MDFG: Multi-Dimensional Fine-Grained Modeling for Fatigue Detection

Mei Wang , Xiaojie Zhu, Ruimin Hu, Dongliang Zhu, Liang Liao, Mang Ye

National Engineering Research Center for Multimedia Software, School of Computer Science, Wuhan University
Hubei Key Laboratory of Multimedia and Network Communication Engineering,Wuhan University
School of Cyber Science and Engineering, Wuhan University, Whuhan, China Cyberspace Security Laboratory,
School of Network and Information Security, Xidian University, Xi'an,China
The School of Computer Science and Engineering, Nanyang Technological University, Singapore

AAAI-25

Introduction

传统方法存在的问题:

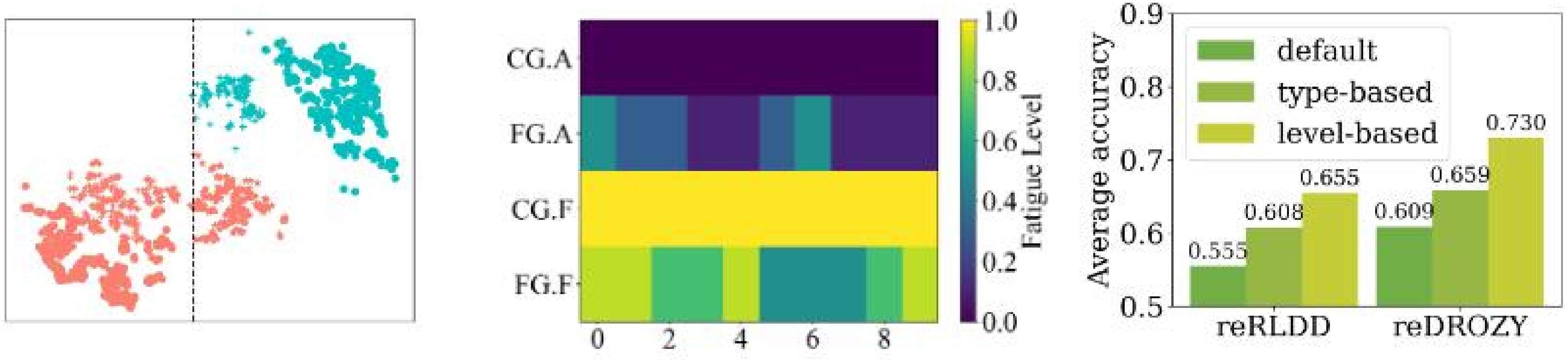
- 粗粒度标签（alert、fatigue）
- 只反映一段时间的状态
- 忽略不同疲劳类型

作者的理解:

可以从时间、类型、程度说明疲劳

Object	Criterion	Temporal Resolution	Complexity	Clarity	Example
CGL	subjective	low, minutes or above	single type	vague, inaccurate	"alert" or "fatigue"
MDFGL	objective	high, up to milliseconds	rich types	accurate, specific	"fatigue score: 0.7"

Table 1: Difference between Coarse-Grained Label and Multi-Dimensional Fine-Grained Label

