Multitask Learning for Emotion and Personality Traits Detection

Yang Li, Amirmohammad Kazameini, Yash Mehta, Erik Cambria

Abstract—In recent years, deep learning-based automated personality trait detection has received a lot of attention, especially now, due to the massive digital footprints of an individual. Moreover, many researchers have demonstrated that there is a strong link between personality traits and emotions. In this paper, we build on the known correlation between personality traits and emotional behaviors, and propose a novel multitask learning framework, SoGMTL that simultaneously predicts both of them. We also empirically evaluate and discuss different information-sharing mechanisms between the two tasks. To ensure the high quality of the learning process, we adopt a MAML-like framework for model optimization. Our more computationally efficient CNN-based multitask model achieves the state-of-the-art performance across multiple famous personality and emotion datasets, even outperforming Language Model based models.

Index Terms—Personality Trait Detection, Multitask Learning, Information Sharing

1 INTRODUCTION

Personality traits refer to the difference among individuals in characteristic patterns of thinking, feeling, and behaving. Specifically, personality traits have been related to individual well being, social-institutional outcomes (e.g., occupational choices, job success), and interpersonal (e.g., relationship satisfaction). In addition, in modern times, people prefer delivering their thoughts, emotions, and complaints on the social media platform, e.g., Facebook and Twitter [2]. Hence, there is a widespread interest to develop models that can use online data on human preferences and behavior (i.e., digital footprints) to automatically predict individuals’ levels of personality traits for use in job screenings [7], recommender-systems [7] and social network analysis. You et al. [7] found that the digital footprint on social media can be used to measure personality traits well. There are different systems in the personality trait description, and the most widely used is called the Five-Factor Model [7]. This system includes five traits that can be remembered by the acronym OCEAN: Openness (OPN), Conscientiousness (CON), Extraversion (EXT), Agreeableness (AGR), and Neuroticism (NEU). With the advancement in machine learning research and the availability of larger amounts of data, the ability and desire to detect user personality and preference are now higher than ever. While the performance of these models is not high enough to allow for the precise distinction of people based on their traits, predictions can still be ‘right’ on average and be utilized for digital mass persuasion [7]. However, automated personality prediction also raises serious concerns with regard to individual privacy and the conception of informed consent [7].

Recent works on this have made significant strides in machine learning-based personality detection [7], [7], [7], [7], [7], [7], [7], [7]. However, all existing works are single-task learning in the supervised learning way. For example, Kalghatgi et al. [7] add an MLP over handcrafted features to do the detection, and Tandera et al. [7] combined LSTM and CNN to make a better feature extraction pipeline, and finally Mehta et al. [7] combine language models with psycholinguistic features for personality prediction.

As we know, emotion has a direct link to personality. According to work from [7], [7], [7], [7], we know that neuroticism predicts higher negative emotion and lower positive affect, and conscientiousness, by contrast, is inversely associate with negative emotions, and agreeableness predicts higher positive emotion and lower negative affect, and extraversion is associated with higher positive affect and more positive subjective evaluations of daily activities, and openness is associate with a mix of positive and negative emotions [7]. Therefore, we build on the results showing that personality trait detection and emotion detection are complementary. After exploring different information-sharing mechanisms (e.g., SiG, SiLG, CAG, and SoG) between personality trait detection and emotion detection, we propose a novel multitask learning framework SoGMTL based on CNN for simultaneously detecting personality traits and emotions, and our model achieves the state of the art across both personality and emotion datasets. Also, to ensure the high quality of the learning procedure, we propose a MAML-like method for model optimization.

The rest of the paper is structured as follows: Section ?? introduces related works about personality trait detection and multitask learning; Section ?? presents the proposed model and discusses several different information sharing gates; Section ?? conducts the experiments on multitask learning; finally, Section ?? concludes the paper and presents future works.
2 RELATED WORKS

This paper mainly focuses on personality trait detection with a multitask learning framework. In the framework, based on the observation that personality and emotion are complementary, we select emotion prediction as to the auxiliary task. Therefore, in the related works, we will only give a comprehensive review of personality trait detection and multitask learning.

