations, $g_1(y_1), g_2(y_2)$ while also deriving accurate mapping predictions. Therefore, we propose the following three training objectives:

$$
\begin{aligned}
L_{\text{map}} &:= \mathcal{C}[y_1, h_1(g_2(y_2))] + \mathcal{C}[y_2, h_2(g_1(y_1))] \\
L_{\text{auto}} &:= \mathcal{C}[y_1, h_1(g_1(y_1))] + \mathcal{C}[y_2, (h_2(g_2(y_2))] \\
L_{\text{sim}} &:= \mathcal{C}[g_1(y_1), g_2(y_2)]
\end{aligned}
$$

where $\mathcal{C}$ denotes the Mean-Squared-Error loss cri-

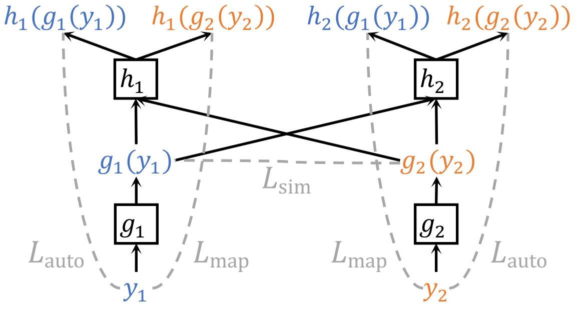


Figure 3: Training the Multi-Way Mapping Model.

terion. L_{map} is the *mapping loss* term where we compare true vs. predicted labels. The two summands represent the two mapping directions, assigning either of the two labels as the source, the other as the target format. The *autoencoder loss*, L_{auto}, captures how well the model can reconstruct the original input label from the hidden emotion representation. It is meant to supplement the mapping loss. Lastly, the *similarity loss*, L_{sim}, directly assesses whether both input label formats end up with a similar intermediate representation. The *total loss* for one instance, finally, is given by

$$L_{\text{total}} := L_{\text{map}} + L_{\text{auto}} + L_{\text{sim}}$$

In practice, we train a matching label encoder $g_1, \ldots, g_N$ for each of our prediction heads $h_1, \ldots, h_N$, thus covering all considered label formats $\mathcal{L}_1, \ldots \mathcal{L}_N$. All label encoders and prediction heads are trained simultaneously on a collection of mapping datasets. This is done as a hierarchical sampling procedure, where we first sample one of the mapping datasets (which determines the encoder and the head to be optimized in this step), then a randomly selected instance. The total loss is computed in a batch-wise fashion and the encoder and head parameters are updated via standard gradient descent-based techniques (see Appendix A for details). We use min-max scaling to normalize value ranges of the labels across datasets: for VAD we choose the interval $[-1, 1]$ and for BE5 the interval $[0, 1]$, reflecting their respective bipolar (VAD) and unipolar (BE5) nature (see Tab. 1).

3.3 Prediction Head Deployment

Following the training of the prediction heads $h_1, \ldots, h_N$, deploying them on top of a base model architecture f is relatively straightforward, resulting in a multi-headed model. The base model's output layer must be resized to the dimensionality of the emotion space $\mathbb{R}^d$ and any present nonlinearity (e.g, softmax or sigmoid activation) must be removed. This modified base model $\bar{f}$ is then optimized to produce emotion embeddings, the heads' input representation (see Fig. 4).

Head parameters are kept constant so that the base model is forced to optimize the representations it provides. Since the heads are specifically trained to treat emotion embeddings consistently, producing suitable representations for *one* head is also likely to produce suitable representations for the remaining heads. Yet, to avoid overfitting the base model to a particular one (i.e., producing representations that are particularly favorable for one head, but much less so for every other), each model $\bar{f}_i$ is trained using multiple heads depending on the available data.

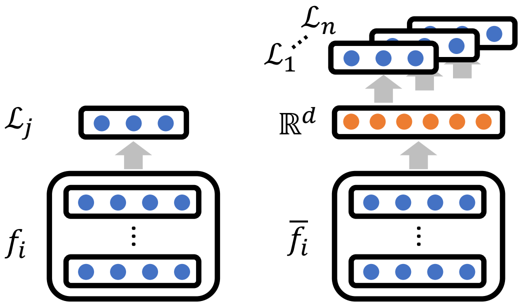


Figure 4: Schematic illustration of a base model before (left) and after (right) head deployment.

If *multiple* datasets are available that match the domain of the base model *and* use different label formats, we train the base model in a multi-task setup: We first draw one of the available datasets and then sample an instance (x, y) from there. Next, we derive a prediction using the matching head h_j as $\hat{y} := h_j(\bar{f}_i(x))$, before computing the *prediction loss*:

$$L_{\text{pred}} := \mathcal{C}[y, \hat{y}]$$

If, on the other hand, only *one* dataset is available which matches the domain of the base model $\bar{f}_i$, we complement the prediction loss with additional error signal using a newly proposed data augmentation technique. This method which we call *emotion label augmentation* synthesizes an alternative label $y^* := h_k(g_j(y))$ for a given instance (x, y) by taking advantage of the label encoder g_j that was trained in the previous step. While g_j translates the label y to the emotion space, the prediction head h_k provides labels in a format different from y. Those artificial labels are then used in place of actual gold labels resulting in the *data augmentation loss*

$$L_{\text{aug}} := \mathcal{C}[y^*, h_k(\bar{f}_i(x))]$$

where the second argument to the loss criterion $\mathcal{C}$ denotes the model's prediction for the previously synthesized labels. Then, $L_{\text{pred}} + L_{\text{aug}}$ yields the final loss.

4 Experimental Setup

The main idea behind our experimental setup is to compare a base model trained with the standard procedure against the same model with portable prediction heads (PPH) attached (cf. Fig. 2 *(a)* vs. *(c)*). Our goal is to show that we obtain the same, if not better, results using PPH compared with the naïve approach.

This study design reflects two purposes. First, comparing the base model with the PPH architecture yields experimental data that allow to indirectly assess the quality of the learned emotion representations. Second, such a comparison may help find evidence that the performance of the PPH approach *scales* with the employed base model—this would suggest that our method is likely to remain valuable even when today's state-of-the-art models are replaced by their successors. Importantly, we train only a single set of prediction heads. Thus, *all* experimental results of the PPH condition are based on the *same* underlying emotion space.

We distinguish two evaluation settings. In the first ("supervised") setting, train and test data come from (different parts of) the same dataset. Without PPH, we train one base model per dataset. Yet, with PPH, base models are shared across datasets of the same domain, whether or not their label formats agree. Consequently, the emotion space needs to store heterogeneous affective information in an easy-to-retrieve way (recall the "lightweight" head design; §3.2). Thus, positive evaluation results would indicate that our method learns a particularly rich representation of emotion. A practical advantage of PPH lies in the reduction of total disk space utilized by the resulting model checkpoints.

The second ("zero-shot") setting assumes that only *one* dataset per language is available, with one particular label format, but one would like to predict ratings in another format as well (e.g., imagine having a VA dataset for Mandarin but you are actually more interested in basic emotions for that language). Doing so with PPH is very simple—one only has to choose the desired head at inference time. Yet, doing so with the base model *per se* is simply impossible. To still be able to offer a quantitative comparison, we resort to an external label mapping component that translates the base model's output into the desired format. We emphasize that this is a very strong baseline due to the high accuracy of the label mapping approach, in general (Buechel and Hahn, 2018a). In this case, the practical advantage of the PPH approach lies in its independence of (possibly unavailable) external post-processors.

