Mitigating Semantic Leakage in Cross-lingual Embeddings via Orthogonality Constraint

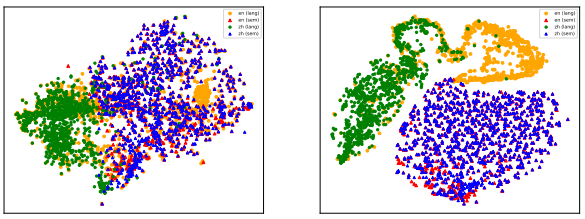
Dayeon Ki[1*] **Cheonbok Park**[2] **Hyunjoong Kim**[2]
[1]University of Maryland [2]NAVER Cloud
dayeonki@umd.edu

Abstract

Accurately aligning contextual representations in cross-lingual sentence embeddings is key for effective parallel data mining. A common strategy for achieving this alignment involves disentangling semantics and language in sentence embeddings derived from multilingual pre-trained models. However, we discover that current disentangled representation learning methods suffer from *semantic leakage*—a term we introduce to describe when a substantial amount of language-specific information is unintentionally leaked into semantic representations. This hinders the effective disentanglement of semantic and language representations, making it difficult to retrieve embeddings that distinctively represent the meaning of the sentence. To address this challenge, we propose a novel training objective, ORthogonAlity Constraint LEarning (ORACLE), tailored to enforce orthogonality between semantic and language embeddings. ORACLE builds upon two components: intra-class clustering and inter-class separation. Through experiments on cross-lingual retrieval and semantic textual similarity tasks, we demonstrate that training with the ORACLE objective effectively reduces semantic leakage and enhances semantic alignment within the embedding space.[1]

1 Introduction

Parallel datasets play a pivotal role in enhancing neural machine translation (NMT) performance (Michel and Neubig, 2018). However, acquiring high-quality parallel texts is challenging, especially for lower-resourced languages where monolingual data is more abundant (Niu et al., 2018). In this context, effective approaches for mining parallel data are essential for applying NMT in practical scenarios (Artetxe and Schwenk, 2019a).



(a) Semantic Leakage (b) ORACLE

Figure 1: Visualization of LaBSE sentence embeddings for 1,000 Chinese-English sentence pairs. Figure 1(a) shows substantial overlap between semantic and language-specific representations. This overlap is effectively mitigated by the proposed ORACLE method, as shown in Figure 1(b).

Recent approaches to this problem utilize cross-lingual sentence embeddings (Schwenk and Douze, 2017; Schwenk, 2018) generated by multilingual pre-trained encoders such as multilingual BERT (Devlin et al. (2019), mBERT) or XLM-RoBERTa (Conneau et al. (2020), XLM-R). These embeddings aim to align semantically similar sentences across languages into a unified latent space, facilitating the extraction of pseudo-parallel pairs (Wang et al., 2022). However, Tiyajamorn et al. (2021) and Kuroda et al. (2022) demonstrate that embeddings of parallel sentences from these encoders form clusters by language rather than by semantics. Building on this, they attempt to disentangle language-specific information from sentence embeddings, thereby distilling language-agnostic semantic embeddings.

In order to achieve this, two premises need to be considered. Given parallel sentence,

(1) How well are the semantic representations **aligned**?
(2) How well are the language-specific representations **separated**?

Prior works have primarily focused on the former, leaving the latter question underexplored. Figure 1 illustrates sentence embeddings of a parallel cor-

*Work done during internship at NAVER Cloud.
[1]Our code and models will be released at publication.

Proceedings of the 9th Workshop on Representation Learning for NLP (RepL4NLP-2024), pages 256–273

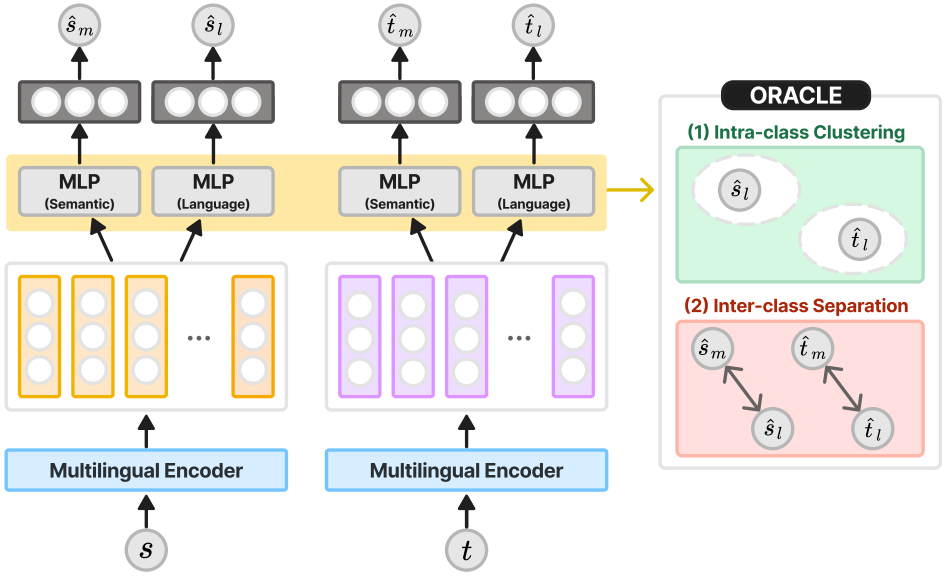


Figure 2: ORACLE objective for training semantic and language MLP networks. ORACLE is composed of two components: **(1) Intra-class clustering** for bringing related components closer in embedding space, **(2) Inter-class separation** for ensuring unrelated components to be distant. s and t represent source and target sentence input respectively. $\hat{s}_m$: source semantic representation; $\hat{s}_l$: source language representation; $\hat{t}_m$: target semantic representation; $\hat{t}_l$: target language representation.

pus pair, indicating that while semantics are well-aligned with previous disentanglement methods, there is still substantial overlap between language-specific and semantic information (Figure 1a). We define this issue as **semantic leakage**, which undermines the effectiveness of cross-lingual embeddings in accurately mining parallel pairs. By constraining orthogonality between semantic and language representations, we facilitate a clearer separation of language-specific information in the embedding space (Figure 1b).

In this work, we introduce **ORACLE** (ORthogonality Constraint LEarning), a training objective aimed at enforcing orthogonality between semantic and language-specific representations. Our goal is to render these two representations independent to each other, thus ensuring their clear differentiation in the embedding space (Mitchell and Steedman, 2015). ORACLE consists of two key components: intra-class clustering and inter-class separation. As shown in Figure 2, intra-class clustering aligns related components more closely, while inter-class separation enforces orthogonality between unrelated components. Our method is designed to be simple and effective, capable of being implemented atop any disentanglement methods.

We explore a range of pre-trained multilingual encoders (LASER (Artetxe and Schwenk, 2019b), InfoXLM (Chi et al., 2021), and LaBSE (Feng et al., 2022)) to generate initial sentence embeddings. Subsequently, we train each semantic and language multi-layer perceptrons (MLPs) with ORACLE to disentangle the sentence embeddings into semantics and language-specific information. Experimental results on both cross-lingual sentence retrieval tasks (Artetxe and Schwenk, 2019b; Zweigenbaum et al., 2017) and the Semantic Textual Similarity (STS) task (Cer et al., 2017) demonstrate higher performance on semantic embeddings and lower performance on language embeddings with ORACLE. This suggests that our method not only resolves semantic leakage but also enhances semantic alignment (§6). Our analysis further reveals that ORACLE leads to robust performance in challenging scenarios such as code-switching (§7.1).

To summarize, our contributions are threefold:
(1) We make the first attempt to address the issue of *semantic leakage*, wherein a substantial amount of language-specific information is leaked into semantic representations.
(2) We mitigate semantic leakage with ORACLE, a simple and effective training objective that improves disentanglement of semantic and language-specific information.
(3) We show that ORACLE leads to robust mining in code-switched scenarios.

2 Related work

2.1 Cross-lingual Sentence Embeddings

Earlier works primarily centered on learning sentence-level representations for mining pseudo-parallel pairs. Initial methods utilized neural machine translation (NMT) systems with a shared encoder (Schwenk and Douze, 2017; Schwenk, 2018). This approach inspired supervised approaches which train neural networks with large parallel datasets. For instance, Lee and Chen (2017) introduced the multilingual Universal Sentence Encoder (mUSE), a dual-encoder model pre-trained on parallel corpora in 16 languages. Similarly, LASER (Artetxe and Schwenk, 2019b) is an encoder-decoder model based on recurrent neural network. More recently, there has been a shift towards using multilingual sentence encoders such as mBERT (Devlin et al., 2019), XLM-R (Conneau et al., 2020), and CRISS (Tran et al., 2020), which are based on single self-attention networks pre-trained on large monolingual datasets. InfoXLM (Chi et al., 2021) extends XLM-R by adding a cross-lingual contrastive pre-training objective to enhance cross-lingual understanding task performance. Subsequently, the Dual Encoder with Anchor Model (DuEAM) (Goswami et al., 2021) incorporates a dual-encoder approach and integrates the word mover's distance to better capture semantic similarity between sentences. LaBSE (Feng et al., 2022) is a state-of-the-art multilingual sentence encoder built upon a dual-encoder framework, pre-trained with both monolingual and bilingual corpora. We leverage several of these multilingual sentence encoders to derive initial cross-lingual sentence embeddings. For our experiments, we specifically focus on three open-source baselines: LASER, InfoXLM, and LaBSE. We investigate the issue of semantic leakage in these encoders and effectively address it by integrating ORACLE.