2.1 Personality Trait Detection

It has been confirmed by researches that online behavior is related to personality [7], [8]. Many works have successfully applied the machine learning methods to detect the personality trait in the content generated over the social media [7], [9]. Especially in the work of You et al. [7], they found that the analysis based on digital footprint was better at measuring personality traits than close others or acquaintances (friends, family, spouse, colleagues, etc.). Personality trait detection can be based on the different types of features, such as text data (self-description, social media content, etc.), demography data (gender, age, followers, etc.), etc. One of the initial models was by Argamon et al. [7], which applied an SVM over the extracted statistical features of functional lexicons to detect the personality trait. Following this work, Farnadi et al. [7] adopted SVM to make personality detection over the features of network size, density, frequency of updating status, etc. Zhusupova et al. [7] detected the personality trait of Portuguese users on the Twitter platform based on demographic and social activity information. Kalghatgi et al. [7] detected the personality trait based on the neural networks (MLP) with the hand-crafted features. Su et al. [7] applied the RNN and HMM to obtain the personality trait based on the Chinese LIWC annotations extracted from the dialogue. Carducci et al. [7] also applied the SVM to do the personality detection, the difference between Farnadi’s work [7] is that the feature they applied is the text data.

Researchers also leveraged some recent developments of NLP in this field. Tandera et al. [7] made personality detection over the text data directly based on deep learning methods (LSTM+CNN). At the same time, Liu et al. [7] built a hierarchical structure based on Bi-RNN to learn the word and sentence representations that can infer the personality traits from three languages, i.e., English, Italian, and Spanish. Majumder et al. [7] proposed a CNN-based model to extract fixed-length features from personal documents, and then connected the learned features with 84 additional features in the Mairesse’s library for personality feature detection. Van et al. [7] tried to infer the personality trait based on the 275 profiles on LinkedIn, a job-related social media platform, and they concluded that extroversion could be well inferred from the self-expression of the profiles. Amirhosseini et al. [7] increased the accuracy of the MBTI dataset with Gradient boosting. Lynn et al. [7] used message level attention instead of word-level attention to improving the result by focusing on the most relevant Facebook posts obtained from Kosinski et al. dataset [10]. Gjurkovic et al. [7] used Sentence BERT [7] to set a benchmark for their huge Reddit dataset named PANDORA, including three personality tests’ labels OCEAN, MBTI, and Enneagram. Pandora tried to improve and address the deficiencies of its older version, MBTI19k [7]. Kazemeini et al. [7] feed BERT embeddings into an SVM based ensemble method to improve the accuracy of Essays dataset of Pennebaker et al. [7]. Finally, Mehta et al. [7] perform a thorough empirical investigation using Language Models (LM) and a variety of psycholinguistic features to identify the best combination and the impact of each feature on predicting the traits, achieving the state-of-the-art results on the Essays as well as the Kaggle MBTI dataset. Recently, Mehta et al. [7] reviewed the latest advances in deep learning-based automated personality with a focus on effective multimodal personality prediction.

2.2 Multitask Learning

Multitask learning has attracted lots of attention in recent years, it is to enhance the performance by learning the commonalities and differences among different tasks [7], [9], [11]. Typically, one of the widely used models is proposed by Carunana [7], [7]. And there is a shared-bottom structure across tasks. This structure reduces the risk of overfitting, but it will cause optimization conflicts due to task differences. Recently, some works attempt to resolve such conflicts by adding constraints on specific task parameters [7], [7]. For example, Duong et al. [7] trained two different tasks with different data sets and added L-2 constraints between the two sets of parameters to enhance the adaptability of the model over the low resource data. Misra et al. [7] proposed the cross-stitch networks to learn a linear combination of task-specific hidden-layer embeddings. And the tasks on the semantic segmentation and surface normal prediction over the image data validate the effectiveness of this method. Yang et al. [7] generated the task-specific parameters by tensor factorization, and it achieved better performance on the task which may lead to conflicts. The drawback of this model relies on its large number of parameters which need more training data. Ma et al. [7] designed a multigate mixture-of-experts based on the work [7], it requires several experts (e.g., there are 8 experts in their settings.) to capture different features of the task, then chooses the highest gate score as the final feature. Same as the works from Yang [7], there is a large number of parameters that need more training data. Unlike previous works, this paper proposes a task-specific multitask learning framework that is used for personality trait detection and emotion detection.

3 MODEL DESCRIPTION

In this section, we describe our proposed model, with the framework shown in Figure ???. Typically, there will be a shared bottom in the structure, and as we have discussed, it is difficult to optimize the model due to different tasks. Therefore, based on CNN, this paper designs an efficient information flow pipeline to realize the information exchange between different tasks.