We conducted experiments on different word and text datasets. For words, we collected ten datasets (cf. Tab. 2) covering five languages. These data are structured as illustrated in the top half of Tab. 1. For text-level experiments we selected three corpora (cf. Tab. 3): Affective Text (AFFT; Strapparava and Mihalcea, 2007), EMOBANK (EMOB; Buechel and Hahn, 2017), and the Chinese Valence Arousal Texts (CVAT; Yu et al., 2016). For an illustration of the type and format of text-level data, see the bottom half in Tab. 1. Since these datasets comprise real-valued annotations, we will use Pearson Correlation r for measuring prediction quality. Datasets were partitioned into fixed train-dev-test splits with ratios ranging between 8-1-1 and 3-1-1; smaller datasets received larger dev and test shares.

The selected data govern how to train a given base model with PPH (§3.3). Since, except for Mandarin, there are always two datasets available per domain, we train the models in the supervised setting using the multi-task approach (but use emotion label augmentation for CVAT). By contrast, in the zero-shot setting, we train a model on *one*, yet test on *another* dataset. Thus, we rely on emotion label augmentation here (and have to exclude CVAT for a lack of a second Mandarin dataset). We emphasize that the zero-shot evaluation has very demanding data requirements: This setting not only requires two datasets of the *same* language domain with *different* label formats (which is already rare) but also additional data to fit mapping models for those particular label formats. To the best of our

ID	Vars	Size	Citation
en1	VAD	1,034	Bradley and Lang (1999)
en2	BE5	1,034	Stevenson et al. (2007)
es1	VA	14,031	Stadthagen-González et al. (2017)
es2	BE5	10,491	Stadthagen-González et al. (2018)
de1	VA	2,902	Võ et al. (2009)
de2	BE5	1,958	Briesemeister et al. (2011)
pl1	VA	2,902	Riegel et al. (2015)
pl2	BE5	2,902	Wierzba et al. (2015)
tr1	VA	2,029	Kapucu et al. (2018)
tr2	BE5	2,029	Kapucu et al. (2018)

Table 2: Word datasets. IDs contain the respective ISO 639-1 language code.

ID	Vars	Size	Lg	Domain
AFFT	BE5	1,250	en	news headlines
EMOB	VAD	10,062	en	genre-balanced
CVAT	VA	2,969	zh	mixed online domains

Table 3: Overview of text datasets.

knowledge, EmoBank and AffT form the only suitable dataset pair on the text-level. At the word-level, such pairs are somewhat easier to get due to highly standardized data collection efforts for affective word norm datasets in psychology (see §2). For this reason, we employ a larger number of word- than text-level datasets in our experiments.

Importantly, only the data requirements for *evaluating* our approach in the zero-shot setting are hard to meet. Yet, *inference* is much easier to provide. We would even argue that the reason why our method is so hard to evaluate is precisely what makes it so valuable. Take the Mandarin CVAT dataset, for example. It is annotated with *Valence* and *Arousal*, but there is, to our knowledge, no compatible Mandarin dataset with basic emotions (thus, CVAT is not used in the zero-shot setting). Our method allows to freely switch between output label formats at inference time without language constraints. That is, we can predict BE5 ratings in Chinese even though there is no such training data.

In terms of base models, we used the Feed-Forward Network developed by Buechel and Hahn (2018b) for the word datasets. This model predicts emotion ratings based on pre-trained embedding vectors (taken from Grave et al., 2018). For text datasets, we chose the BERT$_{\text{base}}$ transformer model by Devlin et al. (2019) using the implementation and pre-trained weights by Wolf et al. (2020). Both (word and text) base models use identical hyper-parameter settings with or without PPH extension. For the word model, we copied the settings of the authors, whereas text model hyperparameters were tuned manually for the base model *without* PPH.

We derived training data for the prediction heads (label mapping datasets) by combining the ratings of the word datasets $en1$ and $en2$. We used the label mapping model from Buechel and Hahn (2018a) as auxiliary label encoders. The dimensionality of the emotion space was set to 100. The label mapping models used as external post-processors in the zero-shot setting were also based on Buechel and Hahn (2018a) and were trained on the same data as the label encoders. Further details beneficial for reproducibility are given in Appendix D.

5 Results

Our main experimental results are summarized in Tables 4 to 7. For conciseness, correlation values are averaged over all target variables per dataset. Per-variable results are given in Appendix B.

Looking at the word datasets in the supervised setup (Tab. 4), we find that attaching portable prediction heads (PPH) not only retains, but often enough slightly increases the performance of the FFN base model (p=.008; two-sided Wilcoxon signed-rank test based on per-dataset results). Since we trained only one base model with PPH per language (but two without PPH), our data suggest that the emotion representations learned with PPH can easily hold affective information from different label formats at the same time. Moreover, PPH here offers the practical benefit of reducing the total disk space used by the resulting model checkpoints due to the smaller number of trained base models. Experiments on the text datasets using BERT as base model show results in line with these findings (see Tab. 5).

In the zero-shot setup, models are tested on datasets with label formats different from the training phase (e.g., $en1$ and $en2$). On the word datasets, using PPH shows small improvements in comparison with the base model as is (p=.003; Tab. 6), again suggesting that the learned emotion representations generalize robustly across label formats. Importantly, the base model is only capable of producing this label format *at all* because we equip it with a label mapping post-processor. While this procedure is very accurate (indeed, it constitutes a very strong baseline), it depends on an external component that may or may not be available for

	Base Model (FFN)		**Base Model + PPH**	
Test Data	Train Data	r	Train Data	r
en1 (VAD)	en1 (VAD)	.818	en1+en2	.824
en2 (BE5)	en2 (BE5)	.898	en1+en2	.898
es1 (VA)	es1 (VA)	.820	es1+es2	.833
es2 (BE5)	es2 (BE5)	.789	es1+es2	.820
de1 (VA)	de1 (VA)	.822	de1+de2	.836
de2 (BE5)	de2 (BE5)	.754	de1+de2	.748
pl1 (VA)	pl1 (VA)	.794	pl1+pl2	.835
pl2 (BE5)	pl2 (BE5)	.814	pl1+pl2	.845
tr1 (VA)	tr1 (VA)	.567	tr1+tr2	.575
tr2 (BE5)	tr2 (BE5)	.607	tr1+tr2	.614
Mean		.768		.783
Disk Use		4.33 MB		2.52 MB

Table 4: Word-level results of supervised setting.

	Base Model (BERT)		**Base Model + PPH**	
Test Data	Train Data	r	Train Data	r
EmoB	EmoB	.630	EmoB+AffT	.619
AffT	AffT	.746	EmoB+AffT	.755
CVAT	CVAT	.737	CVAT	.748
Mean		.704		.707
Disk Use		1.25 GB		0.81 GB

Table 5: Text-level results of supervised setting.

	Base Model (FFN)		Base Model + PPH	
Test Data	Train Data	r	Train Data	r
en1 (VAD)	en2 (BE5)	.801	en2	.810
en2 (BE5)	en1 (VAD)	.834	en1	.839
es1 (VA)	es2 (BE5)	.720	es2	.723
es2 (BE5)	es1 (VA)	.777	es1	.792
de1 (VA)	de2 (BE5)	.681	de2	.684
de2 (BE5)	de1 (VA)	.637	de1	.641
pl1 (VA)	pl2 (BE5)	.812	pl2	.812
pl2 (BE5)	pl1 (VA)	.787	pl1	.807
tr1 (VA)	tr2 (BE5)	.538	tr2	.563
tr2 (BE5)	tr1 (VA)	.550	tr1	.554
Mean		.714		.723
Method	ext. post-processor		built-in	

Table 6: Word-level results of zero-shot setting.