2.2 Disentangled Representation Learning

A high-quality cross-lingual sentence embedding should effectively align semantically similar sentences from different languages in a shared embedding space (Wang et al., 2022). However, embeddings obtained from multilingual sentence encoders are often highly biased by language-specific information (Tiyajamorn et al., 2021). In this context, previous research has largely focused on learning disentangled representations to separate language-specific elements from semantics (Pires et al., 2019; Libovický et al., 2020; Gong et al., 2021; Zhao et al., 2021). One prevalent method involves training semantic and language networks separately, where the former is responsible for extracting meaning while the latter extracts language-specific information (Tiyajamorn et al., 2021; Kuroda et al., 2022; Wu et al., 2022). Specifically, DREAM (Tiyajamorn et al., 2021) utilize a multi-task training approach with a combination of reconstruction, semantic embedding, and language embedding losses, while MEAT (Kuroda et al., 2022) introduces novel loss combinations for more direct disentanglement. The distinct loss components of both methods are outlined in Table 1.

Decomposer	L_R	L_{CR}	L_S	L_L	L_A
DREAM (Tiyajamorn et al., 2021)	✓		✓	✓	
MEAT (Kuroda et al., 2022)	✓	✓		✓	✓

Table 1: Comparison of loss components in DREAM and MEAT. L_R: Reconstruction loss, L_{CR}: Cross-Reconstruction loss, L_S: Semantic embedding loss, L_L: Language embedding loss, L_A: Adversarial loss.

Although disentangled representation learning has been explored previously, existing methods have primarily focused on aligning semantics. We discover that these approaches suffer from semantic leakage, as evidenced by the high performance of language-specific representations. Our work is the first to address this challenge through ORACLE, which enforces orthogonality between semantic and language representations.

3 Background

3.1 DREAM

DREAM (Tiyajamorn et al., 2021) employs two separate multi-layer perceptron (MLP) networks in an autoencoder setup to learn disentangled semantic and language-specific representations. Given a parallel corpus $C = \{(s^1, t^1), ..., (s^n, t^n)\}$, comprising pairs of sentences from a source and target language, each sentence pair (s^i, t^i) is input into a multilingual pre-trained model (PLM). This generates original embeddings for the source $\mathbf{e}_s^i \in \mathbb{R}^d$ and the target sentences $\mathbf{e}_t^i \in \mathbb{R}^d$, where d represents the dimension of the input sentence embeddings. Semantic and language representations are then extracted from these embeddings using a separate semantic MLP network MLP_m (denoted by m to signify "meaning") and a language MLP

network MLP_l.

$$\hat{\mathbf{s}}^i_m = \mathrm{MLP}_m(\mathbf{e}^i_s) \quad (1)$$

$$\hat{\mathbf{s}}^i_l = \mathrm{MLP}_l(\mathbf{e}^i_s) \quad (2)$$

Here, $\hat{\mathbf{s}}^i_m, \hat{\mathbf{s}}^i_l \in \mathbb{R}^d$ represent the semantic and language representations of the source sentence, respectively, and similarly $\hat{\mathbf{t}}^i_m, \hat{\mathbf{t}}^i_l \in \mathbb{R}^d$ for the target sentence. We repeat this process across the entire parallel corpus C.

For each language, the extracted semantic and language representations are element-wise summed to reconstruct the original sentence embedding as the final output. DREAM trains the two MLPs in a multi-task fashion, integrating three loss functions:

$$\mathcal{L}_{\mathrm{DREAM}} = \mathcal{L}_R + \mathcal{L}_S + \mathcal{L}_L \quad (3)$$

where $\mathcal{L}_R$ is the reconstruction loss for reconstructing the original sentence embedding using semantic and language representations. $\mathcal{L}_S$ and $\mathcal{L}_L$ are responsible for extracting semantic and language information, respectively. Furthermore, $\mathcal{L}_L$ comprises both the language embedding loss ($\mathcal{L}^m_L$) and the language classification loss ($\mathcal{L}^i_L$), where $\mathcal{L}^m_L$ minimizes the distance within language embeddings and $\mathcal{L}^i_L$ computes the multi-class cross-entropy loss for the language classification task.

3.2 MEAT

MEAT (Kuroda et al., 2022) builds upon DREAM but incorporates more direct supervision to better disentangle semantic and language representations. MEAT trains the two MLPs with a new combination of four losses:

$$\mathcal{L}_{\mathrm{MEAT}} = \mathcal{L}_R + \mathcal{L}_{CR} + \mathcal{L}_L + \mathcal{L}_A \quad (4)$$

$\mathcal{L}_{CR}$ is the cross-reconstruction loss for reconstructing the original source embedding using semantic from the target and language embedding from the source, and vice versa. $\mathcal{L}_A$ is the adversarial loss designed to reduce language identifiability in semantic representations.

4 ORACLE

The two key ingredients of ORACLE are intra-class clustering (§4.1) and inter-class separation (§4.2). We reformulate the losses originally derived in DREAM and MEAT and impose additional constraints to ensure orthogonality between semantic and language embeddings. Following the setup introduced in Section 3.1, ORACLE also uses semantic (MLP_m) and language MLP (MLP_l) to extract semantics ($\hat{\mathbf{s}}^i_m$, $\hat{\mathbf{t}}^i_m$) and language-specific information ($\hat{\mathbf{s}}^i_l$, $\hat{\mathbf{t}}^i_l$) for each language.

4.1 Intra-class clustering ($\mathcal{L}_{\mathrm{IC}}$)

$\mathcal{L}_{\mathrm{IC}}$ aims to bring relevant representations closer in the multilingual embedding space. As shown in Figure 1a, we notice that previous methods lack a constraint to enforce language embeddings to be clustered within themselves. This causes the language-specific information to leak into the semantics, making it difficult to capture the underlying semantics of the sentence. We constrain this by imposing pairwise cosine distances of each language embeddings:

$$\mathcal{L}_{\mathrm{IC}} = \frac{1}{N}\sum_{i=1}^{N}\big(2 - \phi(\hat{\mathbf{s}}^i_l, \hat{\mathbf{s}}^j_l) - \phi(\hat{\mathbf{t}}^i_l, \hat{\mathbf{t}}^j_l)\big), \quad (5)$$

where $\phi(\cdot)$ denotes pairwise cosine similarity. $\phi(\hat{\mathbf{s}}^i_l, \hat{\mathbf{s}}^j_l)$ and $\phi(\hat{\mathbf{t}}^i_l, \hat{\mathbf{t}}^j_l)$ $(i \neq j)$ measures the pairwise cosine similarity of language embeddings in source and target language respectively. We subtract from 2 to transition each of the similarity metric into distance metric. By minimizing $\mathcal{L}_{\mathrm{IC}}$, we aim to cluster language-specific representation for each language.

4.2 Inter-class separation ($\mathcal{L}_{\mathrm{IS}}$)

Simultaneously, $\mathcal{L}_{\mathrm{IS}}$ enforces irrelevant representations to be clearly separated:

$$\mathcal{L}_{\mathrm{IS}} = \frac{1}{N}\sum_{i=1}^{N}\max(0, \phi(\hat{\mathbf{s}}^i_m, \hat{\mathbf{s}}^i_l) + \max(0, \phi(\hat{\mathbf{t}}^i_m, \hat{\mathbf{t}}^i_l)) \quad (6)$$

where $\phi(\cdot)$ denotes cosine similarity. We impose a minimum value constraint of 0 to ensure the proper enforcement of orthogonality, indicative of unrelatedness, between the source and target language embeddings. Minimizing $\mathcal{L}_{\mathrm{IS}}$ effectively disentangles semantics from language-specific representations by constraining them to be orthogonal in the embedding space.

Combining with the intra-class clustering objective we get the final loss as:

$$\mathcal{L}_{\mathrm{ORACLE}} = \mathcal{L}_{\mathrm{IC}} + \mathcal{L}_{\mathrm{IS}}. \quad (7)$$

We train both MLP networks, MLP_m and MLP_l, with the combined loss $\mathcal{L}_{\mathrm{ORACLE}}$ in a multi-task

learning approach. We integrate $\mathcal{L}_{\text{ORACLE}}$ with the existing loss functions of DREAM or MEAT. This is based on our experiments in Section 7.3 where integrating ORACLE with DREAM or MEAT yields better performance than using it as a stand-alone objective.

5 Experimental setup

5.1 Data

We compile a dataset comprising 12 language pairs sourced from publicly available bilingual corpora[2]. English is chosen as the source language for all pairs. We randomly sample 0.5M sentences for each language pair, which is later split into 0.45M for training and 0.05M for testing. In total, we utilize 6M parallel sentences. We select the language pairs based on diversity in language families, semantic similarity to English, and resource availability. Additional details for each language pair are provided in Table 2.

Language	Family	ISO Code	Similarity	Resource level
English	Germanic	en	-	high
German	Germanic	de	0.81	high
Portuguese	Romance	pt	0.84	high
Italian	Romance	it	0.85	high
Spanish	Romance	es	0.86	high
French	Romance	fr	0.86	high
Chinese	Sino-Tibetan	zh	0.81	high
Arabic	Semitic	ar	0.91	high
Japanese	Japonic	ja	0.69	high
Dutch	Germanic	nl	0.80	medium
Romanian	Romance	ro	0.88	medium
Guaraní	Tupi-Guaraní	gn	0.25	low
Aymara	Andean	ay	0.18	low

Table 2: Summary of 12 languages used for training. Similarity refers to the cosine similarity between 1,000 sample of English and target language sentences measured using LaBSE embeddings.

5.2 Baselines

Our study encompasses three open-source multilingual sentence encoders to generate initial sentence embeddings:

- **LASER**: Multilingual enc-decoder model trained on 93 languages (Artetxe and Schwenk, 2019b).
- **InfoXLM**: XLM-R (Conneau et al., 2020) trained with masked language modeling (MLM), translation language modeling (TLM), and cross-lingual contrastive learning task with monolingual and parallel corpora (Chi et al., 2021).
- **LaBSE**: A dual-encoder framework trained with MLM and TLM on both monolingual and bilingual corpora (Feng et al., 2022).

