
MIDAS: Multi-level Intent, Domain, And Slot Knowledge Distillation for Multi-turn NLU

Yan Li

The University of Sydney
yali3816@uni.sydney.edu.au

So-Eon Kim

Kyung Hee University
sekim0211@khu.ac.kr

Seong-Bae Park

Kyung Hee University
sbpark71@khu.ac.kr

Soyeon Caren Han*

The University of Melbourne
caren.han@unimelb.edu.au

Abstract

Although Large Language Models(LLMs) can generate coherent and contextually relevant text, they often struggle to recognise the intent behind the human user’s query. Natural Language Understanding (NLU) models, however, interpret the purpose and key information of user’s input to enable responsive interactions. Existing NLU models generally map individual utterances to a dual-level semantic frame, involving sentence-level intent and word-level slot labels. However, real-life conversations primarily consist of multi-turn conversations, involving the interpretation of complex and extended dialogues. Researchers encounter challenges addressing all facets of multi-turn dialogue conversations using a unified single NLU model. This paper introduces a novel approach, MIDAS, leveraging a multi-level intent, domain, and slot knowledge distillation for multi-turn NLU. To achieve this, we construct distinct teachers for varying levels of conversation knowledge, namely, sentence-level intent detection, word-level slot filling, and conversation-level domain classification. These teachers are then fine-tuned to acquire specific knowledge of their designated levels. A multi-teacher loss is proposed to facilitate the combination of these multi-level teachers, guiding a student model in multi-turn dialogue tasks. The experimental results demonstrate the efficacy of our model in improving the overall multi-turn conversation understanding, showcasing the potential for advancements in NLU models through the incorporation of multi-level dialogue knowledge distillation techniques.

1 Introduction

Natural Language Understanding (NLU) within the realm of Natural Language Processing (NLP) explores the mechanisms through which computers comprehend human language. Developing a hierarchical semantic framework encompassing domain, intent, and slot has become pivotal in representing the meaning embedded in natural language [1]. We present a conversation example that shows the way of annotation for word-level slots, sentence-level intent, and conversation-level domain from the M2M dataset in Figure 1-(a). The dialogue consists of a total of 9 turns, and each turn includes word-level slot tokens and sentence-level intent information, and the dialogue corresponds to one domain, ‘restaurant’.

Large Language Models (LLM) have received lots of attention in generating human-like text based on user prompts. However, they are still limited when it comes to deeper communication and diverse

*Corresponding Author

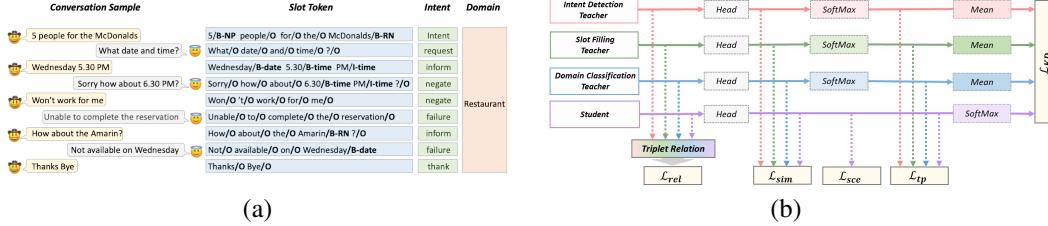


Figure 1: (a) An example of conversations with word-level slots, sentence-level intents, and conversation-level domain annotation from M2M. B-NP (B-Number of People), B-RN (B-Restaurant Name), O (Others) (b) The multi-level teacher knowledge distillation framework for the multi-turn NLU task. Note that we applied three multi-level teachers: Intent Detection, Slot Filling, and Domain Classification. In this framework, we conduct diverse Loss objectives, including \mathcal{L}_{rel} , \mathcal{L}_{sim} , \mathcal{L}_{sce} , \mathcal{L}_{tp} and \mathcal{L}_{KD} , which represent relation loss, similarity loss, student cross-entropy loss, teacher prediction supervise loss, and Kullback-Leibler Divergence loss, respectively.

Model	Year	Token (Slot)	Sentence (Intent)	Document (Domain)	Dialogue Type	Joint Integration
SeqSeq Liu and Lane [2]	2016	○	○	×	Single-Turn	BiRNN + Attention
SDEN Bapna et al. [3]	2017	○	○	○	Multi-Turn	BiRNN + Memory Network
Slot-Gated Goo et al. [4]	2018	○	○	○	Single-Turn	BiLSTM + Slot Gate
BLSTM+attention Tingting et al. [5]	2019	○	○	×	Single-Turn	BiLSTM + Attention
STD Jiang et al. [6]	2021	○	○	○	Single-Turn	Transformer + One-teacher KD
SDJN Chen et al. [7]	2022	○	○	○	Single-Turn	BiLSTM + self KD
XAI Attention Gunaratna et al. [8]	2022	○	○	○	Multi-Turn	eXplainable AI
Tri-level JNLU Weld et al. [9]	2023	○	○	○	Multi-Turn	Cross Transformer
Ours	2024	○	○	○	Multi-Turn	Multi-teacher KD

Table 1: Summary of existing joint NLU models and ours. Token, Sentence, and Document columns indicate whether the relevant information is used for joint integration. KD refers to knowledge distillation. The complete set of summary tables is detailed in Appendix A

key information². Hence, we investigate how to improve the state-of-the-art existing NLU techniques. While existing NLU literature predominantly concentrates on single-turn utterances within a single domain, recent advancements in multi-turn datasets have paved the way for annotations at the dialogue level, spanning across diverse domains. Interpreting more extended and intricate conversations with multiple turns necessitates understanding the ongoing context and retaining previously gathered information. Traditional NLU involves mapping single utterances to a dual-level semantic structure, encompassing sentence-level intent and word-level slot labels. With real-life conversations extending across multiple turns, there is an evident demand for research incorporating dialogue history, as demonstrated by improved performance through dialogue context. The challenge extends beyond dual-level understanding to encompass a three-level comprehension: sentence-level intent, word-level slot, and conversation-level domain classification. However, researchers encounter challenges in handling all aspects of multi-turn dialogue conversations through a single unified NLU model, due to computational complexity and a lack of distillability of multi-level knowledge.

Addressing this need, our paper introduces a novel multi-level multi-teacher knowledge distillation model to enhance NLU understanding in multi-turn dialogues, leveraging diverse levels of knowledge embedded in these datasets. Notably, our model stands as the pioneering approach in multi-teacher knowledge distillation, catering to distinct facets of knowledge within a dialogue. To achieve this, our approach involves the construction of teachers at different levels, specifically focusing on sentence-level intent detection, word-level slot filling, and conversation-level domain classification. We fine-tune these multi-level teachers to acquire the relevant knowledge and combine these to educate the student model in dialogue tasks facilitated by novel multi-level teacher loss functions.

There are two major contributions: First, we introduce a novel multi-level, multi-teacher knowledge distillation model to enhance multi-turn NLU. It outperforms across widely-used multi-NLU datasets, producing superior performance in all intent detection, slot filling, and domain classification, even compared with the LLMs. Secondly, we introduce multi-level teacher loss functions, shedding light on their impact within the multi-teacher knowledge distillation and guiding a student model.

²We tested NLU benchmarks with several LLMs, including LLaMa2, Gemma, and GPT3.5, visualised in Appendix G.1 and G.2

2 Related works

There is a large body of NLU modelling literature, and we briefly introduce the joint NLU models and knowledge distillation models in this section. A more detailed summary of joint NLU models and our model can be found in Table 1.

Natural Language Understanding Early investigations aimed to tackle various Natural Language Understanding tasks by developing models for slot filling and intent detection separately. It has now become commonplace to enhance joint models using transfer learning [10, 11]. One notable strategy involves fine-tuning a language model [12–14] to address the limited generalisation capability caused by insufficient data and leverage high-quality representations from the language model. In these studies, the majority classify intent through the [CLS] representation and slots through each token representation [15, 16]. Another approach to transfer learning involves training a model through knowledge distillation, where a smaller model, known as the student model, is trained to mimic the behaviour of a larger, more complex model, referred to as the teacher model. The predominant method is self-distillation, where both the teacher and student models share the same structures [7, 17]. However, those models mainly focused on single-turn dialogue or adopted only one teacher model. As real-life conversations involve multiple turns, there is a growing need for research that incorporates dialogue history. In multi-turn dialogues, it has been demonstrated that encoding the dialogue history enhances performance [3, 9, 18, 19]. Hence, our model is the first method for multi-teacher knowledge distillation to teach different aspects of knowledge in dialogue. As NLU consists of multiple tasks such as intent classification and slot filling, it is more appropriate to train the student model using a specialised teacher for each task. To the best of our knowledge, there have been no attempts to employ multi-teacher distillation for multi-turn-based NLU.

Knowledge Distillation Knowledge Distillation (KD) defines a learning approach involving using a well-trained network of teachers to guide the training of a student network for various tasks. Early KD transfers knowledge from one teacher to one student model [20]. Multi-teacher KD, inspired by ensemble learning, aims to enhance performance by incorporating knowledge from multiple teacher models into a student model [21–28]. It was common for KD to use multiple teachers to learn the same domain, regardless of whether the teacher had the same or a different architecture. Recent methods have been proposed in which each teacher learns a different domain and imparts knowledge to the student [29, 30]. Additionally, methods for learning different modalities have also been proposed, which are classified into two types: the teacher and student learn different modalities, a concept known as cross-modality [31, 32] and the teacher learns different modalities, and the student receives all modalities [33].

