KDD2021

Alibaba Group 阿里巴巴集团

MeLL: Large-scale Extensible User Intent Classification for Dialogue Systems with Meta Lifelong Learning

Chengyu Wang[1*], Haojie Pan[1*], Yuan Liu[1], Kehan Chen[1], Minghui Qiu[1], Wei Zhou[1], Jun Huang[1], Haiqing Chen[1], Wei Lin[1], Deng Cai[2]

[1] Alibaba Group [2] State Key Lab of CAD & CG, Zhejiang University

Introduction (1)

✓User Intent Classification: **Text-to-label Classifiers**

- Understanding users’ intents based on the input queries issued by users
- Understanding users’ responses to actions previously taken by the systems

✓Extensible User Intent Classification

- The task number is continuously growing through time

✓Challenges

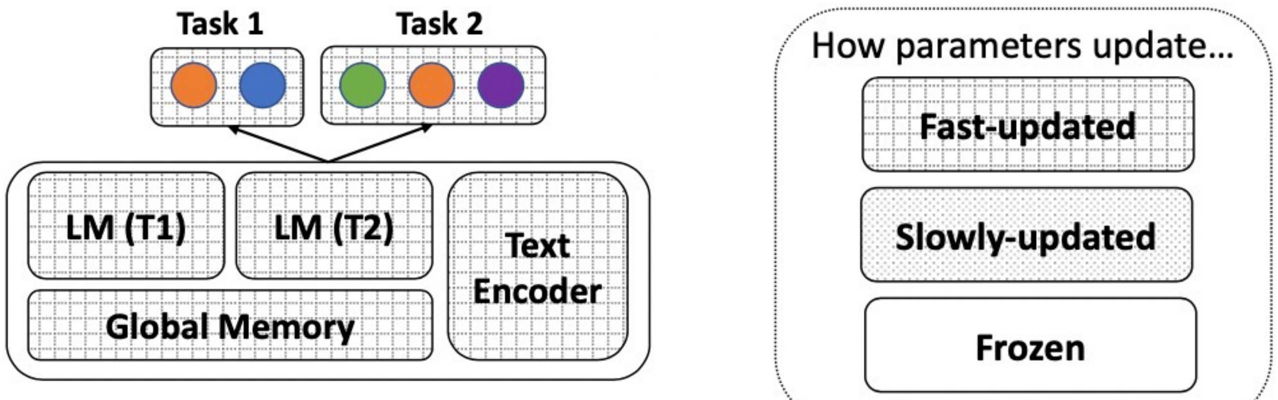
- **Parameter explosion**
- **Catastrophic forgetting**

Introduction (2)

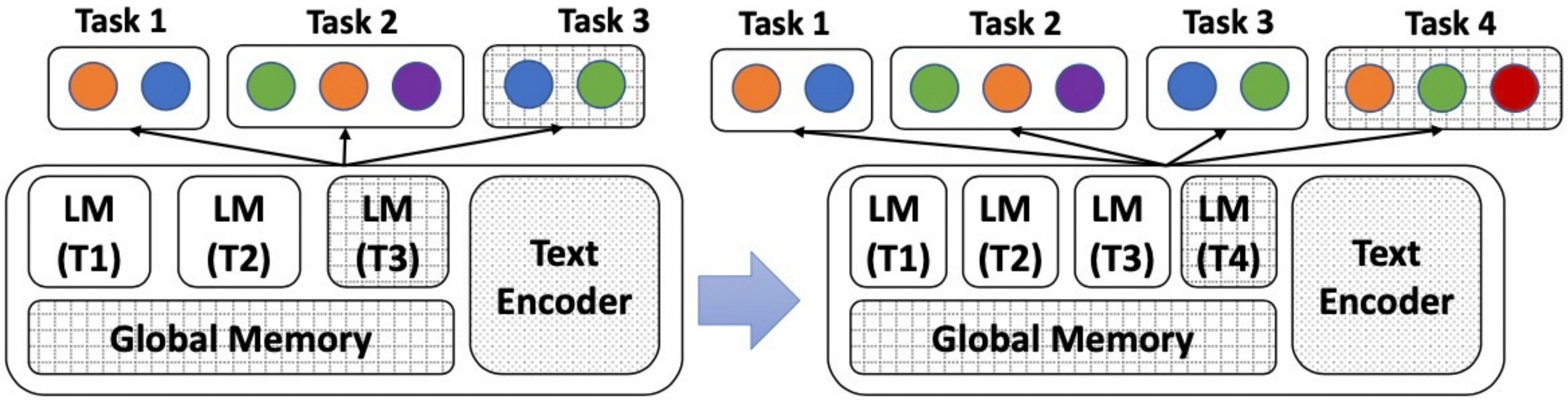
✓Solution: the **Meta Lifelong Learning** (**MeLL**) framework