Each multilingual sentence encoder is pre-trained with different combinations of languages. Consequently, the list of seen and unseen languages from our training data varies for each encoder, as summarized in Appendix Table 6.

5.3 Implementation Details

We train the two MLP layers—a semantic embedding layer and a language embedding layer—to distill semantic and language-specific features while keeping the backbone sentence encoder frozen. The output embedding of the [CLS] token is used for sentence embedding. Further details on training process is detailed in Appendix A.3.

5.4 Evaluation task

Cross-lingual Sentence Retrieval. We evaluate our model on two distinct cross-lingual sentence retrieval tasks: held-out test set and Tatoeba[3] (Artetxe and Schwenk, 2019b). Given a list of bilingual sentences, the cross-lingual sentence retrieval task aims to accurately pair sentences that are in a parallel relationship across languages. The dataset consists of up to 1,000 sentences per language along with their English translations. We follow the evaluation setup proposed by Wang et al. (2022), evaluating accuracy in both Tatoeba-14 and Tatoeba-36 settings, each covering 14 languages from LASER and 36 languages from the XTREME benchmark (Hu et al., 2020). We measure retrieval accuracy using both semantic and language-specific representations. Lower language embedding retrieval results suggest reduced semantic leakage in these representations, while higher semantic retrieval accuracy indicates improved semantic alignment in bilingual sentence pairs.

Semantic Textual Similarity. We also report performance on the SemEval-2017 Semantic Textual Similarity (STS) task (Cer et al., 2017). This task involves 7 cross-lingual and 3 monolingual sentence pairs. We aim to achieve high Spearman's rank correlation coefficients (ρ) with semantic representations, indicating better semantic alignment, while expecting lower coefficients with language representations, indicating effective separation.

[2]Our training corpus is obtained from OPUS (https://opus.nlpl.eu/). Details regarding the training corpus for each language pair are outlined in Appendix A.1.

[3]https://tatoeba.org/

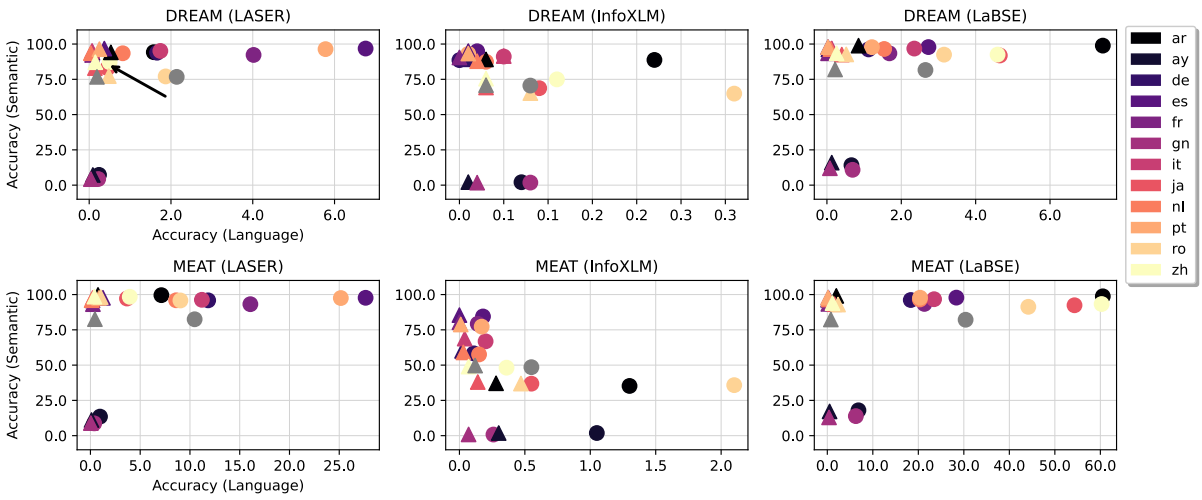


Figure 3: Cross-lingual sentence retrieval performance using our test set, consisting of 0.5M pairs for each language. The optimal representations exhibit high semantic retrieval accuracy and low language embedding retrieval accuracy, aiming for the upper left corner of each plot (indicated by the arrow). ●: vanilla DREAM or MEAT; ▲: with ORACLE objective. Grey: Average accuracy across 12 language pairs. *Top row*: DREAM with each multilingual encoder baselines; *Bottom row*: MEAT with multilingual encoders. Numerical results are in Appendix B.1.

6 Results

6.1 Cross-lingual Sentence Retrieval

Held-out Test Set. Figure 3 illustrates the performance of cross-lingual sentence retrieval on our held-out test set, consisting of 0.5M parallel sentences per language pair. We assess retrieval accuracy using semantic and language-specific representations of these parallel sentences. The optimal representation entails high semantic accuracy and low language embedding accuracy. Notably, applying ORACLE shifts performance towards the upper left quadrant, indicative of higher semantic accuracy and reduced language embedding accuracy across all encoder baselines. We report detailed numerical results in Appendix Table 8.

Tateoba. We draw similar conclusions from another cross-lingual retrieval task, Tatoeba, as shown in Table 3. Utilizing disentangled representations with ORACLE generally yields superior performance compared to representations learned by existing methods such as DREAM and MEAT. One exception is DREAM with LaBSE sentence embeddings, for which the accuracy drops by 0.06 points after integrating ORACLE.

Furthermore, we observe that models exhibit stronger performance from English (EN-XX) than into English (XX-EN) directions. Specifically, for Tatoeba-14, the semantic accuracy difference between the two settings of the vanilla model is smallest for LaBSE at 0.14 points, 0.69 points for LASER, and 15.6 points for InfoXLM on average. We notice a similar trend with the application of ORACLE, with the smallest difference for LaBSE at 0.08 points, 0.22 points for LASER, and 15.78 points for InfoXLM on average. We attribute this to EN-XX setting being similar to our training corpus. We present comprehensive results on Tatoeba in Appendix B.2.

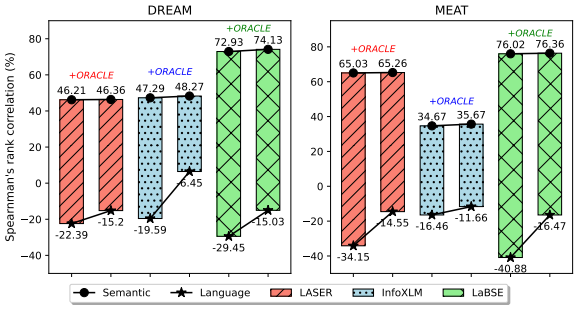


Figure 4: Spearman's rank correlation (%) from the STS task for each multilingual encoder baseline. Length of the bars reflects the performance gap between semantic (●) and language-specific (★) representations. Each set of three bars displays results for LASER, InfoXLM, and LaBSE. Within each color set, the first bar represents the vanilla approach, and the second bar denotes the integration of ORACLE objective.

Seen vs. Unseen. Each multilingual encoder unsurprisingly show lower performance for their unseen target languages, as indicated in Table 6. One exception is the performance of LASER embeddings on Aymara (ay), which shows low per-

Encoder	Objective	Tatoeba-14		Tatoeba-36	
		(EN-XX)	(XX-EN)	(EN-XX)	(XX-EN)
Semantic Embedding (↑)					
LASER	DREAM	68.68	69.53	59.94	62.01
	+ORACLE	**68.82**	**69.66**	**60.14**	**62.11**
	MEAT	88.48	89.00	80.56	79.26
	+ORACLE	88.30	87.70	**81.06**	**79.27**
InfoXLM	DREAM	42.20	51.40	39.51	47.10
	+ORACLE	**42.35**	**51.87**	**39.73**	**47.71**
	MEAT	31.50	53.49	28.21	44.53
	+ORACLE	**32.79**	**54.83**	**29.53**	**45.63**
LaBSE	DREAM	95.57	95.76	95.27	95.09
	+ORACLE	**95.69**	95.75	95.26	95.03
	MEAT	95.67	95.76	95.33	95.06
	+ORACLE	**96.06**	**96.16**	**95.58**	**95.48**
Language Embedding (↓)					
LASER	DREAM	1.58	1.35	1.44	1.21
	+ORACLE	**0.17**	**0.27**	**0.20**	**0.26**
	MEAT	12.52	10.93	10.12	7.86
	+ORACLE	**0.34**	**0.36**	**0.37**	**0.41**
InfoXLM	DREAM	0.31	0.27	0.35	0.37
	+ORACLE	**0.12**	**0.12**	**0.14**	**0.14**
	MEAT	0.33	1.92	0.36	2.32
	+ORACLE	**0.14**	**0.18**	**0.17**	**0.20**
LaBSE	DREAM	18.39	18.09	19.33	19.58
	+ORACLE	**1.26**	**1.36**	**1.50**	**1.70**
	MEAT	87.35	36.66	86.51	40.61
	+ORACLE	**8.48**	**7.00**	**9.92**	**8.41**

Table 3: Cross-lingual retrieval accuracy with Tatoeba dataset. We report the accuracy in both directions (from English and into English). **Bold** denotes better performance than the vanilla approach. All improvements are statistically significant with p-value ≤ 0.001.

formance despite being a seen language. Additionally, we note that ORACLE has a greater impact on the semantic embedding accuracy of unseen languages compared to seen languages. When training with ORACLE, the average semantic accuracy of the seen languages increases from 83.32→83.33 for LASER, 84.29→84.63 for InfoXLM, and 95.43→95.61 for LaBSE. The gap is more significant for unseen languages, increasing from 8.73→8.91 for LASER, 1.93→2.43 for InfoXLM, and 12.51→13.96 for LaBSE. This trend suggests that ORACLE helps bridge the performance gap between seen and unseen languages.