	Base Model (BERT)		Base Model + PPH	
Test Data	Train Data	r	Train Data	r
EmoB	AffT	.385	AffT	.407
AffT	EmoB	.584	EmoB	.582
Mean		.485		.495
Method	ext. post-processor		built-in	

Table 7: Text-level results of zero-shot setting.

the desired mapping direction (the source and the target label format). In contrast, the zero-shot capability is *innate* to ("built-in") the PPH approach. While we need only one prediction head per label format, the number of required mapping components for the base model grows on a quadratic scale with the number of considered formats. Again, text-level experiments show consistent results with word-level ones (Tab. 7).

One may object that the reduction of memory footprint shown in Tables 4 and 5 can also be achieved by traditional multi-task learning (i.e., attaching multiple heads to the base model, training it on two datasets, at once). Likewise, as Tables 6 and 7 indicate, the zero-shot capabilities offered by PPH can, in principle, be provided by additional label mapping components. However, PPH offers a much more elegant solution to combine the advantages of multi-task learning and label mapping without calling for additional (language) resources. Most importantly though, PPH is unique in its ability to embed samples from such heterogeneous datasets in a common representation space—a trait that may offer a general solution to studying emotion across languages, cultures, and individually preferred psychological theory.

6 Visualization of the Emotion Space

To gain first insights into the structure of our learned emotion space, we submitted the weight vectors of the emotion variables to principal component analysis (PCA; recall from §3.2 that each row in a head's weights matrix W corresponds to exactly one variable). Further, we derived emotion embeddings for the samples in Tab. 1 using the PPH-extended models evaluated in the last section. Applying the same PCA transformation to the embedding vectors, we co-locate the samples next to the emotion variables. The results (for the first three PCs) are displayed in Fig. 1. As can be seen, the relative positioning of the samples and variables shows high face validity—samples associated with similar feelings appear close to each other as well as to their akin variable. Appendix C provides additional analyses of the learned embedding space (focusing more deeply on the emotional interpretation of the PC axes and the distribution of emotion embeddings across languages) that further support this positive impression.

7 Conclusions & Future Work

We presented a method for learning a common representation space for the emotional loading of heterogeneous language items. While previous work successfully unified *some* sources' heterogeneity, our emotion embeddings are the first to *comprehensively generalize* over arbitrarily disparate language domains, label formats, and distinct neural network architectures. Our technique is based on a collection of *portable prediction heads* that can be attached to existing state-of-the-art models. Consequently, a model learns to *embed* language items in the common learned emotion space and thus to predict a wider range of emotional meaning facets, yet without sacrificing any predictive power as our experiments on 13 datasets (6 languages) indicate.

Since the resulting emotion representations both generalize across various use cases *and* evidently capture a rich set of affective nuances, we consider this work particularly useful for downstream applications. Thus, future work may build on a concept of *emotion similarity* to, e.g., cluster diverse language items by their associated feeling, retrieve words that evoke emotions similar to a query, or compare the affective meaning of phrases and concepts across cultures.

Acknowledgments

We would like to thank the anonymous reviewers for their helpful suggestions and comments, and Tinghui Duan, doctoral student at the JULIE LAB, for assisting us with the Mandarin gold data.

References

Mohamed Abdalla and Graeme Hirst. 2017. Cross-lingual sentiment analysis without (good) translation. In *IJCNLP 2017 — Proceedings of the 8th International Joint Conference on Natural Language Processing*, volume 1: Long Papers, pages 506–515, Taipei, Taiwan, November 27 – December 1, 2017.

Muhammad Abdul-Mageed and Lyle H. Ungar. 2017. EmoNet: Fine-grained emotion detection with gated recurrent neural networks. In *ACL 2017 — Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*, volume 1: Long Papers, pages 718–728, Vancouver, British Columbia, Canada, July 30 – August 4, 2017.

Cecilia Ovesdotter Alm, Dan Roth, and Richard Sproat. 2005. Emotions from text: Machine learning for text-based emotion prediction. In *HLT-EMNLP 2005 — Proceedings of the Human Language Technology Conference & 2005 Conference on Empirical Methods in Natural Language Processing*, pages 579–586, Vancouver, British Columbia, Canada, October 6–8, 2005.

Saima Aman and Stan Szpakowicz. 2007. Identifying expressions of emotion in text. In *TSD 2007 — Proceedings of the 10th International Conference on Text, Speech and Dialogue*, pages 196–205, Pilsen, Czech Republic, September 3-7, 2007.

Silvio Amir, Ramón F. Astudillo, Wang Ling, Bruno Martins, Mário J. Silva, and Isabel Trancoso. 2015. INESC-ID: A regression model for large scale Twitter sentiment lexicon induction. In *SemEval 2015 — Proceedings of the 9th International Workshop on Semantic Evaluation @ NAACL-HLT 2015*, pages 613–618, Denver, Colorado, USA, June 4-5, 2015.

Isabelle Augenstein, Sebastian Ruder, and Anders Søgaard. 2018. Multi-task learning of pairwise sequence classification tasks over disparate label spaces. In *NAACL-HLT 2018 — Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, volume 1: Long Papers, pages 1896–1906, New Orleans, Louisiana, USA, June 1–6, 2018.

A. Balahur, J. M. Hermida, and A. Montoyo. 2012. Building and exploiting EMOTINET, a knowledge base for emotion detection based on the appraisal theory model. *IEEE Transactions on Affective Computing*, 3(1):88–101.

Jeremy Barnes, Roman Klinger, and Sabine Schulte im Walde. 2018. Bilingual sentiment embeddings: Joint projection of sentiment across languages. In *ACL 2018 — Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics*, volume 1: Long Papers, pages 2483–2493, Melbourne, Victoria, Australia, July 15–20, 2018.

Johan Bollen, Huina Mao, and Xiaojun Zeng. 2011. Twitter mood predicts the stock market. *Journal of Computational Science*, 2(1):1–8.

Laura Ana Maria Bostan, Evgeny Kim, and Roman Klinger. 2020. GOODNEWSEVERYONE: A corpus of news headlines annotated with emotions, semantic roles, and reader perception. In *LREC 2020 — Proceedings of the 12th International Conference on Language Resources and Evaluation*, pages 1554–1566, Marseille, France, May 11–16, 2020.

Laura-Ana-Maria Bostan and Roman Klinger. 2018. An analysis of annotated corpora for emotion classification in text. In *COLING 2018 — Proceedings of the 27th International Conference on Computational Linguistics*, pages 2104–2119, Santa Fe, New Mexico, USA, August 20–26, 2018.

Margaret M. Bradley and Peter J. Lang. 1999. Affective norms for English words (ANEW): Stimuli, instruction manual and affective ratings. Technical Report C-1, The Center for Research in Psychophysiology, University of Florida, Gainesville, Florida, USA.

Benny B. Briesemeister, Lars Kuchinke, and Arthur M. Jacobs. 2011. Discrete Emotion Norms for Nouns: Berlin Affective Word List (DENN–BAWL). *Behavior Research Methods*, 43(2):#441.

Sven Buechel and Udo Hahn. 2017. EMOBANK: Studying the impact of annotation perspective and representation format on dimensional emotion analysis. In *EACL 2017 — Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics*, volume 2: Short Papers, pages 578–585, Valencia, Spain, April 3–7, 2017.

Sven Buechel and Udo Hahn. 2018a. Emotion representation mapping for automatic lexicon construction (mostly) performs on human level. In *COLING 2018 — Proceedings of the 27th International Conference on Computational Linguistics*, pages 2892–2904, Santa Fe, New Mexico, USA, August 20–26, 2018.

Sven Buechel and Udo Hahn. 2018b. Word emotion induction for multiple languages as a deep multi-task learning problem. In *NAACL-HLT 2018 — Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, volume 1: Long Papers, pages 1907–1918, New Orleans, Louisiana, USA, June 1–6, 2018.