✓Components

- Text Encoder
- Global Memory
- Local Memories
- Task-specific Layers



a) Initial Learning Stage

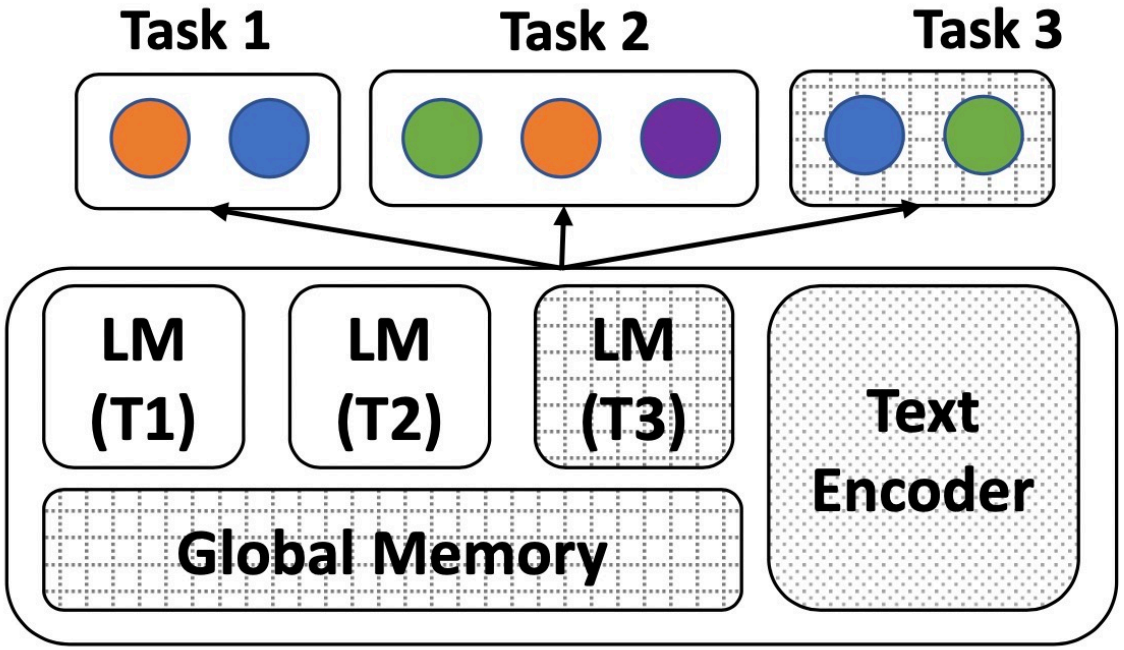


b) Lifelong Learning Stage

Introduction (3)

✓Functionalities of Different Components

- Text Encoder: learning the semantics of input texts (slowly updated)
- Global Memory: storing the class semantics across tasks (fast updated)
- Local Memories: storing the task-specific class semantics (frozen once assigned)
- Task-specific Layers: generating task-specific outputs



Related Work

MeLL: leveraging ideas of both lifelong learning and meta-learning for user intent classification based on pre-trained language models