3.1 Preliminary

Generally, after the embedding layer, each word in the sentence is represented by the vector \( x_i \in R^d \) where \( d \) is the embedding dimension. And the whole sentence is represented by \( X = [x_1, x_2, ..., x_n] \) where \( n \) is the word number. There are three layers in the CNN model, namely...
With these three steps, we can get a powerful model in the we will give a detailed discussion about gate designing. After that we can obtain a feature map with ??.

To evaluate the information sharing, we apply the cosine between the two tasks is useful, but it is difficult to evaluate. While, the disadvantage of SiLG is similar with that in SiG, and it cannot select the useful information from a network by passing all information indiscriminately. 

**SiG:** The first approach is to pass the information from one network to the other one with Sigmoid Gate (SiG) directly, and it can be shown in equation ??.

\[
\begin{align*}
    m_{i,j} &= c_{i}^{1} W_{c} c_{j}^{2} \\
    \alpha_{i,j} &= \frac{\exp m_{i,j}}{\sum_{k} \exp m_{i,k}} \\
    h_{i}^{l2} &= c_{i}^{2} + \left(\sum_{j} \alpha_{i,j} c_{j}^{2}\right)
\end{align*}
\]

(3)

Where \( \sigma \) denotes the Sigmoid function on hidden value, which works as the gate between the two tasks. This gate is simple, and it lets useful and useless information pass indiscriminately between the two tasks. Therefore, it is prone to optimization conflicts, and empirical results have validated this phenomenon.

**CAG:** Another one is the across attention gate (CAG), which is proposed in [7]. CAG considers the contextual information as the extra features in both tasks. The procedures are described in Equation ??.

\[
\begin{align*}
    m_{i,j} &= c_{i}^{1} W_{c} c_{j}^{2} \\
    \alpha_{i,j} &= \frac{\exp m_{i,j}}{\sum_{k} \exp m_{i,k}} \\
    h_{i}^{l2} &= c_{i}^{2} + \left(\sum_{j} \alpha_{i,j} c_{j}^{2}\right)
\end{align*}
\]

(3)

where \( W_{c} \) denotes the model parameter. CAG has considered the features in both tasks in the cross attention calculation, and it should benefit from this cross calculation. However, the features of each task change dynamically during the optimization process, it will be difficult to get the appropriate optimal feature flow between two tasks with CAG. Also, it takes more time in the CAG calculation.

**SiLG:** One another gate we discussed here is the Sigmoid weighted linear gate (SiLG), which is proposed in [7], the procedures are shown in Equation ??.

This gate shows its effectiveness in reinforcement learning as an approximation of ReLU.

\[
\begin{align*}
    m_{i}^{l1} &= \sigma(c_{i}^{1}) \\
    h_{i}^{l2} &= c_{i}^{2} + m_{i}^{l1}
\end{align*}
\]

(4)

While, the disadvantage of SiLG is similar with that in SiG, and it cannot select the useful information from a network by passing all information indiscriminately.

**SoG:** To overcome those problems, we changed the Sigmoid function in SiG to Softmax to construct a selection gate named softmax weight gate SoG, with the result shown in the Equation ??.

\[
\begin{align*}
    m_{i}^{l1} &= \text{softmax}(c_{i}^{1}) \\
    h_{i}^{l2} &= c_{i}^{2} + m_{i}^{l1}
\end{align*}
\]

(5)

With this change, the proposed gate mechanism is more effective: Compared with CAG, SoG has higher time efficiency in the calculation, compared with SiG and SiLG, SoG can better select features with softmax operation.

**3.3 Meta Multitask Training**

It is the multilabel prediction task for the personality trait detection, therefore, we define the objective function with the...
where $C$ is the class number, and it is 5 in our paper.

However, it is the multiclass prediction task for the emotion detection, and we apply the cross entropy as the objective function which is described in Equation (6).

$$
L_{\text{Personality}} = - \frac{1}{C} \sum_{i} y_i \log((1 + \exp(-\hat{y}_i))^{-1}) + (1 - y_i) \log\left(\frac{\exp(-\hat{y}_i)}{1 + \exp(-\hat{y}_i)}\right)
$$

(6)

where $C$ is the class number, and it is 5 in our paper.

To optimize the network in a unified framework, we simply add the two loss functions together as the joint function that shows in Equation (7).

$$
L_{\text{Multi}} = L_{\text{Personality}} + L_{\text{Emotion}}
$$

(7)

To optimize the network in a unified framework, we simply add the two loss functions together as the joint function that shows in Equation (7).