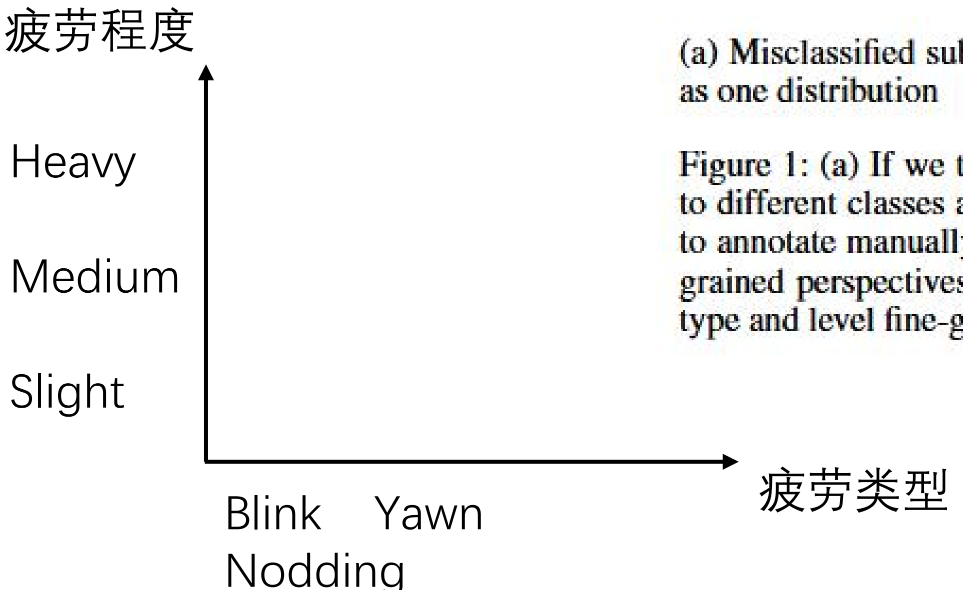


(a) Misclassified sub-types if we treat it as one distribution

(b) Manual fails to capture short-term fatigue changes and quantify fatigue levels

(c) Overlooking fine-grained information substantially reduces performance

Figure 1: (a) If we treat the data as a single type, the sub-types will be misclassified. Different colors and shapes correspond to different classes and types. (b) Demonstrates that the intermediate states have different fatigue levels, which is challenging to annotate manually. Deep purple indicates alert, while yellow indicates fatigue. CG and FG denote coarse-grained and fine-grained perspectives, respectively. (c) The "default" indicates no fine-grained, while "type-based" and "level-based" involve type and level fine-grained, respectively. The latter methods significantly outperform "default," all using the same network ϕ_F.



Method

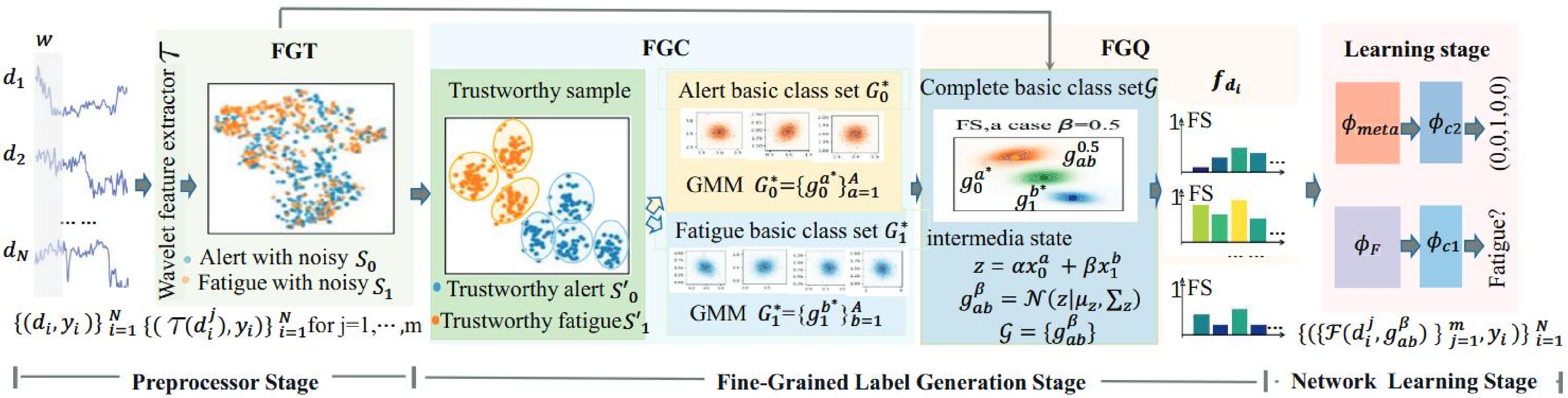


Figure 2: Overview of MDFG. In the preprocessing stage, the wavelet feature extractor $\mathcal{T}$ generates noise-containing FGT samples. In the fine-grained generation stage, we extract base classes from trustworthy samples and then combine the alert base class set G_0^* and the fatigue base class set G_1^* to generate intermediate state base classes. The fatigue score β for $\mathcal{T}(d_i^j)$ is assigned based on the closest base class to that in $\mathcal{G}$. Finally, in the network learning stage, a fatigue detection model is constructed based on the Fatigue Scores (FS) sequence.

