Text-centric Alignment for Bridging Test-time Unseen Modality

Anonymous ACL submission

Abstract

This paper addresses the challenge of handling unseen modalities at test time and dynamic modality combinations with our proposed text-centric alignment method. This training-free in-context-learning alignment approach unifies different input modalities into a single semantic text representation by leveraging in-context learning with Large Language Models and unimodal foundation models. Our method significantly enhances the ability to manage unseen, diverse, and unpredictable modality combinations, making it suitable for both generative and discriminative models to adopt on top. Our extensive experiments primarily evaluates on discriminative tasks, demonstrating that our approach is essential for LLMs to achieve robust modality alignment performance. It also surpasses the limitations of traditional fixed-modality frameworks in embedding representations. This study contributes to the field by offering a flexible and effective solution for real-world applications where modality availability is dynamic and uncertain.

1 Introduction

This work targets the challenge of handling test-time unseen modalities and dynamic modality combinations for multimodal models where the input modality in the testing (or prediction) phase differs from that in training. The motivation for this research arises from the dynamic nature of real-world data, where modalities can unpredictably vary or even be absent at inference time. Consider the following scenario: A hospital has extensive image and text data about its patients, such as X-ray images and doctors' written diagnoses. This data can be used to train an AI model that diagnoses patients based on both image and text inputs. However, to enhance patient satisfaction, the hospital wants to develop a dialog system that can diagnose patients based on their audio descriptions of their symptoms. Typically, achieving this would require collecting audio data and employing transfer learning or domain adaptation techniques to align the information across modalities, a process that demands significant additional effort, cost, and time.

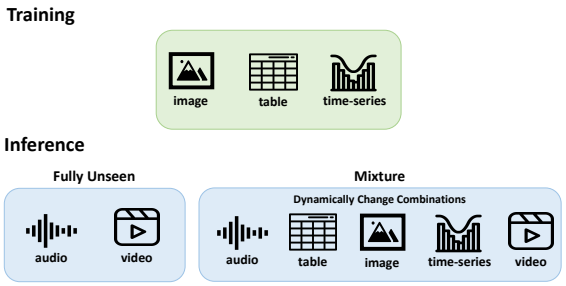

Figure 1: This paper establishes a general method applicable to all mismatch types and combinations. As the figure shows, the model utilizes three modalities during training. Our unified model can handle any combination of modalities during inference with in-context learning, including completely unseen modalities. Additionally, it can deal with the situation that modality combinations dynamically change during inference, covering all possible combinations of both seen and unseen modalities.

Traditional multimodal learning methods, typically fixated on static modality combinations during both training and inference, fall short in such fluid environments. Therefore, this paper explores the possibility of creating a supervised model that utilizes only existing modalities (e.g., images and text in the hospital scenario, and time series and tables in the bank scenario) during training, yet allows for the incorporation of an unseen modality (e.g., audio signals or text inputs) during inference. If successful, this approach would eliminate the need for additional data collection and modality alignment, thereby reducing the associated costs and efforts.

There are two plausible directions to tackle this challenge. First, we can rely on universal models pre-trained on vast datasets across numerous modalities to encode these modalities into embeddings. However, collecting data that have multiple modal-

Training modality Testing modality	table text+image	table text	table image
Embedding-based	0.287	0.289	0.279
Naive Text-centric	0.194	0.193	0.196
GPT-4o few-shot	0.167	0.204	0.168
TAMML	0.360	0.360	0.364

Table 1: A brief summary of the experiment results showing that TAMML alignment are necessary to perform well in text space, particularly in complex scenarios involving multiple modality combinations. Embedding-based and Text-centric are illustrated in Figure 2.

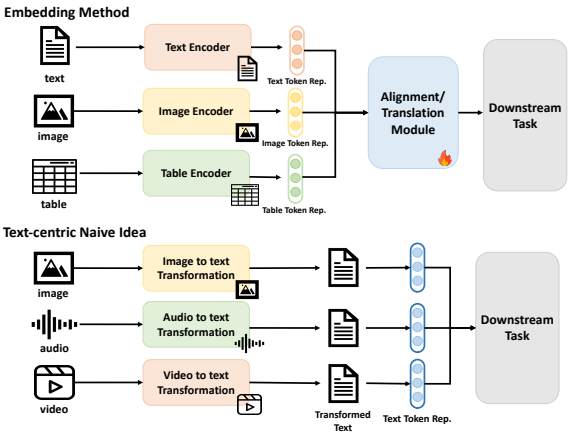

Figure 2: Traditional downstream training relies on embeddings extracted from upstream foundation models, with one foundation model designated for each modality. This approach limits the downstream model's ability to adapt to unseen modalities at test time without undergoing complete retraining. Previous research has addressed this issue by implementing zero-shot cross-modality translations during the inference phase.

ity aligned is extremely rare and costly. Moreover, changes in the input modality necessitate retraining the entire model for accurate predictions. The second direction, which this paper adopts, is to convert every modality into a single modality and build the model based on that unified modality. We argue that converting all modalities into text could be a favorable choice. Text can serve as a unified semantic space, leveraging the extensive zero-shot prediction capabilities of Large Language Models (LLMs). The modality-invariant nature of text provides a versatile bridge across different data types, potentially circumventing issues like modality collapse and extending generalizability to unseen modalities. Furthermore, advanced text analysis tasks such as translation, summarization, and explanation have been extensively researched and integrated into LLMs, offering powerful capabilities to align various modalities effectively.

Our objective is to develop a downstream model that is invariant across modalities. This model should be capable of being trained on data of certain modalities and performing zero-shot predictions on various modality combinations during testing, regardless of whether they were seen during training, as shown in Figure 1 and Figure 3. Note that simply converting all modalities into text for training and inference is insufficient, as the mismatch between different modalities does not translate and align as seamlessly in text, especially in complex scenarios involving multiple modality combinations. A brief summary of the results is shown in Table 1, with more detailed findings presented in the Experiment section. To address these issues, our work has explored challenges such as modality alignment, translation, and augmentation.

TAMML employs LLMs for data transformation across various modalities, with the aim of creating a unified semantic space. This process is conducted exclusively through in-context learning. Initially, we transform different modalities into text. Recently, various solutions have been developed, such as GPT-4 (OpenAI, 2023), Blip2 (Li et al., 2023a) for vision, and TabLLM (Hegselmann et al., 2023) for tabular data. Following this, we engage LLMs in text-style translation across modalities, ensuring that all modalities in their textual representation adopt a consistent linguistic structure, thereby reducing the gap between different modalities. To further align these modalities within a closer semantic space, remove redundant information, and mitigate the heterogeneity inherent in text data from diverse sources, we further conduct modality summarization. This step involves a concise summarization of the translated data. Additionally, TAMML includes a reasoning augmentation step akin to the Chain-of-Thought (Wei et al., 2022) method, where we enhance the data with LLMs to boost prediction and judgment capabilities. Moreover, we leverage LLMs as a source of large-scale external knowledge, enriching the data understanding and interpretative depth (Chen et al., 2023). We aim to answer several hypotheses through extensive experiments. First, whether TAMML is more effective compared to existing solutions in predicting data of unseen modalities. Second, although this work focuses on predicting unseen modalities, we want to understand whether the proposed solution is effective when the modality in testing is already seen during

training. Third, whether the text-as-the-medium strategy is more robust compared to embedding-based cross-modality transfer solutions. We benchmarked TAMML against existing methodologies in closely related tasks, particularly focusing on zero-shot learning cross-modality translation, which involves translating unseen source data to a different target domain. Techniques like MIDiffusion (Wang et al., 2023b) and SDEdit (Meng et al., 2022) demonstrate commendable performance in tasks such as domain translation within images. However, these methods encounter challenges when the source and target domains represent completely different modalities.

Our contributions can be summarized as follows:

- We investigate the potential advantage of using LLMs and text representation for multimodal learning. We propose TAMML , an in-context cross-modality translation method that utilizes foundation models to tackle training/testing modality mismatch and generalize to any unseen modality at test time. TAMML eliminates the need for any pre-training, fine-tuning, and the collection of multi-modality aligned data.

- We demonstrate that TAMML can significantly outperform SOTA approaches by conducting multiple experiments on real-world datasets. We also have an ablation study to analyze the effectiveness of each component in TAMML .

- Additional experiments further verify that even when the testing modality is already seen during training, TAMML can still outperform the competitors by a large margin.

