Revisiting zero-shot cross-lingual topic identification: baselines, languages and evaluation

Anonymous ACL submission

Abstract

In this paper, we revisit cross-lingual topic identification (ID) in zero-shot settings by taking a deeper dive into current datasets, baseline systems and the languages covered. We identify shortcomings in the existing MLDoc evaluation protocol and propose a robust alternative scheme, while also extending the cross-lingual experimental setup to 17 languages. We benchmark several systems that are based on existing multilingual models such as LASER, XLM-R, mUSE, and LaBSE on the new evaluation protocol covering 17 languages. Further, we present a novel Bayesian multilingual document model (MBay) for learning language-independent document embeddings. The model learns to represent the document embeddings in the form of Gaussian distributions, thereby encoding the uncertainty in its covariance. We propagate the learned uncertainties through linear classifiers that benefit in zero-shot cross-lingual topic ID. Our experiments on 17 languages show that the proposed multilingual Bayesian document model performs competitively as compared to other systems based on LASER, XLM-R and mUSE on 8 high resource languages, and outperforms these systems on 9 mid-resource languages. Finally, we consolidate the observations from all our experiments, and discuss points that can potentially benefit the future research works in the area of cross-lingual topic ID.

1 Introduction

The zero-shot cross-lingual topic identification (ID) or document classification aims to classify documents from target languages using a classifier trained on examples from one or more source language(s). This is mainly useful in scenarios where the data from target language(s) have little or no labels to train an in-language classifier. The cross-lingual transfer experiments can also help to analyse and test the capabilities of an underlying multilingual language model.

The common approach is to first train a multilingual language model that aims to capture the semantic relations of words in context, independent of the language (Ammar et al., 2016; Artetxe and Schwenk, 2019; Huang et al., 2019; Conneau et al., 2020; Feng et al., 2020). Such a multilingual model can then later be either (i) fine-tuned for classification (Siddhant et al., 2020) task using labelled examples from source language(s), or (ii) used to extract low-dimensional embeddings (representations) for documents from both source and target languages (Reimers and Gurevych, 2020); the embeddings from source language(s) together with annotated labels are then used for training a light-weight *independent* classifier for cross-lingual topic ID, which is then used to classify embeddings from target languages.

The former approach relying on fine-tuning is not efficient as it would require to keep a copy of the entire multilingual model for every source language, and every down-stream task. The latter approach of extracting language-agnostic document (sentence) embeddings is more practical as it would require only one model, and several light-weight downstream classifiers. This paper entirely focuses on models, experiments and analysis related to the latter scheme relying on language agnostic document embeddings, followed by a light-weight classifier.

1.1 Training multilingual models

Majority, if not all, of the recent works in multilingual representations for cross-lingual transfers have relied on training LSTMs (Schwenk and Douze, 2017; Artetxe and Schwenk, 2019) or transformers (Wu and Dredze, 2019; Conneau et al., 2020) with huge amounts of data (e.g. 227M - 25B sentences) (Wu and Dredze, 2019; Siddhant et al., 2020). The pre-training objectives vary depending on the kind of resources used for training such models. In brief, some require parallel translations of sentences across multiple languages, while others rely on bilingual dictionaries (Ammar et al., 2016) or just monolingual texts covering several languages. Training these large multilingual language models requires enormous computational resources (Strubell et al., 2019), there is a need for alternatives that are computationally efficient. A majority of the large multilingual models share a vocabulary of sub-word units across several (100) *seed* languages. One needs to take care so that all the languages are equally represented in the shared (sub-)word vocabulary to avoid any language bias from the high resource languages. Additionally, such a design choice makes it challenging to extend these models to newer languages having a different orthography. A fair comparison among these language models is nearly impossible as no two models are trained on exactly the same data. The comparisons are only on the downstream tasks while ignoring the affect of the quality and quantity of pre-training data. When training on large amounts of web-data it is possible that some of the down-stream data could have been seen during pre-training. Extensive survey on the aforementioned models/approaches can be found in (Ruder et al., 2019; Doddapaneni et al., 2021). In contrast