Rafael A. Calvo and Sunghwan Mac Kim. 2013. Emotions in text: Dimensional and categorical models. *Computational Intelligence*, 29(3):527–543.

Luna De Bruyne, Pepa Atanasova, and Isabelle Augenstein. 2022. Joint emotion label space modeling for affect lexica. *Computer Speech & Language*, 71:#101257.

Luna De Bruyne, Orphée De Clercq, and Véronique Hoste. 2020. An emotional mess! Deciding on a framework for building a Dutch emotion-annotated corpus. In *LREC 2020 — Proceedings of the 12th International Conference on Language Resources and Evaluation*, pages 1643–1651, Marseille, France, May 11–16, 2020.

Jean-Benoit Delbrouck, Noé Tits, Mathilde Brousmiche, and Stéphane Dupont. 2020. A transformer-based joint-encoding for emotion recognition and sentiment analysis. In *Challenge-HML 2020 — Proceedings of the 2nd Grand Challenge and Workshop on Multimodal Language @ ACL 2020*, pages 1–7, Virtual event, July 10, 2020.

Dorottya Demszky, Dana Movshovitz-Attias, Jeongwoo Ko, Alan Cowen, Gaurav Nemade, and Sujith Ravi. 2020. GoEmotions: A dataset of fine-grained emotions. In *ACL 2020 — Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4040–4054, Virtual event, July 5–10, 2020.

Bart Desmet and Véronique Hoste. 2013. Emotion detection in suicide notes. *Expert Systems with Applications*, 40(16):6351–6358.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *NAACL-HLT 2019 — 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, pages 4171–4186, Minneapolis, Minnesota, USA, June 2–7, 2019.

Paul Ekman. 1992. An argument for basic emotions. *Cognition and Emotion*, 6(3-4):169–200.

Manaal Faruqui, Jesse Dodge, Sujay Kumar Jauhar, Chris Dyer, Eduard H. Hovy, and Noah A. Smith. 2015. Retrofitting word vectors to semantic lexicons. In *NAACL-HLT 2015 — Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1606–1615, Denver, Colorado, USA, May 31 – June 5, 2015.

Bjarke Felbo, Alan Mislove, Anders Søgaard, Iyad Rahwan, and Sune Lehmann. 2017. Using millions of emoji occurrences to learn any-domain representations for detecting sentiment, emotion and sarcasm. In *EMNLP 2017 — Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1615–1625, Copenhagen, Denmark, September 7–11, 2017.

Dehong Gao, Furu Wei, Wenjie Li, Xiaohua Liu, and Ming Zhou. 2015. Cross-lingual sentiment lexicon learning with bilingual word graph label propagation. *Computational Linguistics*, 41(1):21–40.

Radhika Gaonkar, Heeyoung Kwon, Mohaddeseh Bastan, Niranjan Balasubramanian, and Nathanael Chambers. 2020. Modeling label semantics for predicting emotional reactions. In *ACL 2020 — Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4687–4692, Virtual event, Juli 5–10, 2020.

Edouard Grave, Piotr Bojanowski, Prakhar Gupta, Armand Joulin, and Tomáš Mikolov. 2018. Learning word vectors for 157 languages. In *LREC 2018 — Proceedings of the 11th International Conference on Language Resources and Evaluation*, pages 3483–3487, Miyazaki, Japan, May 7–12, 2018.

Jing Han, Zixing Zhang, Zhao Ren, and Björn W. Schuller. 2021. EmoBed : Strengthening monomodal emotion recognition via training with crossmodal emotion embeddings. *IEEE Transactions on Affective Computing*, 12:553–564.

Vasileios Hatzivassiloglou and Kathleen R. McKeown. 1997. Predicting the semantic orientation of adjectives. In *ACL-EACL 1997 — Proceedings of the 35th Annual Meeting of the Association for Computational Linguistics & 8th Conference of the European Chapter of the Association for Computational Linguistics*, pages 174–181, Madrid, Spain, July 7–12, 1997.

Jan Hofmann, Enrica Troiano, Kai Sassenberg, and Roman Klinger. 2020. Appraisal theories for emotion classification in text. In *COLING 2020 — Proceedings of the 28th International Conference on Computational Linguistics*, pages 125–138, Virtual Event, December 8–13, 2020.

Aycan Kapucu, Aslı Kılıç, Yıldız Özkılıç, and Bengisu Sarıbaz. 2018. Turkish emotional word norms for arousal, valence, and discrete emotion categories. *Psychological Reports*, pages 1–22. [Available online Dec 4, 2018].

Sopan Khosla, Niyati Chhaya, and Kushal Chawla. 2018. Aff2Vec : Affect–enriched distributional word representations. In *COLING 2018 — Proceedings of the 27th International Conference on Computational Linguistics*, pages 2204–2218, Santa Fe, New Mexico, USA, August 20–26, 2018.

Yoon Kim. 2014. Convolutional neural networks for sentence classification. In *EMNLP 2014 — Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*, pages 1746–1751, Doha, Qatar, October 25–29, 2014.

Sotiris Lamprinidis, Federico Bianchi, Daniel Hardt, and Dirk Hovy. 2021. Universal Joy : A data set and results for classifying emotions across languages. In *WASSA 2021 — Proceedings of the 11th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis @ EACL 2021*, pages 62–75, Virtual Event, April 19, 2021.

Minglei Li, Qin Lu, Yunfei Long, and Lin Gui. 2017. Inferring affective meanings of words from word embedding. *IEEE Transactions on Affective Computing*, 8(4):443–456.

Shoushan Li, Jian Xu, Dong Zhang, and Guodong Zhou. 2016. Two-view label propagation to semi-supervised reader emotion classification. In *COLING 2016 — Proceedings of the 26th International Conference on Computational Linguistics*, volume Technical Papers, pages 2647–2655, Osaka, Japan, December 11-16, 2016.

Saif Mohammad. 2018. Obtaining reliable human ratings of valence, arousal, and dominance for 20,000 English words. In *ACL 2018 — Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics*, volume 1: Long Papers, pages 174–184, Melbourne, Victoria, Australia, July 15–20, 2018.

Saif Mohammad, Svetlana Kiritchenko, and Xiaodan Zhu. 2013. NRC-CANADA: Building the state-of-the-art in sentiment analysis of tweets. In *SemEval 2013 — Proceedings of the 7th International Workshop on Semantic Evaluation @ NAACL-HLT 2013*, pages 321–327, Atlanta, Georgia, USA, June 14-15, 2013.

Saif M. Mohammad and Peter D. Turney. 2013. Crowdsourcing a word-emotion association lexicon. *Computational Intelligence*, 29(3):436–465.

Telmo Pires, Eva Schlinger, and Dan Garrette. 2019. How multilingual is multilingual BERT? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4996–5001, Florence, Italy, July 28 – August 2, 2019.

Robert Plutchik. 2001. The nature of emotions. Human emotions have deep evolutionary roots, a fact that may explain their complexity and provide tools for clinical practice. *American Scientist*, 89(4):344–350.

Soujanya Poria, Devamanyu Hazarika, Navonil Majumder, Gautam Naik, Erik Cambria, and Rada Mihalcea. 2019. MELD : A multimodal multi-party dataset for emotion recognition in conversations. In *ACL 2019 — Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 527–536, Florence, Italy, July 28 – August 2, 2019.

Ashequl Qadir and Ellen Riloff. 2014. Learning emotion indicators from tweets: Hashtags, hashtag patterns, and phrases. In *EMNLP 2014 — Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*, pages 1203–1209, Doha, Qatar, October 25–29, 2014.