There are different ways to train the model. Generally, each task has its own dataset, therefore, it is necessary to design an adaptive training framework for different subtasks. In this paper, like the $k$ shot in MAML [7], we also select the pseudo $k$ shot (i.e., $k$ batches shot) during training, which indicates select $k$ more batches in one task to form the training data pair. Then we update the parameters in a MAML like way. And the algorithm we designed is shown in Algorithm 1.

Algorithm 1 The optimization method about MAML like in the multitask training.

Require: Personality dataset $D^p$ and Emotion dataset $D^e$.

Require: Model parameters $\theta = [\theta_1, \theta_2, ..., \theta_n]$, and the learning rate $r$.

1: Initialization the model parameters with random values.
2: for select a batch from $D^p$ do
3: Create a list $\Theta$ and $\mathcal{L}$ for parameter update in each step.
4: Sample $k$ batches $D^p_k$ from $D_p$.
5: for select a batch from $D^p_k$ do
6: Obtain the loss $L_{\text{Multi}}$ and its gradient descent $\nabla L_{\text{Multi}}(f_\theta)$.
7: Obtain the updated parameter with $\hat{\theta} = \theta - r \cdot \nabla L_{\text{Multi}}(f(\theta))$ and update it to $\theta$.
8: Stack $\theta$ to $\Theta$, and $L_{\text{Multi}}$ to $\mathcal{L}$
9: end for
10: Calculate the gradient to obtain $\nabla \mathcal{L}(\theta)$ and $\nabla \Theta(\theta)$.
11: Update the parameter $\theta$ with $\nabla \mathcal{L}(\theta) + \nabla \Theta(\theta)$.
12: end for

4 Experiments

4.1 Experiment Settings

The dataset we applied includes ISEAR [7], TEC [7], and Personality [7]. There are 7,666 labeled sentences in ISEAR dataset contains 7 emotions, i.e., anger, disgust, fear, joy, sadness, shame and guilt. TEC includes 21,051 labeled sentences that are selected in tweets by pre-specified hashtags, i.e., joy, anger, disgust, surprise, fear and sadness. Personality dataset contains 9,917 multilabeled sentences in the view of the Big Five personality traits. The details about the datasets are listed in Table 1. In the experiments, all of the dataset are split randomly in a ratio of 8:2, of which 80% are training data, and the rest are the test data. The embedding dimension is set to 300. All the experiments are run on Colab[1] with GPU support, and we run them five times to get the average results in the records. As we have mentioned, it is the multilabel prediction task for the personality trait detection, and the output of this task is a five-dimension vector after the Sigmoid function, with each value representing a trait in the Five-Factor Model. Therefore, when the value is greater than the threshold (i.e., 0.5), we then regard it as having such a trait. While, it is the multiclass prediction task for personality trait detection, and the output of the emotion detection task is a $m$-dimension vector (where $m$ is the emotion categories) after the Softmax function. Therefore, we select the index with the highest value in the $m$-dimension vector as the predicted label. To make a comprehensive evaluation of these two tasks, we adopt Accuracy, Precision, Recall, and F1 measurement in the personality trait detection task, and Accuracy in the emotion detection task, respectively.

4.2 Multitask Learning

Before the discussion, we list the baseline of the single task in Table 2. In the baseline, we include the methods like BERT [7], CNN [7], LSTM [7] those three classical models.

From Table 2, we can see that CNN Single achieved the best performance in almost all datasets with different evaluation measures. That means in the personality trait detection and emotion detection, CNN has a good ability in the feature extraction. And this is also the reason why we choose CNN as the prototype in the design of the multitask model.

Then we carry out experiments to compare the advantages and disadvantages of different variants and models. Firstly, to validate the effectiveness of MAML, we make comparisons with the classical optimization method Adam [2]. In the first column of the Table 2 and 3, the name with “maml” indicates that the model is optimized by maml; otherwise, the model is optimized with Adam directly. In the experiments, the baselines we compared include:

- **SiGMTL**: It applies the Sigmoid function as the gate between two different models.
- **CAGMTL**: It applies the cross-entropy model as the gate between two different models.
- **SiLMTL**: It applies the Sigmoid weighted linear model as the gate between two different models. And it can be treated as that proposed in work [7].
- **SoGMTL**: It is the multitask learning model we proposed.