Fine-Grained Temporality (FGT)

$$D = \{(\mathcal{T}(d_i^j), y_i)\}_{i=1}^{N}$$

Alert with noisy S_0
Fatigue with noisy S_1

Method

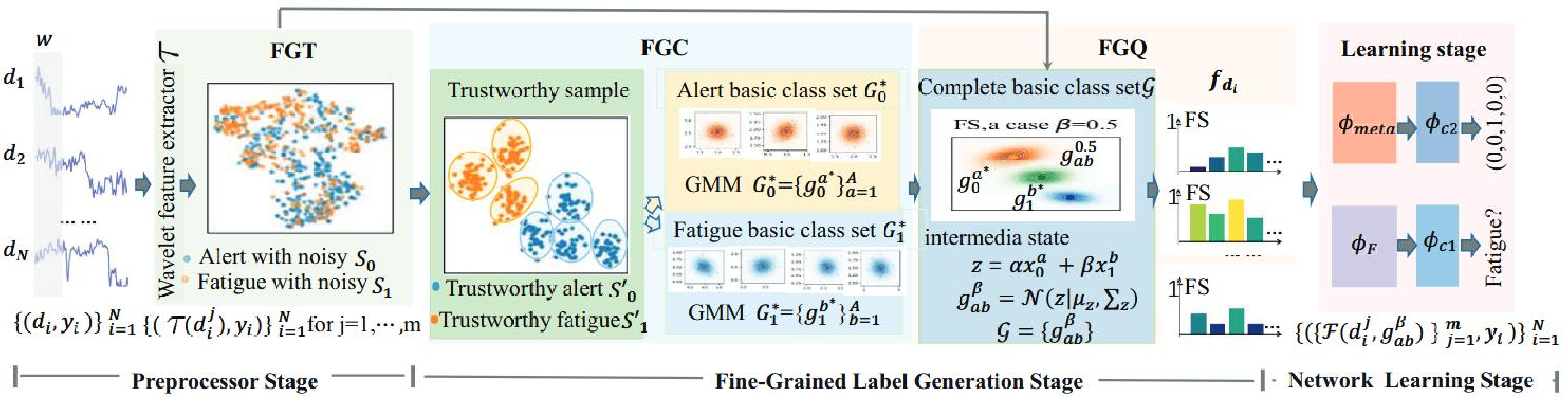


Figure 2: Overview of MDFG. In the preprocessing stage, the wavelet feature extractor $\mathcal{T}$ generates noise-containing FGT samples. In the fine-grained generation stage, we extract base classes from trustworthy samples and then combine the alert base class set G_0^* and the fatigue base class set G_1^* to generate intermediate state base classes. The fatigue score β for $\mathcal{T}(d_i^j)$ is assigned based on the closest base class to that in $\mathcal{G}$. Finally, in the network learning stage, a fatigue detection model is constructed based on the Fatigue Scores (FS) sequence.

Fine-Grained Categorization (FGC)

筛选可信样本

$$S'_0 = \{d_i^j | (\ell(\phi_{\text{trust}}(d_i^j)) < \gamma_0) \text{ if } d_i^j \in S_0\}$$

$$S'_1 = \{d_i^j | (\ell(\phi_{\text{trust}}(d_i^j)) < \gamma_1) \text{ if } d_i^j \in S_1\}$$

使用混合高斯模型对样本建模

$$g_0^a = \sum_{k=1}^{K} (\pi_0^a)_k \mathcal{N}(x_0^a | (\mu_0^a)_k, (\Sigma_0^a)_k)$$