3 MIDAS: Multi-level intent, domain, and slot knowledge distillation for multi-turn NLU

We propose a new multi-level dialogue teacher knowledge distillation framework, MIDAS, that trains the student model S with multi-level teachers to enhance the Natural Language Understanding (NLU) capabilities. Note that we have three multi-level dialogue knowledge teachers, including intent detection, slot filling and domain topic classification. To achieve this, we initially construct teachers with distinct levels of dialogue knowledge, denoted as $T = \{T_{ID}, T_{SF}, T_{DC}\}$, where T is the set of teacher models, and ID , SF , and DC correspond to Intent Detection, Slot Filling, Domain Classification tasks. Then, we fine-tune the teacher models T to acquire knowledge from each task. Finally, a combination of all three multi-level teachers T is employed to instruct the student model S in dialogue tasks using our newly proposed multi-teacher knowledge loss objectives. The comprehensive architecture is depicted in Figure 1-(b).

3.1 Multi-Level teacher construction

We first construct the teachers of different dialogue document component understanding levels, including word-level slot, sentence-level intent, and conversation-level domain knowledge. The inputs for all teachers consist of utterances from each turn in dialogue datasets, denoted by $X^i = x_1^i, x_2^i, \dots, x_l^i$, where X^i represents the i^{th} utterance in the entire dataset, l is the length of the utterance, and x_l^i signifies a word in the utterance.

Word-level slot filling teacher T_{SF} predicts the slot type for each word, providing knowledge to the student model about key slots in the dialogue. The output of T_{SF} is denoted as $\hat{Y}_{SF}^i = \hat{Y}_{SF,1}^i, \hat{Y}_{SF,2}^i, \dots, \hat{Y}_{SF,l}^i$, representing the predicted slot types for each word, where $\hat{Y}_{SF,l}^i \in 0, 1, \dots, k_{SF} - 1$, and k_{SF} is the number of slot types.

Sentence-level intent detection teacher T_{ID} predicts the intent of the utterance, aiding the student model in comprehending the overall intent of each turn. The prediction of T_{ID} is symbolised as \hat{Y}_{ID}^i , where $\hat{Y}_{ID}^i \in 0, 1, \dots, k_{ID} - 1$, and k_{ID} represents the number of intents in the dataset.

Conversation document-level domain classification teacher T_{DC} forecasts the dialogue’s domain, providing knowledge to classify it and understand its background knowledge. The prediction from T_{DC} is indicated as \hat{Y}_{DC}^i , where $\hat{Y}_{DC}^i \in 0, 1, \dots, k_{DC} - 1$, and k_{DC} denotes the number of domains in the dataset.

Using these three levels of teachers, our objective is to instruct the student model to comprehend dialogues from multiple perspectives, incorporating word-level slot knowledge, sentence-level intent, and document-level domain background knowledge. By doing so, we enhance the student model’s grasp of dialogues across various levels. There are two primary reasons for utilising multi-level dialogue knowledge teachers to train a student. First, individually deploying a pre-trained model for each task consumes more computational resources, and some machines may not support running multiple pre-trained models. Instead, the knowledge distillation process leads to more robust models and is resistant to adversarial attacks. Incorporating soft targets from the teacher model can help the student model learn smoother decision boundaries. Secondly, we posit that diverse levels of knowledge derived from multi-turn conversation understanding datasets can enhance the comprehension of each specific natural language understanding task, surpassing the benefits of learning from single-level dialogue knowledge. Note that we use pre-trained models as the foundational structure for our teachers. After experimenting with various backbones, we determined that BERT yields one of the best results overall, as detailed in Section 5.2. These pre-trained models undergo fine-tuning using specific data for each level, resulting in distinct teachers with expertise in intent detection, slot filling, and document classification. Pre-trained models, having been trained on extensive text data, exhibit the capacity to transfer knowledge effectively. Ultimately, we leverage the collective knowledge of these refined teachers to train the student model comprehensively.

3.2 Multi-level Teacher Fine-tuning

We perform separate fine-tuning of pre-trained models on intent detection (ID), slot-filling (SF), and domain classification (DC) tasks. This yields multi-level teachers, T_{ID} , T_{SF} , and T_{DC} respectively, corresponding to sentence-level, word-level and sentence-level knowledge, respectively. Each pre-trained model specialises in learning knowledge at one specific level from the dialogue datasets, resulting in teachers possessing different levels of dialogue document component understanding. It’s important to note that each teacher focuses on one level of dialogue knowledge. This approach is motivated by two factors. First, learning knowledge from a single task is less complex than incorporating knowledge from all tasks, simplifying the fine-tuning of pre-trained models. Secondly, instead of burdening a single model with the challenge of mastering knowledge from all aspects of dialogues, each teacher focuses on a specific level of understanding, such as word-level slot filling, sentence-level intent detection, or document-level domain classification. For each task, we consolidate data from two datasets (MultiWOZ and M2M) by merging split and corresponding label sets. For example, the training set for fine-tuning includes data from both datasets. We apply cross-entropy loss and fine-tune the pre-trained models for a fixed number of epochs, utilising the checkpoint from the last epoch as the teacher model. The process is described as follows:

$$\begin{aligned} X_{j,train} &= X_{j,train}^1, \dots, X_{j,train}^{N_{M2M}}, X_{j,train}^1, \dots, X_{j,train}^{N_{MultiWoz}}, \\ \mathcal{L}_{tce} &= \text{cross_entropy}(T_j(X_{j,train}), Y_j), \\ j &\in \{DC, ID, SF\}, \end{aligned}$$

where N_{M2M} and $N_{MultiWoz}$ are the number of training samples, and Y_j is the ground truth.

Algorithm 1 Triplet Relations

Input: The hidden states of the batch data from the teachers $H_t = \{h_1^1, h_1^2, \dots, h_1^n, \dots, h_j^1, \dots, h_j^n\}$, the hidden states of the batch data from the student $H_s = \{h_s^1, h_s^2, \dots, h_s^n\}$, the teacher model set $T = \{T_1, T_2, \dots, T_j\}$

Parameter: Distance function \mathcal{F}_D

Output: The batch size of triplet relations \mathcal{T}

```

1: Let  $i = 0, \mathcal{T} = \emptyset$ .
2: for  $i < n$  do
3:   Randomly select three samples from the batch and label their
      indexes in the batch as  $r1, r2, r3$ .
4:   Treat the sample indexed  $r1$  as the anchor,  $r2$  as the positive
      sample,  $r3$  as the negative sample.
5:   Let  $l = 0, flag = 0$ 
6:   for  $l < j$  do
7:      $d_{1,2} = \mathcal{F}_D(h_l^{r1}, h_l^{r2}), d_{1,3} = \mathcal{F}_D(h_l^{r1}, h_l^{r3})$ 
8:     if  $d_{1,2} > d_{1,3}$  then
9:        $flag += 1$ 
10:    else
11:       $flag -= 1$ 
12:    end if
13:     $l += 1$ 
14:  end for
15:  if  $flag > 0$  then
16:    Swap the labels of  $h_l^{r2}$  and  $h_l^{r3}$ .
17:  end if
18:   $i += 1$ 
19:   $\mathcal{T} += [h_s^{r1}, h_s^{r2}, h_s^{r3}]$ 
20: end for
21: return  $\mathcal{T}$ 

```

3.3 Multi teachers knowledge distillation

Following the acquisition of multi-level teachers T , we employ a blend of these teachers to instruct the student model S through multi-teacher knowledge distillation. The combination of teachers comprises different levels, such as BERT-Base ID, BERT-Base SF, BERT-Base DC BERT-Base ID, RoBERTa-Base DC, and LLaMa2-7b SF. The student model undergoes separate training for each task, enabling it to grasp the intricacies of individual tasks with the assistance of diverse levels of dialogue knowledge.

We delve into the exploration and introduction of five distinct loss functions to assess their efficacy within the MIDAS. We incorporate Kullback–Leibler Divergence loss and Student Cross Entropy loss, widely utilised knowledge distillation tools. Furthermore, with MIDAS, we explore three specific types of losses tailored for multi-level teacher integration. These encompass relation loss, similarity loss, and teacher-prediction supervised loss, each designed to enhance the learning dynamics in the context of multi-level knowledge distillation.

Kullback–Leibler divergence(KD) loss \mathcal{L}_{KD} : We compute the KD loss [34] by comparing the mean probabilities generated by the combination of teacher models with the probabilities derived from the student model. We use KD loss to align the prediction probability distributions between the student model and the teacher models, facilitating the learning of the student model from multiple teachers.

$$\mathcal{L}_{KD} = KLDivLoss\left(\frac{1}{n_T} \sum_j^{n_T} P_j, P_s\right), P_j = softmax(T_j(X)), P_s = softmax(S(X)),$$

where n_T is the number of teachers.

Student cross entropy(SCE) loss \mathcal{L}_{sce} : This loss function is computed by comparing the student model’s predictions with the ground truth of each task. By employing the cross-entropy loss, the student model receives direct supervisory signals from the ground truth, aiding in its learning process.

$$\mathcal{L}_{sce} = cross_entropy(v_s, Y_{true}), v_s = S(X),$$

where v_s represents the student logit.

Relation loss \mathcal{L}_{rel} : During training, for each batch of data, triplets are randomly generated, and the internal relations of the triplets are determined by aggregating the votes from the combination of teacher models. Employing TripletMarginLoss, the student model learns internal relations among the batch data, aligning its understanding with that of the teacher models and ensuring a consistent perspective on the dataset.

$$\mathcal{L}_{rel} = \frac{1}{N} \sum_i^N TripletMarginLoss(\mathcal{T}_i),$$

where N is the batch size, and triplet \mathcal{T} is generated by and articulated in Algorithm 1.

Similarity loss \mathcal{L}_{sim} : The similarity loss is computed by maximising the logits similarities between the student model and teacher models. With this, the student model can learn the knowledge from the

teacher models in the feature space, not only the prediction probabilities. The loss equation is:

$$\mathcal{L}_{sim} = - \sum_j^{n_T} \mathcal{F}_{sim}(v_j, v_s), v_j = T_j(X), v_s = S(X) \quad (1)$$

Here, \mathcal{F}_{sim} denotes the similarity function, and v_j represents the teacher logit.

