DAPPER: Learning Domain-Adapted Persona Representation Using Pretrained BERT and External Memory

Prashanth Vijayaraghavan
MIT Media lab
pralav@mit.edu

Eric Chu
MIT Media lab
echu@mit.edu

Deb Roy
MIT Media lab
dkroy@media.mit.edu

Abstract

Research in building intelligent agents have emphasized the need for understanding characteristic behavior of people. In order to reflect human-like behavior, agents require the capability to comprehend the context, infer individualized persona patterns and incrementally learn from experience. In this paper, we present a model called DAPPER that can learn to embed persona from natural language and alleviate task or domain-specific data sparsity issues related to personas. To this end, we implement a text encoding strategy that leverages a pretrained language model and an external memory to produce domain-adapted persona representations. Further, we evaluate the transferability of these embeddings by simulating low-resource scenarios. Our comparative study demonstrates the capability of our method over other approaches towards learning rich transferable persona embeddings. Empirical evidence suggests that the learnt persona embeddings can be effective in downstream tasks like hate speech detection.

1 Introduction

With increasing human-machine hybrid technologies, the real-world interactions with AI systems are often stilted. This shortcoming can be attributed to the lack of shared common knowledge about how people will act, communicate and react under different circumstances. Several studies in the field of psychology (Goldberg, 1990; Barrick and Mount, 1993, 1991) have established the role of personas in governing how people process information, attend to and interpret life-experiences, and respond to social situations. Specifically, the relationship between personality and natural language have been widely studied (Digman and Takemoto-Chock, 1981; Pennebaker et al., 2003). For example, a narcissistic person might make frequent use of first-person expressions (I, me, myself, for me, etc.). Therefore, endowing machines with the persona information can lead to the development of psychologically plausible intelligent systems. Though computational models of personality have generally followed prior psychological models or theories (Hjelle and Ziegler, 1992; Costa and PAUL, 1996), multiple definitions of personas have been in use depending on the nature of the domain or task at hand. There has been considerable amount of interest in the past that used NLP tools to conduct persona analysis of fictional characters in literary texts (Flekova and Gurevych, 2015; Mairesse et al., 2007). Motivated by such works, we focus on deriving persona representations that explain human social behavior categorized according to their influences on language, conversations and actions in different social contexts.

In this work, we define persona as the sum total of mental, emotional, and social characteristics of an individual (Soloff, 1985). This broad definition, while basing on theoretical foundations, allows us to learn persona embeddings from annotated text that span across multiple domains and social contexts. Often these persona-annotated domain data are either too small or not representative of all the domain aspects of persona. Therefore, we address these challenges by formulating our representation learning problem through the lens of domain adaptation. We propose a model called DAPPER[1] to learn a domain-adapted persona embedding that promotes positive knowledge transfer across multiple text domains: movies dialogue, forum discussion posts and personal life stories or essays. Towards this goal, we use a pretrained BERT model to extract rich semantic features from text and finetune them by introducing Adaptive Knowledge Transformer that serve as adaptive layers on top of the representations obtained from BERT model.

[1] Short for **D**omain **A**dapted **P**retraining-based **PE**rsona **R**epresentation

Proceedings of the 1st Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 10th International Joint Conference on Natural Language Processing, pages 643–652
December 4 - 7, 2020.

These adaptive layers enrich the representations with domain-related persona knowledge. We explore variants of Transformer encoder layer as our adaptive layers. In our experiments, we compare our Transformer-based DAPPER model with RNN-based techniques on data from three different text domains. Finally, we showcase the advantages of using our representations in a downstream hate speech detection task. Thus, our contributions are as follows:

- We propose a model called DAPPER that integrates pretrained language model with adaptive knowledge Transformer layers to learn better domain-adapted representation of personas.
- We evaluate our model on texts from multiple text domains: Movies dialogue (Chu et al., 2018), forum discussion posts and personal essays or life stories (Pennebaker and King, 1999). Our DAPPER model outperforms the baseline models significantly across these domains.
- We determine how our model performs in domains with limited labeled data by simulating such scenarios within our existing datasets. We show that our domain-knowledge enriched persona representations are capable of adapting to such domains. Further, they show promise in an unrelated downstream hate speech detection task.

2 Related Work

Considering that personality compels a tendency on a lot of aspects of human behavior, there have been several studies intended to model personality traits from text. An earlier work by (Pennebaker and King, 1999) compiled stream-of-consciousness essay dataset for an automated personality detection task. Since the Five Factor Model is widely accepted, several attempts have been made to detect personality from these essays including LIWC features or deep learning techniques (Majumder et al., 2017; Mairesse et al., 2007). (Chaudhary et al., 2013) compared different machine learning models to predict Myers-Brigg Type Indicator. Another line of work (Liu et al., 2016) related to personas focused on developing a language independent and compositional model for personality trait recognition for short tweets. Additionally, there have been other efforts that model personas of movie characters and incorporate speaker persona in dialogue models based on speaking style characterized by natural language sentences (Bamman et al., 2013). We observe that most of these works use different theories and definitions for modeling personas – ranging from widely accepted psychological tests to simple emotion states of people as means of ascertaining personality (Shuster et al., 2018). However, there is very limited work (Li et al., 2016; Chu et al., 2018) focusing on persona embeddings that can be adapted to different domains. In this work, our goal is to produce general purpose persona embeddings computed using texts from various domains .