Method

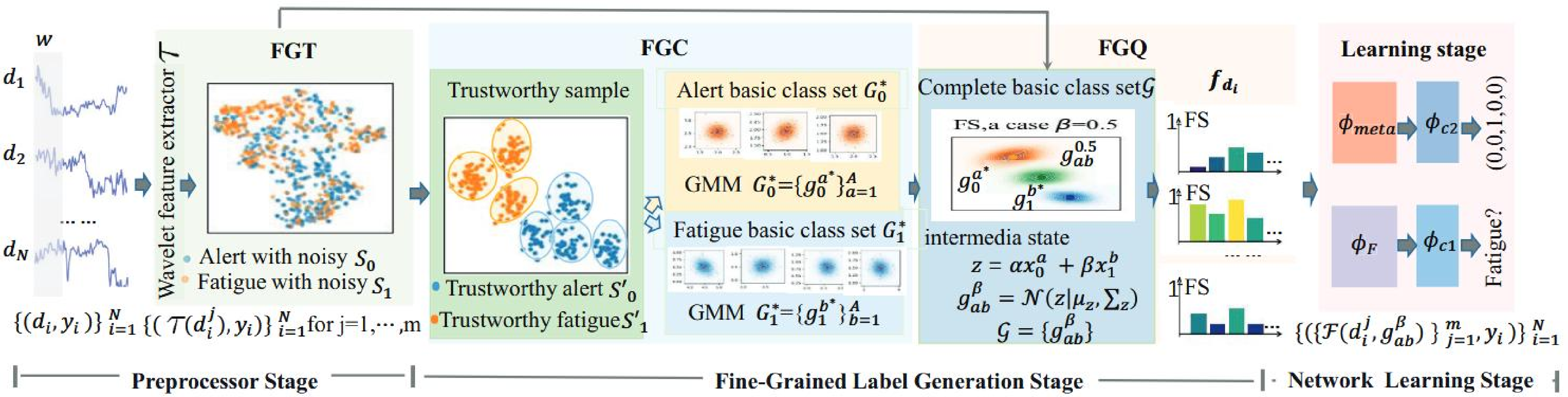


Figure 2: Overview of MDFG. In the preprocessing stage, the wavelet feature extractor $\mathcal{T}$ generates noise-containing FGT samples. In the fine-grained generation stage, we extract base classes from trustworthy samples and then combine the alert base class set G_0^* and the fatigue base class set G_1^* to generate intermediate state base classes. The fatigue score β for $\mathcal{T}(d_i^j)$ is assigned based on the closest base class to that in $\mathcal{G}$. Finally, in the network learning stage, a fatigue detection model is constructed based on the Fatigue Scores (FS) sequence.

Fine-Grained Categorization (FGC)

$$g_0^{a^*} = \mathcal{N}(x_0^a \mid (\mu_0^a)_{k_0^*}, (\Sigma_0^a)_{k_0^*})$$

$$g_1^{b^*} = \mathcal{N}(x_1^b \mid (\mu_1^b)_{k_1^*}, (\Sigma_1^b)_{k_1^*})$$

$$k_0^* = \arg\max_k (\pi_0^a)_k, \quad k_1^* = \arg\max_k (\pi_1^b)_k$$

$$\mu_z = \alpha\,(\mu_0^a)_{k_0^*} + \beta\,(\mu_1^b)_{k_1^*}$$

$$\Sigma_z = \alpha^2\,(\Sigma_0^a)_{k_0^*} + \beta^2\,(\Sigma_1^b)_{k_1^*}$$

Method

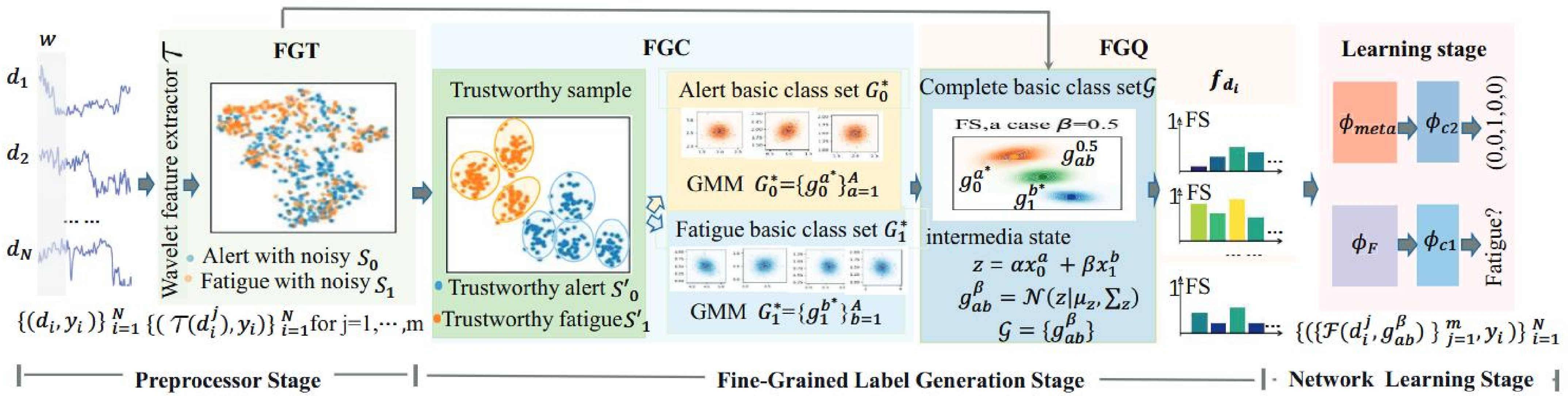


Figure 2: Overview of MDFG. In the preprocessing stage, the wavelet feature extractor $\mathcal{T}$ generates noise-containing FGT samples. In the fine-grained generation stage, we extract base classes from trustworthy samples and then combine the alert base class set G_0^* and the fatigue base class set G_1^* to generate intermediate state base classes. The fatigue score β for $\mathcal{T}(d_i^j)$ is assigned based on the closest base class to that in $\mathcal{G}$. Finally, in the network learning stage, a fatigue detection model is constructed based on the Fatigue Scores (FS) sequence.