6.2 Semantic Textual Similarity

In Figure 4, we present the average Spearman's rank correlation coefficient across 10 STS tasks. The lengths of the bars indicate the performance gap between semantic and language-specific representations. With ORACLE, we observe a stronger positive correlation with STS scores for semantics and a stronger negative correlation for language representations. The extent of improvement in semantic results differs depending on both the encoder and the objective loss function. For DREAM, the highest gain is observed for LaBSE as +1.2 and the lowest for LASER as +0.15. Conversely, for MEAT, the highest gain is observed for InfoXLM as +1.0 and the lowest for LASER as +0.23.

Monolingual vs Cross-lingual. We categorize the STS results into two groups of language pairs: monolingual and cross-lingual. For both DREAM and MEAT, regardless of integrating ORACLE, the semantic embedding performance of monolingual language pairs is superior to that of cross-lingual language pairs. However, while the performance gap between monolingual and cross-lingual language pairs is larger for vanilla DREAM or MEAT, ORACLE can mitigate this gap. When applying ORACLE, the performance gap decreases by approximately 0.73 points for LASER, 1.47 points for InfoXLM, and 0.50 points for LaBSE. We report detailed results for each monolingual and cross-lingual language pairs in Appendix B.3.

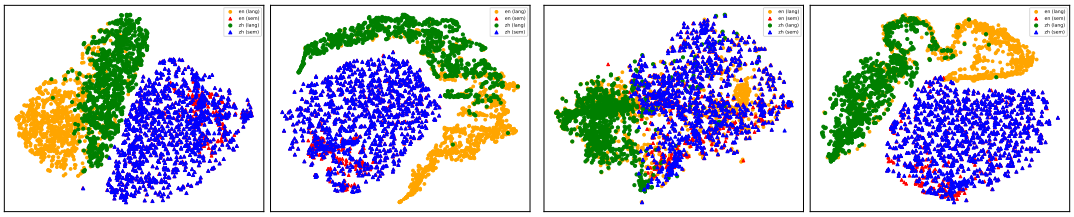
7 Detailed Analysis

7.1 Code-switching

We manually create a code-switched dataset using bilingual dictionaries from MUSE (Conneau et al., 2018). For each language pair, we randomly replace words in the source sentence with corresponding translations in the target language. Further implementation details are provided in Appendix 7.1. As illustrated in Appendix Table 11, our results confirm that integrating ORACLE enhances both semantic and language embedding accuracy, even in practical and challenging scenarios likely encountered during parallel mining, such as code-switching.

7.2 Visualization

In Figure 5, we visualize the LaBSE sentence embedding space using 1,000 English-Chinese sentence pairs from our held-out test set. While previous methods ((a) and (c)) effectively align semantic representations, there is still substantial overlap in the language-specific representations. By applying ORACLE ((b) and (d)), we aim to mitigate the semantic leakage issue, distancing the language representations in parallel sentences while maintaining semantic alignment. We show that this trend is consistent across all language pairs through the visualizations in Appendix D.



(a) DREAM (b) DREAM + ORACLE (c) MEAT (d) MEAT + ORACLE

Figure 5: Visualization of English-Chinese sentence embeddings from our held-out test set. Orange and green denote language embeddings of English and Chinese respectively. Red and blue represent their semantic counterparts. With ORACLE, we can preserve the semantic alignment and clearly divide the language-specific representations.

7.3 ORACLE Components

ORACLE is a multi-task learning objective consisting of two components: intra-class clustering and inter-class separation. Our analysis in Table 4 reveals the distinct impact of each component. Interestingly, using only the inter-class clustering loss demonstrates competitive performance, highlighting its critical role in the effectiveness of ORACLE. However, employing either intra-class clustering or inter-class separation alone presents trade-offs. Combining both components yields the most balanced performance, with highest semantic and lowest language embedding retrieval accuracy.

Furthermore, we discuss the potential of ORACLE as a stand-alone objective. In Figure 6, we illustrate the performance gap when ORACLE is used alone versus alongside DREAM or MEAT losses. We observe that ORACLE alone effectively mitigates semantic leakage with low language retrieval accuracy. However, this is offset by a decrease in semantic alignment compared to its use with DREAM. Therefore, we opt to integrate ORACLE with previous approaches, making it easily adaptable to various frameworks.

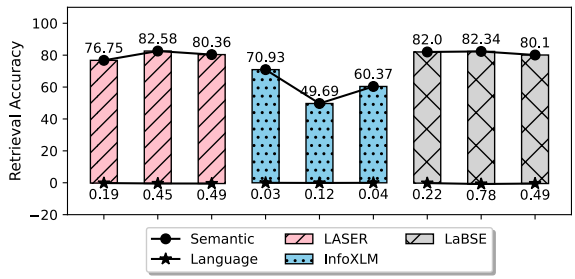


Figure 6: Performance gap between using ORACLE with DREAM (*left*), MEAT (*middle*) or as a stand-alone objective (*right*).

Objective	Tatoeba-14	Tatoeba-36	STS
Semantic Embedding (↑)			
ORACLE	**96.11**	95.53	**74.21**
- $\mathcal{L}_{\text{IC}}$	95.89	95.38	74.13
- $\mathcal{L}_{\text{IS}}$	96.11	**95.54**	72.81
Language Embedding (↓)			
ORACLE	**7.74**	**9.17**	**16.47**
- $\mathcal{L}_{\text{IC}}$	37.78	39.15	30.14
- $\mathcal{L}_{\text{IS}}$	8.07	9.59	18.20

Table 4: Performance change when removing each component of ORACLE from LaBSE sentence embeddings. $\mathcal{L}_{\text{IC}}$: Intra-class clustering; $\mathcal{L}_{\text{IS}}$: Inter-class separation. **Bold** denotes best results for each semantic and language embedding.

8 Conclusion

We explore the issue of semantic leakage, which we define as when language-specific information is leaked into the semantic representations, across various multilingual encoders and objective functions. Addressing this issue is crucial for achieving disentangled semantic and language representations, which is a cornerstone for effective parallel mining. We introduce ORACLE, a simple and effective training objective designed to enforce orthogonality between semantic and language embeddings. Through comprehensive evaluations, we demonstrate that integrating ORACLE not only improves semantic alignment but also ensures clear separation of language representations, as evidenced by embedding space visualization. Further, we conduct detailed analysis to understand the roles of the two key components of ORACLE: intra-class clustering and inter-class separation. While our study primarily focuses on integrating ORACLE with DREAM and MEAT, our method is easily adaptable to various frameworks, offering promising avenues for future work.

9 Limitations

Our work highlights the effectiveness of ORACLE in addressing semantic leakage and improving semantic alignment. While ORACLE demonstrates competitive performance as a stand-alone objective, its integration with DREAM or MEAT losses yields even better results. This limits the usage of ORACLE to be used alongside other methods. This opens many questions for future work to further explore the optimal combination of existing approaches and ORACLE.

Moreover, our study assesses the disentanglement of semantic and language representations in embeddings, focusing on two key aspects: the alignment of semantics in bilingual sentence pairs and the separation of language-specific information. While ORACLE effectively addresses the separation of language-specific information, we notice a trade-off in semantic alignment for certain language pairs. Future works can delve into methods that more efficiently mitigate semantic leakage without compromising semantic representation quality.

Lastly, our experiments are limited to 12 selected language pairs for training. To expand the scope of our study, future work could involve a wider array of language pairs and a broader range of multilingual encoder baselines.

References

Mikel Artetxe and Holger Schwenk. 2019a. Margin-based parallel corpus mining with multilingual sentence embeddings. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics.

Mikel Artetxe and Holger Schwenk. 2019b. Massively multilingual sentence embeddings for zero-shot cross-lingual transfer and beyond. *Transactions of the Association for Computational Linguistics*, 7:597–610.

Daniel Cer, Mona Diab, Eneko Agirre, Iñigo Lopez-Gazpio, and Lucia Specia. 2017. SemEval-2017 task 1: Semantic textual similarity multilingual and crosslingual focused evaluation. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pages 1–14, Vancouver, Canada. Association for Computational Linguistics.

Zewen Chi, Li Dong, Furu Wei, Nan Yang, Saksham Singhal, Wenhui Wang, Xia Song, Xian-Ling Mao, Heyan Huang, and Ming Zhou. 2021. InfoXLM: An information-theoretic framework for cross-lingual language model pre-training. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3576–3588, Online. Association for Computational Linguistics.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2020. Unsupervised cross-lingual representation learning at scale. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8440–8451, Online. Association for Computational Linguistics.

Alexis Conneau, Guillaume Lample, Marc'Aurelio Ranzato, Ludovic Denoyer, and Hervé Jégou. 2018. Word translation without parallel data.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Fangxiaoyu Feng, Yinfei Yang, Daniel Cer, Naveen Arivazhagan, and Wei Wang. 2022. Language-agnostic BERT sentence embedding. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 878–891, Dublin, Ireland. Association for Computational Linguistics.

Hongyu Gong, Vishrav Chaudhary, Yuqing Tang, and Francisco Guzmán. 2021. Lawdr: Language-agnostic weighted document representations from pre-trained models.

Koustava Goswami, Sourav Dutta, Haytham Assem, Theodorus Fransen, and John P. McCrae. 2021. Cross-lingual sentence embedding using multi-task learning. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 9099–9113, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Junjie Hu, Sebastian Ruder, Aditya Siddhant, Graham Neubig, Orhan Firat, and Melvin Johnson. 2020. Xtreme: A massively multilingual multi-task benchmark for evaluating cross-lingual generalization. *CoRR*, abs/2003.11080.