Datasets	Label Type	Size	# Categories
Personal Essays	Big-Five	2,400	5
Forum Posts	MBTI	52,648	16
Movies Dialogue	Tropes	17,342	72

Table 1: Details of the datasets from different domains

3 Datasets

Towards learning a domain-adapted persona embedding, we aggregate different forms of text data: (a) personal stories/essays, (b) dialogues and (c) discussion forum posts. Each of these datasets have distinct persona categories. Table 1 shows the details of the dataset. We elaborate them in the following sections.

3.1 Personal Essays Corpus

Personal stories or reflections explain important parts of one's personality including their goals and values (McAdams and Manczak, 2015). For our purpose, we make use of personal essays originally from Pennebaker et al. (Pennebaker and King, 1999). This corpus consists of 2400 essays collected between 1997 and 2004. Students who produced these texts were assessed based on Big Five[2] Questionnaires. To obtain labels from the self-assessments, z-scores were computed from them by (Mairesse et al., 2007) and the resulting scores were discretized to categories by (Celli et al., 2013).

3.2 Forum Posts Corpus

One of the most commonly administered psychological tests is Myers-Briggs Type Indicator

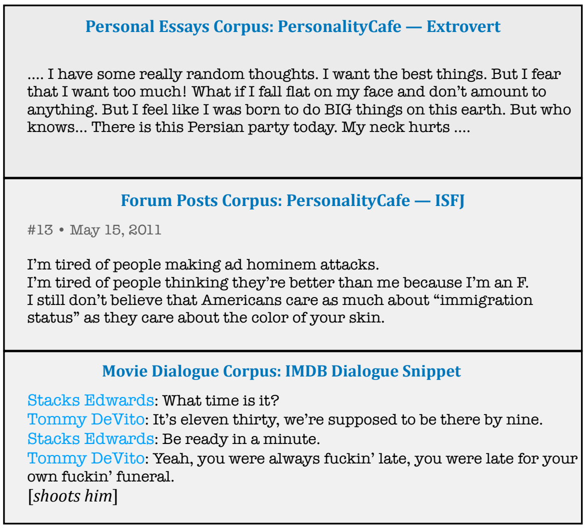
[2] https://en.wikipedia.org/wiki/Big_Five_personality_traits

(MBTI[3]). Based on Jung's theory of psychological types, 16 personality types were recognized as useful reference points to understand one's personality. In this work, we collect a text corpus from a discussion forum called PersonalityCafe[4], that has dedicated communities for each of the 16 MBTI personality types. The members of these communities generally self-identify with the corresponding personality type and post various forms of text including those written in a stream-of-consciousness style. To obtain these posts, we crawled specific sections of the forum related to each personality type. Further, we filter the posts that are too short (i.e. less than 75 characters in length) and replace explicit mentions of their personality type in the text with markers. Though the prevalence of MBTI personality types in general population is highly disproportional, the forum posts might not always reflect that distribution. Therefore, we create a more or less balanced dataset to avoid any skewed representation of personality types. In total, our Forum Posts dataset contains 52,648 posts. The dataset will be made publicly available.

3.3 Movies Dialogue Corpus

In a contrast to prior datasets which has well-defined persona categories based on personality tests/theories, we use a dataset that views character tropes as a proxy for persona labels. In the context of fiction, character trope refers to the aspects of a story that conveys information about a character including its role in the plot, personality, motivations and perceived behavior. Thus, we utilize the IMDB dialogue snippet dataset[5] (Chu et al., 2018) containing utterances of characters in movies obtained from CMU Movie Summary datasets (Bamman et al., 2013). Each of the 433 characters in the dataset is associated with one among 72 different trope labels. Additionally, we collect more persona-related domain-specific knowledge from TVTropes. TVTropes is a wiki that collects document descriptions about plot conventions and devices. It also contains useful notes describing MBTI [6] and Big Five [7] personality traits with references to character tropes that closely relate to each of those categories.

Figure 1 displays samples from the datasets used in this work. Using these datasets and persona category knowledge, we focus on developing domain-adapted persona embeddings.



Personal Essays Corpus: PersonalityCafe — Extrovert

.... I have some really random thoughts. I want the best things. But I fear that I want too much! What if I fall flat on my face and don't amount to anything. But I feel like I was born to do BIG things on this earth. But who knows... There is this Persian party today. My neck hurts ....

Forum Posts Corpus: PersonalityCafe — ISFJ

#13 • May 15, 2011

I'm tired of people making ad hominem attacks.
I'm tired of people thinking they're better than me because I'm an F.
I still don't believe that Americans care as much about "immigration status" as they care about the color of your skin.