2 Related Works

2.1 Multimodal Foundation Models

Recent advances in foundation models have greatly improved multimodal generation. However, aligning the semantic spaces of independently trained models remains challenging, limiting seamless modality transfer at test time. Multimodal LLMs (MLLMs) have shown strong reasoning and generation abilities (Yin et al., 2023), but large-scale pretraining across multiple modalities still demands extensive data.

To address this, many works convert modality inputs into text for LLM alignment. LLaVA (Liu et al., 2023) and VideoChat-Text (Li et al., 2023b) turn images or videos into captions. Cosmos (Agarwal et al., 2025) uses video summaries for retrieval (Blog, 2024), while ChatCAD (Wang et al., 2023a) and OphGLM (Gao et al., 2023) generate diagnostic text from X-rays. NExT-GPT (Wu et al., 2023) also builds a general-purpose multimodal LLM via modality adaptors, but unlike TAMML , it requires training projection layers. In contrast, we achieve any-to-any alignment through text alone using in-context learning.

2.2 Zero-shot Learning for Cross-Modality Translation

Zero-shot learning (ZSL) offers a promising approach for cross-modality translation when source modality data is unavailable. A key challenge for learning-based methods is their limited generalization to unseen classes (Wang et al., 2021; Bucher et al., 2017; Kuchibhotla et al., 2022). Traditional ZSL methods map features to a shared semantic space via discriminative (Palatucci et al., 2009; Akata et al., 2015) or generative models (Long et al., 2017; Wang et al., 2018).

In modality translation, GAN-based approaches perform latent space manipulation through GAN inversion (Zhu et al., 2020; Shi et al., 2022; Abdal et al., 2020), while diffusion-based methods (Ho et al., 2020; Kawar et al., 2022; Meng et al., 2022) enable zero-shot alignment by perturbing features toward target distributions. Despite strong results when domains are numerically aligned (Cheng et al., 2023), performance often degrades with large appearance mismatches.

2.3 Training-based Modality Binding

Recent works such as ImageBind (Girdhar et al., 2023) and LanguageBind (Zhu et al., 2023) reduce the complexity of aligning modality pairs by leveraging a central anchor—text in LanguageBind and image in ImageBind. This design simplifies cross-modality alignment from requiring explicit pairwise supervision across all modalities to aligning each modality with a single anchor. This idea aligns closely with our motivation in TAMML , which also uses text as a unifying interface.

However, both ImageBind and LanguageBind rely heavily on large-scale supervised training across diverse modality pairs. While they reduce the need for all modalities to be jointly present during training, they still require extensive paired data between each modality and the anchor (text

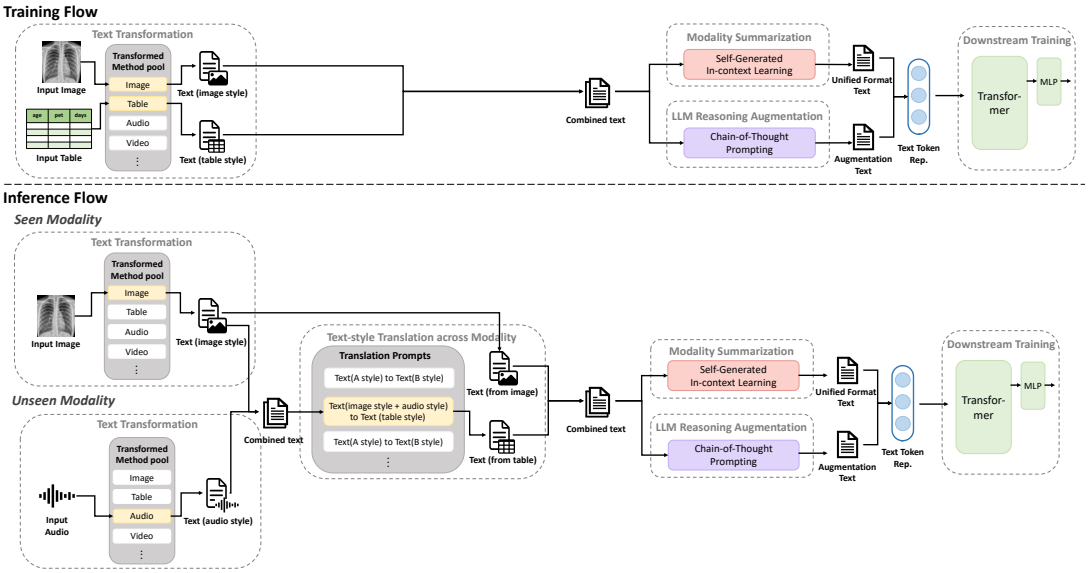

Figure 3: In the training phase, each raw input modality is transformed into text representations using a corresponding foundation model. Following the modality transformation, summarization, and augmentation are applied in parallel. Finally, the output texts are concatenated as the training inputs to a transformer model for downstream prediction. The inference phase follows a similar pattern, with the exception of utilizing an LLM for the text-style translation after the text transformation module. We apply a one-shot in-context learning approach to adapt the linguistic style as anticipated during training.

or image). In contrast, we proposed a training-free framework that operates solely using pre-trained modality-to-text converters and in-context LLM reasoning, making it more lightweight and deployable under resource-constrained settings.

3 Methodologies

This section describes how TAMML enables the generalization to unseen testing modalities and unseen modality combinations. We explicitly separate our pipeline into two phases: **Training-Free Alignment Method**, which includes modules that do not require any model training (Sections 3.2.1 to 3.2.4), and **Downstream Prediction**, which requires minimal supervised training (Section 3.3). Figure 3 presents an overview of the entire process.

In Section 3.1, we define the problem setup and notations. Sections 3.2.1–3.2.4 explain the components of the training-free alignment method: text transformation, text-style translation, modality summarization, and LLM reasoning augmentation. Section 3.3 describes the downstream task setup with minimal supervised training.

3.1 Problem Formalization

Suppose we have a set M of p modalities, $M = \{m_1, m_2, \ldots, m_p\}$. In the training phase, a subset of modalities $M_T \subseteq M$ is used. In the inference phase, a different subset $M_I \subseteq M$ is utilized. This subset meets the critical condition $M_T \cap M_I = \emptyset$, ensuring no overlap in modalities between training and inference.

Within this framework, we define two distinct datasets: one for the training phase and another for the inference phase. The training dataset D_T consists of n_T samples. Each sample x is restricted to M_T, denoted as $D_T = \{(x^i_{M_T}, y^i)\}_{i=1}^{n_T}$. Similarly, the inference dataset D_I consists of n_I samples, each restricted to M_I, formalized as $D_I = \{(x^i_{M_I}, y^i)\}_{i=1}^{n_I}$. Our algorithms are designed to build the model F on D_T and evaluate unseen data and modality combinations in D_I. This evaluation measures the model's ability to generalize knowledge in zero-shot multimodal learning.

3.2 Training-Free Alignment Method

3.2.1 Text Transformation

We map heterogeneous modalities into a shared textual space using pre-trained modality-to-text models. These transformations are performed without training. For image modality, we use image captioning models to generate descriptions. For tables, we follow the TabLLM (Hegselmann et al., 2023) template-based serialization. Text is retained in its original form. This conversion harmonizes inputs at the representation level without requiring task-specific finetuning. Real-world Example in Appendix C.6.1