We then conduct experiments to make comparisons with the different variants and models. The experiment results on dataset ISEAR are shown in Table 2, and the experiment results on dataset TEC are shown in Table 3. As can be seen from the two tables, SoGMTL+ MAML achieves the best results in personality trait detection under almost all conditions, and the average improvement is about 0.75% for training with the ISEAR dataset and about 2.09%.

1. https://colab.research.google.com/
### Table 1
The details of the datasets this paper involved.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Total Number</th>
<th>Label Type</th>
<th>Label</th>
</tr>
</thead>
<tbody>
<tr>
<td>ISEAR</td>
<td>7,666</td>
<td>Single</td>
<td>{anger, disgust, fear, joy, sadness, shame, guilt}</td>
</tr>
<tr>
<td>TEC</td>
<td>21,051</td>
<td>Single</td>
<td>{joy, anger, disgust, surprise, fear, sadness}</td>
</tr>
<tr>
<td>Personality</td>
<td>9,917</td>
<td>Multiple</td>
<td>{openness, conscientiousness, extraversion, agreeableness, and neuroticism}</td>
</tr>
</tbody>
</table>

### Table 2
The baselines results over the datasets we adopted.

<table>
<thead>
<tr>
<th>Models</th>
<th>Accuracy</th>
<th>Precision</th>
<th>Recall</th>
<th>F1</th>
<th>Average</th>
<th>Accuracy</th>
<th>Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>BERT Single</td>
<td>60.79%</td>
<td>61.53%</td>
<td>77.60%</td>
<td>68.39%</td>
<td>67.08%</td>
<td>59.28%</td>
<td>54.98%</td>
</tr>
<tr>
<td>LSTM Single</td>
<td>61.84%</td>
<td>64.06%</td>
<td>75.10%</td>
<td>69.13%</td>
<td>67.53%</td>
<td>54.96%</td>
<td>55.51%</td>
</tr>
<tr>
<td>CNN Single</td>
<td>62.23%</td>
<td>63.17%</td>
<td>82.60%</td>
<td>71.54%</td>
<td>69.89%</td>
<td>59.60%</td>
<td>56.59%</td>
</tr>
</tbody>
</table>

### Table 3
The experiment results over the Personality+ISEAR datasets.

<table>
<thead>
<tr>
<th>Models</th>
<th>Accuracy</th>
<th>Precision</th>
<th>Recall</th>
<th>F1</th>
<th>Average</th>
<th>Accuracy</th>
<th>Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>SiGMTL</td>
<td>61.29%</td>
<td>63.01%</td>
<td>71.58%</td>
<td>66.61%</td>
<td>65.87%</td>
<td>57.20%</td>
<td>58.92%</td>
</tr>
<tr>
<td>SiGMTL+maml</td>
<td>61.70%</td>
<td>63.62%</td>
<td>74.01%</td>
<td>67.45%</td>
<td>66.70%</td>
<td>58.17%</td>
<td>59.12%</td>
</tr>
<tr>
<td>CAGMTL</td>
<td>62.18%</td>
<td>64.71%</td>
<td>73.39%</td>
<td>69.08%</td>
<td>67.47%</td>
<td>58.13%</td>
<td>59.47%</td>
</tr>
<tr>
<td>CAGMTL+maml</td>
<td>62.41%</td>
<td>64.96%</td>
<td>75.22%</td>
<td>68.00%</td>
<td>67.66%</td>
<td>58.35%</td>
<td>59.21%</td>
</tr>
<tr>
<td>SiLGMTL</td>
<td>60.49%</td>
<td>63.49%</td>
<td>68.79%</td>
<td>65.96%</td>
<td>64.68%</td>
<td>55.98%</td>
<td>56.18%</td>
</tr>
<tr>
<td>SiLGMTL+maml</td>
<td>60.81%</td>
<td>63.46%</td>
<td>70.34%</td>
<td>66.15%</td>
<td>65.19%</td>
<td>54.21%</td>
<td>54.69%</td>
</tr>
<tr>
<td>SoGMTL</td>
<td>62.57%</td>
<td>64.05%</td>
<td>79.96%</td>
<td>71.09%</td>
<td>69.42%</td>
<td>59.03%</td>
<td>59.71%</td>
</tr>
<tr>
<td>SoGMTL+maml</td>
<td>62.77%</td>
<td>64.22%</td>
<td>83.12%</td>
<td>72.45%</td>
<td>70.64%</td>
<td>59.71%</td>
<td>59.71%</td>
</tr>
</tbody>
</table>