Movie Dialogue Corpus: IMDB Dialogue Snippet

Stacks Edwards: What time is it?
Tommy DeVito: It's eleven thirty, we're supposed to be there by nine.
Stacks Edwards: Be ready in a minute.
Tommy DeVito: Yeah, you were always fuckin' late, you were late for your own fuckin' funeral.
[*shoots him*]

Figure 1: Samples from different datasets used for learning domain-adapted persona embeddings.

4 Problem Setup

The overall goal of our model is to learn persona embeddings using documents from different domains $\mathcal{D}$: dialogue utterances, forum posts and personal essays. This persona representation learning problem is formulated as a supervised classification problem. Let us denote the i^{th} input document as $\mathcal{I}^{(i)} = [\mathcal{I}_1^{(i)}, \mathcal{I}_2^{(i)}, ..., \mathcal{I}_{|I|}^{(i)}]$. Here, a document refers to a list of sentences from the personal essays or forum Posts corpus and dialogue snippets in case of movies dialogue corpus (explained in Section 3). Each input $\mathcal{I}^{(i)}$ in our data is associated with their domain-specific persona label $p_k^{(i)}$ where $k \in \{1, 2, ..|\mathcal{D}|\}$, $p_k^{(i)} \in \mathcal{Y}_k$ and $\mathcal{Y}_k$ is the personal categories related to the k^{th}-domain.

5 Proposed Model

In this work, we explore the idea of leveraging a pretrained BERT model towards our goal of learning domain-adaptive persona embeddings. Instead of relying only on the domain-specific training data, we allow additional domain knowledge to be injected into our model using an external memory. Our model architecture is illustrated in Figure 2.

5.1 Input Processing

The input to our DAPPER model can take different forms depending on the domain under considera-

[3] https://en.wikipedia.org/wiki/Myers–Briggs_Type_Indicator

[4] https://www.personalitycafe.com

[5] https://pralav.github.io/emnlp_personas/

[6] https://tvtropes.org/pmwiki/pmwiki.php/UsefulNotes/MyersBriggs

[7] https://tvtropes.org/pmwiki/pmwiki.php/UsefulNotes/BigFivePersonalityTraits

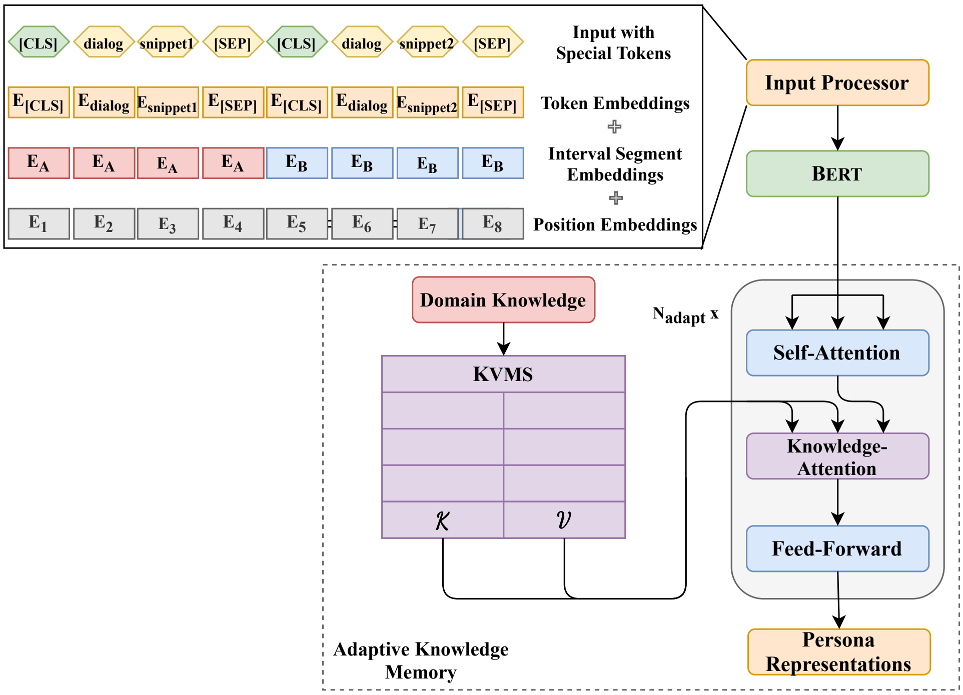


Figure 2: Illustration of our DAPPER model.

tion: (a) long essays or forum posts containing several sentences representing personal details, goals and values, and (b) dialogue snippets having character's own lines and additional contextual information such as narrator or interacting characters' lines. The varying nature of the data from these domains can pose a challenge to our modeling objective. In order to represent data from these domains, we define the following procedure:

- For Personal Essays and Forum Posts Corpus, we insert a special $[CLS]$ token at the beginning of each sentence s_j in an essay or post with an intention that each $[CLS]$ token will accumulate the features of the tokens following it.
- For Dialogue Corpus, we introduce a $[CLS]$ before every dialogue snippet d_j while the character's own lines and additional context are separated by a $[SEP]$ token.
- Next, we apply interval segment embeddings, E_A or E_B, to distinguish sentences or dialogue snippets in our data. This is done by alternating assignments between two consecutive sentences or dialogue snippets. For example, we would assign $[E_A, E_B, E_A, E_B]$ to a list of dialogue snippets denoted as $[d_1, d_2, d_3, d_4]$.