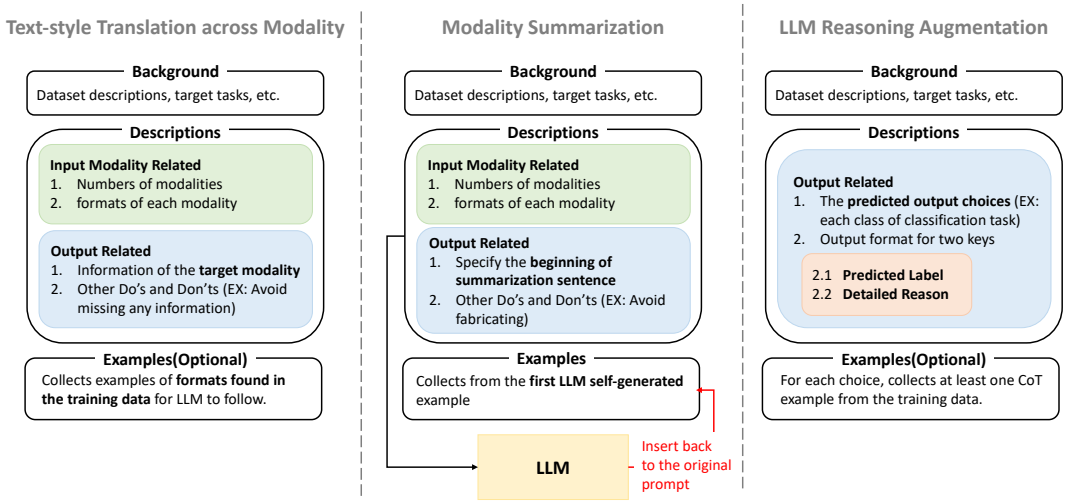

Figure 4: Examples of prompt templates for each modules

3.2.2 Text-style Translation across Modality

Despite converting all inputs to text, style and semantic gaps remain due to modality-specific formatting. We address this with in-context learning using LLMs, which learn to translate textual inputs from inference-time modalities into the style of training-time modalities. This reduces distribution shift without model finetuning. Prompts use three few-shot examples from D_T. Real-world Example in Appendix C.6.2

3.2.3 Modality Summarization

To enhance alignment and eliminate redundancy, we use LLMs to summarize the transformed modality inputs. This improves representation uniformity and highlights shared content. The summarization prompt includes a demonstration constructed from D_T, and one-shot in-context learning is applied for each sample. Real-world Example in Appendix C.6.3

3.2.4 LLM Reasoning Augmentation

We augment the inputs with LLM-generated reasoning traces to incorporate external knowledge and explain predictions. Prompts include task-specific instructions and examples. No model training is involved; all augmentation is via in-context LLM calls. Real-world Example in Appendix C.6.4

3.3 Downstream Prediction

Following the training-free multimodal alignment, we evaluate performance on a range of downstream tasks, including regression, classification, and ranking. The aligned and summarized text representations are processed by a transformer encoder (Longformer (Beltagy et al., 2020)), followed by mean pooling and a lightweight Multilayer Perceptron (MLP) for final prediction. Only the MLP is trained during this phase, while all preceding components remain frozen. We use Cross Entropy Loss for classification tasks and Mean Squared Error for regression. This minimal training setup isolates the contribution of the alignment method.

4 Experiments

4.1 Experiment Setup

In all experiments, we primarily present results from GPT-4-Vision for image captioning, unless specified otherwise. For additional results involving other image caption models, please refer to Table 10. The detailed information of the dataset, metrics, computation resource are in Appendix A.4.

4.1.1 Competitors

We compare TAMML against several zero-shot cross-modality translation baselines, including embedding-based approaches and a naive GPT-4-vision method. A schematic of embedding-based pipelines is shown in Figure 2. We include two perturbation-diffusion models—SDEdit (Meng et al., 2022) and DDRM (Kawar et al., 2022)—and one GAN-based method, Idinvert (Zhu et al., 2020). These methods train generative models (diffusion or GANs) to map embeddings from unseen modalities into the training distribution. At test time, modality-specific encoders extract embeddings, which are transformed via generative models into aligned representations. For diffusion-based methods, we train a score-based model (Ho et al., 2020)

Training	Testing	SDEdit	DDRM	Idinvert	Naive MLLM Transformation	TAMML Methods LLaMa 3 8B	Mistral 7B	Mixtral 8x7B	Mixtral 8x22B	GPT-3.5
PetFinder | Accuracy ↑										
text+image	tabular	0.282	0.291	0.279	0.310	0.309	0.301	0.317	0.332	**0.348**
text+tabular	image	0.289	0.277	0.286	0.329	0.306	0.322	0.335	0.323	**0.380**
image+tabular	text	0.281	0.297	0.279	0.305	0.303	0.304	0.320	0.313	**0.355**
text	image+tabular	0.291	0.283	0.289	0.282	0.315	0.304	0.330	**0.368**	0.344
text	image	0.289	0.276	0.287	0.293	0.355	0.341	0.307	0.360	**0.374**
text	tabular	0.293	0.259	0.277	0.297	0.295	0.286	0.325	0.341	**0.357**
image	text+tabular	0.290	0.297	0.284	0.314	0.310	0.322	**0.346**	0.342	0.341
image	text	0.288	0.282	0.280	0.306	0.323	0.325	0.329	**0.330**	0.319
image	tabular	0.291	0.287	0.284	0.300	0.322	0.302	0.341	0.319	**0.348**
tabular	text+image	0.290	0.271	0.285	0.194	0.314	0.309	0.327	0.333	**0.360**
tabular	text	0.289	0.265	0.280	0.193	0.295	0.294	0.302	0.317	**0.364**
tabular	image	0.289	0.263	0.277	0.196	0.294	0.305	0.338	0.311	**0.364**
Average		0.289	0.279	0.282	0.277	0.312	0.310	0.326	0.332	**0.355**
Airbnb | MSE ↓										
text+image	tabular	0.935	0.600	0.799	0.365	0.303	0.371	0.326	**0.313**	0.367
text+tabular	image	0.656	0.778	0.643	0.957	0.626	0.466	0.451	**0.447**	0.508
image+tabular	text	0.514	0.565	0.781	0.695	0.413	0.325	**0.312**	0.359	0.332
text	image+tabular	1.548	0.914	0.915	0.438	0.315	0.368	0.323	**0.284**	0.421
text	image	1.513	0.895	1.010	0.524	0.537	0.521	0.439	**0.404**	0.520
text	tabular	1.061	0.824	0.931	0.759	0.308	0.348	0.345	**0.297**	0.448
image	text+tabular	0.556	0.530	0.602	0.457	0.431	**0.368**	0.382	0.392	0.395
image	text	0.678	0.589	0.759	0.423	0.439	0.389	**0.375**	0.421	0.391
image	tabular	0.592	0.538	0.516	0.668	0.459	0.452	0.487	**0.405**	0.414
tabular	text+image	0.637	0.675	0.662	0.480	0.467	0.347	0.310	0.379	**0.280**
tabular	text	0.569	0.693	0.707	0.477	0.481	0.341	0.313	0.339	**0.301**
tabular	image	0.609	0.715	0.615	0.913	0.627	0.461	**0.431**	0.535	0.551
Average		0.822	0.693	0.745	0.596	0.451	0.396	**0.375**	0.381	0.411
Avito | MSE ↓										
text+image	tabular	0.103	0.113	0.126	0.051	0.045	0.045	0.043	**0.041**	0.044
text+tabular	image	0.130	0.133	0.142	0.051	0.048	0.048	0.047	0.048	**0.046**
image+tabular	text	0.113	0.125	0.137	**0.040**	0.045	0.045	0.045	**0.043**	0.046
text	image+tabular	0.124	0.123	0.131	0.050	0.046	0.047	0.046	0.046	**0.045**
text	image	0.124	0.122	0.129	0.052	0.048	0.050	**0.047**	0.048	**0.047**
text	tabular	0.127	0.124	0.134	0.052	0.045	0.046	0.046	0.046	**0.044**
image	text+tabular	0.123	0.126	0.134	**0.044**	**0.044**	**0.044**	**0.044**	**0.044**	**0.044**
image	text	0.118	0.124	0.129	0.045	0.045	0.046	0.047	0.046	**0.045**
image	tabular	0.119	0.126	0.134	0.049	0.044	0.044	0.045	**0.043**	0.044
tabular	text+image	0.128	0.139	0.137	0.044	0.046	0.045	0.046	**0.044**	0.046
tabular	text	0.124	0.131	0.138	0.046	0.046	**0.044**	0.047	**0.044**	0.045
tabular	image	0.126	0.137	0.140	**0.044**	0.050	0.047	0.048	0.046	0.048
Average		0.122	0.127	0.135	0.048	0.046	0.046	0.046	**0.045**	**0.045**

Table 2: This table presents a detailed comparison, highlighting TAMML 's performance against all baseline models under modality mismatch scenarios. The PetFinder dataset uses accuracy as the key evaluation metric. The Airbnb dataset and the Avito dataset both use Mean Squared Error (MSE) as the key evaluation metric.

using DDIM (Song et al., 2020) as the backbone. For GANs, we adopt StyleGAN (Karras et al., 2019). All baselines use modality-specific foundation model encoders, followed by alignment layers and a transformer for downstream prediction after fine-tuning. We also include a LanguageBind-style training as baseline, which serves as an upper bound for our method when sufficient paired data is available.