✓User Intent Classification

✓Lifelong Learning

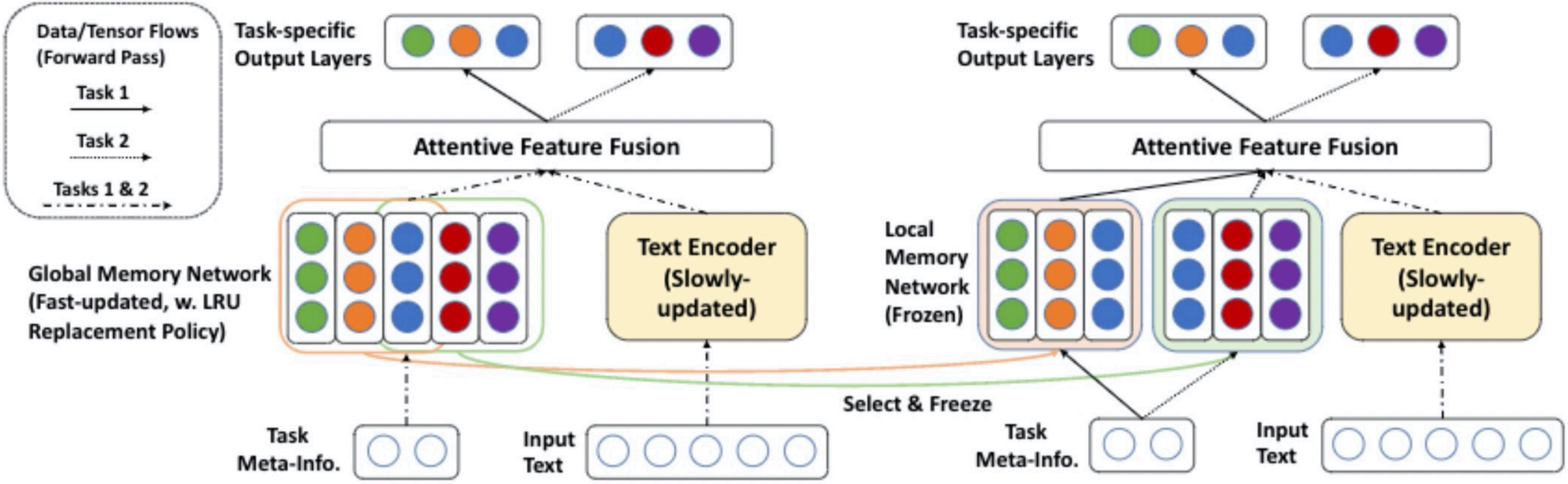
- Solving an unlimited sequence of tasks with the help of previously learned tasks

✓Meta-learning

- Training meta-learners that can adapt to a variety of tasks with little training data available