Yuto Kuroda, Tomoyuki Kajiwara, Yuki Arase, and Takashi Ninomiya. 2022. Adversarial training on disentangling meaning and language representations for unsupervised quality estimation. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 5240–5245, Gyeongju, Republic of Korea. International Committee on Computational Linguistics.

Guang-He Lee and Yun-Nung Chen. 2017. MUSE: Modularizing unsupervised sense embeddings. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 327–337, Copenhagen, Denmark. Association for Computational Linguistics.

Jindřich Libovický, Rudolf Rosa, and Alexander Fraser. 2020. On the language neutrality of pre-trained multilingual representations. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 1663–1674, Online. Association for Computational Linguistics.

Paul Michel and Graham Neubig. 2018. MTNT: A testbed for machine translation of noisy text. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 543–553, Brussels, Belgium. Association for Computational Linguistics.

Jeff Mitchell and Mark Steedman. 2015. Orthogonality of syntax and semantics within distributional spaces. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1301–1310, Beijing, China. Association for Computational Linguistics.

Xing Niu, Michael Denkowski, and Marine Carpuat. 2018. Bi-directional neural machine translation with synthetic parallel data. In *Proceedings of the 2nd Workshop on Neural Machine Translation and Generation*, pages 84–91, Melbourne, Australia. Association for Computational Linguistics.

Telmo Pires, Eva Schlinger, and Dan Garrette. 2019. How multilingual is multilingual BERT? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4996–5001, Florence, Italy. Association for Computational Linguistics.

Holger Schwenk. 2018. Filtering and mining parallel data in a joint multilingual space. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 228–234, Melbourne, Australia. Association for Computational Linguistics.

Holger Schwenk and Matthijs Douze. 2017. Learning joint multilingual sentence representations with neural machine translation. In *Proceedings of the 2nd Workshop on Representation Learning for NLP*, pages 157–167, Vancouver, Canada. Association for Computational Linguistics.

Nattapong Tiyajamorn, Tomoyuki Kajiwara, Yuki Arase, and Makoto Onizuka. 2021. Language-agnostic representation from multilingual sentence encoders for cross-lingual similarity estimation. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 7764–7774, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Chau Tran, Yuqing Tang, Xian Li, and Jiatao Gu. 2020. Cross-lingual retrieval for iterative self-supervised training.

Yaushian Wang, Ashley Wu, and Graham Neubig. 2022. English contrastive learning can learn universal cross-lingual sentence embeddings. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 9122–9133, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Linjuan Wu, Shaojuan Wu, Xiaowang Zhang, Deyi Xiong, Shizhan Chen, Zhiqiang Zhuang, and Zhiyong Feng. 2022. Learning disentangled semantic representations for zero-shot cross-lingual transfer in multilingual machine reading comprehension. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 991–1000, Dublin, Ireland. Association for Computational Linguistics.

Wei Zhao, Steffen Eger, Johannes Bjerva, and Isabelle Augenstein. 2021. Inducing language-agnostic multilingual representations. In *Proceedings of *SEM 2021: The Tenth Joint Conference on Lexical and Computational Semantics*, pages 229–240, Online. Association for Computational Linguistics.

Pierre Zweigenbaum, Serge Sharoff, and Reinhard Rapp. 2017. Overview of the second BUCC shared task: Spotting parallel sentences in comparable corpora. In *Proceedings of the 10th Workshop on Building and Using Comparable Corpora*, pages 60–67, Vancouver, Canada. Association for Computational Linguistics.

A Implementation Details

A.1 Training Corpus

In this section, we discuss the implementation details of our ORACLE objective. We describe the specific training corpus utilized for each language pair in Table 5.

A.2 Seen vs. Unseen Languages

In Table 6, we present the list of seen and unseen languages for each multilingual sentence encoder baseline, listed in alphabetical order. Across all encoders, Guaraní (gn) is categorized as an unseen language, while Aymara (ay) is classified as an unseen language for InfoXLM and LaBSE.

A.3 Training Details

Size of each MLP layer is embedding size of the encoder (1024 for LASER and 768 for XLM-R and LaBSE) by the number of language pairs (12). For training, we use Adam optimizer with an initial learning rate as 1e-5 and a batch size of 512.

We train the model for 10,000 iterations, evaluating the model's performance on the validation set at the end of each iteration. We implement early stopping to halt training when there is no improvement over 10 consecutive iterations. We find that DREAM converges in approximately 250 iterations and MEAT in 20 iterations.

B Detailed Results

B.1 Held-out Test set

In Table 8, we present detailed results for the cross-lingual sentence retrieval task using our held-out test set. The top section shows the performance of the initial sentence embeddings from LASER, InfoXLM, and LaBSE. In the middle section, we detail the accuracy of extracted semantic embeddings, while the bottom rows represent the language embedding accuracy. ORACLE, particularly for LaBSE, notably reduces language embedding accuracy, indicating a mitigation of semantic leakage compared to the vanilla DREAM or MEAT frameworks. Additionally, we observe an improvement in the semantic retrieval accuracy across all encoder baselines on average.

B.2 Tatoeba

In our analysis of the Tatoeba dataset detailed in Table 9, we exclude two language pairs (en-ay and en-gn) as Tatoeba does not support them. We show that a similar trend is observed: training MLP networks with ORACLE not only improves semantic alignment but also effectively addresses the semantic leakage issue in the vanilla DREAM or MEAT. Also, we observe that training LaBSE sentence embeddings with ORACLE yields state-of-the-art semantic retrieval accuracy compared to previous methods.

B.3 Semantic Textual Similarity

We present detailed numerical results for the monolingual and cross-lingual STS benchmark in Table 10. The results support our observation from the cross-lingual retrieval tasks that ORACLE helps address both semantic alignment and the semantic leakage issue.

C Code-switching

C.1 Dataset Construction

For our code-switching evaluation, we utilize bilingual dictionaries sourced from MUSE (Conneau et al., 2018). MUSE provides dictionaries in both the to English (XX-EN) and from English (EN-XX) directions. Specifically, we focus on dictionaries with the XX-EN direction. These dictionaries comprise root words in the source language paired with their corresponding translations in the target language. As noted by Conneau et al. (2018), the translations are generated using an internal translation tool, which accounts for word polysemy, resulting in some root words having multiple translations.

For each language pair listed in Table 3, we randomly substitute words in the source sentences with their corresponding translations in the target language, utilizing the dictionaries from MUSE. We ensure that the selected sentences of our code-switching evaluation contain at least one code-switched word. The resulting dataset comprises 1,000 sentences per language pair. We show examples of the manually created code-switched dataset in Table 7.

Training corpus	Language pair
Europarl	en-de, en-es, en-fr, en-it, en-nl, en-pt
Wikimatrix	en-ar, en-ja, en-ro, en-zh
Tatoeba	en-gn
NLLB	en-ay

Table 5: Summary of training corpus for each language pair.

Encoder	Seen	Unseen
LASER	ar, ay, de, en, es, fr, it, ja, nl, pt, ro, zh	gn
InfoXLM	ar, de, en, es, fr, it, ja, nl, pt, ro, zh	ay, gn
LaBSE	ar, de, en, es, fr, it, ja, nl, pt, ro, zh	ay, gn

Table 6: Seen and unseen languages for each pre-trained multilingual encoder. Note that seen refers to languages used during pre-training.

C.2 Results

In Table 11, we present the retrieval accuracy achieved on our code-switched dataset. Similar to the trends observed in other tasks, integrating ORACLE consistently improves both semantic and language embedding accuracy across all multilingual encoder baselines.

D Visualizations

From Figures 7 to 16, we provide visualizations of semantic and language embeddings for each language pair, complementing the discussion in Section 7.2. We use LaBSE to generate the initial sen-

Language pair	Code-switched Example
De-En	Source: Wie *long should* Tom *and I* hierbleiben? Target: How long are Tom and I supposed to stay here?
Fr-En	Source: Je *am here* jusqu'à *three* heures. Target: I will stay here till three o'clock.
It-En	Source: Fadil sparò al *dog* di Dania. Target: Fadil shot Dania's dog.
Ro-En	Source: E *traditional* să gates *black* la înmormântare. Target: It is traditional to wear black to a funeral.

Table 7: Examples of code-switched dataset manually created using bilingual dictionaries from MUSE (Conneau et al., 2018). *Italic* represent words that are code-switched in the source sentence.

tence embeddings, with 1,000 parallel sentences sampled from our held-out test set for each language pair. When solely using DREAM or MEAT (depicted in (a) and (c) for each visualization), we observe a notable amount of overlap in language embeddings between the source and target language, indicating semantic leakage. However, the integration of ORACLE effectively mitigates this issue, resulting in clearer separation and reduced overlap in language embeddings (depicted in (b) and (d)). This improvement is consistent across all language pairs.