4.2 Main Results

Here, we articulate our hypotheses and address the research questions to evaluate the effectiveness of TAMML . Q1: Under test-time unseen modality scenarios, is TAMML better than the embedding-based SOTA zero-shot cross modality translation? Q2 (follow Q1): Is TAMML still effective for situations in which the testing modality has been involved during training? (i.e. training: all modalities, testing: some of the modalities) and other modality mismatch combinations? The following three questions are presented in Appendix C. Q3: Is text representation generally more robust than embedding representation for cross-modality translation? Q4: What is the performance of text-based solutions versus embedding-based solutions when training and testing modalities are exact identical? Q5: How about comparing TAMML to non-zero-shot transferring methods, such as domain adaptation? Q6: How does LanguageBind-Style training serves as a upper vound performance compared to TAMML when sufficient paired multimodal data is available?

Additional inquiries, Q3, Q4, Q5, Q6 and de-

Training	Testing	Pet | Acc ↑				Airbnb | MSE ↓				Avito | RMSE ↓			
		SDEdit	DDRM	Idinvert	TAMML	SDEdit	DDRM	Idinvert	TAMML	SDEdit	DDRM	Idinvert	TAMML
all	tabular	0.282	0.269	0.252	**0.338**	0.428	0.621	0.732	**0.270**	0.108	0.123	0.133	**0.041**
all	image	0.285	0.286	0.267	**0.356**	0.566	0.649	0.711	**0.486**	0.114	0.123	0.136	**0.044**
all	text	0.284	0.284	0.274	**0.349**	0.502	0.601	0.695	**0.253**	0.113	0.123	0.131	**0.044**
all	image+tabular	0.307	0.276	0.256	**0.382**	0.394	0.556	0.683	**0.251**	0.118	0.124	0.129	**0.042**
all	text+tabular	0.315	0.306	0.283	**0.377**	0.353	0.470	0.544	**0.185**	0.124	0.124	0.134	**0.041**
all	text+image	0.292	0.286	0.244	**0.378**	0.489	0.537	0.673	**0.212**	0.110	0.115	0.125	**0.043**
all	all	0.334	0.304	0.281	**0.395**	0.345	0.463	0.542	**0.178**	0.109	0.114	0.123	**0.042**
	Average	0.300	0.287	0.265	**0.368**	0.440	0.557	0.654	**0.262**	0.112	0.121	0.130	**0.042**
all	comb	0.294	0.285	0.263	**0.362**	0.455	0.572	0.673	**0.299**	0.115	0.122	0.131	**0.043**
text+image	comb	0.282	0.290	0.277	**0.320**	0.643	0.623	0.747	**0.331**	0.108	0.120	0.132	**0.043**
text+tabular	comb	0.290	0.278	0.282	**0.341**	0.517	0.645	0.674	**0.318**	0.120	0.127	0.134	**0.044**
image+tabular	comb	0.296	0.296	0.269	**0.358**	0.452	0.524	0.666	**0.236**	0.116	0.121	0.132	**0.042**
	Average	0.291	0.287	0.273	**0.345**	0.517	0.591	0.690	**0.286**	0.115	0.123	0.132	**0.043**

Table 3: This table presents a detailed comparison, highlighting TAMML 's performance against embedding-based translation baselines when the model is trained on all modalities and tested on different subset modalities. The result shows the effectiveness of TAMML even when the testing modality has been involved during training. Mixtral 8x7B is used in this experiment.

tailed descriptions of our experimental setup, including model checkpoints, hyperparameters, and dataset specifics, are provided in Appendix C.

Q1: Under Modality Mismatch Scenarios, How Does TAMML Compare To the SOTA Zero-shot Cross Modality Translation? In Q1, we focus on situations where training and testing modalities are completely different. We mainly compare our results to several zero-shot cross-modality data translation methods. The key findings outlined in Table 2 underscore the superior performance of TAMML , which achieves substantial gains over competing baselines across various modality combinations and different foundation models. Specifically, with the best-performing GPT-3.5 on the PetFinder dataset, TAMML enhances accuracy by an average of approximately **21%**, significantly outperforming the best-performing baseline methods. Similarly, in the Airbnb dataset, TAMML achieves an average reduction in mean square error of around **54%**, dwarfing the maximum 16% error reduction seen with alternative baselines. Further examination of the differences among various foundation models within the TAMML framework underscored the impact of model size on quality. For instance, Mixtral 8x22B improved accuracy by 7% on the PetFinder dataset compared to Mistral 7B. For complex tasks such as summarization and translation, larger models performed better. However, even smaller models showed improvement compared to baselines in mismatch scenarios. These results suggest that the proposed strategy, which integrates LLMs' in-context learning with foundation models, holds a decisive edge over all existing methods.

Q2: Is the Proposed Solution Effective When There is No Train/Test Modality Mismatch or Only Partial Mismatch in Various Modality Combinations? The key findings outlined in Table 3 underscore the superior performance of TAMML , which still achieves substantial gains over competing baselines across various modality combinations. These results suggest that despite no modality mismatching, our strategy holds a decisive edge over embedding-based methods. Specifically, on the PetFinder dataset, our technique enhances accuracy by an average of approximately **22.6%**, significantly outperforming the best-performing embedding-based methods. Similarly, in the Airbnb dataset, TAMML achieves a decrease of approximately **40.5%** in mean squared error, indicating a significant improvement in prediction accuracy. Moreover, in the Avito dataset, the decrease is even more pronounced, with a reduction of approximately **62.5%** in mean squared error when applying TAMML .

4.3 Ablation Studies

This section evaluates the contribution of individual components in TAMML through a series of ablation studies using GPT-3.5 as the foundation LLM. We incrementally add each module and summarize the performance impact in Table 14.

We first observe that converting modality features into text (*Text Transformation*) improves performance by 2% over embedding-based methods like SDEdit, suggesting reduced modality mismatch. However, tabular data sees a 10% drop, likely due to rigid text formatting that diverges from natural language style. Adding *Modality Summarization* significantly boosts tabular accuracy from

Training	Testing	PetFinder | Accuracy ↑				
		SDEdit	Text Transformation	+Modality Summarization	+Reasoning Augmentation	+Text-style Translation
text+image	tabular	0.282	0.310	0.321	0.338	**0.348**
text+tabular	image	0.289	0.329	0.365	0.363	**0.380**
image+tabular	text	0.281	0.305	0.295	0.321	**0.355**
text	image+tabular	0.291	0.282	0.296	0.343	**0.344**
text	image	0.289	0.293	0.298	0.341	**0.374**
text	tabular	0.293	0.297	0.318	0.315	**0.357**
image	text+tabular	0.290	0.314	0.289	0.325	**0.341**
image	text	0.288	0.306	0.330	**0.336**	0.319
image	tabular	0.291	0.300	0.307	0.303	**0.348**
tabular	text+image	0.290	0.194	**0.366**	0.341	0.360
tabular	text	0.289	0.193	0.306	0.327	**0.364**
tabular	image	0.289	0.196	0.357	0.353	**0.364**
Average $\pm$ Variance ($\times 10^{-4}$)		0.289 ± 0.12	0.277 ± 25.91	0.321 ± 7.2	0.334 ± 2.5	$\mathbf{0.355 \pm 2.4}$

Table 4: Ablation studies on various components of TAMML . Our observations reveal that text transformations significantly enhance performance across all modality combinations except for tabular data, which is in fixed formatted text. The formatting issue is effectively solved by incorporating a summarization module, resulting in a substantial enhancement in performance. Furthermore, the inclusion of both the translation module and the reasoning augmentation module leads to further improvements in overall performance.

0.277 to 0.321, helping normalize structure and reduce formatting inconsistencies. Incorporating *LLM Reasoning Augmentation* further improves the average score to 0.334, while also reducing performance variance across different modality pairs. Finally, applying *Text-Style Translation* contributes the largest gain, raising the average to 0.355. This step proves especially effective when there is a persistent style gap between training and inference, as in image-to-tabular translation, by helping the model maintain consistent mapping across phases.

5 Computational Cost and Practicality Analysis

We report the average runtime latency for each component of TAMML and compare it with existing baselines in Table 5. While our alignment pipeline introduces some overhead (1.76s per sample), it remains comparable to SDEdit (1.63s) and DDRM (1.89s), and is significantly more practical than methods like Idinvert (2.01s). The overhead primarily stems from LLM in-context operations, which are fully parallelizable.