✓Pre-trained Language Models

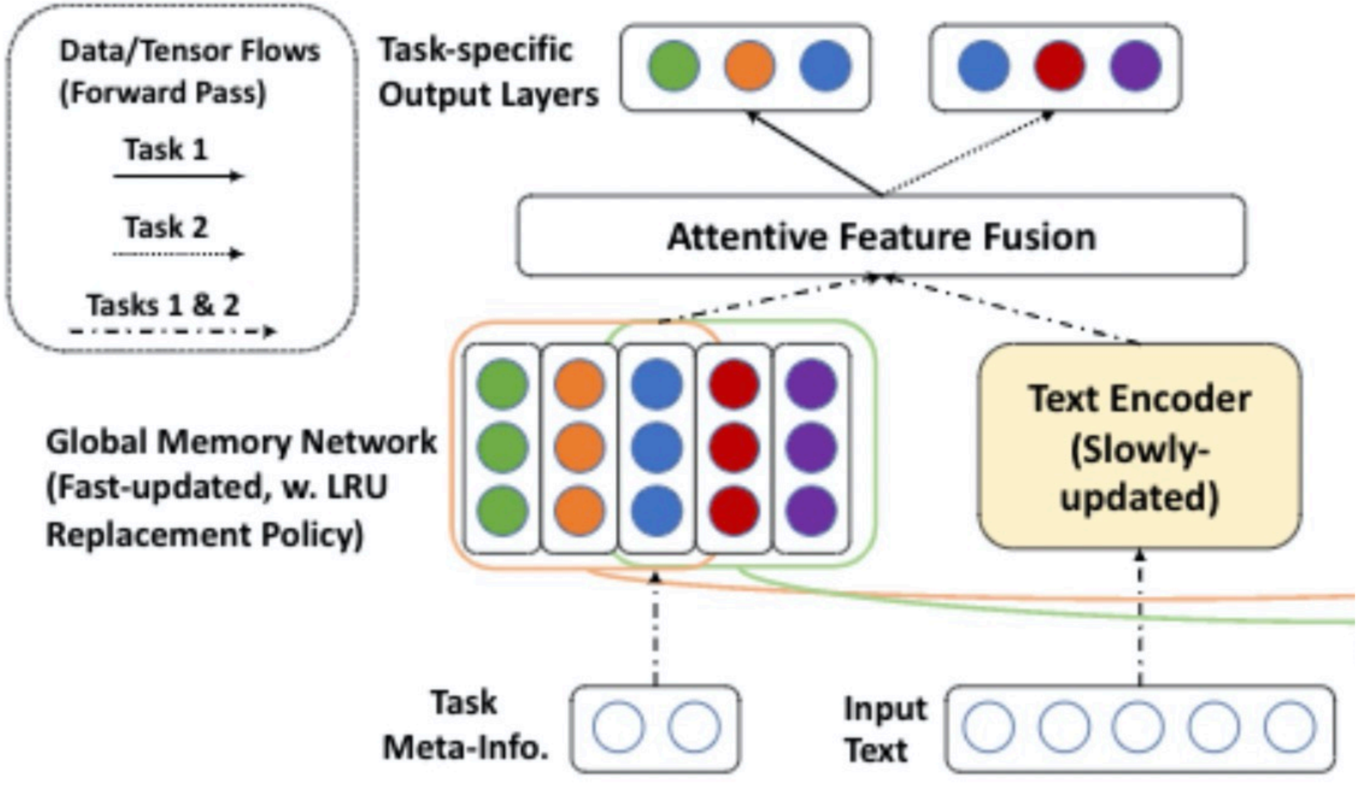
MeLL: Basic Model Structure



a) MeLL (Training Time)

b) MeLL (Inference Time)

MeLL (Training Time)



a) MeLL (Training Time)