Encoder	Objective	en-ar	en-ay	en-de	en-es	en-fr	en-gn	en-it	en-ja	en-nl	en-pt	en-ro	en-zh	Avg.
						Original Embedding								
LASER	-	99.87	11.36	96.13	97.88	93.38	5.12	96.78	98.86	96.48	97.83	99.34	99.39	82.70
InfoXLM*	-	21.24	1.40	25.53	29.31	28.41	0.69	29.07	10.48	14.72	23.80	15.59	13.06	17.78
LaBSE	-	99.00	16.33	96.16	97.82	93.37	12.82	96.77	92.30	96.21	97.66	87.60	94.05	81.67
						Semantic Embedding (↑)								
LASER	DREAM	94.22	7.27	93.85	96.81	92.28	4.33	95.13	82.94	93.46	96.39	77.05	87.10	76.74
	+ORACLE	94.10	7.23	**93.87**	96.79	92.28	4.26	95.13	**82.95**	**93.54**	96.39	**77.27**	**87.13**	**76.75**
	MEAT	99.58	13.50	95.91	97.71	93.18	8.73	96.29	97.20	95.95	97.54	95.82	98.42	82.49
	+ORACLE	**99.78**	11.11	**95.92**	97.71	**93.21**	**8.91**	**96.33**	**97.57**	**96.02**	**97.67**	**98.16**	**98.54**	**82.58**
InfoXLM	DREAM	88.74	2.06	88.61	94.89	90.47	1.80	91.14	68.71	87.44	93.06	64.87	74.96	70.56
	+ORACLE	**89.11**	**2.89**	**89.20**	**95.00**	**90.50**	**1.96**	**91.49**	**69.21**	**87.85**	**93.27**	**65.23**	**75.46**	**70.93**
	MEAT	35.24	1.87	58.32	84.47	79.28	0.79	66.85	36.85	57.57	77.28	35.81	48.18	48.54
	+ORACLE	**37.13**	1.87	**60.09**	**85.46**	**80.35**	**0.87**	**68.79**	**38.02**	**59.00**	**78.80**	**36.92**	**48.96**	**49.69**
LaBSE	DREAM	98.88	14.14	96.20	97.85	93.42	10.87	96.79	91.90	96.50	97.79	92.49	92.52	81.61
	+ORACLE	98.87	**15.94**	**96.22**	97.84	**93.44**	**11.97**	**96.86**	**92.52**	96.46	**97.80**	**92.66**	**93.45**	**82.00**
	MEAT	98.75	17.97	96.14	97.83	93.38	13.85	96.71	92.42	96.53	97.77	91.32	93.30	82.16
	+ORACLE	**99.08**	17.18	**96.29**	**97.87**	**93.41**	12.96	**96.86**	**93.17**	**96.54**	**97.79**	**92.95**	**93.99**	**82.34**
						Language Embedding (↓)								
LASER	DREAM	1.58	0.24	1.66	6.76	4.02	0.22	1.74	0.45	0.82	5.78	1.87	0.50	2.14
	+ORACLE	**0.53**	**0.09**	**0.08**	**0.37**	**0.07**	**0.04**	**0.07**	**0.14**	**0.04**	**0.25**	**0.48**	**0.15**	**0.19**
	MEAT	7.13	0.96	11.83	27.65	16.07	0.38	11.20	3.67	8.61	25.16	9.04	3.95	10.47
	+ORACLE	**0.76**	**0.11**	**0.21**	**1.26**	**0.25**	**0.04**	**0.18**	**0.22**	**0.12**	**0.86**	**1.04**	**0.40**	**0.45**
InfoXLM	DREAM	0.22	0.07	0.00	0.02	0.01	0.08	0.05	0.09	0.03	0.01	0.31	0.11	0.08
	+ORACLE	**0.03**	**0.01**	0.01	**0.01**	**0.00**	**0.02**	0.05	**0.03**	**0.02**	0.01	**0.08**	**0.03**	**0.03**
	MEAT	1.30	1.05	0.11	0.18	0.14	0.26	0.20	0.55	0.15	0.17	2.10	0.36	0.55
	+ORACLE	**0.28**	**0.30**	**0.02**	**0.00**	**0.00**	**0.07**	**0.04**	**0.14**	**0.03**	**0.01**	**0.47**	**0.07**	**0.12**
LaBSE	DREAM	7.42	0.66	1.12	2.73	1.68	0.69	2.35	4.65	1.54	1.21	3.15	4.59	2.65
	+ORACLE	**0.85**	**0.13**	**0.04**	**0.03**	**0.03**	**0.08**	**0.13**	**0.40**	**0.10**	**0.03**	**0.52**	**0.26**	**0.22**
	MEAT	60.54	6.84	18.34	28.36	21.33	6.27	23.46	54.32	20.33	20.38	44.17	60.28	30.39
	+ORACLE	**1.98**	**0.50**	**0.18**	**0.22**	**0.22**	**0.38**	**0.31**	**1.59**	**0.35**	**0.16**	**2.36**	**1.14**	**0.78**

Table 8: Cross-lingual sentence retrieval accuracy with our test set, comprising 0.5M pairs for each language. We expect the semantic retrieval accuracy to be higher and lower with language embedding. **Bold** represents when our method surpass the vanilla approach and **highlight** denotes when the average value is higher. vanilla: original DREAM or MEAT approach; ORACLE: incorporation of our objective. *: We use mean pooling to compute sentence embedding. All average improvements are statistically significant with p-value ≤ 0.001.

Encoder	Objective	en-ar	en-de	en-es	en-fr	en-it	en-ja	en-nl	en-pt	en-ro	en-zh	Avg.
						Original Embedding						
MUSE ❖	-	-	-	95.40	93.50	94.30	93.80	94.00	94.90	30.00	94.30	86.90
CRISS ♥	-	-	-	96.30	92.70	92.50	84.80	93.40	-	-	85.60	90.20
DuEAM ♠	-	-	-	93.00	91.50	85.70	84.20	-	91.20	88.50	90.20	87.90
LASER	-	91.95	99.05	98.00	95.65	95.30	95.35	96.30	95.15	97.40	95.45	95.96
InfoXLM*	-	20.95	38.50	30.85	32.35	24.85	28.20	19.85	36.90	30.40	34.05	29.69
LaBSE	-	89.75	99.20	98.10	96.05	94.75	96.40	96.90	95.55	97.40	96.20	96.03
						Semantic Embedding (↑)						
LASER	DREAM	60.35	89.85	83.40	76.55	80.95	71.70	80.10	82.15	80.60	74.20	77.99
	+ORACLE	60.30	**90.00**	83.40	**76.65**	**81.00**	**72.05**	**80.35**	**82.30**	80.60	**74.60**	**78.13**
	MEAT	86.95	96.55	96.00	91.35	91.80	90.65	91.75	93.45	94.80	92.95	92.63
	+ORACLE	**87.30**	**98.05**	**96.65**	**92.75**	**92.55**	87.95	**93.70**	**94.25**	**95.40**	**93.80**	**93.24**
InfoXLM	DREAM	44.05	57.65	68.65	62.80	54.80	56.45	62.70	66.50	58.15	67.10	59.89
	+ORACLE	**44.80**	**58.00**	**68.85**	62.65	54.80	**57.05**	62.30	**66.65**	57.95	**67.20**	**60.03**
	MEAT	31.60	67.25	70.75	67.30	63.70	42.05	68.95	74.05	59.05	57.70	60.24
	+ORACLE	**31.80**	**69.00**	**71.60**	**68.35**	**64.15**	**43.10**	**70.85**	**75.25**	**60.45**	**60.30**	**61.49**
LaBSE	DREAM	89.90	99.10	98.50	95.80	95.05	95.75	97.35	95.45	97.50	95.35	95.98
	+ORACLE	89.70	**99.15**	98.50	**95.90**	94.85	**95.90**	97.30	**95.50**	**97.70**	**95.55**	**96.01**
	MEAT	90.30	99.20	98.15	98.90	94.55	96.15	97.30	95.55	97.55	95.50	96.32
	+ORACLE	**90.95**	**99.40**	**98.50**	**96.30**	**95.20**	**96.40**	**97.40**	**95.75**	**97.85**	**95.80**	**96.36**
						Language Embedding (↓)						
LASER	DREAM	1.20	1.50	2.95	1.50	4.45	1.15	2.00	3.70	1.85	1.70	2.20
	+ORACLE	**0.25**	**0.10**	**0.30**	**0.20**	**0.25**	**0.10**	**0.35**	**0.30**	**0.20**	**0.05**	**0.21**
	MEAT	19.40	9.75	13.55	6.70	14.30	5.65	9.90	16.15	16.20	10.85	12.25
	+ORACLE	**0.60**	**0.20**	**0.45**	**0.30**	**0.55**	**0.30**	**0.65**	**0.40**	**0.75**	**0.45**	**0.47**
InfoXLM	DREAM	0.10	0.10	0.25	0.15	0.45	0.20	0.40	0.20	0.15	0.20	0.22
	+ORACLE	0.10	0.10	**0.10**	**0.10**	**0.10**	**0.10**	**0.10**	**0.10**	**0.10**	**0.10**	**0.10**
	MEAT	0.35	0.55	2.00	0.90	2.75	0.30	3.35	0.85	1.15	0.40	1.26
	+ORACLE	**0.10**	**0.15**	**0.25**	**0.15**	**0.10**	**0.10**	**0.20**	**0.20**	**0.15**	**0.15**	**0.16**
LaBSE	DREAM	24.50	11.50	18.95	17.85	24.25	11.20	14.20	9.70	12.35	24.70	16.92
	+ORACLE	**2.15**	**1.20**	**1.00**	**0.30**	**1.45**	**1.15**	**1.30**	**9.30**	**0.70**	**1.00**	**1.96**
	MEAT	64.25	55.30	64.10	64.30	64.50	65.45	60.25	58.15	57.60	60.45	61.44
	+ORACLE	**8.30**	**7.25**	**5.05**	**5.00**	**8.15**	**9.70**	**7.20**	**3.90**	**6.75**	**5.65**	**6.70**

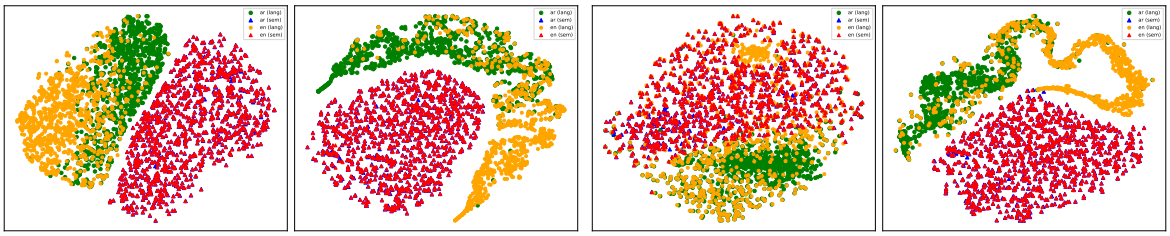
Table 9: Cross-lingual retrieval accuracy with Tatoeba task. For each language pair, we report the average accuracy of both directions (from English and into English). **Bold** represents when our method surpass the vanilla approach and **highlight** denotes when the average value is higher. ❖: results from Lee and Chen (2017) (*supervised*) ; ♥: results from Tran et al. (2020) (*weakly supervised*); ♠: results from Goswami et al. (2021) (*self-supervised*). *: We use mean pooling to compute sentence embedding. All average improvements are statistically significant with p-value ≤ 0.001.