Teacher prediction supervised loss \mathcal{L}_{tp} : In addition to utilising the ground truth for each task, we incorporate the predictions made by the teacher models as pseudo-labels to facilitate the training of the student model. We employ the probabilities assigned by the teacher models for each class, ensuring that the student comprehensively acquires the knowledge embedded in the teacher models.

$$\mathcal{L}_{tp} = \sum_j^{n_T} cross_entropy(v_s, P_j),$$

We experiment with diverse combinations of the aforementioned loss functions to assess their impact on the student’s performance across various datasets and NLU tasks. The summary of the loss function is described in Appendix B and the detailed result analysis can be found in Section 5.4.

4 Experimental setup

4.1 Datasets and Baselines

In this paper, we focus on multi-turn dialogue analysis in the dialogue state tracking (DST) domain, which consists of all three natural language understanding tasks, including intent classification, slot filling, and domain (topic) classification. Following by [9], we utilise two widely used benchmark datasets in multi-turn dialogue Natural Language Understanding (NLU): Multi-Domain Wizard-of-Oz 2.2 (MWOZ) and Machines Talking To Machines (M2M), especially used as benchmarks in the DST field. Details for each dataset are described in Appendix C.

Due to the limited number of baselines available for Multi-turn Dialogue Understanding, we initially adopted the three published results as baselines [2, 4, 9]. Additionally, we fine-tuned the pre-trained language models commonly used in the NLU domain. **BERT-Base**³ is representative transformer encoder-based language model. **RoBERTa-Base**⁴ builds on BERT and adjusts hyperparameters by eliminating the next-sentence prediction objective. It also trains with larger mini-batches and learning rates. **ALBERT-Base**⁵ is a BERT-based model that demonstrated superior performance by reducing the model size with factorised embedding parameterisation and cross-layer parameter sharing. **SeqSeq** [2] is an RNN with attention mechanisms, designed for the joint tasks of ID and SF. **Slot-Gated** [4] introduces a slot gate to capture the relationship between intent and slot, aiming to improve semantic understanding through global optimisation. **Tri-level JNLU** [9] is a pioneering model that incorporates domain information in the joint modelling of ID and SF.

4.2 Metrics and implementations

This paper evaluates the performance of baseline models and MIDAS in all three multi-turn dialogue tasks, including ID, SF, and DC for each dataset. Following by [2, 4, 9], the metrics for each task are shown as follows: *Accuracy* for ID and DC and *F1 score* for SF. Accuracy is the most commonly used metric for **Intent Detection (ID)** as determining the intent of an utterance is typically framed as a classification task. Accuracy is calculated as the ratio of correct predictions to the total number of tests. **Domain Classification (DC)** also employs accuracy as it is a classification task. On the other hand, **Slot Filling (SF)** employs F1 score. F1 score is directed towards assessing the prediction effectiveness for slot tokens. It computes an F1 score for each class and determines the token-based micro-averaged F1 score across all classes.

We introduce some implementation details in this section and the complete details in Appendix H. For **Multi-teacher fine-tuning**, we use BERT-Base, RoBERTa-Base and LLaMA2-7b⁶ as the

³<https://huggingface.co/bert-base-uncased>

⁴<https://huggingface.co/roberta-base>

⁵<https://huggingface.co/albert-base-v2>

⁶<https://huggingface.co/meta-LLaMA/LLaMA-2-7b>

teacher backbones and fine-tune them on each task. For fine-tuning LLaMa2-7b, we adopt an unmask strategy used in [35]. We use AdamW [36] and CrossEntropy loss to fine-tune the pre-trained models for 3 epochs. The learning rate is 5e-5 and is warm-uped linearly from 0 to 5e-5 during the first 10% training steps. The batch size is 32. For **Multi-level Distillation**, we use AdamW and the aforementioned losses to train the student with multi-level teachers. We use Squared Euclidean distance in algorithm 1 and cosine similarity in equation 1. For the combination of these losses, we sum them without any weight. We use the same optimiser, learning rate, warm-up strategy, and batch size as the one used in teacher fine-tuning, and use a vanilla Transformer encoder as a student.

5 Results

5.1 Overall performance

We compare MIDAS with fine-tuned PLM baselines and published pioneering model results for two mainstream multi-turn natural language understanding tasks, Intent Detection and Slot Filling, with the same evaluation setup. Table 2 shows that MIDAS remarkably outperforms other baselines. To demonstrate the improvement achieved through MIDAS, we conduct experiments utilising two widely recognised multi-turn dialogue understanding datasets, MWOZ and M2M. Note that all baselines and MIDAS are fine-tuned for each of the two tasks individually. As detailed in Section 3.1, our approach involves the utilisation of pre-trained models, BERT or RoBERTa, for the fine-tuning of our three multi-level teacher models. These teachers encompass Intent Detection (ID), Slot Filling (SF), and Domain Classification (DC). It is important to highlight that MIDAS undergoes knowledge distillation from three distinct multi-level teachers, each specialising in sentence-level intent, word token-level slot, and conversation-level domain topic. Thus, Table 2 shows the results *MIDAS (BERT)* and *MIDAS (RoBERTa)* that all teachers are constructed using either the BERT or RoBERTa architecture. Two versions of MIDAS exhibit superior performance across both datasets, presenting outstanding outcomes with a slot-filling error rate below 2%. While the RoBERTa-Base model demonstrates superiority in MWOZ, the BERT-Base model excels in M2M. What should be noted is the performance difference between these models is not substantial, with both consistently outperforming other baseline models. In Intent Detection (ID) and Slot Filling (SF) tasks, MIDAS showcases notably higher performance compared to baselines. We also conduct experiments on the Domain Classification (DC) task with the same datasets to better compare the differences between MIDAS and other PLM baselines. However, while surpassing BERT-Base and ALBERT-Base, the performance difference is marginal. We assume that this discrepancy is attributed to the small number of the domain class. In contrast to other baseline models, Seq2Seq and Slot-Gated lack a structure incorporating domain information, making them unable to assess domain classification performance.

Overall, the observation highlights that bolstering multi-level conversation knowledge substantially improves the comprehension of each Natural Language Understanding (NLU) task. Specifically, enhancing results in intent detection is achievable by refining a student model through the distillation of multi-level knowledge, encompassing sentence-level intent, word-level slots, and conversation-level domain knowledge. The following two sections (Sections 5.2 and 5.3) delve into a more comprehensive exploration of multi-level teacher models and the combination of multi-level teachers.

5.2 Effect of pretrained model for teachers

We then evaluate the efficacy of different pre-trained models for our multi-level teachers. As detailed in Section 5.1 and illustrated in Table 2, we employed all three multi-level teachers (ID, SF, and DC)

	ID		SF		DC			ID		SF		DC	
	MWOZ (ACC)	M2M (ACC)	MWOZ (F1)	M2M (F1)	MWOZ (ACC)	M2M (ACC)		MWOZ (ACC)	M2M (ACC)	MWOZ (F1)	M2M (F1)	MWOZ (ACC)	M2M (ACC)
MIDAS (BERT)	0.8464	0.9427	0.9928	0.9850	0.8780	0.8952	ID-only	0.8406	0.9366	0.8590	0.9684	0.7977	0.7159
MIDAS (RoBERTa)	0.8502	0.9377	0.9928	0.9813	0.8816	0.8945	SF-only	0.8310	0.9377	0.9619	0.9718	0.2425	0.8930
MIDAS (LLaMa)	0.8403	0.9392	0.9912	0.9833	0.8702	0.8804	DC-only	0.8408	0.9321	0.8888	0.9534	0.6330	0.8915
MIDAS (Mixed 1)	0.8472	0.9411	0.9839	0.9745	0.8808	0.8929	ID+SF	0.8422	0.9399	0.9924	0.9835	0.8760	0.8939
MIDAS (Mixed 2)	0.8473	0.9401	0.9928	0.9764	0.8769	0.8925	ID+DC	0.8400	0.9292	0.9923	0.9848	0.8756	0.8929
							SF+DC	0.8376	0.9416	0.9923	0.9825	0.8760	0.8940
							ID+SF+DC	0.8464	0.9427	0.9928	0.9850	0.8780	0.8952

(a)

(b)

Table 3: (a) The performance based on the type of teacher models. The *MIDAS (BERT)* and *MIDAS (RoBERTa)* are identical to those presented in the table 2 whose all teachers are either BERT or RoBERTa. *MIDAS (LLaMa)* refers to the outcome of utilising the LLaMa2-7b model as teacher models of all tasks. The *MIDAS (Mixed 1 and 2)* represents the outcome of mixed type teacher combination; Mixed 1: BERT (ID), LLaMa (SF) and RoBERTa (DC); Mixed 2: BERT (ID), RoBERTa (SF) and RoBERTa (DC). (b) The performance based on the type of teacher models. The first column indicates the type of teacher used. For example, *ID+SF+DC* uses all intent classification, slot filling, and domain classification teachers, while *ID-only* uses only the intent classification teacher. Only BERT is utilised as the teacher model. Results using RoBERTa are presented in Appendix D.

\mathcal{L}_{KD}	\mathcal{L}_{sce}	\mathcal{L}_{sim}	\mathcal{L}_{rel}	\mathcal{L}_{tp}	ID		SF		DC	
					MWOZ (ACC)	M2M (ACC)	MWOZ (F1)	M2M (F1)	MWOZ (ACC)	M2M (ACC)
○	○	○	×	×	0.8429	<u>0.9411</u>	0.9928	0.9856	0.8750	0.8928
○	○	×	○	×	0.8427	<u>0.9411</u>	0.9928	0.9791	0.8750	0.8927
○	○	○	○	×	0.8464	0.9427	0.9928	<u>0.9850</u>	0.8780	0.8952
○	○	○	○	○	0.8462	0.9373	0.9927	0.9761	0.8793	0.8903

Table 4: The comparison of the diverse loss function combinations. Only BERT is utilised as the teacher model, and the results of RoBERTa are presented in Appendix E. The full names of each loss can be found in Section 3.3. We adopt two \mathcal{L}_{KD} and \mathcal{L}_{sce} as compulsory knowledge distillation loss and also explore three \mathcal{L}_{rel} , \mathcal{L}_{sim} , and \mathcal{L}_{tp} for MIDAS. Scores in bold indicate leadership among the metrics, and underlined scores indicate the second-best.

based on BERT and/or RoBERTa, resulting in a superb performance. In this section, we investigate how various pre-trained language models can impact the knowledge distillation ability of our multi-level teachers in instructing the student model. In addition to using BERT or RoBERTa, we also incorporate LLaMa2-7b, a decoder-only based pre-trained model, into our analysis.