### Table 4
The experiment results over the Personality+TEC datasets.

<table>
<thead>
<tr>
<th>Models</th>
<th>Accuracy</th>
<th>Precision</th>
<th>Recall</th>
<th>F1</th>
<th>Average</th>
<th>Accuracy</th>
<th>Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>SiGMTL</td>
<td>62.24%</td>
<td>64.35%</td>
<td>72.83%</td>
<td>68.74%</td>
<td>66.10%</td>
<td>55.45%</td>
<td>56.49%</td>
</tr>
<tr>
<td>SiGMTL+maml</td>
<td>61.87%</td>
<td>64.76%</td>
<td>72.79%</td>
<td>68.43%</td>
<td>66.96%</td>
<td>55.38%</td>
<td>56.34%</td>
</tr>
<tr>
<td>CAGMTL</td>
<td>62.22%</td>
<td>64.23%</td>
<td>75.81%</td>
<td>69.51%</td>
<td>67.94%</td>
<td>56.49%</td>
<td>57.64%</td>
</tr>
<tr>
<td>CAGMTL+maml</td>
<td>62.86%</td>
<td>64.45%</td>
<td>79.29%</td>
<td>71.10%</td>
<td>69.43%</td>
<td>57.64%</td>
<td>59.32%</td>
</tr>
<tr>
<td>SiLGMTL</td>
<td>61.54%</td>
<td>63.68%</td>
<td>72.19%</td>
<td>67.62%</td>
<td>66.26%</td>
<td>55.32%</td>
<td>56.04%</td>
</tr>
<tr>
<td>SiLGMTL+maml</td>
<td>62.21%</td>
<td>64.26%</td>
<td>77.74%</td>
<td>69.12%</td>
<td>67.58%</td>
<td>55.04%</td>
<td>55.94%</td>
</tr>
<tr>
<td>SoGMTL</td>
<td>62.65%</td>
<td>63.95%</td>
<td>79.40%</td>
<td>70.82%</td>
<td>69.21%</td>
<td>56.29%</td>
<td>57.15%</td>
</tr>
<tr>
<td>SoGMTL+maml</td>
<td>63.08%</td>
<td>64.79%</td>
<td>86.11%</td>
<td>73.93%</td>
<td>71.98%</td>
<td>57.15%</td>
<td>58.34%</td>
</tr>
</tbody>
</table>

for training with the TEC dataset compared with the single task learning with CNN. And it is the same in emotion detection, on the ISEAR dataset, our model improves by 1.32% compared to single-task learning, and on the TEC dataset, our model improves by 0.56% compared to single-task learning. Especially in the measurement of recall, both datasets showed significant improvement. Compared with the single-task learning using CNN, ISEAR improved by 0.52% and TEC improved by 3.51%. From these observations, we can conclude that our SoGMTL improves the performance of both tasks in multitask learning through the designed gated mechanism. Compared to the baselines of the different variants and models in the MTL, we can see that our design achieves the best results in almost all cases, except for the precise measurement of personality trait detection trained with the ISEAR dataset. When we make a horizontal comparison between TEC and ISEAR, we find that the performance on TEC dataset is generally greater than that on the ISEAR. One reason may be that TEC has three times as much data as ISEAR. In terms of optimization methods, we can see that the MAML method is more effective than the normal method, improving in almost all cases.

To validate the computation efficiency with these three gate methods, the running time for one epoch is recorded in Figure ??.

![Fig. 3. The running time with different gates, y axis denotes seconds.](image-url)
4.3 Ablation Study

To validate the features that learned by the SoG in different levels (i.e., convolutional layer (Conv for short), max-pooling layer (Max-Pooling) and dense layer (Dense), etc.), we only use one SoG gate in the model, and the results are shown in Table ??.

It can be seen from the table that the performance of SoG in the convolutional layer is the best, followed by the performance in the max-pooling layer. That means that the SoG in the convolutional layer is the most important in the information flow control. Furthermore, to quantify the degree of SoG's control over information flows at different locations, we calculate the cosine similarity between the latent vectors of two tasks in the convolutional layer (i.e., Sim1) and dense layer (i.e., Sim2) respectively. And the results are shown in Table ??.