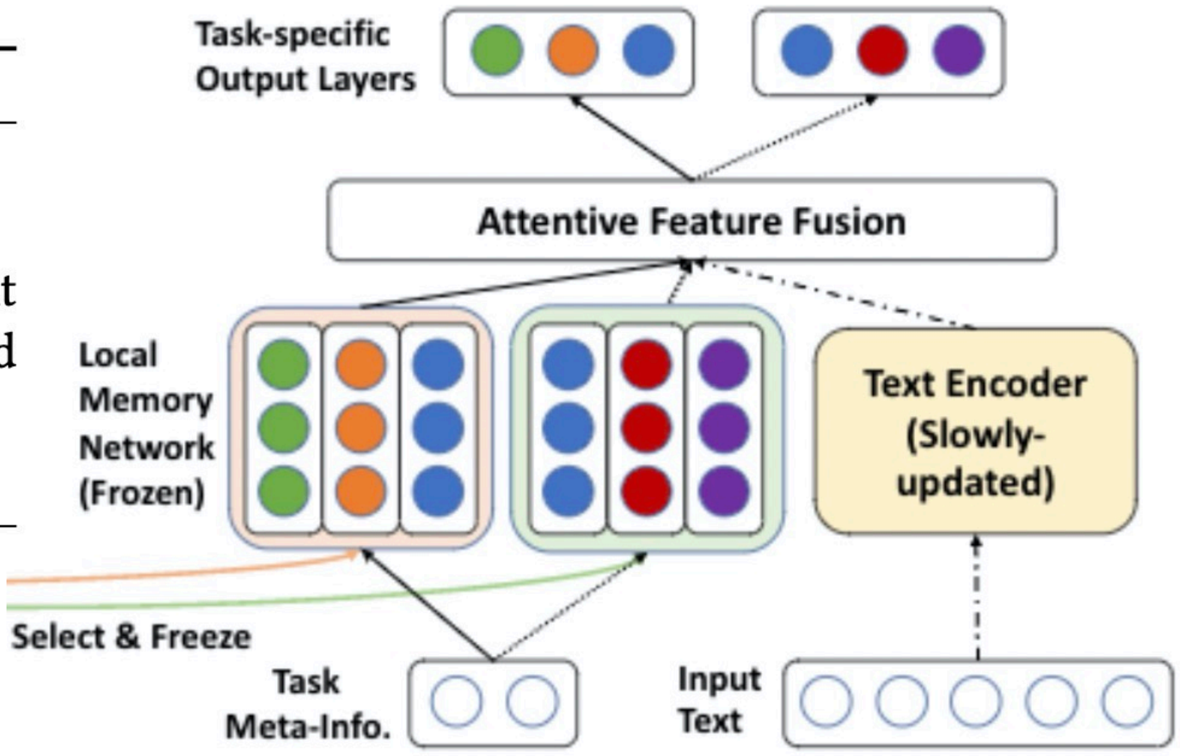
Algorithm 1 MeLL Training Procedure

// Initial Learning Stage
Initialize global memory G based on $\mathcal{D}_1, \mathcal{D}_2, \cdots \mathcal{D}_N$.
while not converge **do**
  Sample a task $\mathcal{T}_n$ from $\mathcal{T}_1, \mathcal{T}_2, \cdots \mathcal{T}_N$.
  Read a batch $\{(x_{n,i}, y_{n_i})\}$ from $\mathcal{D}_n$.
  Run through BERT to obtain representations $\{Q(x_{n,i})\}$.
  Read global memory G with the task meta-info. $\mathcal{Y}_n$ and text representations $\{Q(x_{n,i})\}$ to generate features $\{Att(x_{n,i})\}$ and pass them to the output layer f_n.
  Update parameters of f_n, G and the text encoder by back propagation.
end while
Create local memories $L_1, L_2, \cdots, L_N$ for $\mathcal{T}_1, \mathcal{T}_2, \cdots \mathcal{T}_N$, with all parameters frozen.
// Lifelong Learning Stage (Assume task $\mathcal{T}_j$ arrives, $j > N$.)
Update global memory G based on $\mathcal{D}_j$ w. LRU replacement.
Train the model with a new task-specific output layer f_j and a smaller learning rate on BERT. Parameters of f_n, G and BERT are updated.
Create local memory L_j for $\mathcal{T}_j$ with all parameters frozen.

MeLL (Inference Time)

Algorithm 2 MeLL Inference Procedure

1: Read a batch $\{(x_{n,i})\}$ from an unlabeled dataset of task $\mathcal{T}_n$.
2: Run through BERT to obtain representations $\{Q(x_{n,i})\}$.
3: Read local memory L_n with the task meta-info. $\mathcal{Y}_n$ and text representations $\{Q(x_{n,i})\}$ to generate features $\{Att(x_{n,i})\}$ and pass them to the task-specific output layer f_n.
4: Make predictions $\{\hat{y}_{n,i}\}$ based on $f_n(Att(x_{n,i}))$.



b) MeLL (Inference Time)

Global and Local Memory Networks

✓Global Memory Network

- Each "slot" stores the "centroid" representation for each class.