Table 3-(a) shows the results of the effectiveness of using various pre-trained models as base models for all three multi-level teachers⁷. Compared to the high-achieving two encoder-based models, BERT and RoBERTa, the MIDAS (LLaMa) multi-level teachers produce lower performance⁸. We assume a decoder-only model like LLaMa, primarily used for generating coherent and contextually relevant text, whereas BERT and RoBERTa are encoder-based models that have a deep understanding of context and relationships between words and excel in classification tasks. In addition to having multi-level teachers using a single pre-trained model, we adopt a mixed type of pre-trained model for preparing multi-level teachers. For instance, we can apply BERT as a pre-trained model for teaching sentence-level intent knowledge, utilise RoBERTa as a teacher model for word-level slots, and adopt LLaMa as a conversation-level domain topic teacher model. As shown in Table 3-(a), the result shows that using mixed types of pre-trained teacher models is less effective than employing a consistent single pre-trained model as the teacher. This implies that knowledge distillation from teachers with inconsistencies in their feature spaces may impede the learning process for a single student model.

5.3 Effect of combination of multi-level teachers

We explore the impact of incorporating each multi-level teacher (ID, SF, DC) in all three multi-turn dialogue understanding tasks. MIDAS is evaluated with individual teachers (ID, SF, DC), all possible pairs from {ID, SF, DC}⁹, and then with all three teachers. Table 3 presents the results for each

⁷Note that the *MIDAS (BERT)* and *MIDAS (RoBERTa)* models are identical to those presented in the Table 2.

⁸Any decoder-only LM produces a similar low performance.

⁹Note that we do not adopt \mathcal{L}_{rel} since it is not possible to adopt when there are two teachers.

combination of teacher models for three different dialogue understanding tasks. Note that the table demonstrates the outcome of *MIDAS (BERT)* teachers, and we produce that of *MIDAS (RoBERTa)* in Appendix D. The experimental findings highlight that the **ID+SF+DC** combination attains the highest performance, underscoring the advantage of the student model integrating knowledge from all teachers for each task.

5.4 Effect of knowledge distillation loss function

As mentioned in Section 3.3, we conducted the loss function ablation study for MIDAS. This comprehensive evaluation aims to identify the most effective combinations that enhance the student model’s proficiency in handling different aspects of dialogue understanding across multiple NLU tasks. Note that we use \mathcal{L}_{KD} and \mathcal{L}_{sce} as compulsory knowledge distillation losses, and conduct an ablation study of three newly proposed multi-level teacher losses: \mathcal{L}_{sim} , \mathcal{L}_{rel} , and \mathcal{L}_{tp} . As shown in Table 4, the results indicate that incorporating \mathcal{L}_{sim} with \mathcal{L}_{rel} achieves the best or the second best performance across all tasks and datasets. Although \mathcal{L}_{rel} and \mathcal{L}_{sim} share a similar trend, their impact on model learning may be somewhat superior when employed independently, particularly when utilising \mathcal{L}_{sim} . While incorporating \mathcal{L}_{tp} with the others led to a slight performance increase, it did not match the effectiveness observed with the sole application of the earlier losses.¹⁰

5.5 Qualitative analysis: case study

We further evaluate MIDAS with a qualitative assessment of the three NLU tasks on M2M. As shown in Table 5, we assume to have a three-utterance conversation¹¹, ‘boat is fine’, ‘sure, i found this one: the view’, ‘how about the ivy or boats?’. Based on the given conversation, we test all three NLU tasks, including intent classification, slot filling, and domain classification. The domain would be classified across the entire conversation, so we placed it only once to the end. We mainly compare with two models: 1) **MIDAS (BERT)**, trained with three teachers $BERT_{ID}$, $BERT_{SF}$, and $BERT_{DC}$, and 2) **BERT-only** refers a single fine-tuned BERT (*BERT-Base*), which focus on only one task for each prediction.

Although the single fine-tuned BERT (*BERT-only*) can sometimes predict the slots correctly, it does not communicate/integrate with the word level and domain level classification. For example, (*BERT-only*) correctly tags B-RN (Restaurant Name) in the turn 3, while no-integration would produce a domain prediction as ‘movie’. Nevertheless, the student model exhibits the ability to identify (-RN)Restaurant Names even in the absence of domain labels when trained independently on the SF task and concurrently on the ID and DC tasks. Instances such as these validate our hypothesis that leveraging diverse knowledge levels from multi-turn conversation datasets can improve the understanding of individual natural language understanding tasks, outperforming the advantages of learning with single-level dialogue knowledge.

6 Conclusion and Limitation

This paper introduces a novel multi-level teacher knowledge distillation framework to enhance multi-turn natural language understanding (NLU). By fine-tuning pre-trained models at word, sentence, and document levels, we construct multi-level teachers, imparting their knowledge to a student

¹⁰We conducted testing with \mathcal{L}_{tp} only, it produces much lower performance than any others, which is expected.

¹¹The original conversation is longer but reduced in the visualisation due to page limitation.

model. Various loss functions are introduced and explored, and the experiment results demonstrate the framework’s effectiveness in improving the student model’s understanding across diverse NLU tasks. It shows better than the LLM result. However, there are some spaces for future work, including covering multilingual multi-turn dialogue. We believe this work will provide valuable insights into various aspects of dialogue knowledge for NLU and multi-level knowledge distillation.

References

- [1] Henry Weld, Xiaoqi Huang, Siqu Long, Josiah Poon, and Soyeon Caren Han. A survey of joint intent detection and slot filling models in natural language understanding. *ACM Computing Surveys*, 55(8):1–38, 2022.
- [2] Bing Liu and Ian Lane. Attention-based Recurrent Neural Network Models for Joint Intent Detection and Slot Filling. In *Proceedings of the INTERSPEECH 2016*, pages 685–689, 2016.
- [3] Ankur Bapna, Gokhan Tur, Dilek Hakkani-Tur, and Larry Heck. Sequential Dialogue Context Modeling for Spoken Language Understanding. In *Proceedings of the 18th SIGDIAL*, pages 103–114, 2017.
- [4] Chih-Wen Goo, Guang Gao, Yun-Kai Hsu, Chih-Li Huo, Tsung-Chieh Chen, Keng-Wei Hsu, and Yun-Nung Chen. Slot-Gated Modeling for Joint Slot Filling and Intent Prediction. In *Proceedings of the 2018 NAACL*, pages 753–757, 2018.
- [5] Chen Tingting, Lin Min, and Li Yanling. Joint Intention Detection and Semantic Slot Filling Based on BLSTM and Attention. In *Proceedings of the 2019 ICCCDBA*, pages 690–694, 2019.
- [6] Yidi Jiang, Bidisha Sharma, Maulik Madhavi, and Haizhou Li. Knowledge Distillation from BERT Transformer to Speech Transformer for Intent Classification. In *Proceedings of the INTERSPEECH 2021*, pages 4713–4717, 2021.
- [7] Lisong Chen, Peilin Zhou, and Yuexian Zou. Joint Multiple Intent Detection and Slot Filling via Self-Distillation. In *Proceedings of the 2022 ICASSP*, pages 7612–7616, 2022.
- [8] Kalpa Gunaratna, Vijay Srinivasan, Akhila Yerukola, and Hongxia Jin. Explainable Slot Type Attentions to Improve Joint Intent Detection and Slot Filling. In *Findings of the EMNLP 2022*, pages 3367–3378, 2022. doi: 10.18653/v1/2022.findings-emnlp.245. URL <https://aclanthology.org/2022.findings-emnlp.245>.
- [9] Henry Weld, Sijia Hu, Siqu Long, Josiah Poon, and Soyeon Han. Tri-level Joint Natural Language Understanding for Multi-turn Conversational Datasets. In *Proceedings of the INTERSPEECH 2023*, pages 700–704, 2023.
- [10] Subendhu Rongali, Beiyi Liu, Liwei Cai, Konstantine Arkoudas, Chengwei Su, and Wael Hamza. Exploring Transfer Learning For End-to-End Spoken Language Understanding. In *Proceedings of the 35th AAAI*, volume 35, pages 13754–13761, 2021.
- [11] Meryem M’hamdi, Doo Soon Kim, Franck Dernoncourt, Trung Bui, Xiang Ren, and Jonathan May. X-METRA-ADA: Cross-lingual Meta-Transfer learning Adaptation to Natural Language Understanding and Question Answering. In *Proceedings of the 2021 NAACL*, pages 3617–3632, 2021.
- [12] Mai Hoang Dao, Thinh Hung Truong, and Dat Quoc Nguyen. Intent Detection and Slot Filling for Vietnamese. In *Proceedings of the INTERSPEECH 2021*, pages 4698–4702, 2021.
- [13] Waheed Ahmed Abro, Guilin Qi, Muhammad Aamir, and Zafar Ali. Joint Intent Detection and Slot Filling using Weighted Finite State Transducer and BERT. *Applied Intelligence*, 52(15):17356–17370, 2022.
- [14] Jie Mei, Yufan Wang, Xinhui Tu, Ming Dong, and Tingting He. Incorporating BERT With Probability-Aware Gate for Spoken Language Understanding. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 31:826–834, 2023.
- [15] Qian Chen, Zhu Zhuo, and Wen Wang. BERT for Joint Intent Classification and Slot Filling. *arXiv preprint arXiv:1902.10909*, 2019.