Monika Riegel, Małgorzata Wierzba, Marek Wypych, Łukasz Żurawski, Katarzyna Jednoróg, Anna Grabowska, and Artur Marchewka. 2015. Nencki Affective Word List (NAWL): The cultural adaptation of the Berlin Affective Word List–Reloaded (BAWL–R) for Polish. *Behavior Research Methods*, 47(4):1222–1236.

Sascha Rothe, Sebastian Ebert, and Hinrich Schütze. 2016. Ultradense word embeddings by orthogonal transformation. In *NAACL 2016 — Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 767–777, San Diego, California, USA, June 12–17, 2016.

James A. Russell and Albert Mehrabian. 1977. Evidence for a three-factor theory of emotions. *Journal of Research in Personality*, 11(3):273–294.

Klaus R. Scherer. 2000. Psychological models of emotion. In Joan C. Borod, editor, *The Neuropsychology of Emotion*, pages 137–162. Oxford University Press.

João Sedoc, Daniel Preoţiuc-Pietro, and Lyle H. Ungar. 2017. Predicting emotional word ratings using distributional representations and signed clustering. In *EACL 2017 — Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics*, volume 2: Short Papers, pages 564–571, Valencia, Spain, April 3–7, 2017.

Roman Shantala, Gennadiv Kyselov, and Anna Kyselova. 2018. Neural dialogue system with emotion embeddings. In *SAIC 2018 — Proceedings of the 1st IEEE International Conference on System Analysis & Intelligent Computing*, pages 1–4, Kyiv, Ukraine, October 8–12, 2018.

Richard Socher, Alex Perelygin, Jean Y. Wu, Jason Chuang, Christopher D. Manning, Andrew Y. Ng, and Christopher Potts. 2013. Recursive deep models for semantic compositionality over a sentiment treebank. In *EMNLP 2013 — Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1631–1642, Seattle, Washington, USA, October 18-21, 2013.

Hans Stadthagen-González, Pilar Ferré, Miguel A. Pérez-Sánchez, Constance Imbault, and José Antonio Hinojosa. 2018. Norms for 10,491 Spanish words for five discrete emotions: Happiness, disgust, anger, fear, and sadness. *Behavior Research Methods*, 50(5):1943–1952.

Hans Stadthagen-González, Constance Imbault, Miguel A. Pérez-Sánchez, and Marc Brysbært. 2017. Norms of valence and arousal for 14,031 Spanish words. *Behavior Research Methods*, 49(1):111–123.

Jacopo Staiano and Marco Guerini. 2014. DEPECHE MOOD: A lexicon for emotion analysis from crowd annotated news. In *ACL 2014 — Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics*, volume 2: Short Papers, pages 427–433, Baltimore, Maryland, USA, June 22-27, 2014.

Ryan A. Stevenson, Joseph A. Mikels, and Thomas W. James. 2007. Characterization of the Affective

Norms for English Words by discrete emotional categories. *Behavior Research Methods*, 39(4):1020–1024.

Carlo Strapparava and Rada Mihalcea. 2007. SemEval 2007 Task 14: Affective text. In *SemEval 2007 — Proceedings of the 4th International Workshop on Semantic Evaluations @ ACL 2007*, pages 70–74, Prague, Czech Republic, June 23–24, 2007.

Maite Taboada, Julian Brooke, Milan Tofiloski, Kimberly Voll, and Manfred Stede. 2011. Lexicon-based methods for sentiment analysis. *Computational Linguistics*, 37(2):267–307.

Shabnam Tafreshi and Mona Diab. 2018. Emotion detection and classification in a multigenre corpus with joint multi-task deep learning. In *COLING 2018 — Proceedings of the 27th International Conference on Computational Linguistics*, volume 1, Technical Papers, pages 2905–2913, Santa Fe, New Mexico, USA, August 20–26, 2018.

Enrica Troiano, Sebastian Padó, and Roman Klinger. 2019. Crowdsourcing and validating event-focused emotion corpora for German and English. In *ACL 2019 — Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4005–4011, Florence, Italy, July 28 – August 2, 2019.

Peter D. Turney and Michael L. Littman. 2003. Measuring praise and criticism: Inference of semantic orientation from association. *ACM Transactions on Information Systems*, 21(4):315–346.

Melissa L.-H. Võ, Markus Conrad, Lars Kuchinke, Karolina Urton, Markus J. Hofmann, and Arthur M. Jacobs. 2009. The Berlin Affective Word List Reloaded (BAWL–R). *Behavior Research Methods*, 41(2):534–538.

Shuo Wang, Aishan Maoliniyazi, Xinle Wu, and Xiaofeng Meng. 2020. EMO2VEC: Learning emotional embeddings via multi-emotion category. *ACM Transactions on Internet Technology*, 20(2):1–17.

Xiangyu Wang and Chengqing Zong. 2021. Distributed representations of emotion categories in emotion space. In *ACL-IJCNLP 2021 — Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics & 11th International Joint Conference on Natural Language Processing*, volume 1: Long Papers, pages 2364–2375, Virtual Event, August 1-6, 2021.

Małgorzata Wierzba, Monika Riegel, Marek Wypych, Katarzyna Jednoróg, Paweł Turnau, Anna Grabowska, and Artur Marchewka. 2015. Basic emotions in the Nencki Affective Word List (NAWL BE): New method of classifying emotional stimuli. *PLoS ONE*, 10(7):#e0132305.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander M. Rush. 2020. Huggingface's transformers: State-of-the-art natural language processing. *arXiv:1910.03771 [cs]*.

Peng Xu, Andrea Madotto, Chien-Sheng Wu, Ji Ho Park, and Pascale Fung. 2018. EMO2VEC : Learning generalized emotion representation by multi-task training. In *WASSA 2018 — Proceedings of the 9th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis @ EMNLP 2018*, pages 292–298, Brussels, Belgium, October 31, 2018.

Liang-Chih Yu, Lung-Hao Lee, Shuai Hao, Jin Wang, Yunchao He, Jun Hu, K. Robert Lai, and Xuejie Zhang. 2016. Building Chinese affective resources in valence-arousal dimensions. In *NAACL-HLT 2016 — Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 540–545, San Diego, California, USA, June 12–17, 2016.

Liang-Chih Yu, Jin Wang, K. Robert Lai, and Xuejie Zhang. 2017. Refining word embeddings for sentiment analysis. In *EMNLP 2017 — Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 534–539, Copenhagen, Denmark, September 9–11, 2017.

Amir Zadeh, Minghai Chen, Soujanya Poria, Erik Cambria, and Louis-Philippe Morency. 2017. Tensor fusion network for multimodal sentiment analysis. In *EMNLP 2017 — Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1103–1114, Copenhagen, Denmark, September 7–11, 2017.

Peixiang Zhong, Di Wang, and Chunyan Miao. 2019. Knowledge-enriched transformer for emotion detection in textual conversations. In *EMNLP-IJCNLP 2019 — Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing & 9th International Joint Conference on Natural Language Processing*, pages 165–176, Hong Kong, China, November 3-7, 2019.