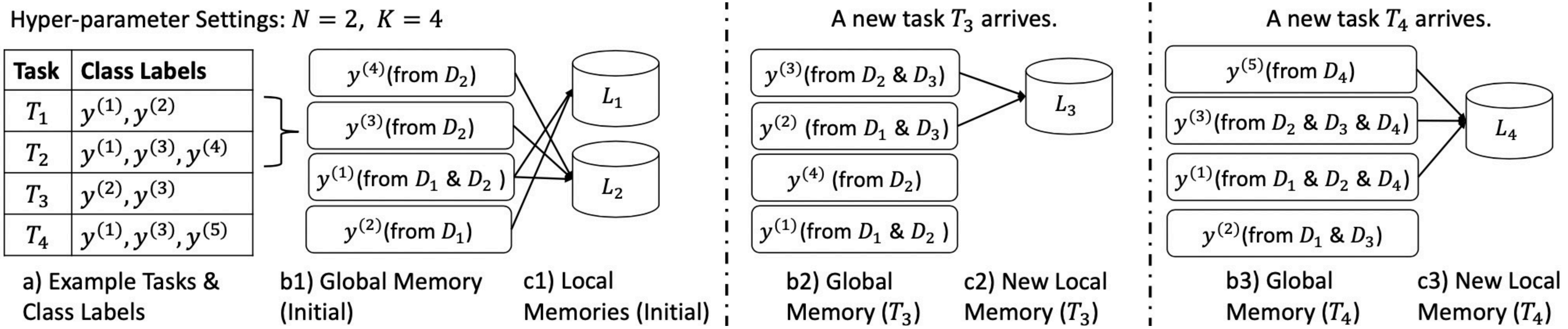
Initial Stage

$$G_N^{(m)} = \frac{1}{|\mathcal{T}^{(m)}|} \sum_{\mathcal{T}_n \in \mathcal{T}^{(m)}} \frac{1}{|\mathcal{D}_n^{(m)}|} \sum_{(x_{n,i}, y_{n,i}) \in \mathcal{D}_n^{(m)}} Q(x_{n,i})$$

Update Rule

$$G_j^{(m)} = (1-\gamma) G_{j-1}^{(m)} + \frac{\gamma}{|\mathcal{D}_j^{(m)}|} \sum_{(x_{n,i}, y_{n,i}) \in \mathcal{D}_j^{(m)}} Q(x_{n,i})$$

- Replacement policy for "slots": Least Recently Used (LRU)



Feature Fusion and Model Output

✓Feature Fusion

- Attentive score $\alpha^{(m)}(x_{n,i}) = \mathrm{softmax}(Q(x_{n,i})^T \cdot G_n^{(m)})$

- Attentive features $Att(x_{n,i}) = Q(x_{n,i}) + \sum_{y^{(m)} \in \mathcal{Y}_n} \alpha^{(m)}(x_{n,i}) \cdot G_n^{(m)}$

Results from BERT encoder

Results from global memory

✓Model Output

- Each task has its own task-specific output layer.

Experiments (1)

✓Datasets

- TaskDialog-EUIC: built from three public query intent classification datasets
- Hotline-EUIC: a real-world e-commerce dataset for response intent classification in hotline agents

✓Experimental Settings

- bert-base-en (uncased) for TaskDialog-EUIC
- roberta-tiny-chinese for Hotline-EUIC

	TaskDialog-EUIC	Hotline-EUIC
#Train.	12,845	90,594
#Dev.	2,569	10,114
#Test	2,569	11,803
#Tasks	90	90
#Base tasks	30	30
#Distinct labels	26	71

Experiments (2)

- Examples of Hotline-EUIC

Domain	Task Description	User Response Intents
Map	Check whether the shop name is correct Check whether the shop is still open	{Yes, No, Other} {Open, Close, Not sure}
Health	Ask about the medication history Ask about the fasting plasma glucose	{1 Year, 1-3 Years, >3 Years} {Normal, Pre-diabetes, Diabetes}
Food takeout	Check if the customer is available to pick up the takeout Satisfaction survey	{ Available, Not available, Deliver as soon as possible } { Satisfied, Slow delivery, Food spilled, Not received }
Express delivery	Check if the customer is available to pick up the delivery Satisfaction survey	{ Available, Not available, Collect the parcels by others } { Satisfied, Slow delivery, Package damaged, Not received }