<table>
<thead>
<tr>
<th>Variants</th>
<th>Sim1</th>
<th>Sim2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dense</td>
<td>0.21 (-0.13)</td>
<td>0.47 (-0.05)</td>
</tr>
<tr>
<td>Max-Pooling</td>
<td>0.14 (-0.20)</td>
<td>0.33 (-0.19)</td>
</tr>
<tr>
<td>Conv</td>
<td>0.32 (-0.02)</td>
<td>0.47 (-0.05)</td>
</tr>
<tr>
<td>SoGMTL</td>
<td>0.34</td>
<td>0.52</td>
</tr>
</tbody>
</table>

TABLE 5

The cosine similarity between the latent vector of the two tasks in the convolutional layer and dense layer respectively when SoG is placed at different levels.

SoGMTL is assumed to be the "best" status of information control in multitask learning. Therefore, it can be concluded from the table that when SoG is only placed in the dense layer, the difference between Sim1 and the best is -0.13, and the difference between Sim2 and the best is -0.05. However, both gaps become extremely small when SoG is only placed in the convolutional layer. Thus, we can know that more useful information is shared via the SoG in the convolutional layer, followed by the max-pooling layer, and less of them are shared with the dense layer. That is why the average performance is the best when SoG is only placed in the convolutional layer shown in Table ??.

4.4 Case Study

To make a clear understanding of SoGMTL’s performance in multitask learning, a case study is conducted, and the results are shown in Table ??.

Personality trait detection is a multilabel classification problem, so there are multiple labels in a detection. When the value is greater than a threshold (i.e., 0.5), it indicates the personality has this trait. And the cases in the Table ?? show the effectiveness of SoGMTL in this task by having clear boundaries in the prediction. However, emotion detection is a multiclass classification problem [?]. In the prediction, the label is determined by the maximum value. As can be seen from the image in Table ??, there is an obvious difference between the maximum value and other values. In other words, our proposed model can be well applied to personality trait detection and emotion detection in a multitask learning approach.

5 Conclusion

In this work, we designed a multitask learning framework in personality trait detection and emotion detection. Also, we have discussed and validated the effective sharing way between two tasks. Finally, we designed a MAML-like training method when there is more than one data source. Empirical results show the effectiveness of our proposed framework and optimization method. In future work, more contextual information like personal preference, current location, etc., will be considered.
### Table 5
The performance of SoG in different levels that trained with TEC dataset.

<table>
<thead>
<tr>
<th>Settings</th>
<th>Accuracy</th>
<th>Precision</th>
<th>Recall</th>
<th>F1</th>
<th>Average</th>
<th>Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dense</td>
<td>62.45%</td>
<td>63.94%</td>
<td>79.30%</td>
<td>70.80%</td>
<td>69.12%</td>
<td>56.55%</td>
</tr>
<tr>
<td>Max-Pooling</td>
<td>62.45%</td>
<td>63.34%</td>
<td>80.82%</td>
<td>71.13%</td>
<td>69.48%</td>
<td>55.81%</td>
</tr>
<tr>
<td>Conv</td>
<td>62.90%</td>
<td>64.86%</td>
<td>81.05%</td>
<td>72.03%</td>
<td>70.21%</td>
<td>56.95%</td>
</tr>
</tbody>
</table>

### Table 7
The case study about the SoGMTL’s performance in the personality trait detection and emotion detection. The dashed line in the predicted image indicates the threshold in the personality trait detection, it is 0.5 in our paper.

<table>
<thead>
<tr>
<th>Case</th>
<th>Category</th>
<th>Sentence</th>
<th>True Label</th>
<th>Predicted</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Personality</td>
<td>Is still awake at 3:30. oh me.</td>
<td>Neuroticism, Openness</td>
<td><img src="image1" alt="Predicted Image" /></td>
</tr>
<tr>
<td></td>
<td>Emotion</td>
<td>I’m home watchin this sad movie. Missing college.</td>
<td>Sadness</td>
<td><img src="image2" alt="Predicted Image" /></td>
</tr>
<tr>
<td>2</td>
<td>Personality</td>
<td>Damn you’s a sexy bitch DAMN GIRL!!!</td>
<td>Neuroticism, Agreeableness, Openness</td>
<td><img src="image3" alt="Predicted Image" /></td>
</tr>
<tr>
<td></td>
<td>Emotion</td>
<td>Holding my fucking tongue</td>
<td>Anger</td>
<td><img src="image4" alt="Predicted Image" /></td>
</tr>
</tbody>
</table>