Fine-Grained Quantification (FGQ)

$$g_{ab}^{\beta} = \mathcal{N}(z \mid \mu_z, \Sigma_z)$$

β is set to multi-group FS, specifically {0, 0.1, 0.3, 0.5, 0.7, 0.9, 1}
由 alert subtype a + fatigue subtype b 混合出来的“β程度疲劳状态”

→ 造出很多“不同疲劳程度”的模板

Method

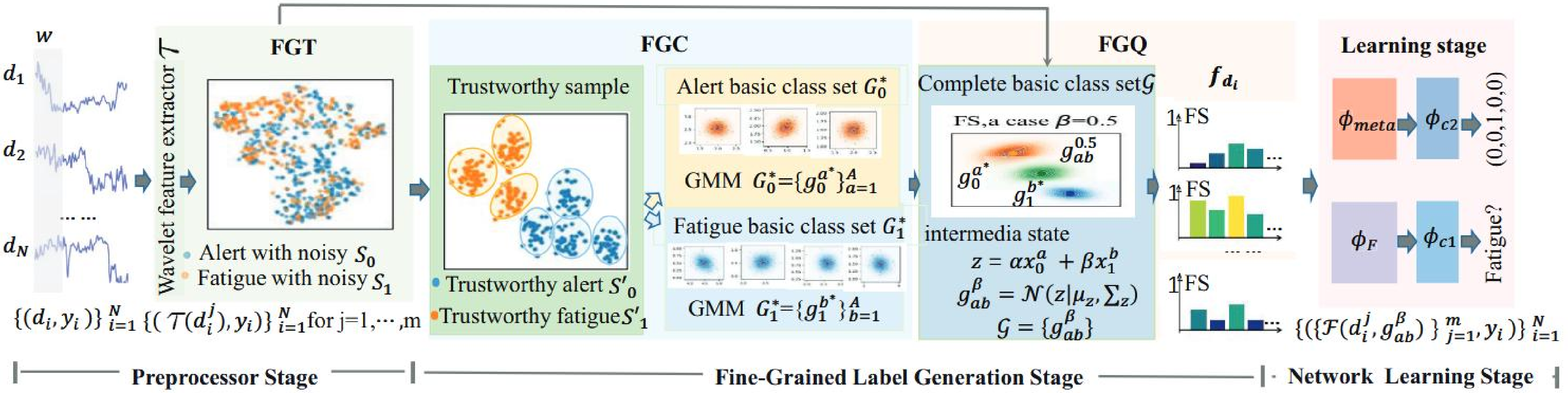


Figure 2: Overview of MDFG. In the preprocessing stage, the wavelet feature extractor $\mathcal{T}$ generates noise-containing FGT samples. In the fine-grained generation stage, we extract base classes from trustworthy samples and then combine the alert base class set G_0^* and the fatigue base class set G_1^* to generate intermediate state base classes. The fatigue score β for $\mathcal{T}(d_i^j)$ is assigned based on the closest base class to that in $\mathcal{G}$. Finally, in the network learning stage, a fatigue detection model is constructed based on the Fatigue Scores (FS) sequence.

Fine-Grained Quantification (FGQ)

$$p(d_i^j | g_{ab}^{\beta})$$

看一个样本最像哪个模板
就把这个模板的程度当作
它的疲劳分数

$$f_{d_i^j} = \mathcal{F}(d_i^j, g_{ab}^{\beta}) \qquad \beta^*, a^*, b^* = \arg\max_{\beta,a,b} p(d_i^j \mid g_{ab}^{\beta})$$