We also incorporate position embeddings into our input data processing step. Thus, we obtain a uniform way of representing our inputs texts from different domains. This allows us to hierarchically learn abstract persona representations.

5.2 Encoder

Our input document $\mathcal{I}^{(i)}$ is passed to our input processing module $f_{\mathcal{I}}(\cdot)$. The output of this module is a document representation augmented with special tokens and processed with interval segment and position embeddings. The processed input is passed to the pretrained BERT model. Formally, this is computed as:

$$H^{(i)} = \text{BERT}(f_{\mathcal{I}}(\mathcal{I}^{(i)})) \qquad (1)$$

where $f_{\mathcal{I}}$ is the input processing function, $H^{(i)}$ contains contextualized embeddings related to each token in the processed input document. We obtain j^{th} sentence or snippet embeddings by extracting the corresponding vector of j^{th} $[CLS]$ token from the topmost BERT layer. We denote this as $R^{(i)} \in \mathcal{R}^{|I| \times d_h}$, d_h is the set to the hidden dimensions of the BERT model.

5.3 Adaptive Knowledge Transformer

Inspired by a prior work by (Miller et al., 2016; Zhang et al., 2017), we integrate an external memory module with the Transformer architecture and refer it as Adaptive Knowledge Transformer (AKT).

This component aids to create persistent latent embeddings related to persona categories and further accumulate more knowledge as we process data from new domains. We conceptualize this component to be composed of: (a) a Key-Value Memory Store (KVMS) that specifically facilitates adaptivity to new domains or data (b) Transformer-based adaptive layers that attends over the contents of the memory to enrich the representation with persona-related domain knowledge. By feeding the computed $R^{(i)}$ into our AKT, we obtain domain-knowledge enriched persona embeddings. This is given as:

$$\mathcal{P}^{(i)} = \text{AKT}(R^{(i)}) \tag{2}$$

5.3.1 KVMS: Key-Value Memory Store

Our KVMS module consists of a mutable key matrix ($\mathcal{K} \in \mathcal{R}^{N_M \times d_K}$) that accumulates persona-related knowledge across multiple domains and a non-updatable value matrix ($\mathcal{V} \in \mathcal{R}^{N_M \times d_V}$) containing a learnable persona category embedding. The key matrix, $\mathcal{K}$, is initialized with representations of text descriptions of character tropes, MBTI types and Big-Five traits collected from TVTropes wiki (explained in 3.3) while the value matrix, $\mathcal{V}$, is set to their corresponding learnable persona category embeddings. We feed the text descriptions through the input processing model and compute the sum of the sentence embeddings obtained from the topmost layer of BERT.

5.3.2 Knowledge-Attention

Conventionally, a Transformer encoder layer consists of two sub-layers: (a) a multi-headed self-attention network and (b) a point-wise fully-connected network. Each sub-layer has a residual connection followed by layer normalization. For the sake of brevity, we avoid the residual connections and layer normalization functions in our model illustration (Figure 2) and explanation.

Our Transformer-based adaptive layers contain an additional sub-layer to integrate the persona-relevant domain knowledge into the contextual representation obtained from the encoder. We refer to this sub-layer as Knowledge-Attention. This is fine-tuned using domain-specific categories based on a supervised classification objective. The steps involved in Transformer adaptive layers are given as follows:

$$Q^{(n)} = \text{MHA}(C^{(n-1)}, C^{(n-1)}, C^{(n-1)}) \tag{3}$$

$$A^{(n)} = \text{MHA}(Q^{(n)}, \mathcal{K}, \mathcal{V}) \tag{4}$$

$$C^{(n)} = \text{FFN}(A^{(n)}) \tag{5}$$

$$\mathcal{P}^{(i)} = C^{N_{adapt}} \tag{6}$$

where MHA is a multi-head attention function as explained in (Vaswani et al., 2017), $n = \{1, 2, .., N_{adapt}\}$, $C^{(0)} = R^{(i)}$, $C^{(n-1)}$ is the output from the previous Transformer layer, $A^{(n)}$ is the output from the knowledge-attention sub-layer. Our knowledge-attention mechanism identifies the most correlated and relevant knowledge from the KVMS component with respect to the input document embeddings. The resulting domain knowledge-enhanced representations are fed to the point-wise feed-forward sub-layer (FFN). We stack such adaptive layers on top of each other and the output from N_{adapt}^{th} layer is our final domain-adapted persona representation, $\mathcal{P}^{(i)}$.