Encoder	Objective	ar-ar	en-en	es-es	en-ar	en-de	en-tr	en-es	en-fr	en-it	en-nl	Avg.
					Original Embedding							
LASER	-	68.85	66.55	57.93	77.62	79.68	64.20	71.98	69.05	70.83	68.68	69.54
InfoXLM*	-	19.11	50.20	36.17	12.89	16.31	24.86	9.10	25.12	28.10	30.55	25.24
mSimCSE	-	69.06	74.50	65.71	79.45	80.83	73.85	72.07	76.98	76.98	75.22	74.47
					Semantic Embedding (↑)							
LASER	DREAM	57.10	53.98	46.36	43.22	41.92	40.60	32.88	48.58	49.94	47.47	46.21
	+ORACLE	**57.16**	**54.14**	**46.64**	**43.55**	**42.11**	**40.67**	32.83	**48.70**	**49.98**	**47.80**	**46.36**
	MEAT	66.87	71.95	79.16	62.41	60.44	65.06	54.63	61.94	66.27	63.90	65.26
	+ORACLE	**67.14**	**72.69**	78.75	**63.59**	60.03	**66.19**	**55.20**	61.38	65.80	63.76	**65.45**
InfoXLM	DREAM	50.28	56.39	56.16	43.35	39.54	42.71	38.42	48.02	47.80	50.18	47.29
	+ORACLE	50.25	56.38	56.16	43.35	**49.55**	42.61	38.40	47.98	**47.82**	**50.19**	**48.27**
	MEAT	35.83	61.23	51.14	11.09	25.58	33.32	20.04	29.38	41.58	37.50	34.67
	+ORACLE	**35.87**	**61.40**	50.73	**11.26**	**28.15**	**34.83**	**21.41**	**31.55**	**42.72**	**38.77**	**35.67**
LaBSE	DREAM	69.84	74.78	79.82	70.97	70.82	71.30	64.22	75.67	76.28	75.56	72.93
	+ORACLE	**70.65**	**76.03**	**81.06**	**72.37**	**72.49**	**73.33**	**66.18**	**76.13**	**76.76**	**76.29**	**74.13**
	MEAT	72.03	80.34	83.66	74.71	75.40	73.59	70.48	77.82	78.18	77.43	76.36
	+ORACLE	**72.05**	**80.41**	**83.86**	**75.09**	**75.67**	**74.56**	**70.98**	77.75	**78.57**	**77.44**	**76.64**
					Language Embedding (↓)							
LASER	DREAM	45.12	34.87	39.95	18.94	11.89	21.50	17.64	8.29	14.91	10.81	22.39
	+ORACLE	**21.43**	**12.77**	**18.40**	20.85	**11.65**	**17.12**	**15.30**	**6.14**	16.87	11.51	**15.20**
	MEAT	52.51	40.35	55.29	35.78	22.93	27.26	27.89	22.73	30.74	25.99	34.15
	+ORACLE	**21.26**	**14.97**	**21.75**	**21.48**	**6.09**	**15.71**	**15.51**	**4.86**	**15.37**	**8.49**	**14.55**
InfoXLM	DREAM	39.62	51.03	49.74	3.87	2.66	4.53	12.95	9.94	9.37	12.21	19.59
	+ORACLE	**24.62**	**31.07**	**36.01**	**-10.84**	**-7.12**	**-9.51**	**2.85**	**-3.18**	**-2.06**	**2.67**	**6.45**
	MEAT	33.00	50.01	51.02	-6.10	8.21	-2.50	-1.97	8.44	13.58	10.87	16.46
	+ORACLE	33.13	**46.96**	51.63	**-11.63**	**0.14**	**-8.14**	**-7.91**	**1.57**	**7.25**	**3.59**	**11.66**
LaBSE	DREAM	44.32	40.35	50.81	24.12	22.56	28.29	18.76	22.86	20.38	22.02	29.45
	+ORACLE	**33.10**	**19.88**	**28.60**	**1.59**	**1.14**	**17.39**	**15.57**	**12.17**	**8.36**	**12.53**	**15.03**
	MEAT	52.11	68.57	68.18	38.57	28.94	35.87	31.27	28.66	29.40	27.21	40.88
	+ORACLE	**37.25**	**27.30**	**33.61**	**0.08**	**0.79**	**16.94**	**16.60**	**10.98**	**8.01**	**13.12**	**16.47**

Table 10: Spearman's rank correlation coefficients (ρ) of monolingual and cross-lingual STS task. **Bold** represents when our method surpass the vanilla approach and **highlight** indicates when the average value is higher. *: We use mean pooling to compute sentence embedding. All average improvements are statistically significant with p-value ≤ 0.001.

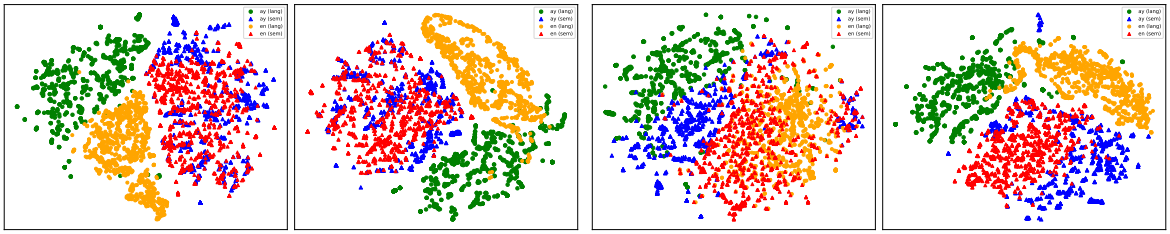
Encoder	Objective	en-ar	en-de	en-es	en-fr	en-it	en-nl	en-pt	en-ro	Avg.
					Original Embedding					
LASER	-	90.82	98.75	98.26	95.17	94.63	95.22	95.91	98.37	95.89
InfoXLM*	-	13.71	39.04	37.05	36.20	29.44	29.54	41.44	31.37	32.22
LaBSE	-	90.06	99.48	98.49	95.60	93.46	97.02	96.56	98.24	96.11
					Semantic Embedding (↑)					
LASER	DREAM	59.75	88.58	84.67	73.90	84.46	80.66	82.02	82.43	79.56
	+ORACLE	59.62	**88.79**	**85.02**	73.90	84.46	80.45	**82.24**	82.18	**79.58**
	MEAT	84.91	97.82	96.28	92.70	93.34	93.84	95.05	96.86	93.85
	+ORACLE	**86.54**	97.09	**97.79**	92.59	93.22	92.67	**95.26**	96.11	**93.91**
InfoXLM	DREAM	28.05	55.14	58.07	59.72	54.09	53.99	54.90	56.46	52.55
	+ORACLE	27.80	**56.39**	**58.19**	**60.15**	**54.67**	**54.84**	**56.62**	56.34	**53.13**
	MEAT	15.35	42.26	60.51	58.22	56.43	56.43	57.70	50.69	49.70
	+ORACLE	**16.35**	42.26	**61.32**	**59.94**	**58.06**	**57.17**	**59.31**	**52.07**	**50.81**
LaBSE	DREAM	87.78	99.27	97.33	95.38	92.87	96.60	96.45	97.99	95.46
	+ORACLE	**88.30**	99.27	**98.14**	95.38	**93.22**	**96.81**	**96.66**	**98.11**	**95.74**
	MEAT	88.43	99.38	98.03	94.95	92.87	96.49	96.12	97.74	95.50
	+ORACLE	**89.56**	**99.69**	**98.37**	**96.60**	**93.34**	**97.13**	**96.34**	**98.24**	**96.16**
					Language Embedding (↓)					
LASER	DREAM	2.01	4.15	7.08	3.01	7.36	6.06	9.36	3.76	5.35
	+ORACLE	**0.25**	**0.83**	**1.39**	**0.43**	**0.93**	**0.85**	**2.48**	**0.75**	**0.99**
	MEAT	35.72	23.88	35.31	17.72	27.57	31.77	49.62	22.84	30.55
	+ORACLE	**1.51**	**1.66**	**3.02**	**1.07**	**2.45**	**2.34**	**5.06**	**1.25**	**2.30**
InfoXLM	DREAM	0.13	0.52	0.93	0.64	0.93	2.34	0.54	0.53	0.82
	+ORACLE	0.13	**0.10**	**0.46**	**0.11**	**0.23**	**0.85**	**0.11**	**0.13**	**0.27**
	MEAT	1.89	10.38	20.44	13.64	20.56	36.03	23.90	15.93	17.85
	+ORACLE	**0.38**	**1.04**	**3.37**	**1.40**	**3.15**	**8.93**	**3.12**	**1.76**	**2.89**
LaBSE	DREAM	11.57	14.54	21.24	23.52	27.69	27.21	19.38	17.44	20.32
	+ORACLE	**1.26**	**1.25**	**1.39**	**2.69**	**1.99**	**2.98**	**0.75**	**0.88**	**1.65**
	MEAT	48.81	37.80	53.31	51.34	56.54	53.35	43.27	34.00	47.30
	+ORACLE	**6.42**	**7.06**	**6.04**	**7.63**	**9.11**	**10.52**	**6.14**	**6.40**	**7.42**

Table 11: Retrieval accuracy with our code-switching dataset. **Bold** represents when our method surpasses the vanilla approach and **highlight** denotes when the average value is higher. vanilla: original DREAM or MEAT approach; ORACLE: incorporation of our objective. *: We use mean pooling to compute sentence embedding. All average improvements are statistically significant with p-value ≤ 0.001.