TAMML does not require model retraining and allows flexible module use, enabling practical deployment across different latency budgets. Though our method incurs additional token usage (400-600 tokens per sample), this remains well within modern LLM context limits (8k-32k). Overall, TAMML balances scalability, robustness, and deployment simplicity.

Table 5: Average Latency per Sample (in seconds) for Each Component and Baseline Methods.

Method / Operation	**Latency (s)**	**Method / Operation**	**Latency (s)**
TAMML		Baseline	
Text Transformation	0.42	SDEdit	1.63
Text-style Translation	0.37	DDRM	1.89
Modality Summarization	0.45	Idinvert	2.01
LLM Augmentation	0.52	Direct Input	0.25
Total	**1.76**		

6 Conclusion and Future Directions

Our study has effectively harnessed Large Language Models (LLMs) for multimodal learning, creating a unified semantic space that integrates various data modalities through text in a complete in-context learning manner. Through techniques such as text transformation, text-style translation, summarization, and reasoning augmentation, we have demonstrated our proposed TAMML alignment that the operations performed in the text domain using in-context learning with LLMs can achieve comparable performance to traditional methods operating in embedding space. This approach not only opens new avenues in multimodal learning but also underscores the significant potential and advantages of text as a unifying medium. Future efforts will focus on refining TAMML for broader multimodal tasks as well as other challenges in multimodal learning through text-centric approach, such as modality robustness, modality collapse and modality competition.

7 Limitation

One of the key limitations of our study is the inherent randomness of the LLM text generation. Due to cost constraints, we only performed three run for each of our experiments. While this approach provides a general indication of performance, it may not fully capture the variability and could lead to less accurate conclusions. More extensive experimentation with a larger number of runs would be necessary to achieve a higher degree of confidence in the results. In addition, we cannot guarantee to reproduce the results on the closed-source LLMs.

Use of AI Assistants

ChatGPT was utilized to refine paper writing. The authors paid careful attention to ensuring that AI-generated content is accurate and aligned with the author's intentions.

References

2023. Inside airbnb : Hawaii. Accessed on: 10 September, 2023.

Rameen Abdal, Yipeng Qin, and Peter Wonka. 2020. Image2stylegan++: How to edit the embedded images? In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8296–8305.

Mongrel Jedi Addison Howard, MichaelApers. 2018. Petfinder.my adoption prediction.

Niket Agarwal, Arslan Ali, Maciej Bala, Yogesh Balaji, Erik Barker, Tiffany Cai, Prithvijit Chattopadhyay, Yongxin Chen, Yin Cui, Yifan Ding, et al. 2025. Cosmos world foundation model platform for physical ai. *arXiv preprint arXiv:2501.03575*.

Zeynep Akata, Florent Perronnin, Zaid Harchaoui, and Cordelia Schmid. 2015. Label-embedding for image classification. *IEEE transactions on pattern analysis and machine intelligence*, 38(7):1425–1438.

Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katherine Millican, Malcolm Reynolds, et al. 2022. Flamingo: a visual language model for few-shot learning. *Advances in Neural Information Processing Systems*, 35:23716–23736.

Iz Beltagy, Matthew E Peters, and Arman Cohan. 2020. Longformer: The long-document transformer. *arXiv preprint arXiv:2004.05150*.

NVIDIA Blog. 2024. Ai agents summarize videos to supercharge search. Accessed: 10-Feb-2025.

Maxime Bucher, Stéphane Herbin, and Frédéric Jurie. 2017. Generating visual representations for zero-shot classification. In *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pages 2666–2673.

Haoxing Chen, Yaohui Li, Yan Hong, Zhuoer Xu, Zhangxuan Gu, Jun Lan, Huijia Zhu, and Weiqiang Wang. 2023. Boosting audio-visual zero-shot learning with large language models. *arXiv preprint arXiv: 2311.12268*.

Shin-I Cheng, Yu-Jie Chen, Wei-Chen Chiu, Hung-Yu Tseng, and Hsin-Ying Lee. 2023. Adaptively-realistic image generation from stroke and sketch with diffusion model. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 4054–4062.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. 2010. An image is worth 16x16 words: Transformers for image recognition at scale. arxiv 2020. *arXiv preprint arXiv:2010.11929*.

Weihao Gao, Zhuo Deng, Zhiyuan Niu, Fuju Rong, Chucheng Chen, Zheng Gong, Wenze Zhang, Daimin Xiao, Fang Li, Zhenjie Cao, et al. 2023. Ophglm: Training an ophthalmology large language-and-vision assistant based on instructions and dialogue. *arXiv preprint arXiv:2306.12174*.

Rohit Girdhar, Alaaeldin El-Nouby, Zhuang Liu, Mannat Singh, Kalyan Vasudev Alwala, Armand Joulin, and Ishan Misra. 2023. Imagebind: One embedding space to bind them all. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 15180–15190.

Yury Gorishniy, Ivan Rubachev, Valentin Khrulkov, and Artem Babenko. 2021. Revisiting deep learning models for tabular data. *Advances in Neural Information Processing Systems*, 34:18932–18943.

Ivan Guz, Julia Elliott, Myagkikh Konstantin, Sohier Dane, Vladislav Kassym, and Wendy Kan. 2018. Avito demand prediction challenge.

Stefan Hegselmann, Alejandro Buendia, Hunter Lang, Monica Agrawal, Xiaoyi Jiang, and David Sontag. 2023. Tabllm: Few-shot classification of tabular data with large language models. In *International Conference on Artificial Intelligence and Statistics*, pages 5549–5581. PMLR.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. 2020. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851.

Tero Karras, Samuli Laine, and Timo Aila. 2019. A style-based generator architecture for generative adversarial networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4401–4410.

Bahjat Kawar, Michael Elad, Stefano Ermon, and Jiaming Song. 2022. Denoising diffusion restoration models. *Advances in Neural Information Processing Systems*, 35:23593–23606.

Hari Chandana Kuchibhotla, Sumitra S Malagi, Shivam Chandhok, and Vineeth N Balasubramanian. 2022. Unseen classes at a later time? no problem. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9245–9254.

Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. 2023a. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. *arXiv preprint arXiv:2301.12597*.

KunChang Li, Yinan He, Yi Wang, Yizhuo Li, Wenhai Wang, Ping Luo, Yali Wang, Limin Wang, and Yu Qiao. 2023b. Videochat: Chat-centric video understanding. *arXiv preprint arXiv:2305.06355*.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. 2023. Visual instruction tuning.

Yang Long, Li Liu, Fumin Shen, Ling Shao, and Xuelong Li. 2017. Zero-shot learning using synthesised unseen visual data with diffusion regularisation. *IEEE transactions on pattern analysis and machine intelligence*, 40(10):2498–2512.

Leland McInnes, John Healy, and James Melville. 2018. Umap: Uniform manifold approximation and projection for dimension reduction. *arXiv preprint arXiv:1802.03426*.

C. Meng, Y. He, Y. Song, J. Song, J. Wu, J.-Y. Zhu, and S. Ermon. 2022. Sdedit:guided image synthesis and editing with stochastic differential equations. *International Conference on Learning Representations*.

OpenAI. 2023. Openai models api.

Mark Palatucci, Dean Pomerleau, Geoffrey E Hinton, and Tom M Mitchell. 2009. Zero-shot learning with semantic output codes. *Advances in neural information processing systems*, 22.

Zhiliang Peng, Wenhui Wang, Li Dong, Yaru Hao, Shaohan Huang, Shuming Ma, and Furu Wei. 2023. Kosmos-2: Grounding multimodal large language models to the world. *arXiv preprint arXiv:2306.14824*.

Yichun Shi, Xiao Yang, Yangyue Wan, and Xiaohui Shen. 2022. Semanticstylegan: Learning compositional generative priors for controllable image synthesis and editing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11254–11264.

Jiaming Song, Chenlin Meng, and Stefano Ermon. 2020. Denoising diffusion implicit models. *arXiv:2010.02502*.

Eric Tzeng, Judy Hoffman, Kate Saenko, and Trevor Darrell. 2017. Adversarial discriminative domain adaptation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7167–7176.

Sheng Wang, Zihao Zhao, Xi Ouyang, Qian Wang, and Dinggang Shen. 2023a. Chatcad: Interactive computer-aided diagnosis on medical image using large language models. *arXiv preprint arXiv:2302.07257*.