5.3.3 Memory Update

Intuitively, accumulation of persona-related knowledge extracted from the training documents into our memory store can enhance the quality of the learned persona embeddings. Therefore, we perform a memory update operation on selective rows in the key matrix $\mathcal{K}$ based on the persona-related features derived from the input document and its corresponding ground truth persona labels. The update step is defined as follows:

$$\lambda = \sigma(W_k \mathcal{K}[g_j] + W_r \phi(\mathcal{P}^{(i)}) \tag{7}$$

$$\mathcal{K}[g_j] = \lambda \odot \mathcal{K}[g_j] + (1 - \lambda) \odot \phi(\mathcal{P}^{(i)}) \tag{8}$$

where g_j refers to the indices of the rows in KVMS containing knowledge about ground truth persona label $p_k^{(i)}$, ϕ is aggregation function that compresses the information from $\mathcal{P}^{(i)}$ into a single vector. We find from preliminary experiments that the mean $[CLS]$ token embedding serves as an effective alternative to computing an average embedding related to the tokens in the input document.

5.4 Training Objective

Our model learns persona embeddings using a supervised classification objective. We feed the output of the aggregation function ϕ to a domain-specific softmax layer to get q, where $q =$

$softmax(f_q(\phi(\mathcal{P}^{(i)})))$. Note that the categories vary across each domain.

$$\mathcal{L}_{CE} = \sum_{j=1}^{N_k} -p_j log(q_j) \tag{9}$$

$$\mathcal{L}_{attn} = \frac{1}{M} \sum_{j=1}^{M} -log(r_j[g_j]) \tag{10}$$

$$\mathcal{L} = \alpha_1 \mathcal{L}_{CE} + \alpha_2 \mathcal{L}_{attn} \tag{11}$$

where $\mathcal{L}_{CE}$ is the cross-entropy loss, α_1, α_2 are learnable parameters, $p_j \in \{0, 1\}$ denotes the ground-truth label that reflects if the input document belongs to j^{th} persona category, $\mathcal{L}_{attn}$ is the attention loss that promotes focus on rows with ground truth persona, $r_j[g_j]$ is the attention score for the row in $\mathcal{K}$ reflecting p_k^i's knowledge.

6 Experiments

In this section, we describe the various evaluations settings: datasets, baselines, our model variants, modes and metrics. Our experiments are designed to study the following research questions:

RQ1: How well does our DAPPER model perform in comparison to baselines and its variants on domain-specific persona classification task?

RQ2: Is our model capable of adapting to new domains with limited labeled data?

RQ3: How good are the learned persona embeddings? Do they exhibit transfer capability to a downstream task?

6.1 Dataset Preparation

We evaluate our models using persona-related datasets from different domains: movies dialogue, forum posts and personal essays as explained in Section 3. Using a 70-10-20 split, we divide our persona dataset associated with each domain into training, validation and test sets.

6.2 Baselines & Model Variants (RQ1)

Since we collect persona datasets from different domains, we also compare our model's performance to domain-specific baseline methods. All these methods are enlisted as follows:

- AFF2VEC (Khosla et al., 2018) is a method for enriched word embeddings that are representative of affective interpretations of words.

- CNN (Kim, 2014) is a single-layer CNN where the input document is passed in entirety without any additional knowledge. For Personal Essays corpus, we report the best results from (Majumder et al., 2017) as they use additional features to improve persona classification task.

- AMN (Chu et al., 2018) learns persona embeddings from movies dialogue using a multi-level attention mechanism augmented with prior knowledge about persona categories. Note that this model is one of the closest relevant work to our model. For movies dialogue corpus, we report scores only for the best performing configuration, i.e., $ndialog = 32$. For the remaining datasets, we treat each sentence from the text as a character utterance and train the model accordingly.

- TTS is a non-pretrained Transformer baseline trained with the same settings as (Vaswani et al., 2017). We do not feed additional domain knowledge to this model. It is randomly initialized and trained for our task from the scratch.

- BERT FT (Devlin et al., 2018) is a fine-tuned (FT) version of BERT$_{base}$ model. We do not feed additional domain knowledge to this model. We refrain from training BERT$_{large}$ due to memory constraints.

- BERT + GRU FT (Devlin et al., 2018; Chung et al., 2014) is a similar to our DAPPER model, but applies GRU-based adaptive for persona classification task. For this setting, we experiment with and without additional knowledge using a suffix "+K". In "+K" setting, we use GRU as the controller and apply an approach similar to AMN to enrich the learnt embeddings with domain knowledge. Without the suffix, GRU is used for fine-tuning only.

- DAPPER is our complete model by default. We also experiment with its variants using suffix "-K" indicating no knowledge attention.

The various BERT-based models can be considered as variants of our DAPPER model. While we report F_1-scores for movies dialogue and forum discussion post datasets, we report accuracy scores for personal essays corpus in order to remain consistent with prior work evaluation metrics (Majumder et al., 2017).