Experiments (3)

✓Overall Model Performance

Task	TaskDialog-EUIC				Hotline-EUIC			
Results	All tasks		New tasks		All tasks		New tasks	
	Accuracy	F1	Accuracy	F1	Accuracy	F1	Accuracy	F1
MTL (Upper-bound)*	0.9597	0.9590	0.9568	0.9562	0.9788	0.9480	0.9832	0.9523
Single*	0.9006	0.8974	0.9005	0.8969	0.9196	0.8685	0.9239	0.8814
Lifelong-freeze	0.9214	0.9194	0.9015	0.8988	0.9401	0.8798	0.9259	0.8501
Lifelong-seq	0.3140	0.2043	0.3447	0.2455	0.4517	0.3485	0.5272	0.4238
Lifelong-replay*	0.6225	0.5481	0.5485	0.4573	0.8215	0.8260	0.9420	0.8553
MeLL	**0.9379**	**0.9342**	**0.9271**	**0.9224**	**0.9673**	**0.9341**	**0.9675**	**0.9319**

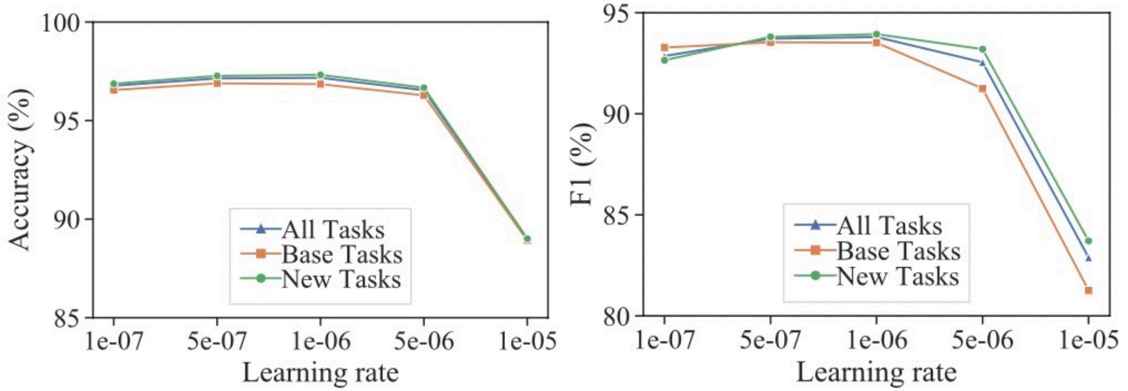
Experiments (4)

✓Ablation Study

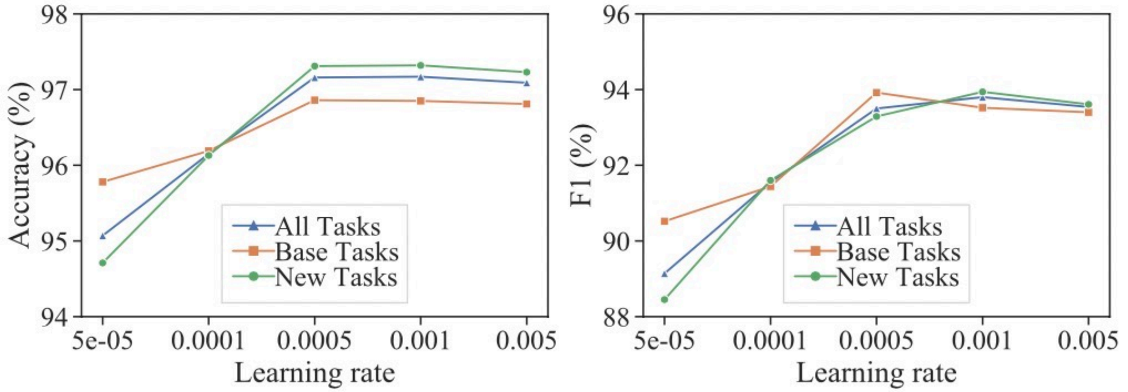
- The meta knowledge plays an important role in overall model performance.