Wenlin Wang, Yunchen Pu, Vinay Verma, Kai Fan, Yizhe Zhang, Changyou Chen, Piyush Rai, and Lawrence Carin. 2018. Zero-shot learning via class-conditioned deep generative models. In *Proceedings of the AAAI conference on artificial intelligence*, volume 32.

Zihao Wang, Clair Vandersteen, Charles Raffaelli, Nicolas Guevara, François Patou, and Hervé Delingette. 2021. One-shot learning for landmarks detection. In *Deep Generative Models, and Data Augmentation, Labelling, and Imperfections: First Workshop, DGM4MICCAI 2021, and First Workshop, DALI 2021, Held in Conjunction with MICCAI 2021, Strasbourg, France, October 1, 2021, Proceedings 1*, pages 163–172. Springer.

Zihao Wang, Yingyu Yang, Maxime Sermesant, Hervé Delingette, and Ona Wu. 2023b. Zero-shot-learning cross-modality data translation through mutual information guided stochastic diffusion. *arXiv preprint arXiv:2301.13743*.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. 2022. Chain-of-thought prompting elicits reasoning in large language models. *Advances in Neural Information Processing Systems*, 35:24824–24837.

Shengqiong Wu, Hao Fei, Leigang Qu, Wei Ji, and Tat-Seng Chua. 2023. Next-gpt: Any-to-any multimodal llm. *arXiv preprint arXiv:2309.05519*.

Shukang Yin, Chaoyou Fu, Sirui Zhao, Ke Li, Xing Sun, Tong Xu, and Enhong Chen. 2023. A survey on multimodal large language models. *arXiv preprint arXiv:2306.13549*.

Bin Zhu, Bin Lin, Munan Ning, Yang Yan, Jiaxi Cui, HongFa Wang, Yatian Pang, Wenhao Jiang, Junwu Zhang, Zongwei Li, et al. 2023. Languagebind: Extending video-language pretraining to n-modality by language-based semantic alignment. *arXiv preprint arXiv:2310.01852*.

Jiapeng Zhu, Yujun Shen, Deli Zhao, and Bolei Zhou. 2020. In-domain gan inversion for real image editing. In *European conference on computer vision*, pages 592–608. Springer.

A Experiment Detail Setup

A.1 GPU Settings and Computational Resources

All experiments were conducted on NVIDIA A6000 GPUs with GB memory. Most evaluations used single-GPU inference for LLM components.

The latency analysis in Table 5 was benchmarked using batch size 1 and greedy decoding for all LLM components. Prompt construction and tokenization were handled using Hugging Face Transformers and OpenAI APIs (GPT-3.5/GPT-4-V). All runs were executed on Linux systems with CUDA 12.1 and PyTorch 2.1.

A.2 Model Checkpoints

We conduct all experiments with GPT-3.5-turbo as the LLM and GPT-4-vision as the image caption model through OpenAI APIs (OpenAI, 2023), except for the analysis experiment that compares different LLMs and foundation models.

Model	Checkpoints
GPT-3.5-turbo	gpt-3.5-turbo-0613
GPT-4-vision	gpt-4-vision-preview
BLIP2	huggingface: Salesforce/blip-image-captioning-large
Kosmos2	huggingface: microsoft/kosmos-2-patch14-224
Vision Transformer	huggingface: google/vit-base-patch16-224
Flamingo	huggingface: openflamingo/OpenFlamingo-9B-vitl-mpt7b
Longformer	huggingface: allenai/longformer-base-4096
LLAMA-2-7b-chat	huggingface: meta-llama/Llama-2-7b-chat
LLAMA-2-13b-chat	huggingface: meta-llama/Llama-2-13b-chat
LLAMA-2-70b-chat	huggingface: meta-llama/Llama-2-70b-chat
Mixtral-8x7b	huggingface:mistralai/Mixtral-8x7B-Instruct-v0.1

Table 6: Model checkpoints.

A.3 Hyperparameters

Model	Hyperparameters
GPT-3.5-turbo	temperature=1, max_tokens=4096
GPT-4-vision	temperature=0.8, max_tokens=300
BLIP2	default parameter
Kosmos2	default parameter
Vision Transformer	default parameter
Flamingo	default parameter
Longformer	max_length=2048
LLAMA-2-7b-chat	temperature=1, max_tokens=4096
LLAMA-2-13b-chat	temperature=1, max_tokens=4096
LLAMA-2-70b-chat	temperature=1, max_tokens=4096
Mixtral	temperature=1, max_tokens=4096
SDEdit	batch_size=1, sample_step=3, noise_scale=150
DDRM	batch_size=1, degredation_type=deno, noise=1.5
Idinvert	batch_size=64, gradient_accumulate=8, network_capacity=32

Table 7: Hyper parameters.

A.4 Dataset

A.4.1 PetFinder.my Adoption Prediction (Addison Howard, 2018)

examines what factors predict how quickly a pet is adopted after being listed. The dataset is a composite of the following modalities:

- Text: contains the description of the status of the pet
- Image: contains a profile photo of the pet
- Tabular: contains basic information, such as gender and breed.

A.4.2 Airbnb Pricing Prediction (ins, 2023)

is composed of the following modalities used for making a regression prediction of housing prices:

- Text: contains the human-written description of the homestay, the neighborhood description, and the host's profile.
- Image: contains images of the homestay
- Tabular: delivers essential details such as location, rating score, and review counts.

A.4.3 Avito Demand Prediction (Guz et al., 2018)

predicts the likelihood of an ad selling something based on user item and context features:

- Text: contains the ad title and description.
- Image: contains a profile photo of the item.
- Tabular: contains basic information, such as region, city, item category, etc.

PetFinder	
Field	Value
url	https://www.kaggle.com/competitions/petfinder-adoption-prediction
# instances	13453
tabular columns	23

Airbnb	
Field	Value
url	http://insideairbnb.com/get-the-data/
# instances	12184
tabular columns	30

A.5 Foundation Models

For image modality, we utilize the embedding layer and tokenization method of the Vision Transformer (Dosovitskiy et al., 2010). This process splits the image into fixed-size patches and then projects each patch to obtain embeddings. For tabular modality, we employ the FT-Transformer(Gorishniy et al., 2021) method to encode, dividing tabular features into numeric and categorical with separate projection layers for dimension enhancement. For text modality, the embedding layer of Longformer(Beltagy et al., 2020) is used for projection.

Avito	
Field	Value
url	https://www.kaggle.com/competitions/avito-demand-prediction/data
# instances	7000
tabular columns	18

Table 8: Dataset Meta Info

B Analysis and Discussion

In this section, we delve into a series of analyses and discussions, extracting valuable insights from our discoveries. Specifically, we provide more supportive evidence with visualization and distribution distance measurements.

B.1 Visualization for Distribution Alignment

In Section C.1, we have validated the effectiveness of text transformation in TAMML through experimental performance. Furthermore, we visualized 1,400 data points in these modalities with their position-aware embeddings using UMAP (McInnes et al., 2018) in Figure. 5. The left figure illustrates the original distributions of image and text embeddings, while the right figure displays the corresponding distributions after the summarization module in TAMML . We observe that the distribution boundaries between image and text modalities become less distinct, which indicates they are closer in the semantic space. To be more precise, TAMML significantly reduces the average instance Euclidean distance between image and text in the semantic space from 10.213 to 0.411 as shown in Table 9.