Algorithm 1 Training the Multi-Way Mapping Model

```
1:  $(Y_{1,1}, Y_{1,2}), (Y_{2,1}, Y_{2,2}), \ldots (Y_{n,1}, Y_{n,2}) \leftarrow$ Mapping datasets used for training
2:  $g_{1,1}, h_{1,1}, g_{1,2}, h_{1,2}, \ldots, g_{n,1}, h_{n,1}, g_{n,2}, h_{n,2} \leftarrow$ randomly initialized label encoders and prediction heads [†]
3:  $n_{\text{steps}} \leftarrow$ total number of training steps
4:  for all $i_{\text{step}}$ in $1, \ldots, n_{\text{steps}}$ do
5:      $(Y_{i,1}, Y_{i,2}) \leftarrow$ randomly sample a mapping dataset
6:      $(y_1, y_2) \leftarrow$ randomly sample a batch s.t. $y_1 \subset Y_{i,1}$ and $y_2 \subset Y_{i,2}$ with identical indices
7:      $(e_1, e_2) \leftarrow (g_{i,1}(y_1), g_{i,2}(y_2))$
8:      $\hat{y}_{1,1} \leftarrow h_{i,1}(e_1)$
9:      $\hat{y}_{1,2} \leftarrow h_{i,2}(e_1)$
10:     $\hat{y}_{2,1} \leftarrow h_{i,1}(e_2)$
11:     $\hat{y}_{2,2} \leftarrow h_{i,2}(e_2)$
12:     $L_{\text{map}} \leftarrow \mathcal{C}(y_1, \hat{y}_{2,1}) + \mathcal{C}(y_2, \hat{y}_{1,2})$ [‡]
13:     $L_{\text{auto}} \leftarrow \mathcal{C}(y_1, \hat{y}_{1,1}) + \mathcal{C}(y_2, \hat{y}_{2,2})$
14:     $L_{\text{sim}} \leftarrow \mathcal{C}(e_1, e_2)$
15:     $L_{\text{total}} \leftarrow L_{\text{map}} + L_{\text{auto}} + L_{\text{sim}}$
16:     compute $\nabla L_{\text{total}}$ and update weights
17: end for
```

[†] If two sets of labels $Y_{a,b}, Y_{c,d}$ follow the same label format, then they use the same label encoders (i.e, $g_{a,b} = g_{c,d}$) and prediction heads ($h_{a,b} = h_{c,d}$).

[‡] $\mathcal{C}$ denotes Mean-Squared-Error Loss.

A Algorithmic Details for Training the Multi-Way Mapping Model

The intuition behind Algorithm 1 is as follows: We simultaneously train multiple label encoders and prediction heads on several mapping datasets using three distinct objective functions. First, of course, we consider the quality of the label mapping (*mapping loss*; line 12). Second, we propose an *autoencoder loss* (line 13) where the model must learn to reconstruct the original input from the emotion embedding. Third, we propose an *embedding similarity loss* (line 14) which enforces the similarity of the hidden representation of both formats for a given instance since they supposedly describe the same emotion. Our training loop starts by first sampling one of the mapping datasets and then a batch from the chosen dataset (lines 5–6). To compute the loss efficiently, we first cache the encoded representations of both label formats (line 7) before applying all relevant prediction heads (lines 8–11).

B Per-Variable Results

For readability reasons, the experimental results reported in §5 only give the average performance score over all emotional target variables for a given dataset. To complement this, the full set of per-variable results are given in Tab. 8.

					Val	Aro	Dom	Joy	Ang	Sad	Fea	Dis	Mean
Level	Test	Setting	Model	Train									
word	en1	supervised	FFN	en1	.920	.704	.829	—	—	—	—	—	.818
			FFN+PPH	en1+en2	.936	.700	.836	—	—	—	—	—	.824
		zeroshot	FFN	en2	.932	.664	.808	—	—	—	—	—	.801
			FFN+PPH	en2	.927	.701	.802	—	—	—	—	—	.810
	en2	supervised	FFN	en2	—	—	—	.929	.900	.898	.890	.873	.898
			FFN+PPH	en1+en2	—	—	—	.936	.890	.895	.901	.869	.898
		zeroshot	FFN	en1	—	—	—	.918	.822	.805	.864	.759	.834
			FFN+PPH	en1	—	—	—	.914	.835	.850	.843	.751	.839
	es1	supervised	FFN	es2	.848	.792	—	—	—	—	—	—	.820
			FFN+PPH	es1+es2	.870	.795	—	—	—	—	—	—	.833
		zeroshot	FFN	es2	.873	.567	—	—	—	—	—	—	.720
			FFN+PPH	es2	.872	.575	—	—	—	—	—	—	.723
	es2	supervised	FFN	es2	—	—	—	.768	.793	.834	.803	.745	.789
			FFN+PPH	es1+es2	—	—	—	.817	.832	.857	.838	.754	.820
		zeroshot	FFN	es1	—	—	—	.808	.795	.823	.775	.685	.777
			FFN+PPH	es2	—	—	—	.811	.805	.839	.810	.695	.792
	de1	supervised	FFN	de1	.867	.776	—	—	—	—	—	—	.822
			FFN+PPH	de1+de2	.892	.780	—	—	—	—	—	—	.836
		zeroshot	FFN	de2	.832	.530	—	—	—	—	—	—	.681
			FFN+PPH	de2	.836	.532	—	—	—	—	—	—	.684
	de2	supervised	FFN	de2	—	—	—	.812	.766	.738	.798	.653	.754
			FFN+PPH	de1+de2	—	—	—	.842	.788	.655	.795	.662	.748
		zeroshot	FFN	de1	—	—	—	.824	.717	.500	.733	.411	.637
			FFN+PPH	de1	—	—	—	.824	.720	.489	.749	.424	.641
	pl1	supervised	FFN	pl1	.852	.735	—	—	—	—	—	—	.794
			FFN+PPH	pl1+pl2	.907	.764	—	—	—	—	—	—	.835
		zeroshot	FFN	pl2	.919	.705	—	—	—	—	—	—	.812
			FFN+PPH	pl2	.918	.707	—	—	—	—	—	—	.812
	pl2	supervised	FFN	pl2	—	—	—	.819	.807	.815	.810	.821	.814
			FFN+PPH	pl1+pl2	—	—	—	.897	.835	.820	.826	.846	.845
		zeroshot	FFN	pl1	—	—	—	.877	.786	.749	.763	.761	.787
			FFN+PPH	pl1	—	—	—	.893	.798	.777	.779	.789	.807
	tr1	supervised	FFN	tr1	.556	.577	—	—	—	—	—	—	.567
			FFN+PPH	tr1+tr2	.571	.579	—	—	—	—	—	—	.575
		zeroshot	FFN	tr2	.561	.514	—	—	—	—	—	—	.538
			FFN+PPH	tr2	.576	.549	—	—	—	—	—	—	.563
	tr2	supervised	FFN	tr1	—	—	—	.607	.603	.628	.627	.568	.607
			FFN+PPH	tr1+tr2	—	—	—	.611	.608	.628	.634	.589	.614
		zeroshot	FFN	tr1	—	—	—	.547	.566	.563	.579	.495	.550
			FFN+PPH	tr1	—	—	—	.583	.533	.575	.588	.488	.554
text	EmoB	supervised	BERT	EmoB	.801	.562	.527	—	—	—	—	—	.630
			BERT+PPH	EmoB+AffT	.798	.550	.509	—	—	—	—	—	.619
		zeroshot	BERT	AffT	.660	.200	.295	—	—	—	—	—	.385
			BERT+PPH	AffT	.686	.238	.297	—	—	—	—	—	.407
	AffT	supervised	BERT	AffT	—	—	—	.730	.634	.818	.836	.712	.746
			BERT+PPH	EmoB+AffT	—	—	—	.776	.659	.823	.841	.675	.755
		zeroshot	BERT	EmoB	—	—	—	.727	.485	.727	.689	.290	.584
			BERT+PPH	EmoB	—	—	—	.724	.491	.736	.704	.255	.582
	CVAT	supervised	BERT	CVAT	.878	.596	—	—	—	—	—	—	.737
			BERT+PPH	CVAT	.878	.617	—	—	—	—	—	—	.748

Table 8: Full experimental results per dataset and target variable in Pearson's r. "Mean" column corresponds to data given in Tabs. 4, 5, 6, and 7.

C Further Analysis of the Emotion Space

Building on the PCA transformation described in §6, we illustrate the position of *all* emotion variables in Fig. 5.