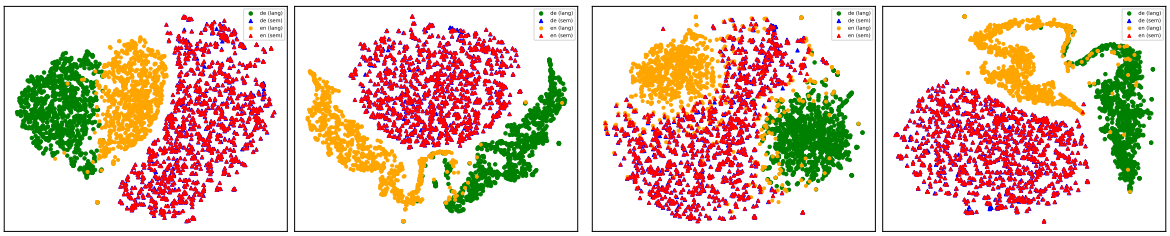
(a) DREAM (b) DREAM + ORACLE (c) MEAT (d) MEAT + ORACLE

Figure 7: LaBSE sentence embeddings for English-Arabic sentence pair.



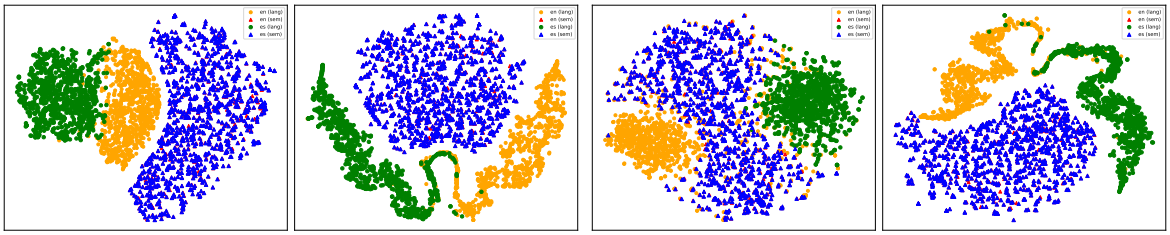
(a) DREAM (b) DREAM + ORACLE (c) MEAT (d) MEAT + ORACLE

Figure 8: LaBSE sentence embeddings for English-Aymara sentence pair.



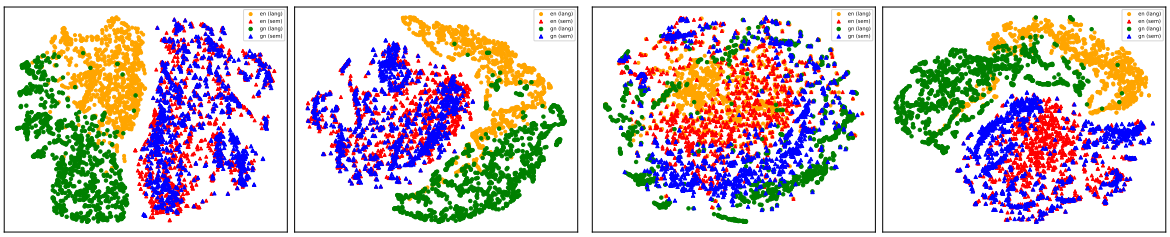
(a) DREAM (b) DREAM + ORACLE (c) MEAT (d) MEAT + ORACLE

Figure 9: LaBSE sentence embeddings for English-German sentence pair.



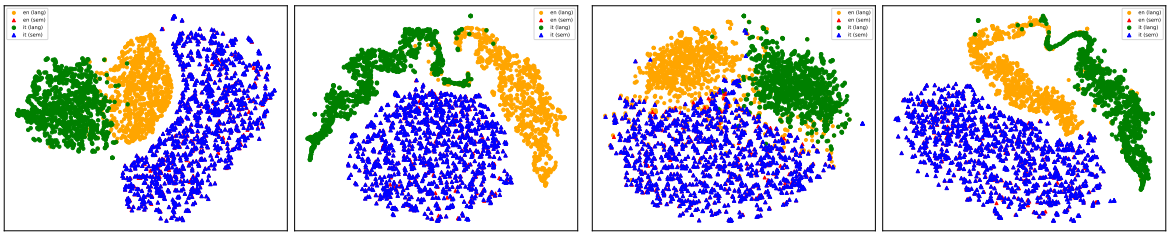
(a) DREAM (b) DREAM + ORACLE (c) MEAT (d) MEAT + ORACLE

Figure 10: LaBSE sentence embeddings for English-Spanish sentence pair.



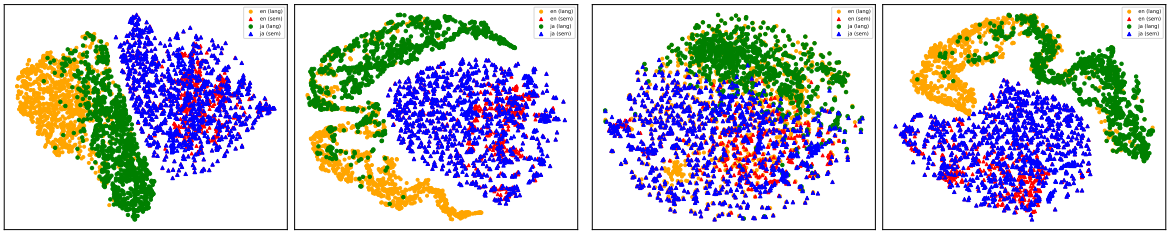
(a) DREAM (b) DREAM + ORACLE (c) MEAT (d) MEAT + ORACLE

Figure 11: LaBSE sentence embeddings for English-Guaraní sentence pair.



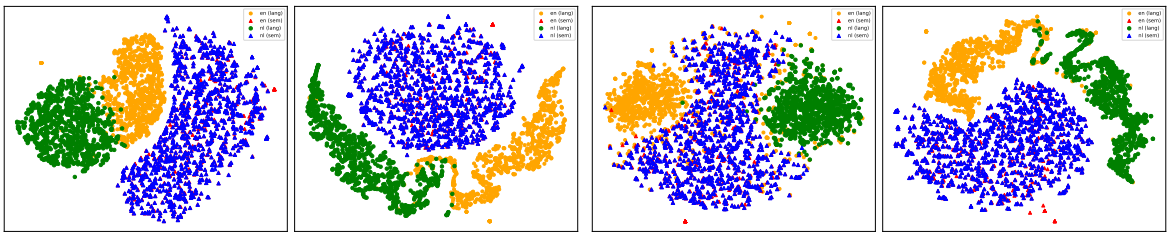
(a) DREAM (b) DREAM + ORACLE (c) MEAT (d) MEAT + ORACLE

Figure 12: LaBSE sentence embeddings for English-Italian sentence pair.



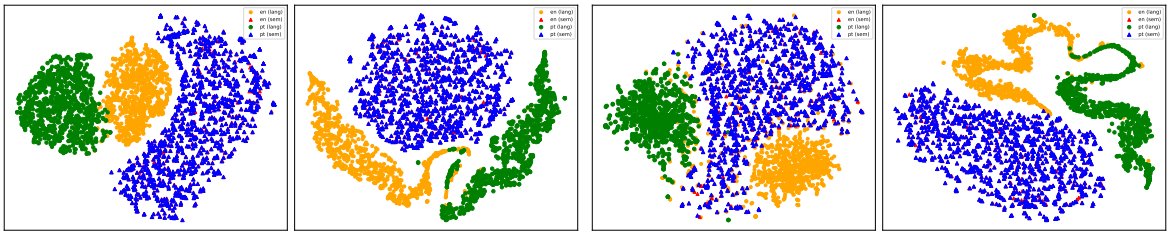
(a) DREAM (b) DREAM + ORACLE (c) MEAT (d) MEAT + ORACLE

Figure 13: LaBSE sentence embeddings for English-Japanese sentence pair.



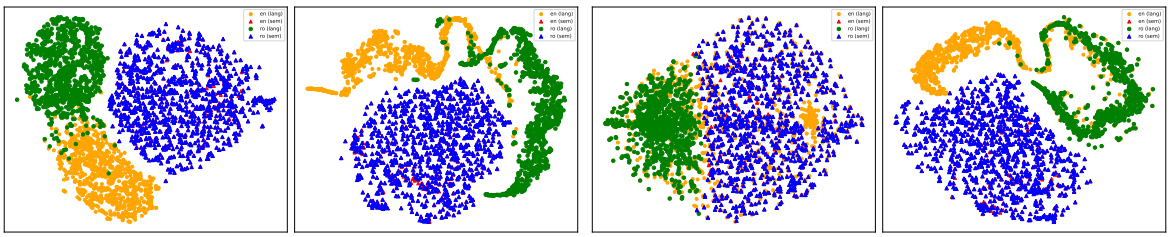
(a) DREAM (b) DREAM + ORACLE (c) MEAT (d) MEAT + ORACLE

Figure 14: LaBSE sentence embeddings for English-Dutch sentence pair.



(a) DREAM (b) DREAM + ORACLE (c) MEAT (d) MEAT + ORACLE

Figure 15: LaBSE sentence embeddings for English-Portuguese sentence pair.



(a) DREAM (b) DREAM + ORACLE (c) MEAT (d) MEAT + ORACLE

Figure 16: LaBSE sentence embeddings for English-Romanian sentence pair.