Method

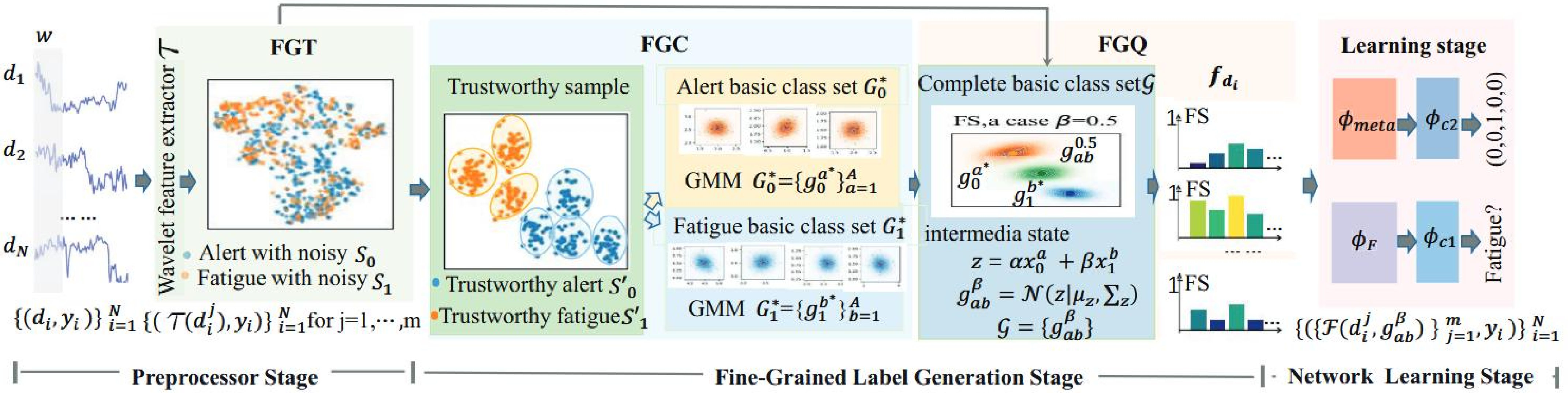


Figure 2: Overview of MDFG. In the preprocessing stage, the wavelet feature extractor $\mathcal{T}$ generates noise-containing FGT samples. In the fine-grained generation stage, we extract base classes from trustworthy samples and then combine the alert base class set G_0^* and the fatigue base class set G_1^* to generate intermediate state base classes. The fatigue score β for $\mathcal{T}(d_i^j)$ is assigned based on the closest base class to that in $\mathcal{G}$. Finally, in the network learning stage, a fatigue detection model is constructed based on the Fatigue Scores (FS) sequence.

$\emptyset_F$： 以alert和fatigue为标签进行学习

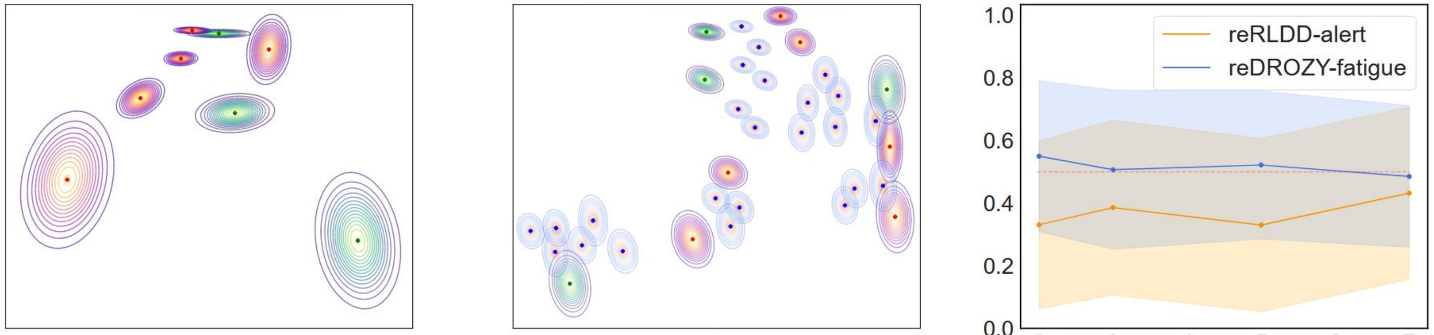
使用相同网络

$\emptyset_{meta}$： 构造正负样本对，学习不同样本之间的相似程度（为了提升模型泛化性）

$$s_{j,l} = \phi_{c2}(\phi_{\text{meta}}(C(\phi_F(f_{d_l}), \phi_F(f_{d_j})))),\ (l \neq j) \quad (9)$$

$$\mathcal{L}_{\text{meta}} = -\frac{1}{N_b} \sum_{l \neq j} y_j' \log(s_{j,l}) + (1 - y_j') \log(1 - s_{j,l}) \quad (10)$$