Within the first three principal components, two major groups can be visually discerned: the negative basic emotions of *Sadness*, *Fear*, and *Anger* forming the first group, and *Joy* and the two affective dimensions of *Valence* and *Dominance* forming the second. Intuitively speaking, this stands to reason, as *Valence* and *Dominance* typically show a very high positive correlation in annotation studies. The same holds for *Valence* and *Joy*. Likewise, *Sadness*, *Fear*, and *Anger* usually correlate positively with each other. Yet, between these groups of variables, studies show a negative correlation (cf. studies listed in Tab. 2). Interestingly, these observations indicate that the first principal component of the emotion space may represent a *Polarity* axis.

The remaining two variables, *Disgust* and *Arousal*, position themselves relatively far from the aforementioned groups and opposite of each other in the second principal component. While it is less obvious what this component represents, it is worth noting that both *Arousal* and *Disgust* generalize poorly across label formats. That is, while *Joy*, *Anger*, *Sadness*, and *Fear* are relatively easy to predict from VAD ratings in a label mapping experiment, and, likewise, *Valence* and *Dominance* can well be estimated from BE5 ratings, the variables of *Arousal* and *Disgust* seem to carry information more specific to their respective label format (Buechel and Hahn, 2018a). In the light of these observations, it may not come as a surprise that these variables receive positions that demarcate them clearly from the remaining ones.

The third principal component seems to be linked to the intensity or action potential of a feeling. Here, *Arousal*, *Dominance*, and *Disgust* and, less pronounced, *Fear* and *Anger* score highly, while *Sadness* and *Joy* receive comparatively low values.

Next, we examine whether the learned representations are sufficiently language-agnostic, i.e., that samples with similar emotional load receive similar embeddings independent of their language domain. We derived emotion embeddings for all entries in all of our word datasets (cf. Tab. 2) using the base models with portable prediction heads from the "supervised" setting of our main experiments. Again building on the previously established PCA transformation, we plotted the position of these multilingual samples in 2D (see Fig. 6).

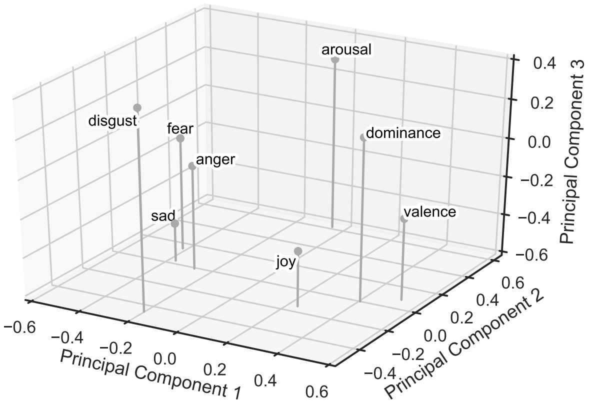


Figure 5: Position of emotion variables in PCA space.

It is noteworthy that entries in our emotion space seem to form clusters according to their affective meaning and not within their dataset or language. As a result, items from different languages overlap so heavily that their respective markers (○, △,□,◇, and ☆) become hard to differentiate.

Furthermore, we selected the highest- and lowest-rated words for *Valence* and *Arousal* and the highest-rated word for *Disgust* in each language. We locate these words in the PCA space and give translations for non-English entries. As can be seen, their position shows high face validity relative to each other and the emotion variables, supporting our claim that the learned emotion space is indeed language-independent.

We emphasize that monolingual, rather than crosslingual, word embeddings were used and that samples from each language were embedded using a separate base model. Hence, the observed alignment of words in PCA space may safely be attributed to our proposed training scheme using portable prediction heads.

D Further Details for Reproducibility

D.1 Description of Computing Infrastructure

All experiments were conducted on a single machine with a Debian 4 operating system. The hardware specifications are as follows:

- 1 GeForce GTX 1080 with 8 GB graphics memory
- 1 Intel i7 CPU with 3.60 GHz
- 64 GB RAM

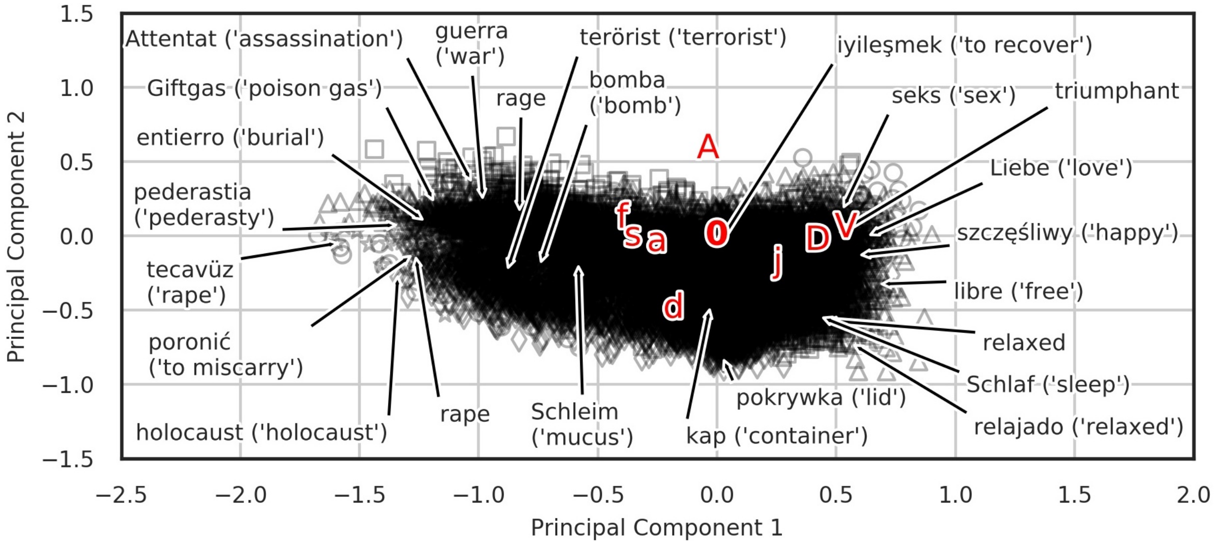


Figure 6: Position of the emotion variables **V***alence*, **A***rousal*, **D***ominance* and **j***oy*, **a***nger*, **s***adness*, **f***ear*, and **d***isgust* in the learned emotion space $\mathbb{R}^d$ (first two PCA dimensions; origin marked with "**0**") together with entries from English (◯), Spanish (△), German (□), Polish (◇), and Turkish (☆) word datasets, as well as highest and lowest *Valence* and *Arousal* word, and highest *Disgust* word per language (arrows).

D.2 Runtime of the Experiments

Training the multi-way mapping model takes about one minute. Training time for the base models varies depending on the dataset. In the following, we report training and inference times for the *largest* dataset per condition, respectively, describing an upper bound of the time requirements.

Regarding the word models, it takes about ten minutes to train a base model without portable prediction heads (PPH) and about 15 minutes to train one with PPH. Since the latter base model replaces two of the former ones in our experiments, the overall training time is reduced by using PPH. Training a word model with emotion label augmentation (the alternative technique for fitting a model with PPH) takes 10 minutes, about as long as training it without PPH. Inference is completed in 1.5 minutes in either case. However, most of that time is needed for loading the language-specific word embeddings. Once this task is done, actually computing the predictions takes only about one second.

Regarding the text models, a baseline model without PPH is trained in about 15 minutes. This number increases with PPH to 30 minutes using the multi-task approach (but again, one PPH model replaces two of the baseline models). In line with the runtime results of the word models, training the text base model with emotion label augmentation takes 15 minutes, about as long as training it without PPH. In either case, inference is completed in well under a minute.