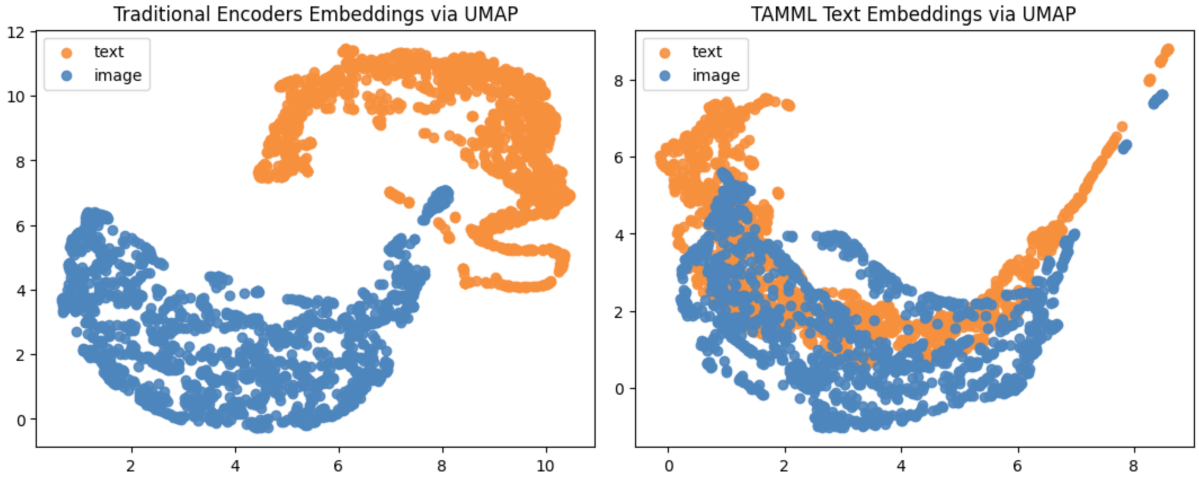

Figure 5: The left and right pictures illustrate the visualizations of embeddings for image and text data, respectively, before and after our processes.

	w/o	Normalization	Standardization
Embedding	10.213	5.444	39.151
Text	**0.411**	**0.101**	**0.584**

Table 9: Averaged Euclidean distance between all modalities. Text representation shows a more aligned distribution between modalities compared to embedding representation.

B.2 Effects of the Image Caption Models

Some might argue that the improvement in text transformation in TAMML could be attributed to the superior GPT-4 model. To investigate this, we replaced the different image caption models in our architecture with smaller open-source models. We conducted ablation studies focusing on the performance of four image foundation models. Specifically, we showed that our approach maintains strong performance even with smaller models. Table 10 showcases the results averaged across twelve training-inference modality combinations. The results suggest that using smaller image caption models does not necessarily result in significantly inferior performance with TAMML .

Image Caption Models				
Pet | Acc ↑	Blip2	Kosmos2	Flamingo	GPT4
Average	0.303	0.299	0.293	**0.307**

Table 10: Image caption model comparison: Each number presented here is an average derived from twelve modality combination experiments. In general, we can infer that the foundation model has only a limited impact on TAMML .

B.3 In-context Modality Transfer Outperforms Zero-shot Learning Based Methods

Text-style translation across modalities in TAMML transforms the training modality combination into the testing modality combination to reduce the semantic gap between them using LLMs. Similar concepts are used in zero-shot learning baselines, which create a generative model for modality translation. For comparison, we collected different pairs of training and testing data and created visualizations for each one of them.

Orange is the source modality, blue is the target modality, and purple is the source modality after transformation. Visualization results of Ours are shown in Figure 6. Visualization results of SDEdit are shown in Figure 7. As the results indicate, our translation effectively maps to closely align with the target modality in semantic space.

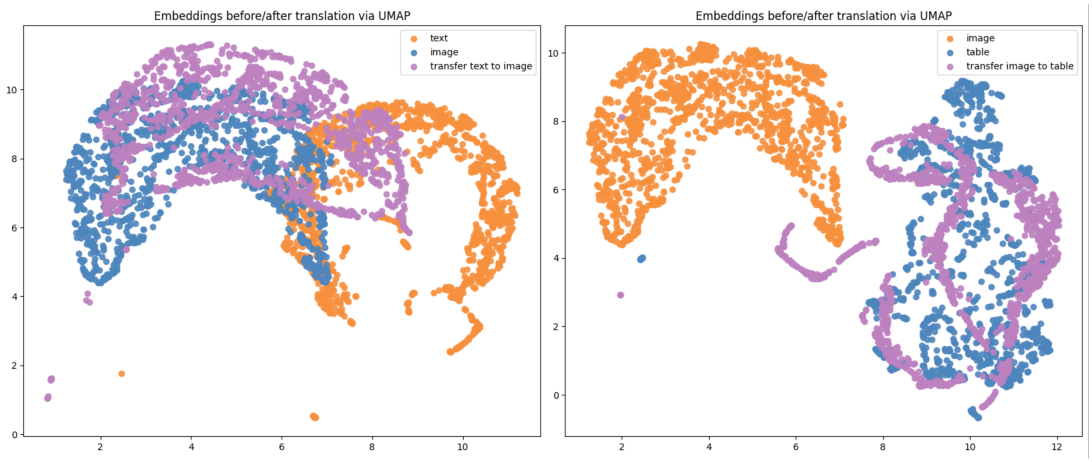

Figure 6: Cross Modality Translation (Ours): training data map to the distribution of target modality.

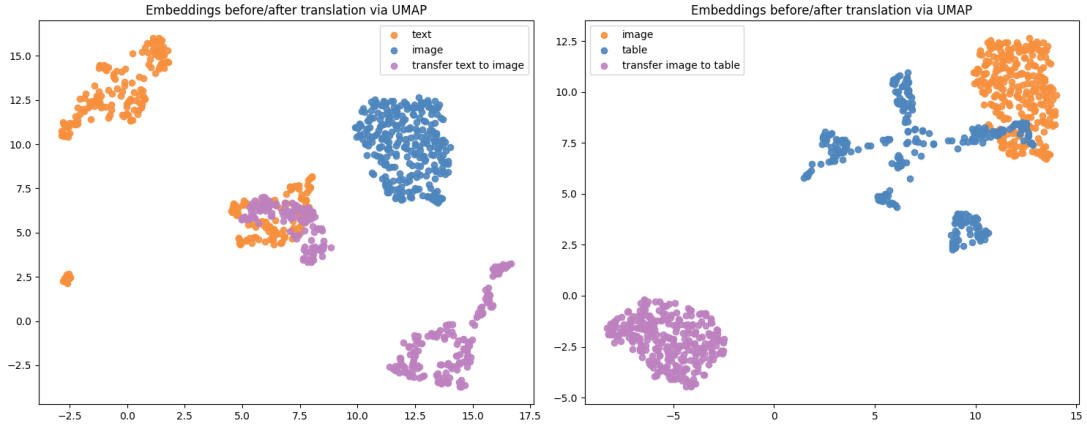

Figure 7: Cross Modality Translation (PetFinder / SDEdit): training data map to the distribution of target modality.

C Extra Experiment Results

In this section, we try to answer additional inquiries, Q4 and Q5, where Q4 explores the performance of text-based solutions versus embedding-based solutions when training and testing modalities are identical, and Q5 compares our zero-shot in-context learning approach to non-zero-shot methods, such as domain adaptation.

C.1 Q3: Is Text Representation Generally More Robust Than Embedding Representation For Cross Modality Translation?

In this section, we aimed to understand the trade-off between performance and flexibility when converting various modalities from embedding into text, especially under modality mismatch conditions.

C.1.1 MLLMs Baseline

Previous experiments in Section 4.2 and Section 4.2 cannot compare text representation and embedding representation since converting modalities into text involves different foundation models, each with different capabilities. For a fair comparison of performance between text representation and embedding representation, the most appropriate approach is to utilize multimodal Language Model Models (MLLMs). This ensures fairness in the comparison because all modalities are converted from the same foundation model. Therefore, we applied the following SOTA MLLMs in our experiments: Kosmos-2 (Peng et al., 2023) and Flamingo (Alayrac et al., 2022). In our experiments, we leverage MLLMs as both pre-trained feature extractors and text decoders. We then employ mean pooling to aggregate representations, followed by using an MLP as a backbone model to generate predictions. We aim to compare the performance gaps between a downstream model trained on images and another trained on image captions (attributed to the dataset).

C.1.2 Results

Results in Table 11 consistently reveal that downstream models trained on image captions exhibit less performance degradation compared to those trained on image embeddings in scenarios of modality mismatch. This observation holds true across all state-of-the-art multimodal LLMs we investigated. Such results strongly suggest that cross-modality translation within text representations, as facilitated by TAMML , proves to be a more effective and robust strategy than utilizing embedding representations when faced with modality mismatch conditions.

Pet | Acc ↑	Flamingo		Kosmos2	
Test/Train	caption	image	caption	image
text	**-0.07**	-0.10	**-0.09**	-0.11
tabular	**-0.08**	-0.10	**-0.12**	-0.21
text+tabular	**-0.08**	-0.11	**-0.10**	-0.15
Air | MSE ↓	**Flamingo**		**Kosmos2**	
Test/Train	caption	image	caption	image
text	**-0.00**	-0.06	**-0.01**	-0.03
tabular	**-0.03**	-0.07	**-0.05**	-0.05
text+tabular	**-0.02**	-0.07	**-0.04**	-0.03

Table 11: Text representation shows consistently less performance degradation for cross-modality translation when explicitly compared to embedding representation. Both representations are derived from the same Multimodal LLMs for fair comparison. Nevertheless, transforming from image to caption has a slight performance reduction.