- [16] Seong-Hwan Heo, WonKee Lee, and Jong-Hyeok Lee. mcBERT: Momentum Contrastive Learning with BERT for Zero-Shot Slot Filling. In *Proceedings of the INTERSPEECH 2022*, pages 1243–1247, 2022.
- [17] Xuxin Cheng, Zhihong Zhu, Wanshi Xu, Yaowei Li, Hongxiang Li, and Yuexian Zou. Accelerating Multiple Intent Detection and Slot Filling via Targeted Knowledge Distillation. In *Findings of the EMNLP 2023*, pages 8900–8910, 2023.
- [18] Ting-Wei Wu and Biing Juang. Infusing Context and Knowledge Awareness in Multi-turn Dialog Understanding. In *Findings of the EACL 2023*, pages 254–264, 2023.
- [19] Nguyen Anh Tu, Hoang Thi Thu Uyen, Tu Minh Phuong, and Ngo Xuan Bach. Joint Multiple Intent Detection and Slot Filling with Supervised Contrastive Learning and Self-Distillation. In *Proceedings of the 26th ECAI*, pages 333–343, 2023.
- [20] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the Knowledge in a Neural Network. In *Advances in NeurIPS*, pages 1–9, 2014.
- [21] Chuhan Wu, Fangzhao Wu, and Yongfeng Huang. One Teacher is Enough? Pre-trained Language Model Distillation from Multiple Teachers. In *Findings of the ACL 2021*, pages 4408–4413, 2021.
- [22] Fei Yuan, Linjun Shou, Jian Pei, Wutao Lin, Ming Gong, Yan Fu, and Dixin Jiang. Reinforced Multi-Teacher Selection for Knowledge Distillation. In *Proceedings of the 35th AAAI*, pages 14284–14291, 2021.
- [23] Yeongseo Jung, Eunseo Jung, and Lei Chen. Towards a Unified Conversational Recommendation System: Multi-task Learning via Contextualized Knowledge Distillation. In *Proceedings of the 2023 EMNLP*, pages 13625–13637, 2023.
- [24] Kai Wang, Yu Liu, Qian Ma, and Quan Z Sheng. Mulde: Multi-Teacher Knowledge Distillation for Low-Dimensional Knowledge Graph Embeddings. In *Proceedings of the Web Conference 2021*, pages 1716–1726, 2021.
- [25] Kuan-Po Huang, Tzu-hsun Feng, Yu-Kuan Fu, Tsu-Yuan Hsu, Po-Chieh Yen, Wei-Cheng Tseng, Kai-Wei Chang, and Hung-yi Lee. Ensemble Knowledge Distillation of Self-Supervised Speech Models. In *Proceedings of the 2023 ICASSP*, pages 1–5, 2023.
- [26] Abdollah Amirkhani, Amir Khosravian, Masoud Masih-Tehrani, and Hossein Kashiani. Robust Semantic Segmentation with Multi-teacher Knowledge Distillation. *IEEE Access*, 9:119049–119066, 2021.
- [27] Seyed Iman Mirzadeh, Mehrdad Farajtabar, Ang Li, Nir Levine, Akihiro Matsukawa, and Hassan Ghasemzadeh. Improved Knowledge Distillation via Teacher Assistant. In *Proceedings of the 34th AAAI*, pages 5191–5198, 2020.
- [28] Wonchul Son, Jaemin Na, Junyong Choi, and Wonjun Hwang. Densely Guided Knowledge Distillation Using Multiple Teacher Assistants. In *Proceedings of the 2021 ICCV*, pages 9395–9404, 2021.
- [29] Haojie Pan, Chengyu Wang, Minghui Qiu, Yichang Zhang, Yaliang Li, and Jun Huang. Meta-KD: A Meta Knowledge Distillation Framework for Language Model Compression across Domains. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing*, pages 3026–3036. Association for Computational Linguistics, 2021. doi: 10.18653/v1/2021.acl-long.236. URL <https://aclanthology.org/2021.acl-long.236>.
- [30] Zhong Ji, Jingwei Ni, Xiyao Liu, and Yanwei Pang. Teachers Cooperation: Team-Knowledge Distillation for Multiple Cross-domain Few-shot Learning. *Frontiers of Computer Science*, 17(2):172312, 2023.
- [31] Quan Kong, Ziming Wu, Ziwei Deng, Martin Klinkigt, Bin Tong, and Tomokazu Murakami. MMAct: A Large-Scale Dataset for Cross Modal Human Action Understanding. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8658–8667, October 2019.

- [32] Jianyuan Ni, Raunak Sarbjna, Yang Liu, Anne H.H. Ngu, and Yan Yan. Cross-Modal Knowledge Distillation For Vision-To-Sensor Action Recognition. In *2022 IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 4448–4452, 2022. doi: 10.1109/ICASSP43922.2022.9746752.
- [33] Woojeong Jin, Maziar Sanjabi, Shaoliang Nie, Liang Tan, Xiang Ren, and Hamed Firooz. Modality-specific Distillation. In *Proceedings of the Third Workshop on Multimodal Artificial Intelligence*, pages 42–53. Association for Computational Linguistics, 2021. URL <https://aclanthology.org/2021.maiworkshop-1.7>.
- [34] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.
- [35] Zongxi Li, Xianming Li, Yuzhang Liu, Haoran Xie, Jing Li, Fu lee Wang, Qing Li, and Xiaoqin Zhong. Label supervised llama finetuning, 2023.
- [36] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In "*Proceedings of the ICLR 2018*", 2018.
- [37] Libo Qin, Tailu Liu, Wanxiang Che, Bingbing Kang, Sendong Zhao, and Ting Liu. A Co-Interactive Transformer for Joint Slot Filling and Intent Detection. In *Proceedings of the 2021 ICASSP*, pages 8193–8197, 2021.
- [38] Libo Qin, Fuxuan Wei, Tianbao Xie, Xiao Xu, Wanxiang Che, and Ting Liu. GL-GIN: Fast and Accurate Non-Autoregressive Model for Joint Multiple Intent Detection and Slot Filling. In *Proceedings of the 59th ACL*, pages 178–188, 2021.
- [39] Jixuan Wang, Kai Wei, Martin Radfar, Weiwei Zhang, and Clement Chung. Encoding Syntactic Knowledge in Transformer Encoder for Intent Detection and Slot Filling. In *Proceedings of the 35th AAAI*, volume 35, pages 13943–13951, 2021.
- [40] Ting-Wei Wu, Ruolin Su, and Biing-Hwang Juang. A Context-Aware Hierarchical BERT Fusion Network for Multi-Turn Dialog Act Detection. In *Proceedings of the INTERSPEECH 2021*, pages 1239–1243, 2021.
- [41] Dongsheng Chen, Zhiqi Huang, Xian Wu, Shen Ge, and Yuexian Zou. Towards Joint Intent Detection and Slot Filling via Higher-order Attention. In *Proceedings of the 35th IJCAI*, pages 4072–4078, 2022.
- [42] Bowen Xing and Ivor Tsang. Group is better than individual: Exploiting Label Topologies and Label Relations for Joint Multiple Intent Detection and Slot Filling. In *Proceedings of the 2022 EMNLP*, pages 3964–3975, 2022.
- [43] Thanh Tran, Kai Wei, Weitong Ruan, Ross McGowan, Nathan Susanj, and Grant P. Strimel. Adaptive Global-Local Context Fusion for Multi-Turn Spoken Language Understanding. In *Proceedings of the 36th AAAI*, volume 36, pages 12622–12628, 2022.
- [44] Thinh Pham, Chi Tran, and Dat Quoc Nguyen. MISCA: A Joint Model for Multiple Intent Detection and Slot Filling with Intent-Slot Co-Attention. In *Findings of the EMNLP 2023*, pages 12641–12650, 2023.
- [45] Liang Huang, Senjie Liang, Feiyang Ye, and Nan Gao. A Fast Attention Network for Joint Intent Detection and Slot Filling on Edge Devices. *IEEE Transactions on Artificial Intelligence*, pages 1–11, 2023. doi: 10.1109/TAI.2023.3309272.
- [46] Soyeon Caren Han, Siqu Long, Huichun Li, Henry Weld, and Josiah Poon. Bi-directional joint neural networks for intent classification and slot filling. In "*Proceedings of the INTERSPEECH*", Brno, Czech Republic, 08 2021. ISCA.
- [47] Xiaoxue Zang, Abhinav Rastogi, Srinivas Sunkara, Raghav Gupta, Jianguo Zhang, and Jindong Chen. MultiWOZ 2.2 : A dialogue dataset with additional annotation corrections and state tracking baselines. In *Proceedings of the 2nd Workshop on Natural Language Processing for Conversational AI*, pages 109–117, 2020.

- [48] Bing Liu, Gokhan Tür, Dilek Hakkani-Tür, Pararth Shah, and Larry Heck. Dialogue learning with human teaching and feedback in end-to-end trainable task-oriented dialogue systems. In *Proceedings of the 2018 NAACL*, pages 2060–2069, 2018.