D.3 Number of Parameters in Each Model

The number of parameters per model is given in Tab. 9.

Model (Component)	No. Parameters
Portable Prediction Heads	0.8K
Label Encoders (per format)	18.8K
Label Encoders (in total)	53.4K
Word-Level FFN (per model)	110.6K
BERT$_{\text{base}}$ (per model)	110.0M

Table 9: Number of parameters in each model.

D.4 Validation Performance

Tables 10 – 13 show the dev set results corresponding to the test set results in Tables 4 – 7, respectively. As can be seen, the former are consistent with the latter, yet overall slightly higher, as is usually the case.

D.5 Evaluation Metric

Prediction quality is evaluated using Pearson correlation defined as

$$r_{x,y} := \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2}\,\sqrt{\sum_{i=1}^{n}(y_i - \bar{y})^2}}$$

where $x = x_1, x_2, \ldots, x_n$, $y = y_1, y_2, \ldots, y_n$ are real-valued number sequences and $\bar{x}$, $\bar{y}$ are their

	Base Model (FFN)		**Base Model + PPH**	
Test Data	Train Data	r	Train Data	r
en1 (VAD)	en1 (VAD)	.800	en1+en2	.806
en2 (BE5)	en2 (BE5)	.876	en1+en2	.877
es1 (VA)	es1 (BE5)	.832	es1+es2	.850
es2 (BE5)	es2 (BE5)	.783	es1+es2	.820
de1 (VA)	de1 (BE5)	.825	de1+de2	.835
de2 (BE5)	de2 (BE5)	.780	de1+de2	.792
pl1 (VA)	pl1 (BE5)	.794	pl1+pl2	.841
pl2 (BE5)	pl2 (BE5)	.784	pl1+pl2	.835
tr1 (VA)	tr1 (BE5)	.600	tr1+tr2	.611
tr2 (BE5)	tr2 (BE5)	.613	tr1+tr2	.628
Mean		.769		.790
Disk Use		4.33 MB		2.52 MB

Table 10: Validation word-level results in the supervised setting.

	Base Model (BERT)		**Base Model + PPH**	
Test Data	Train Data	r	Train Data	r
EmoB	EmoB	.610	EmoB+AffT	.600
AffT	AffT	.783	EmoB+AffT	.790
CVAT	CVAT	.748	CVAT	.749
Mean		.714		.713
Disk Use		1.25 GB		0.81 GB

Table 11: Validation text-level results in the supervised setting.

	Base Model (FFN)		**Base Model + PPH**	
Test Data	Train Data	r	Train Data	r
en1 (VAD)	en2 (BE5)	.762	en2	.778
en2 (BE5)	en1 (VAD)	.814	en1	.815
es1 (VA)	es2 (BE5)	.759	es2	.758
es2 (BE5)	es1 (VA)	.767	es1	.779
de1 (VA)	de2 (BE5)	.692	de2	.672
de2 (BE5)	de1 (VA)	.696	de1	.696
pl1 (VA)	pl2 (BE5)	.806	pl2	.829
pl2 (BE5)	pl1 (VA)	.776	pl1	.796
tr1 (VA)	tr2 (BE5)	.556	tr2	.571
tr2 (BE5)	tr1 (VA)	.556	tr1	.565
Mean		.719		.726
Method		ext. post-processor		built-in

Table 12: Validation word-level results in the zero-shot setting.

	Base Model (BERT)		**Base Model + PPH**	
Test Data	Train Data	r	Train Data	r
EmoB	AffT	.353	AffT	.368
AffT	EmoB	.636	EmoB	.664
Mean		.495		.516
Method		ext. post-processor		built-in

Table 13: Validation text-level results in the zero-shot setting.

respective means. We rely on the implementation provided in the SciPy package.[2]

D.6 Model and Hyperparameter Selection

As described in §4, we mostly relied on hyperparameter choices by the authors of our base models. Hence, we performed only a relatively small amount of tuning throughout this work.

For the word base model and the label encoder, no further hyperparameter selection was required. For the text base model (BERT), we verified via a first round of development experiments that default settings yield satisfying prediction quality on our datasets. The learning rate of the AdamW optimizer was set to 10^{-5} based on established recommendations. Besides the number of training epochs (see below), the only dataset-specific hyperparameter choice had to be made for the batch size which we set according to constraints in GPU memory. (The samples in the CVAT dataset are significantly longer than in AffT so that fewer samples of the former can be placed in one batch.) We used the pre-trained weights “bert-base-uncased” and “bert-base-chinese” from Wolf et al. (2020) for the English and Mandarin datasets, respectively. The dimensionality of the emotion space $\mathbb{R}^d$ was initially set to 100 and remained unchanged after verifying that the Multi-Way Mapping Model indeed showed good label mapping performance.

For each (word or text) dataset, we trained the models well beyond convergence, recording their dev set performance after each epoch (number of epochs differs between datasets). We then chose the best-performing checkpoint (according to Pearson correlation) for the final test set evaluation.

Hyperparameter choices were identical between base models with and without PPH. We emphasize that for each base model, hyperparameters were set (by us or by the respective authors) with respect to base model *without* PPH, thus forming a challenging testbed for our approach. We see an extensive hyperparameter search as a fruitful venue for future work.

D.7 Data Access

Below, we list URLs for all datasets used in our experiments.

en1 https://osf.io/2k97q/download (ratings must be extracted from PDF)

en2 https://static-content.springer.com/esm/art%3A10.3758%2FBF03192999/MediaObjects/Stevenson-BRM-2007.zip

es1 https://static-content.springer.com/esm/art%3A10.3758%2Fs13428-015-0700-2/MediaObjects/13428_2015_700_MOESM1_ESM.csv

[2] https://docs.scipy.org/doc/scipy/reference/generated/scipy.stats.pearsonr.html

es2 https://static-content.springer.com/esm/art%3A10.3758%2Fs13428-017-0962-y/MediaObjects/13428_2017_962_MOESM1_ESM.csv

de1 https://www.ewi-psy.fu-berlin.de/einrichtungen/arbeitsbereiche/allgpsy/Download/BAWL/index.html

de2 https://static-content.springer.com/esm/art%3A10.3758%2Fs13428-011-0059-y/MediaObjects/13428_2011_59_MOESM1_ESM.xls

pl1 https://static-content.springer.com/esm/art%3A10.3758%2Fs13428-014-0552-1/MediaObjects/13428_2014_552_MOESM1_ESM.xlsx

pl2 https://doi.org/10.1371/journal.pone.0132305.s004

tr1 https://osf.io/rxtdm

tr2 https://osf.io/rxtdm

AFFT http://web.eecs.umich.edu/~mihalcea/affectivetext/

EMOB https://github.com/JULIELab/EmoBank

CVAT http://nlp.innobic.yzu.edu.tw/resources/cvat.html

D.8 Details of Train-Dev-Test Splits

EMOB comes with a stratified split with ratios of about 8-1-1 (exactly 8062 train, 1000 dev, 1000 test samples). Since the samples of AFFT are mostly also included in EMOB, we decided to use the data split of the latter for the former, too. Samples of AFFT that were not included in EMOB (about 5% of the data) were removed before the experiments. CVAT features a 5-fold data split but without assigning the resulting parts to train, dev, or test utilization. We used the first three for training, the fourth for development/validation, and the fifth for testing.

The word datasets in Tab. 2 do not come with a fixed data split. Instead, we defined splits ourselves with ratios ranging between 3-1-1 to 8-1-1, depending on the number of samples. Instances were randomly assigned to train, dev, and test split using fixed random seeds. The resulting partitions were stored as JSON files and placed under version control.