C.2 Q4: Text-based Solutions Versus Embedding-based Solutions When Training And Testing Modalities Are Identical

Table 12 provides the experiment results under no train/test modality mismatch.

Train & Test	Regular (Embedding)	TAMML (Text)
text	0.352	**0.382**
image	0.273	**0.369**
tabular	**0.429**	0.394
text+image	0.286	**0.400**
text+tabular	**0.411**	0.404
image+tabular	0.403	**0.408**

Table 12: Experiment results under no train/test modality mismatch condition. Under this condition, TAMML does not show performance degradation and even performs better in several modality combinations. The regular method means the downstream model is trained on embedding representations. Note that this result differs from the result in Table 11 because the foundation models used for generating embedding and text representations are not the same.

C.3 Q5: How does TAMML compare to non-zero-shot methods?

Table 13 provides the experiment results under modality mismatch with different test time finetuning settings (not zero-shot). The settings are as follows:

- no finetuning: complete mismatch scenario same as main result experiments.
- unsupervised domain adaptation: finetune- the downstream model given the information of inference modality but without labels. We adopted the ADDA (Tzeng et al., 2017) method.
- supervised training (with all modalities): the downstream model given the information of paired train/inference time modality with labels. This means that the modality used in testing is fully trained.

Train	Test	no finetuning: Emb	no finetuning: TAMML	unsupervised domain adaptation	supervised training (all modalities)
text	image	0.288	**0.374**	0.195	0.338
text	tabular	0.289	0.357	0.281	**0.359**
image	text	0.270	**0.319**	0.276	0.306
image	tabular	0.273	0.348	0.276	**0.359**
tabular	text	0.289	**0.364**	0.195	0.306
tabular	image	0.279	**0.364**	0.195	0.338

Table 13: The experiment results showed a condition with other non-zero-shot methods. Under this condition, TAMML shows no performance degradation and even performs better in several modality combinations in zero-shot.

C.4 Q6: Upper-Bound Performance with LanguageBind-style Training

To contextualize the performance of TAMML , we conducted a small-scale experiment following the LanguageBind (**?**) training strategy. Using identical modality encoders, we fine-tuned them on the PetFinder dataset to establish an upper-bound reference for supervised, embedding-based approaches. As expected, the fine-tuned model achieved higher accuracy across modality translation settings:

These results confirm that training-based methods can achieve higher performance when sufficient paired data is available. However, our in-context, zero-shot method still performs competitively without any fine-tuning or additional supervision, demonstrating its practical utility in resource-constrained settings.

C.5 Ablation Studies

This section explores the contribution of individual components within TAMML by conducting ablation studies. We incrementally add modules to evaluate their impact on performance, with findings summarized in Table 14. In this section, our TAMML framework employs the GPT-3.5 as the foundation LLM.

Training	Testing	PetFinder | Accuracy ↑				
		SDEdit	Text Transformation	+Modality Summarization	+Reasoning Augmentation	+Text-style Translation
text+image	tabular	0.282	0.310	0.321	0.338	**0.348**
text+tabular	image	0.289	0.329	0.365	0.363	**0.380**
image+tabular	text	0.281	0.305	0.295	0.321	**0.355**
text	image+tabular	0.291	0.282	0.296	0.343	**0.344**
text	image	0.289	0.293	0.298	0.341	**0.374**
text	tabular	0.293	0.297	0.318	0.315	**0.357**
image	text+tabular	0.290	0.314	0.289	0.325	**0.341**
image	text	0.288	0.306	0.330	**0.336**	0.319
image	tabular	0.291	0.300	0.307	0.303	**0.348**
tabular	text+image	0.290	0.194	**0.366**	0.341	0.360
tabular	text	0.289	0.193	0.306	0.327	**0.364**
tabular	image	0.289	0.196	0.357	0.353	**0.364**
Average $\pm$ Variance ($\times 10^{-4}$)		0.289 ± 0.12	0.277 ± 25.91	0.321 ± 7.2	0.334 ± 2.5	$\mathbf{0.355 \pm 2.4}$

Table 14: Ablation studies on various components of TAMML . Our observations reveal that text transformations significantly enhance performance across all modality combinations except for tabular data, which is in fixed formatted text. The formatting issue is effectively solved by incorporating a summarization module, resulting in a substantial enhancement in performance. Furthermore, the inclusion of both the translation module and the reasoning augmentation module leads to further improvements in overall performance.

C.5.1 Text Transformation

Compared to the embedding-based methods SDEdit, Table 14 shows converting modality features into text enhances performance by approximately 2%, indicating less modality mismatch during training and inference compared to embedding representations. This improvement is consistent across most data modalities, except for tabular data, which sees a decline of about 10%. This discrepancy is attributed to the fixed format of tabular text transformation, highlighting a significant style gap with more fluid, human-like writing, particularly impacting tabular data's inference performance.

C.5.2 Modality Summarization

Table 14 results indicate modality summarization improves tabular data accuracy significantly from 0.277 to 0.321 on average. After this stage, TAMML has already outperformed the strongest competitor SDEdit. This suggests that summarization effectively standardizes text formats into a cohesive style, mitigating heterogeneity in text transformation and enhancing data format alignment.

C.5.3 Reasoning Augmentation

Table 14 indicates that augmentation enhanced our average performance from 0.321 to 0.334. Additionally, we have observed that it contributes to a more stable performance across different scenarios. The variance value with augmentation is substantially lower than that without it.

C.5.4 Text-Style Translation across Modality

According to Table 14, text-style translation bridges training and inference phase gaps, with about 6% improvement from 0.334 to 0.355. This enhancement is particularly notable when the gap in textual style remains consistent across phases, as seen in the image-to-table scenarios. Such consistency aids in more accurate mapping function determination by the model.

C.6 Examples for Methodology Components

C.6.1 Text Transformation

A Real-world Example: When predicting diseases, we often have access to patients' pathology table reports, medical imaging, and audio of patient narration. First, we will perform text transformation on these data. For the images, we transfer it into captions such as "*The patient has sigmoid colon cancer causing an obstruction, which has led to dilation in the descending colon.*" For the tables, we transform it into statements like "*Histologic Type is Adenocarcinoma*" and "*Histologic Grade is Moderately differentiated.*"

Train Modalities	Test Modality	Accuracy
Text + Image	Tabular	0.369
Text + Tabular	Image	0.387
Image + Tabular	Text	0.374
Text	Image + Tabular	0.371
Text	Image	0.386
Text	Tabular	0.383
Image	Text + Tabular	0.379
Image	Text	0.384
Image	Tabular	0.385
Tabular	Text + Image	0.389
Tabular	Text	0.388
Tabular	Image	0.386
Average		**0.382**

Table 15: Fine-tuned LanguageBind-style embedding results on the PetFinder dataset.

For the audio files, we perform speech recognition and acquire descriptions such as "*I've been a little bloated for two weeks, and I have had only three bowel movements.*"

C.6.2 Text-style Translation across Modality

A Real-world Example: When the training combination for disease prediction includes table modality, and only video modality data is available at inference, we will perform text-style translation on the textual representation of audio data. Continuing the example from Section 3.2.1, the textual representation of the audio, "*I've been a little bloated for two weeks, and I have had only three bowel movements*,", is translated as "*Symptom is Bloating. The symptom is difficulty with bowel movements. Duration is Two weeks.*"

C.6.3 Modality Summarization

A Real-world Example: Building on the example from Section 3.2.1, now the input includes two modalities: image and table. We summarize the textual representations from these modalities. Here is how the summarization looks: "*The patient has moderately differentiated adenocarcinoma of the sigmoid colon, causing an obstruction and dilation of the descending colon.*"

C.6.4 LLM Reasoning Augmentation

A Real-world Example: Building on the example from Section 3.2.1, now the input includes two modalities: image and table. The current goal is to determine whether a patient requires hospital observation. The results after augmentation are as follows: "*The obstruction in the sigmoid colon can lead to increased risks of bowel perforation, where the colon wall might rupture due to increased pressure. This complication is serious and requires immediate medical intervention.*"