Experiments



(a) Basic class in reRLDD (b) Synthetic basic class in reDROZY (c) FS statistics for misclassified samples

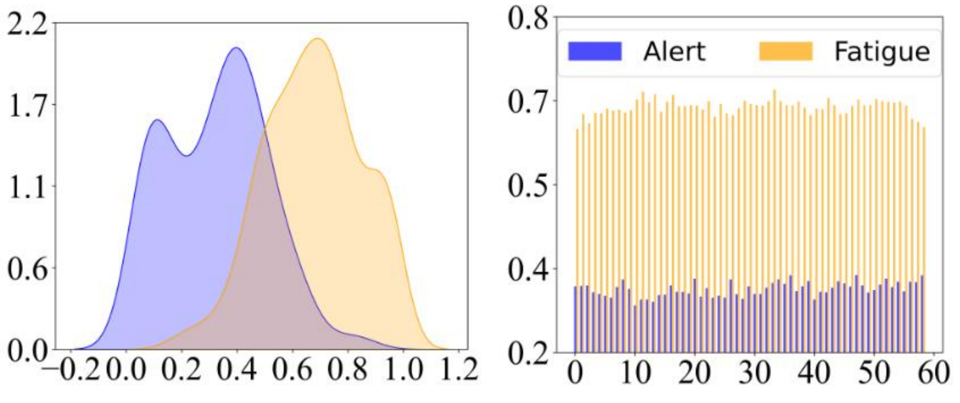
Figure 3: (a) illustrates the base classes extracted from the three alert subtypes and five fatigue subtypes. (b) primarily showcases the TSEN visualization of the 24 synthetic base classes in reDROZY under FS $\beta = 0.5$, These base classes are derived from the base class of four alert and six fatigue. (c) The closer the average FS of misclassified samples is to 0.5, the better the methods' performance. When the Group count is 7, the average FS is closer to 0.5. The blue and orange bands represent the variance.

- 中间状态建模是有效的
- 0.4 和 0.6离 0.5 太近，会导致边界混乱，这个时候状态没有区分度

Multi-group fatigue score β	Group count	reRLDD		reDROZY	
		Acc (Av acc)	F1 (Av F1)	Acc (Av acc)	F1 (Av F1)
[0,1]	2	0.664 (0.608)	0.656 (0.588)	0.706 (0.659)	0.714 (0.665)
[0,0.5,1]	3	0.680 (0.644)	0.670 (0.629)	0.744 (0.723)	0.742 (0.717)
[0,0.3,0.5,0.7,1]	5	0.633 (0.611)	0.609 (0.584)	0.769 (0.736)	0.756 (0.724)
[0,0.1,0.3,0.5,0.7,0.9,1]	**7**	**0.695 (0.655)**	**0.689 (0.639)**	**0.775 (0.730)**	**0.772 (0.719)**
[0,0.1,0.2,0.3,0.4,0.5,0.6,0.7,0.8,0.9,1]	11	0.648 (0.617)	0.636 (0.605)	0.756 (0.691)	0.750 (0.684)

Table 2: Results of different FS β groups. The optimal results and the average results (shown in parentheses) are reported.

Experiments



(a) Density estimation (b) Average FS per window.

Figure 4: FGQ Related Results. (a) Density estimation of reDROZY. The x-axis is fatigue score. (b) Average FS for segments in sliding windows across different states.