6.3 Model Modes (RQ2)

We attribute the domain adaptive capability of our DAPPER model to three main aspects: pretrained language model, domain knowledge enrichment and joint training across multiple datasets. However, this ability can be demonstrated only when we apply it to domains with limited labeled data. Therefore, we run our model in "ADAPT" mode which simulates low-data regimes to analyze the importance of some of the above mentioned aspects. In ADAPT mode, we restrain the amount of training data for only one of the domains while retaining the complete set for the remaining domains. Further, we vary the percentage of training examples from one domain to understand how early our models adapt to that domain (with decent performance). We refer to the default model mode for experiments in Section 6.2 as "FULL". For this experiment, we plot the average prediction performance (F_1) for varying percentages of domain-specific training set.

6.4 Other Experimental Settings

For baselines, we initialize our word embedding layers using GloVe (Pennington et al., 2014) embeddings. We use the publicly released pre-trained model parameters for BERT variants. We perform a grid-search and optimize the hyperparameters using the validation set. In our experiments, $N_{adapt} = 3$, resulted in best outcomes. We use Adam (Kingma and Ba, 2014) as our optimizer. In FULL mode, the model achieves the best performance after training for 50 epochs with a learning rate of $\alpha = 0.00001$. For ADAPT mode, we perform a fixed number of epochs to train each variant. We use PyTorch to implement our model and train it on on 4 GPUs. In order to alleviate the problem of unbalanced datasets, we utilize class weights in categorical cross-entropy loss for each domain based on the training and validation sets.

6.5 Results

6.5.1 DAPPER Performance (RQ1)

Table 2 presents the results of our evaluation under complete training data settings (FULL). Our DAPPER model achieves an absolute improvement of 14.53% over previously reported model baseline (AMN) in the dialogues domain. While several models have shown only marginal improvement in prediction performance on Personal Essays corpus, our model shows promise by recording an improvement of 8.67% in comparison to the previously reported CNN baseline. Overall, our DAPPER model outperforms the baselines across all the three datasets significantly.

Effect of Architecture Choices (RQ1): Pretrained BERT-based models have consistently outperformed all the previous baselines including the non-pretrained TTS model. Moreover, the Transformer-based adaptive layers, with an average improvement of 6.1% (with knowledge-attention) and 4.2% (without knowledge-attention), are much more powerful than RNN-based adaptive layers. Further, we observe that BERT + GRU FT records only marginal gains over BERT when there is no knowledge-attention.

Effect of Knowledge-Attention (RQ1): From our results in Table 2, we analyze the importance of the knowledge-attention to the overall performance gain. We compute percentage performance gain between similar models with and without knowledge-attention sub-layer(eg. DAPPER, DAPPER $- K$). We find that the performance boost provided by the knowledge-attention module is noteworthy. We posit that the higher percentage gain (7.38%) for Forum Posts dataset is due to the additional domain knowledge (MBTI-related) ingested into our KVMS (explained in Section 3). Inspecting further within individual domain, the percentage increase in prediction performance almost doubles[8] for Transformer-based adaptive layers (as in DAPPER) in comparison with RNN-based adaptive layers (BERT + GRU FT + K). The reason for this phenomenon can be ascribed to the multi-hop knowledge enrichment facilitated by N_{adapt} encoder layers commonly observed in Memory networks literature (Miller et al., 2016).

6.5.2 ADAPT Mode Performance (RQ2)

Figure 3a and 3b show the mean prediction performance on movies dialogue and forum posts datasets respectively. We measure the domain adaptive capability of models based on the distance from its lifetime best performance. By varying the percentage of training data, we notice that our DAPPER model stabilizes early and outperforms the other variants with limited amount of training data. Notably, AMN model performs better than TTS model under low-data regimes. The improved performance of AMN is due to the domain knowledge enrichment via an external memory module.

[8] % increase-RNN vs Transformer-based adaptive layers: Movies dialogue corpus: 1.6% vs 3.24% (dialogue), 5.86% vs 8.9% (posts), 1.6% to 2.4% (essays)

Models	Domain-related Persona Datasets		
	Movies Dialogues (F_1)	Forum Posts (F_1)	Personal Essays ($Acc.$)
AFF2VEC	_	_	0.579*
CNN	0.628	0.391	0.588*
AMN	0.750*	0.453	0.591
TTS	0.776	0.496	0.593
BERT FT	0.804	0.539	0.607
BERT + GRU FT + K	0.820	0.579	0.616
BERT + GRU FT	0.807	0.547	0.608
DAPPER	**0.859**	**0.636**	**0.639**
DAPPER − K	0.832	0.584	0.624

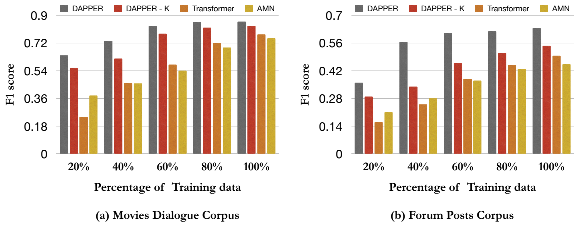
(a)

Models	F_1
Text Only	
BCA	**0.744***
CNN-CHAR	0.735*
1-Extra Feature	
BCA + $\mathcal{P}$	0.776
BCA + SC	**0.784***
All Features	
BCA + SC + $\mathcal{P}^{(>att)}$	0.812
BCA + SC + $\mathcal{P}^{(<att)}$	**0.824**

(b)

Table 2: Evaluation results of different models on: (a) three different Persona-related domain datasets in FULL mode, and (b) a downstream application – Hate Speech detection. Results with * are taken from prior studies using the model on that dataset.