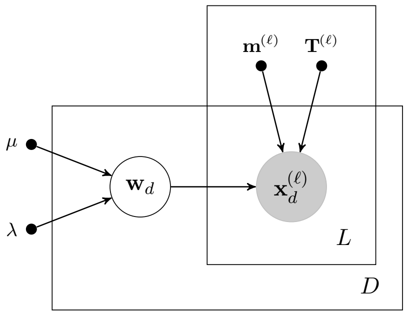

Figure 1: Graphical representation of the proposed multilingual Bayesian model, where L represents number of languages and D denotes number of L-way parallel documents (translations). $\{\mathbf{m}^{(\ell)}, \mathbf{T}^{(\ell)}\} \forall \ell$ are document-independent, language-specific model parameters, whereas $\mathbf{w}_d$ is document-specific but language-independent random variable (embedding), and $\mathbf{x}_d^{(\ell)}$ is the observed vector of word counts representing document d from language ℓ.

to the neural models, there is also work on classical multilingual topic models (Mimno et al., 2009; Yang et al., 2019), which are suitable for topic ID and document clustering. While these models are budget-friendly in terms of computation, the downstream evaluation datasets and tasks (Schwenk and Li, 2018; Kakwani et al., 2020; Hu et al., 2020) do not overlap between neural and classical models, hence it is difficult to ascertain the advantages of the latter over former.

1.2 Contributions of the paper

- We propose a simple, yet efficient multilingual Bayesian (MBay) model for learning language-agnostic document (sentence) embeddings, that enables to train robust downstream linear classifiers for zero-shot cross-lingual topic ID.

- The proposed model can be easily extended to newer languages without requiring to retrain from scratch (continual learning), while constraining only on a subset of existing parameters, thus making it computation-budget-friendly.

- We re-visit the zero-shot cross-lingual document classification task, and make the following contributions: (i) we identify the shortcomings in evaluation, and propose a robust alternative, (ii) we setup and evaluate zero-shot transfer systems on a new set of 9 languages from IndicNLP suite (Kakwani et al., 2020), in addition to the existing 8 from MLDoc (Schwenk and Li, 2018), (iii) we benchmark several pre-trained models, and also the proposed model on the revised datasets covering 17 languages (128 transfer directions[1], and (iv) we provide an in depth analysis of the downstream classification systems, that can best make use of the language-agnostic document (sentence) embeddings from various models.

2 MBay: Multilingual Bayesian Model

Like majority of the probabilistic topic and document models (Blei, 2012; Miao et al., 2016), the presented model also relies on *bag-of-words* representation of documents. Let $V^{(\ell)}$ represent the vocabulary size in language $\ell \in \mathcal{M}$, where $L = |\mathcal{M}|$ denotes the number of languages. Let $\{\mathbf{m}^{(\ell)}, \mathbf{T}^{(\ell)}\} \forall \ell$ represent the language-specific model parameters, where $\mathbf{T}^{(\ell)}$ is a low-rank matrix of size $V^{(\ell)} \times K$ $(K \ll V^{(\ell)})$ that defines the subspace of document specific unigram distributions, and $\mathbf{m}^{(\ell)} \in \mathbb{R}^{V^{(\ell)}}$ represents bias or offset. The multilingual model assumes that the L-way

[1] 9 languages from IndicNLP news articles dataset resulting in $9 \times 8 = 72$, and 8 languages from MLDoc resulting in $8 \times 7 = 56$ transfer directions $(72 + 56 = 128)$.

parallel data (translations of *bag-of-words*) are generated according to the following process:

First, a K-dimensional language-independent, document-specific embedding is sampled from an isotropic Gaussian distribution with precision λ

$$\mathbf{w}_d \sim \mathcal{N}(\mathbf{w} \mid \mathbf{0}, (\lambda \mathbf{I})^{-1}). \tag{1}$$