Model	Year	Token (Slot)	Sentence (Intent)	Document (Domain)	Dialogue Type	Joint Integration
SeqSeq Liu and Lane [2]	2016	○	○	×	Single-Turn	BiRNN + Attention
SDEN Bapna et al. [3]	2017	○	○	○	Multi-Turn	BiRNN + Memory Network
Slot-Gated Goo et al. [4]	2018	○	○	×	Single-Turn	BiLSTM + Slot Gate
BLSTM+attention Tingting et al. [5]	2019	○	○	×	Single-Turn	BiLSTM + Attention
Co-Interactive Transformer Qin et al. [37]	2021	○	○	×	Single-Turn	BiLSTM + Attention
GL-GIN Qin et al. [38]	2021	○	○	×	Single-Turn	BiLSTM + GAT
SyntacticTF Wang et al. [39]	2021	○	○	×	Single-Turn	Transformer
STD Jiang et al. [6]	2021	○	○	×	Single-Turn	Transformer + One-teacher KD
JointIDSF Dao et al. [12]	2021	○	○	×	Single-Turn	CRF + Attention
CaBERT-SLU Wu et al. [40]	2021	○	○	○	Multi-Turn	Attention
SDJN Chen et al. [7]	2021	○	○	×	Single-Turn	BiLSTM + self KD
HAN Chen et al. [41]	2022	○	○	×	Single-Turn	BiLSTM + Attention
ReLA-NET Xing and Tsang [42]	2022	○	○	×	Single-Turn	BiLSTM + GAT
XAI Attention Gunaratna et al. [8]	2022	○	○	×	Multi-Turn	XAI
WFST-BERT Abro et al. [13]	2022	○	○	×	Single-Turn	WFST
Contextual SLU Tran et al. [43]	2022	○	○	○	Multi-Turn	BiLSTM + Attention
TKDF Cheng et al. [17]	2023	○	○	×	Single-Turn	SSRAN + One-teacher KD
MISCA Pham et al. [44]	2023	○	○	×	Single-Turn	BiLSTM + Attention
PAGM Mei et al. [14]	2023	○	○	×	Single-Turn	Gate
FAN Huang et al. [45]	2023	○	○	×	Single-Turn	Attention
Tri-level JNLU Weld et al. [9]	2023	○	○	○	Multi-Turn	Transformer
CKA-NLU Wu and Juang [18]	2023	○	○	○	Multi-Turn	Attention
BiSLU Tu et al. [19]	2023	○	○	×	Single-Turn	self KD
MIDAS	2024	○	○	○	Multi-Turn	Multi-teacher KD

Table 6: Summary of previous joint NLU models and MIDAS. Token, Sentence, and Document columns indicate whether the relevant information is used for joint integration. GAT in the Joint Integration column refers to the graph attention network, KD refers to knowledge distillation, and WFST refers to Weighted Finite-State Transducers.

A Related works

Table 6 presents a comparison of MIDAS with 23 previous joint NLU models. Recently, most NLU studies have embraced a joint learning model capable of handling all NLU tasks to mitigate error propagation inherent in pipelined approaches [39, 46, 8, 45]. The initial joint models employed traditional neural networks like RNN [2] and LSTM [5, 38, 41–44] with attention mechanisms.

All models leverage slot-level knowledge and intent-level knowledge, but only five previous works incorporate domain-level knowledge. This implies that only five prior studies utilised a multi-turn dialogue dataset.

Only one previous study [9] conducted tests on domain classification. Hence, we chose [9] as a representative baseline. What sets the proposed model apart is its utilisation of multi-teacher knowledge distillation. While two previous works employed self-knowledge distillation and another two adopted one-teacher knowledge distillation, MIDAS represents the first attempt at employing multi-teacher knowledge distillation for joint learning in natural language understanding.

B Each losses of proposed loss function

The following losses are utilized to train the student model, each playing a distinct role:

- L_{kd} : This loss facilitates the transfer of knowledge from the teacher models to the student model, enabling the student to mimic the general behavior of multiple teachers.
- L_{rel} : This loss is designed to capture the relationships between different samples in the input data. It helps to align the student’s understanding with that of the teacher models and ensures a consistent perspective on the dataset.
- L_{sim} : This loss encourages the student model to generate outputs that are similar to those of the teacher models in terms of their overall structure and distribution. It helps to maintain consistency between the student and teacher predictions.
- L_{sce} : This loss function serves as the fundamental mechanism for training the student model. It entails the student learning to predict the correct labels associated with the input data.

	ID		SF		DC	
	MWOZ (ACC)	M2M (ACC)	MWOZ (F1)	M2M (F1)	MWOZ (ACC)	M2M (ACC)
ID-only	0.8339	0.8097	0.9079	0.9326	0.6183	0.7147
SF-only	0.8403	0.8945	0.9620	0.9434	0.2471	0.8917
DC-only	0.8469	0.8929	0.9547	0.9251	0.7521	0.8913
ID+SF	0.8451	0.9083	0.9928	0.9802	0.8734	0.8923
ID+DC	0.8373	0.9114	0.9921	0.9797	0.8763	0.8888
SF+DC	0.8453	0.9147	0.9927	0.9805	0.8707	0.8912
ID+SF+DC	0.8502	0.9377	0.9928	0.9813	0.8816	0.8945

Table 7: The performance based on the type of teacher models. The first column indicates the type of teacher used. For example, *ID+SF+DC* uses all intent classification, slot filling, and domain classification teachers, while *ID-only* uses only the intent classification teacher. Only RoBERTa is utilised as the teacher model.

- L_{tp} : This loss leverages the predictions of the teacher models to provide additional supervision signals to the student model. It helps to guide the student towards making predictions that align with those of the teachers.

C Details of datasets

MWOZ [47] is specifically designed for Dialogue State Tracking (DST) and adopts the conventional human-vs-human Wizard of Oz approach across diverse domains, including attraction, bus, hospital, hotel, police, restaurant, taxi, and train. It incorporates 30 slot types and 11 intent types. The dataset comprises 8,437 dialogues, with an average of 5.68 turns per dialogue and 14.07 tokens per turn. Following by [2, 4, 9], we do not consider any multi-label samples but utilise the data with a single domain and intent.

M2M [48] is introduced with virtual agents and user-generated interactions to emulate goal-directed conversations through paraphrasing with templated utterances. M2M has movies and restaurant domains. The slots and intents are categorical, with 21 slot types and 15 intent types. The dataset comprises 1,500 dialogues, with an average of 9.86 turns per dialogue and 8.25 tokens per turn.

D Combination-based ablation study

We explore the impact of incorporating each multi-level teacher (ID, SF, DC) in all three multi-turn dialogue understanding tasks. Table 7 presents the results for each combination of teacher models for three different dialogue understanding tasks. The experimental results are when only RoBERTa is adopted as the teacher model. MIDAS is evaluated with individual teachers (ID, SF, DC), all possible pairs from {ID, SF, DC}, and then with all three teachers. For example, ID+SF+DC uses all intent classification, slot filling, and domain classification teachers, while ID-only uses only the intent classification teacher. Note that we do not adopt \mathcal{L}_{rel} while two models are used since it is not possible to adopt when there are two teachers. The experimental findings highlight that the **ID+SF+DC** combination attains the highest performance, underscoring the advantage of the student model integrating knowledge from all teachers for each natural language understanding task.

No.	Model	Tokens (Slot)	Intent	Domain
1	Utterance	near kirkland, and, i, don, 't, care, about, the ratings	-	-
	Ground Truth	O, B-location, O, O, O, O, O, O, O, O	inform	restaurant
	MIDAS (BERT)	O, B-location, O, O, O, O, O, O, O, O	inform	restaurant
	BERT-Only	O, B-restaurant_name, O, O, O, O, O, O, O, O	infrom	movie
2	Utterance	let, 's, go, with, the, view	-	-
	Ground Truth	O, O, O, O, B-restaurant_name, I-restaurant_name	affirm	restaurant
	MIDAS (BERT)	O, O, O, O, B-restaurant_name, I-restaurant_name	affirm	restaurant
	BERT-Only	O, O, O, O, O, O	affirm	movie
3	Utterance	then, find, me, one, in, the, expensive, price, range.	-	-
	Ground Truth	O, O, O, O, O, O, O, O, O	find_hotel	hotel
	MIDAS (BERT)	O, O, O, O, O, O, O, O, O	find_hotel	hotel
	BERT-Only	O, O, O, O, O, O, O, O, O	find_restaurant	restaurant
4	Utterance	which, ever, is, nice., i, will, need, some, info, on, it, too.	-	-
	Ground Truth	O, O, O, O, O, O, O, O, O, O	find_attraction	attraction
	MIDAS (BERT)	O, O, O, O, O, O, O, O, O, O	find_attraction	attraction
	BERT-Only	O, O, O, O, O, O, O, O, O, O	find_hotel	restaurant
5	Utterance	great, we, are, meeting, friends, at, wandlebury, country, park, before, we, eat,, can, you, tell, me, about, that, place, and, where, it, is?	-	-
	Ground Truth	O, O, O, O, O, B-attraction-name, I-attraction-name, I-attraction-name, O, O	find_attraction	attraction
	MIDAS (BERT)	O, O, O, O, O, B-attraction_name, I-attraction_name, I-attraction_name, O, O	find_attraction	attraction
	BERT-Only	O, O, O, O, O, B-attraction-name, I-hotel-name, O, O	find_restaurant	restaurant
6	Utterance	yes, may, i, have, the, address, post, code, and, phone, number, for, golden, house?, i'll, book, it, myself.	-	-
	Ground Truth	O, O, O, O, O, O, O, O, B-restaurant_name, I-restaurant_name, O, O, O, O	find_restaurant	restaurant
	MIDAS (BERT)	O, O, O, O, O, O, O, O, B-restaurant_name, I-restaurant_name, O, O, O, O	find_restaurant	restaurant
	BERT-Only	O, O	find_hotel	hotel
7	Utterance	can, you, book, for, arrival, closer, to, 17:30, for, one, person, and, give, me, the, reference, number, also, i, would, like, to, see, a, college, in, centre.	-	-
	Ground Truth	O, O	find_attraction	attraction
	MIDAS (BERT)	O, O	find_attraction	attraction
	BERT-Only	O, O	book_train	train
8	Utterance	yes, please., i, need, an, address, and, phone, number, too.	-	-
	Ground Truth	O, O, O, O, O, O, O, O, O	find_attraction	attraction
	MIDAS (BERT)	O, O, O, O, O, O, O, O, O	find_attraction	attraction
	BERT-Only	O, O, O, O, O, O, O, O, O	find_restaurant	restaurant
9	Utterance	just, need, to, know, what, area, "its", in.	-	-
	Ground Truth	O, O, O, O, O, O, O	find_attraction	attraction
	MIDAS (RoBERTa)	O, O, O, O, O, O, O	find_attraction	attraction
	RoBERTa-Only	O, O, O, O, O, O, O	find_hotel	hotel
10	Utterance	i, would, actually, like, to, book, 5, people, and, would, like, to, know, the, reference, number, for, the, tickets, and, the, address, of, la, tasca, restaurant.	-	-
	Ground Truth	O, B-restaurant-name, I-restaurant-name, O	find_restaurant	restaurant
	MIDAS (RoBERTa)	O, B-restaurant-name, I-restaurant-name, O	find_restaurant	restaurant
	RoBERTa-Only	O, B-restaurant-name, B-hotel-name, O	book_train	restaurant
11	Utterance	"its", not, a, restaurant,, "its", an, attraction., nusha.	-	-
	Ground Truth	O, O, O, O, O, O, O, B-attraction-name	find_attraction	attraction
	MIDAS (RoBERTa)	O, O, O, O, O, O, O, B-attraction-name	find_attraction	attraction
	RoBERTa-Only	O, O, O, O, O, O, O	find_attraction	restaurant
12	Utterance	will, you, give, me, the, phone, number,, address,, and, postcode, for, graffiti,, please?	-	-
	Ground Truth	O, O, O, O, O, O, O, O, B-restaurant-name, O	find_restaurant	restaurant
	MIDAS (RoBERTa)	O, O, O, O, O, O, O, O, B-restaurant-name, O	find_restaurant	restaurant
	RoBERTa-Only	O, O, O, O, O, O, O, O, O, O	find_attraction	restaurant
13	Utterance	i, am, looking, to, get, to, the, rajmahal, restaurant, please., how, do, i, get, there?	-	-
	Ground Truth	O, O, O, O, O, O, B-restaurant_name, O, O, O, O, O, O	find_restaurant	restaurant
	MIDAS (RoBERTa)	O, O, O, O, O, O, B-restaurant_name, O, O, O, O, O, O	find_restaurant	restaurant
	RoBERTa-Only	O, O, O, O, O, O, B-restaurant_name, O, O, O, O, O, O	find_taxi	taxi