对fatigue标签和alert标签分别统计疲劳分数，验证疲劳分数有效性

Method	reRLDD		reDROZY	
	Acc (Av acc)	F1 (Av F1)	Acc (Av acc)	F1 (Av F1)
Eye4feature'16	0.530 (-)	0.448 (-)	0.538 (-)	0.530 (-)
Eye16feature'18	0.604 (-)	0.585 (-)	0.624 (-)	0.611 (-)
FacialUints'19	0.604 (-)	0.585 (-)	0.645 (-)	0.636 (-)
HMLSTM'19	0.567 (0.539)	0.443 (0.433)	0.684 (0.622)	0.757 (0.709)
Lee et al. (eye)'22	0.552 (0.552)	0.343 (0.343)	0.760 (0.697)	**0.838 (0.795)**
Khandare et al.'23	0.523 (0.477)	0.412 (0.123)	0.678 (0.613)	<u>0.806</u> (<u>0.749</u>)
Mehmood et al. (eye)_5fps'24	0.547 (0.503)	0.442 (0.386)	0.744 (0.668)	0.724 (0.659)
Mehmood et al. (eye)_10fps	0.563 (0.525)	0.477 (0.424)	0.744 (0.670)	0.722 (0.663)
Mehmood et al. (eye)_15fps	0.563 (0.514)	0.485 (0.398)	0.719 (0.646)	0.705 (0.645)
iTransformer'24	0.573 (0.561)	0.552 (0.539)	0.754 (0.692)	0.750 (0.696)
TimesNet'23	0.631 (0.498)	0.627 (0.462)	0.754 (0.690)	0.739 (0.684)
TimesURL'24	0.529 (0.511)	0.510 (0.481)	0.684 (0.673)	0.695 (0.684)
MILLET'24	0.573 (0.489)	0.572 (0.456)	0.754 (0.694)	0.731 (0.688)
MDFG (w/o meta)	<u>0.695 (0.655)</u>	<u>0.689 (0.639)</u>	<u>0.775 (0.730)</u>	0.772 (0.719)
MDFG	**0.719 (0.688)**	**0.716 (0.681)**	**0.781 (0.746)**	0.777 (0.737)

Note: (-) indicates the use of SVM in the text without the final average value.

Table 3: Comparison with eye movement-related methods. The best results and the final five-epoch average results (shown in parentheses) are reported. Optimal results are bolded and suboptimal results are underlined.

Experiments

可信阈值超参数实验

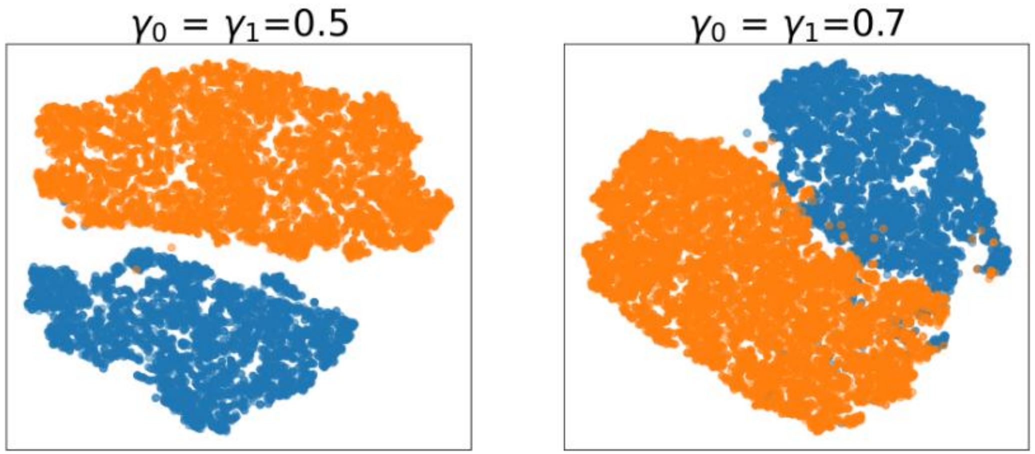


Figure 6: 0.5 is a reliable threshold, with increased alert-fatigue overlap at 0.7. Blue is alert, orange is fatigue.

元学习样本对超参数实验

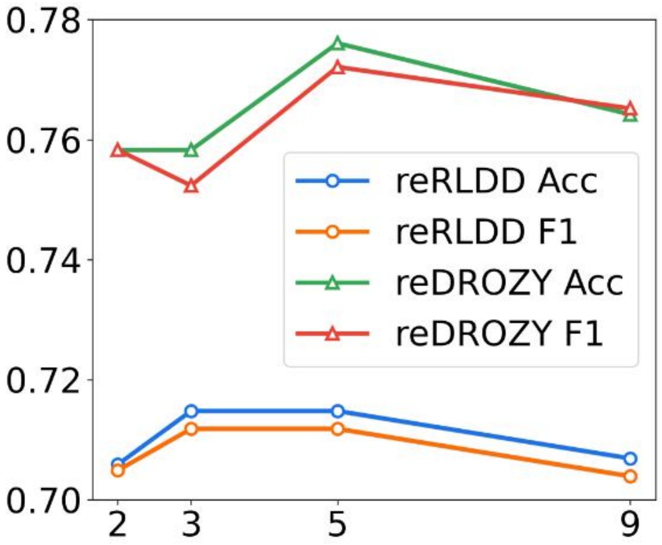


Figure 7: Acc for meta-learning pairwise numbers.

Thanks