$\mathbf{w}_d$ can be interpreted as vector representing higher-level semantic concepts (such as topic) of a document, independent of any language. For each language $\ell \in \mathcal{M}$, a vector of word counts $\mathbf{x}_d^{(\ell)}$ is generated by the following two steps: The document-specific unigram distribution $\boldsymbol{\phi}_d^{(\ell)}$ is computed using the language-specific parameters

$$\boldsymbol{\phi}_d^{(\ell)} = \mathrm{softmax}(\mathbf{m}^{(\ell)} + \mathbf{T}^{(\ell)}\,\mathbf{w}_d), \tag{2}$$

and the vector of word counts $\mathbf{x}_d^{(\ell)}$ is sampled $\mathbf{x}_d^{(\ell)} \sim \mathrm{Multinomial}(\boldsymbol{\phi}_d^{(\ell)}, N_d^{(\ell)})$, where $N_d^{(\ell)}$ are the number of word tokens in document d. $\mathbf{x}^{(1)} \ldots \mathbf{x}^{(L)}$ represent L-way parallel *bag-of-words* statistics.

The above steps describe the generative process of the proposed multilingual document model. However, in reality, we do not generate any data, instead we invert the generative process: given the training (observed) data $\mathbf{x}_d^{(\ell)}\,\forall\,\ell \in \mathcal{M}, \forall\, d = 1 \ldots D$, we estimate the language-specific model parameters $\{\mathbf{m}^{(\ell)}, \mathbf{T}^{(\ell)}\}$ and also the posterior distributions of language-independent document embeddings $p(\mathbf{w}_d|\mathbf{x}_d^{(1)} \ldots \mathbf{x}_d^{(L)})\,\forall\, d$. Moreover, given an unseen document $\mathbf{x}_u^{(\ell)}$ from any of the L languages, we infer the corresponding posterior distribution of the document embedding $p(\mathbf{w}_u \mid \mathbf{x}_u^{(\ell)})$. Note that such a posterior distribution also carries the uncertainty about the estimate.

Although we describe the model assuming L-way parallel data, in practice the model can be trained with parallel text (translations) between language pairs (bi-texts) covering all the L languages.

2.1 Variational Bayes training

The proposed model is trained using the variational Bayes framework, i.e., we approximate the intractable true posterior with the variational distribution $q(\mathbf{w}_d) = \mathcal{N}(\mathbf{w}_d \mid \boldsymbol{\nu}_d, \mathrm{diag}(\boldsymbol{\gamma}_d)^{-1})$ and optimize the evidence lower-bound (Bishop, 2006). Further, we use Monte Carlo samples via the re-parametrization trick (Kingma and Welling, 2014; Rezende et al., 2014) to approximate the expectation over log-sum-exp (log normalizer) term which appears in the lower-bound (Kesiraju et al., 2020). The resulting lower-bound for a single set of L-way parallel documents is

$$\begin{aligned}\mathcal{L}(q_d) \approx \sum_{\forall \ell \in \mathcal{M}} \sum_{i=1}^{V^{(\ell)}} x_{di}^{(\ell)} \Big[(m_i^{(\ell)} + \mathbf{t}_i^{(\ell)}\,\boldsymbol{\nu}_d) \\ - \frac{1}{R}\sum_{r=1}^{R} \log \Big(\sum_{j=1}^{V} \exp\{m_j^{(\ell)} + \mathbf{t}_j^{(\ell)}\, g(\boldsymbol{\epsilon}_{dr})\}\Big)\Big] \\ - D_{\mathrm{KL}}(q_d \,||\, p),\end{aligned} \tag{3}$$

where $D_{\mathrm{KL}}(q_d \,||\, p)$ is the Kullback-Leibler divergence from variational distribution $q(\mathbf{w})$ to the prior (1) and, $g(\boldsymbol{\epsilon}_{dr}) = \boldsymbol{\nu} + \boldsymbol{\gamma} \odot \tilde{\boldsymbol{\epsilon}}_{dr}$, with $\tilde{\boldsymbol{\epsilon}}_{dr} \sim \mathcal{N}(\boldsymbol{\epsilon} \mid \mathbf{0}, \mathbf{I})$. R are the number of Monte Carlo samples used for empirically approximating the expectation over log-sum-exp.