Table 8: 13 Prediction examples with both datasets on slot filling, intent detection, and domain classification results of each model. The first two utterances are from M2M, while the rest are from MultiWOZ 2.2 (MWOZ). The first eight results come from *MIDAS (BERT)* and BERT-Only, whereas the remaining five results pertain to *MIDAS (RoBERTa)* and RoBERTa-Only. The green cell represents a result that matches the ground truth, the red cell indicates incorrect results, and the yellow cell indicates partially correct results.

E Loss function ablation study

We conducted the loss function ablation study for MIDAS with RoBERTa Teacher. This comprehensive evaluation aims to identify the most effective combinations that enhance the student model's proficiency in handling different aspects of dialogue understanding across multiple NLU tasks. Note

\mathcal{L}_{KD}	\mathcal{L}_{sce}	\mathcal{L}_{sim}	\mathcal{L}_{rel}	\mathcal{L}_{tp}	ID		SF		DC	
					MWOZ (ACC)	M2M (ACC)	MWOZ (F1)	M2M (F1)	MWOZ (ACC)	M2M (ACC)
○	○	○	×	×	0.8441	0.9362	0.9928	0.9842	0.8744	0.8945
○	○	×	○	×	0.8459	0.9377	0.9610	0.8415	0.8816	0.8914
○	○	○	○	×	0.8502	0.9376	0.9928	0.9813	0.8803	0.8945
○	○	○	○	○	0.8488	0.9264	0.9912	0.9704	0.8811	0.8922

Table 9: The comparison of the diverse loss function combinations. Only RoBERTa is utilised as the teacher model. We adopt two \mathcal{L}_{KD} and \mathcal{L}_{sce} as compulsory knowledge distillation loss and also explore three \mathcal{L}_{rel} , \mathcal{L}_{sim} , and \mathcal{L}_{tp} for MIDAS. Scores in bold indicate leadership among the metrics, and underlined scores indicate the second-best.

	ID		SF		DC	
	MWOZ (ACC)	M2M (ACC)	MWOZ (F1)	M2M (F1)	MWOZ (ACC)	M2M (ACC)
LLaMa2-7b-chat	0.4751	0.3363	0.0217	0.0751	0.6528	0.5231
LLaMa2-13b-chat	0.1679	0.2013	0.0891	0.1092	0.5602	0.4468
LLaMa2-70b-chat	0.3896	0.3275	0.0619	0.0883	0.6987	0.6012
Gemma-7b	0.6515	0.4588	0.6653	0.4357	0.7227	0.5426
GPT3.5	0.6971	0.5100	0.8175	0.5516	0.7739	0.7740
GPT4o	0.6789	0.6410	0.8418	0.6616	0.7877	0.8503
GPT4o with demonstration	0.7614	0.7510	0.8525	0.7132	0.7941	0.7051
Our best model	0.8502	0.9427	0.9928	0.9856	0.8816	0.8952

Table 10: The comparison of the proposed models with prompt tuning methods using Large Language Models. ID, SF and DC indicate intent detection, slot filling and domain classification, respectively, as mentioned in Section 4.2. ACC and F1 stand for accuracy and micro F1, respectively, and scores in bold indicate leadership among the metrics.

that we use \mathcal{L}_{KD} and \mathcal{L}_{sce} as compulsory knowledge distillation losses, and conduct an ablation study of three newly proposed multi-level teacher losses: \mathcal{L}_{sim} , \mathcal{L}_{rel} , and \mathcal{L}_{tp} . As shown in Table 9, the results indicate that incorporating \mathcal{L}_{sim} with \mathcal{L}_{rel} achieves the best or the second best performance across all tasks and datasets. Although \mathcal{L}_{rel} and \mathcal{L}_{sim} share a similar trend, their impact on model learning may be somewhat superior when employed independently, particularly when utilising \mathcal{L}_{sim} . While incorporating \mathcal{L}_{tp} with the others led to a slight performance increase, it did not match the effectiveness observed with the sole application of the earlier losses.

F More case studies

We further evaluate MIDAS with a qualitative assessment of the three NLU tasks on MWOZ and M2M. In Table 8, we test all three NLU tasks, including intent classification, slot filling, and domain classification. The first two utterances are from M2M, while the rest are from MultiWOZ 2.2 (MWOZ). The first eight results come from **MIDAS (BERT)**, trained with three teachers $BERT_{ID}$, $BERT_{SF}$, and $BERT_{DC}$, and **BERT-only** refers a single fine-tuned BERT (*BERT-Base*), whereas the remaining five results pertain to **MIDAS (RoBERTa)**, trained with three teachers $RoBERTa_{ID}$, $RoBERTa_{SF}$, and $RoBERTa_{DC}$, and **RoBERTa-only** refers a single fine-tuned RoBERTa (*RoBERTa-Base*). Although the single fine-tuned BERT (*BERT-only*) or RoBERTa (*RoBERTa-only*) can sometimes predict the slots correctly, it does not communicate/integrate with the word level and domain level classification. Instances such as these validate our hypothesis that leveraging diverse knowledge levels from multi-turn conversation datasets can improve the understanding of individual natural language understanding tasks, outperforming the advantages of learning with single-level dialogue knowledge.

G Prompt method

G.1 Quantitative analysis

We measured the performance using the zero-shot prompt method to compare performance with Large Language Model (LLM). The LLM LLaMa, Gemma, and GPT3.5 were utilized. The prompt for Intent Detection (ID) and Domain Classification (DC) was given as follows: In this task, you

Conversation Sample	MIDAS	Gemma-7b	GPT3.5
Find a restaurant for breakfast	Slot Token: Find/O a/restaurant/O for/O breakfast/B-Meal Intent: Greeting	Slot Token: Find/B-Cat a/B-Date restaurant/B-Loc for/B-Meal Intent: Inform	Slot Token: Find/O a/B-Meal restaurant/O for/O breakfast/O Intent: Request
In what area and what type of rating for the place?	Slot Token: In/O what/J area/O and/O what/O type/O of/O Intent: Request	Slot Token: In/B-Loc what/B-Cat area/I-Rating and/N what/N type/N of/N rating/N for/N the/N place/N Intent: Inform	Slot Token: In/O what/O area/O and/O what/O type/O of/O Intent: Request
One in Redmond and is Michelin rated	Slot Token: One/O in/O Redmond/B-Loc and/O is/O Intent: Inform	Slot Token: One/B-RN in/B-Loc Redmond/B-Rating and/N is/N Intent: Inform	Slot Token: One/Loc in/O Redmond/O and/B-Rating is/O Intent: Inform
I found the following place: the ivy, acorn or deep blue	Slot Token: Michelin/B-Rating rated/I-Rating Intent: Select	Slot Token: Michelin/N rated/N Intent: Inform	Slot Token: Michelin/B-RN rated/I-Rating Intent: Select
Show me info for deep blue	Slot Token: I/O found/O the/O following/O place/O -/O the/B-RN ivy/I-RN , /O acorn/B-RN or/O deep/B-RN blue/I-RN Intent: Affirm	Slot Token: I/O found/B-RN the/B-RN following/O place/O ./O the/O, ivy/O , /O acorn/O or/O deep/O blue/O Intent: Inform	Slot Token: I/O found/o the/O following/B-RN place/O :/B-RN the/I-RN ivy/I-RN , /O acorn/N or/N deep/N blue/N Intent: Request
Domain	Restaurant	Movie	Movie
Slots type: O (Other), Meal, Loc (Location), RN (Restaurant Name), Cat (Category), Date, Movie, Time, N (None, LLM doesn't answer)			
Intent type: Greeting, Request, Inform, Select, Affirm			
Domain type: Movie, Restaurant			