This feature is absent in TTS. Furthermore, we note that DAPPER − K model is able to maintain a good performance even under low-data settings. We intuit that pretraining involved in DAPPER − K model is one of the reasons behind this behavior. Therefore, we find that our DAPPER model is able to learn general purpose persona embeddings that can adapt to low-data settings. Moreover, the combination of pretraining and adaptive knowledge transformer facilitates domain adaptation effectively.



(a) Movies Dialogue Corpus

(b) Forum Posts Corpus

Figure 3: Evaluation of DAPPER model in ADAPT mode. We report the mean prediction performance (F_1) on Movies Dialogue and Forum Posts dataset.

6.6 Cluster Analysis (RQ3)

In order to demonstrate the capabilities of our persona embedding, we first perform a simple cluster analysis. Following prior studies (Bamman et al., 2013; Chu et al., 2018), we measure the ability to recover persona-based clusters using our embeddings through the purity scores as in (Bamman et al., 2013). We compute the overlap between clusters as: $Purity = \frac{1}{N}\sum_n max_j|y_n \cap c_j|$, where y_n is the n-th ground truth cluster, N is total number of characters, c_j is the j^{th} predicted cluster. By applying simple agglomerative clustering on our persona embeddings (k clusters), we report these purity scores for movies dialogue corpus. Specifically, we compare the results with AMN. Results in Table 3 indicate that our DAPPER model sharpens the persona embeddings so as to form much better clusters.

k	AMN	DP	DAPPER
25	48.4	39.63	**68.6**
50	48.1	31.0	**65.3**
100	45.2	24.4	**63.4**

Table 3: Cluster purity scores. DP is the Dirichlet Persona as reported in (Bamman et al., 2013)

7 Application: Hate Speech Detection

With concerns about hate crimes, harassment, and intimidation on the rise, the role of online hate in exacerbating such violence cannot be discounted. Hence, there is an growing need to identify and counter the problem of hateful content on social media. While most prior modeling approaches have attempted to capture the semantics of hate from text, a few of them (Vijayaraghavan et al.) have used multi-modal information to detect hateful content. Few attempts have been made to study the personality of targets and instigators of hate. Since our DAPPER model learns persona embeddings from different forms of text such as dialogues, posts or personal essays, we deem it fit to explore how well our persona embeddings transfer knowledge to a hate speech detection task involving texts from a different domain (in our case, Twitter).

There are several publicly available labeled hate speech datasets (de Gibert et al., 2018; Waseem, 2016) but very few include author metadata or tweets. In this work, we take advantage of the models and datasets introduced by (Vijayaraghavan et al.) (hereafter referred as MM-HATE). This weakly-labeled dataset contains author information and additional metadata about potential hate groups. Instead of training a powerful hate speech system from the scratch, we augment their base architecture with our persona embeddings and evaluate the prediction performance on the task at hand. We compute persona representations ($\mathcal{P}$) for an author based on their past tweets. We train MM-HATE's best performing model, BIGRU+CHAR+ATTN (BCA), under the following settings: (a) BCA + $\mathcal{P}$, which combines our persona embeddings with the extracted text features, (b) BCA + SC + $\mathcal{P}^{(>att)}$, which integrates the persona embeddings at the penultimate layer. Note that the text and socio-cultural (SC) features are already fused at that layer, and (c) BCA + SC + $\mathcal{P}^{(<att)}$ fuses the extracted text and socio-cultural features with persona embeddings using an attention layer (as in MM-HATE).

Table 2b summarizes the results of our evaluation on hate speech detection task. We observe that SC-fused model (BCA + SC) performs marginally better than our persona-fused model (BCA + $\mathcal{P}$). This result can be ascribed to the domain specificity of SC features. We also note that the combination of all the extracted features leads to a marked improvement in prediction performance, and even more so when the persona embeddings are fed to the fusion layer (BCA + SC + $\mathcal{P}^{(<att)}$). Thus, our DAPPER model is able to extract behavioral features from user texts allowing positive knowledge transfer to various domains and applications.

8 Conclusion

We proposed a DAPPER model that learns a domain adapted pretraining-based persona representation. Our DAPPER model leverages pretrained BERT model and fine-tunes it with additional domain-adaptive layers. By introducing a knowledge-attention mechanism, we allow the domain knowledge to be integrated into our persona embeddings. The proposed model achieves significant gains across persona classification task in different domains. Our evaluations validate that our model is capable of adapting to a new domain with limited labeled data. We also highlight the transferability of our persona embeddings in a downstream hate speech detection task.

References

David Bamman, Brendan O'Connor, and Noah A Smith. 2013. Learning latent personas of film characters. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 352–361.