Ablation	F1	Improv. Rate
MeLL	0.9341	N/A
w/o Meta knowledge	0.9178	-1.63%
w/o Slow learner	0.9269	-0.72%
w/o LRU replacement policy	0.9380	+0.39%

✓Parameter Analysis



(a) Accuracy w.r.t the learning rate of the slow leaner. **(b) F1 w.r.t the learning rate of the slow leaner.**



(c) Accuracy w.r.t the learning rate of the fast leaner. **(d) F1 w.r.t the learning rate of the fast leaner.**

Experiments (5)

✓Online Deployment

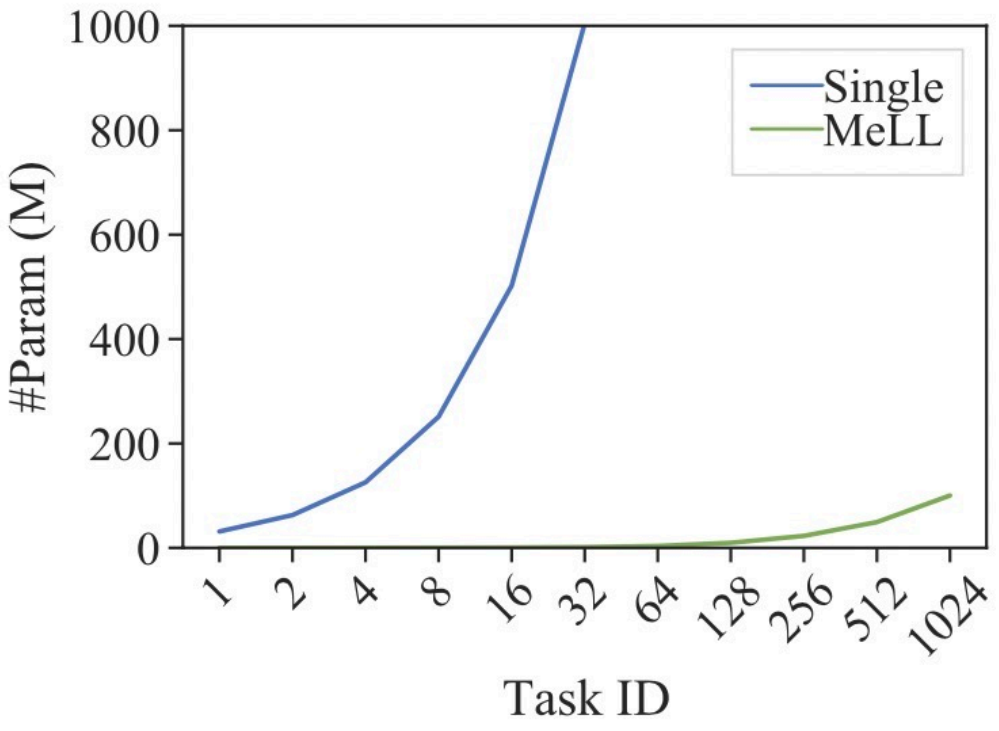
- A/B test on AliMe hotline system
- Online system
  - Task-specific TextCNN models

Method	F1	Relative Improv.
Online system (Single)	0.8359	N.A.
MeLL (w. LRU)	**0.9079**	8.61%

ALIME | 专注服务的会话机器人

✓Salability Analysis

- Number of parameters w. the number of tasks



Conclusion

✓We present the MeLL framework to address large-scale extensible user intent classification.

✓Experiments and online A/B test show that MeLL consistently outperforms strong baselines.

✓Future work:

- How MeLL be employed to solve other tasks and support other applications.

THANKS

--------- Q&A Section --------