Conversation Sample	MIDAS	Gemma-7b	GPT3.5
Looking for a train	Slot Token: Looking/O for/O a/O train/O Intent: Find Train	Slot Token: Looking/O for/O a/O train/O Intent: Find Train	Slot Token: Looking/O for/O a/O train/O Intent: Find Train
What's the time?	Slot Token: What's/O the/I-TL Intent: Find Train	Slot Token: What's/O the/I-TL Intent: Find Train	Slot Token: What's/O the/I-TL Intent: Find Train
I want to depart after 19:45	Slot Token: I/O want/O to/O depart/O after/O 19:45/B-TL Intent: Find Train	Slot Token: I/O want/O to/O depart/O after/O 19:45/O Intent: Find Train	Slot Token: I/O want/O to/O depart/O after/O 19:45/O Intent: Find Train
How many tickets?	Slot Token: How many/O tickets/O Intent: Find Train	Slot Token: How many/O tickets/O Intent: Find Train	Slot Token: How many/O tickets/O Intent: Find Train
I would like to book a book	Slot Token: I/O would/O like/O to/O book/O 4/O Intent: Find Train	Slot Token: I/O would/O like/O to/O book/O 4/O Intent: Find Train	Slot Token: I/O would/O like/O to/O book/O 4/O Intent: Book Train
Do you need any other services?	Slot Token: I/O need/O any/O other/O services/O Intent: Find Train	Slot Token: I/O need/O any/O other/O services/O Intent: Find Train	Slot Token: I/O need/O any/O other/O services/O Intent: Find Train
I would like to know the address of La Tasac Restaurant	Slot Token: I/O would/O like/O to/O know/O the/O address/O of/O La/B-RN Tasac/I-RN Restaurant/O Intent: Find Res.	Slot Token: I/O would/O like/O to/O know/O the/O address/O of/O La/Tasac/O Restaurant/O Intent: Find Res.	Slot Token: I/O would/O like/O to/O know/O the/O address/O of/O La/B-RN Tasac/O Restaurant/O Intent: Find Res.
It is located at 14-16 Bridge Street, Would you like a reservation?	Slot Token: It is located at 14-16 Bridge Street, Would you like a reservation? Intent: Find Res.	Slot Token: It is located at 14-16 Bridge Street, Would you like a reservation? Intent: Find Res.	Slot Token: It is located at 14-16 Bridge Street, Would you like a reservation? Intent: Find Res.
I would like to book a table for 5 people	Slot Token: I/O would/O like/O to/O book/O a/O table/O for/O 5/O people/O Intent: Book Res.	Slot Token: I/O would/O like/O to/O book/O a/O table/O for/O 5/B-RF people/B-RN Intent: Book Res.	Slot Token: I/O would/O like/O to/O book/B-RB a/O table/O for/O 5/O people/O Intent: Book Res.
What is the date?	Slot Token: What/I-TL is/O the/I-DT date/O Intent: Book Res.	Slot Token: What/I-TL is/O the/I-DT date/O Intent: Book Res.	Slot Token: What/I-TL is/O the/I-DT date/O Intent: Book Res.
At 15:30 on the same day	Slot Token: At/O 15:30/B-RB on/O the/O same/O day/O Intent: Book Res.	Slot Token: At/O 15:30/O on/O the/O same/O day/O Intent: Book Res.	Slot Token: At/O 15:30/O on/O the/O same/O day/O Intent: Book Res.
Domain	Train, Restaurant	Train	Train
Slots type: O (Other), TL (Train Leaveat), RN (Restaurant Name), RB (Restaurant Booktime), RF (Restaurant Food)			
Intent type: Find Train, Book Train, Find Res. (Find Restaurant), Book Res. (Book Restaurant)			
Domain type: Train, Restaurant			

(a)

(b)

Figure 2: Two examples for qualitative analysis: (a) shows the results on the M2M dataset, and (b) shows the results on the MWOZ dataset. Each example shows the results when MIDAS matches the ground truth. The three cells below each example display the type lists for slot, intent, and domain and red text in each column of the results table indicates errors.

are given a dialogue. Your job is to classify the following dialogue into one of the {number of classes} different {intents or domains}. The {intents or domains} are: {name of classes}. Input: [{input}]. Output(only output the {intent or domain}):. The prompt for Slot Filling (SF) is: In this task, you are given a dialogue. Your job is to classify the following dialogue into one of the {number of classes} different slots. The slots are: {name of classes}. Input: [{input}]. Output(Only output slot types. And the slot types should be output as a list without any explanation):.

Table 10 shows the experimental results of each baseline with the performance of our best model. We can see that GPT3.5 performs best on all tasks in the zero-shot testing, but still falls significantly short of our model’s performance. In the ID and SF tasks, the performance of LLaMa is significantly worse than that of Gemma and GPT. This suggests that factors such as architecture, training data, and training methods also impact LLM performance, in addition to the number of parameters.

Even within the LLaMa series, the number of model parameters doesn’t always determine performance; the 7b model sometimes outperforms the 13b and 70b models. Note that only the 70b model was used with 4-bit quantisation.

Across all three tasks, LLMs occasionally generate out-of-scope class names, despite having all class names provided. Additionally, in the SF task, LLMs don’t always output answers corresponding to the length of the original text. Despite our prompt stating that no explanation is needed for efficiency, LLMs sometimes still generate explanations. These observations indicate that LLMs don’t fully grasp the input.

G.2 Qualitative analysis

In the qualitative analysis, we focus on two representative LLMs, Gemma-7b and GPT3.5, as shown in Figure 2. From the M2M conversation as shown in Figure 2-(a), we found that both LLMs can not predict slot types based on the context. For example, GPT3.5 predicts “Michelin/B-RN

rated/I-Rating” instead of “Michelin/B-Rating rated/I-Rating”. Except for the wrong understanding of the conversation, we found that both LLMs can not follow the prompt all the time. For example, both LLMs do not predict the slot type for each token, where the missing predictions are represented by “N”. From the Multi-Domain Wizard-of-Oz 2.2 (MWOZ) conversation as shown in Figure 2-(b), we can see that both LLMs can not make predictions in terms of the whole conversation, resulting the conflicts of the predictions of the domains and intents. For example, GPT3.5 predicts “Book Train” after “Book Restaurant” and Gemma-7b predicts “Find Restaurant” after “Book Restaurant”. Another example is that both LLMs failed to predict the domain “Restaurant” of the last turn dialogue, even the Gemma-7b already predicted the intent as “Find Restaurant”.

H Implementation details

H.1 Experiment hyperparameters

Table 11 presents the hyperparameters, used in our proposed Multi-level Teacher Fine-tuning, as well as Multi-Teacher Knowledge Distillation. The Implementation details can be found in Section 4.2. of the main submission.

We further present the results of various experiments conducted to select hyperparameters, particularly the learning rate, in Table 12. In all tests, the temperature is fixed at 20, and only the learning rate is changed to 0.0005, 0.00005, and 0.000005. In the experiments on the M2M dataset, the performance of Gemma-7b alongside LLaMa2-7b is also measured to compare performance with the generative model. The highest accuracy is shown when the learning rate was 0.00005, and Gemma-7b shows similar performance to LLaMa2-7b, but LLaMa2-7b is slightly superior. The best performance is observed when the learning rate is 0.00005, which is also the case in experiments on the MWOZ dataset.

Hyper-parameter	Value in Fine-tuning	Value in Knowledge Distillation
Learning Rate	5e-5	5e-5
Batch Size	32	32
Warm-up Steps	10% of Max epoch	10% of Max epoch
Max epoch	3	100
Stop Strategy	Max Epoch	Early Stopping on validation loss
Stop Patience	-	10
Optimizer	AdamW	AdamW
Optimizer Weight Decay	1e-2	1e-2
Optimizer Betas	0.9, 0.999	0.9, 0.999
Margin in \mathcal{L}_{rel}	-	0.2
Norm in \mathcal{L}_{rel}	-	2
\mathcal{F}_D in \mathcal{L}_{rel}	-	L2-Norm
Similarity in \mathcal{L}_{sim}	-	Cosine Similarity
Max Token Length	512	512

Table 11: The hyper-parameters used in our experiments.

Model	Task	Learning Rate	Accuracy
M2M			
LLaMa2-7b	ID	0.0005 0.00005 0.000005	0.9121 0.9392 0.9093
	SF	0.0005 0.00005 0.000005	0.9696 0.9833 0.9349
	DC	0.0005 0.00005 0.000005	0.8895 0.8804 0.8375
Gemma-7b	ID	0.0005 0.00005 0.000005	0.9204 0.9357 0.9102
	SF	0.0005 0.00005 0.000005	0.9693 0.9816 0.9429
	DC	0.0005 0.00005 0.000005	0.8799 0.8840 0.7890
MWOZ			
LLaMa2-7b	ID	0.0005 0.00005 0.000005	0.8021 0.8403 0.7952
	SF	0.0005 0.00005 0.000005	0.9776 0.9912 0.9740
	DC	0.0005 0.00005 0.000005	0.8411 0.8702 0.7026

Table 12: Summary of performance changes according to learning rate changes.

H.2 Model details

We display the visualisation of teacher models and our student model Vanilla Transformer Encoder together. Those two summarises can be found in Table 13. Note that we use LoRA to fine-tune LLaMa 2-7b.

	BERT	RoBERTa	LLaMa	Student
Architecture	Encoder	Encoder	Decoder	Encoder
Parameters	110M	125M	7B	58M
Layers	12	12	32	6
Heads	12	12	32	8
Hidden Dim.	768	768	4096	768
Feed Forward Dim.	3072	3072	11008	2048
Dropout Rate	0.1	0.1	0.0	0.3
Rank of LoRA	-	-	64	
Alpha of LoRA	-	-	16	
Dropout of LoRA	-	-	0.1	

Table 13: The details of the models used in our work.

H.3 Hardware information

Our experiments are run on the Linux platform with an A6000 Nvidia graphic card and an AMD Ryzen Threadripper PRO 5955WX 16-core CPU, and the RAM is 128G.