Murray R Barrick and Michael K Mount. 1991. The big five personality dimensions and job performance: a meta-analysis. *Personnel psychology*, 44(1):1–26.

Murray R Barrick and Michael K Mount. 1993. Autonomy as a moderator of the relationships between the big five personality dimensions and job performance. *Journal of applied Psychology*, 78(1):111.

Fabio Celli, Fabio Pianesi, David Stillwell, and Michal Kosinski. 2013. Workshop on computational personality recognition: Shared task. In *Seventh International AAAI Conference on Weblogs and Social Media*.

Shristi Chaudhary, Ritu Singh, Syed Tausif Hasan, and Ms Inderpreet Kaur. 2013. A comparative study of different classifiers for myers-brigg personality prediction model. *Linguistic analysis*, page 21.

Eric Chu, Prashanth Vijayaraghavan, and Deb Roy. 2018. Learning personas from dialogue with attentive memory networks. *arXiv preprint arXiv:1810.08717*.

Junyoung Chung, Caglar Gulcehre, KyungHyun Cho, and Yoshua Bengio. 2014. Empirical evaluation of gated recurrent neural networks on sequence modeling. *arXiv preprint arXiv:1412.3555*.

JR Costa and T PAUL. 1996. of personality theories: Theoretical contexts for the five-factor model. *The five-factor model of personality: Theoretical perspectives*, 51.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

John M Digman and Naomi K Takemoto-Chock. 1981. Factors in the natural language of personality: Re-analysis, comparison, and interpretation of six major studies. *Multivariate behavioral research*, 16(2):149–170.

Lucie Flekova and Iryna Gurevych. 2015. Personality profiling of fictional characters using sense-level links between lexical resources. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1805–1816.

Ona de Gibert, Naiara Perez, Aitor García-Pablos, and Montse Cuadros. 2018. Hate speech dataset from a white supremacy forum. *arXiv preprint arXiv:1809.04444*.

Lewis R Goldberg. 1990. An alternative" description of personality": the big-five factor structure. *Journal of personality and social psychology*, 59(6):1216.

Larry A Hjelle and Daniel J Ziegler. 1992. *Personality theories: Basic assumptions, research, and applications*. McGraw-Hill Book Company.

Sopan Khosla, Niyati Chhaya, and Kushal Chawla. 2018. Aff2vec: Affect–enriched distributional word representations. *arXiv preprint arXiv:1805.07966*.

Yoon Kim. 2014. Convolutional neural networks for sentence classification. *arXiv preprint arXiv:1408.5882*.

Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Jiwei Li, Michel Galley, Chris Brockett, Georgios P Spithourakis, Jianfeng Gao, and Bill Dolan. 2016. A persona-based neural conversation model. *arXiv preprint arXiv:1603.06155*.

Fei Liu, Julien Perez, and Scott Nowson. 2016. A language-independent and compositional model for personality trait recognition from short texts. *arXiv preprint arXiv:1610.04345*.

François Mairesse, Marilyn A Walker, Matthias R Mehl, and Roger K Moore. 2007. Using linguistic cues for the automatic recognition of personality in conversation and text. *Journal of artificial intelligence research*, 30:457–500.

Navonil Majumder, Soujanya Poria, Alexander Gelbukh, and Erik Cambria. 2017. Deep learning-based document modeling for personality detection from text. *IEEE Intelligent Systems*, 32(2):74–79.

Dan P McAdams and Erika Manczak. 2015. Personality and the life story.

Alexander Miller, Adam Fisch, Jesse Dodge, Amir-Hossein Karimi, Antoine Bordes, and Jason Weston. 2016. Key-value memory networks for directly reading documents. *arXiv preprint arXiv:1606.03126*.

James W Pennebaker and Laura A King. 1999. Linguistic styles: Language use as an individual difference. *Journal of personality and social psychology*, 77(6):1296.

James W Pennebaker, Matthias R Mehl, and Kate G Niederhoffer. 2003. Psychological aspects of natural language use: Our words, our selves. *Annual review of psychology*, 54(1):547–577.

Jeffrey Pennington, Richard Socher, and Christopher D Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543.

Kurt Shuster, Samuel Humeau, Antoine Bordes, and Jason Weston. 2018. Engaging image chat: Modeling personality in grounded dialogue. *arXiv preprint arXiv:1811.00945*.

Paul H Soloff. 1985. Personality disorders. In *Diagnostic interviewing*, pages 131–159. Springer.

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

Prashanth Vijayaraghavan, Hugo Larochelle, and Deb Roy. Interpretable multi-modal hate speech detection.

Zeerak Waseem. 2016. Are you a racist or am i seeing things? annotator influence on hate speech detection on twitter. In *Proceedings of the first workshop on NLP and computational social science*, pages 138–142.

Jiani Zhang, Xingjian Shi, Irwin King, and Dit-Yan Yeung. 2017. Dynamic key-value memory networks for knowledge tracing. In *Proceedings of the 26th international conference on World Wide Web*, pages 765–774.