The complete lower-bound is just the summation over all the documents. Additionally, we use ℓ_2 regularization term with weight ω for language-specific model parameters $\{\mathbf{T}^{(\ell)}\}\,\forall\,\ell$. Thus, the final objective is

$$\mathcal{L} = \sum_{d=1}^{D} \mathcal{L}(q_d) - \omega \sum_{\forall \ell \in \mathcal{M}} \sum_{i=1}^{V^{(\ell)}} ||\,\mathbf{t}_i^{(\ell)}\,||_2\,. \tag{4}$$

In practice, we follow batch-wise stochastic optimization of (4) using ADAM (Kingma and Ba, 2015). For a batch of documents $d \in \mathcal{B}$ covering a subset of languages $\mathcal{M}_B \subseteq \mathcal{M}$, we update the all model parameters $\{\mathbf{m}^{(\ell)}, \mathbf{T}^{(\ell)}\}\,\forall\,\ell \in \mathcal{M}_B$ and the variational posterior distribution of document embeddings $q(\mathbf{w}_d)\,\forall\, d \in \mathcal{B}$.

2.2 Extending to newer languages

Since the model uses language-specific parameters and vocabulary, it is possible to extend the model to a new set languages (denoted by $\bar{\mathcal{M}}$) without re-training from scratch. The necessary conditions are that every new language ($\bar{\ell}$) should have parallel text with at least one other language from $\mathcal{M} \cup \bar{\mathcal{M}}$ subject to the constraint that there exists at least one parallel pair between $\mathcal{M}$ and $\bar{\mathcal{M}}$. This can be seen as continual learning, and requires only to learn the parameters corresponding to the newer languages $\{\mathbf{m}^{(\bar{\ell})}, \mathbf{T}^{(\bar{\ell})}\}\,\forall\,\bar{\ell} \in \bar{\mathcal{M}}$. It also means that the performance on existing seed languages is unaffected with the addition of newer languages. In this paper, we show the results from experiments where we start with a seed model covering 6 languages, which is then extended to 11 newer languages. Similar approaches are also explored for multilingual neural machine translation (Bérard, 2021).

2.3 Inferring embeddings

Given a bag-of-word statistics from an unseen document from any of the $\ell \in \mathcal{M} \cup \bar{\mathcal{M}}$ languages, we can infer (extract) the corresponding document embedding along with its uncertainty. This is done by keeping the language-specific model parameters $\{\mathbf{m}^{(\ell)}, \mathbf{T}^{(\ell)}\}$ constant, and iteratively optimizing the objective in (3) with respect to the parameters of the variational distribution. In the resulting variational posterior $q(\mathbf{w}) = \mathcal{N}(\mathbf{w} \mid \boldsymbol{\nu}, \operatorname{diag}(\boldsymbol{\gamma})^{-1})$, the mean $\boldsymbol{\nu}$ represents the (most likely) document embedding, and variance $\operatorname{diag}(\boldsymbol{\gamma})^{-1}$ encodes the uncertainty around the mean $\boldsymbol{\nu}$. Since all the documents and language-specific model parameters are independent (Fig. 1), inferring the embeddings can be parallelized and is computationally cheaper.

3 Classification exploiting uncertainties

In a typical setting where we have only point estimates of embeddings, all the embeddings are considered equally important by a classifier. This may not be true all the time. For example, shorter and documents with many rare words can result in poor estimates of the embeddings; which can affect parameters of the classifier during training, and also the performance during prediction. Additionally, there might be noise while projecting embeddings from multiple languages into the same semantically aligned latent space. The proposed model yields document embeddings represented by Gaussian distributions, with the uncertainty about the embedding encoded in the covariance. These uncertainties are specific to each example and can be seen as *heteroscedastic aleatoric* uncertainties (Kendall and Gal, 2017). We present two linear classifiers that can exploit this uncertainty. The first one is the generative Gaussian linear classifier with uncertainty (GLCU) (Kesiraju et al., 2020). The second one is the discriminative multi-class logistic regression with uncertainty (MCLRU).

3.1 Generative classifier

In generative classifiers, the posterior probability of class label ($\mathcal{C}_k$) given a feature vector (embedding) $\mathbf{w}$ is computed from the joint distribution

$$p(\mathcal{C}_k \mid \mathbf{w}) = \frac{p_\theta(\mathbf{w} \mid \mathcal{C}_k)\, p(\mathcal{C}_k)}{\sum_j p_\theta(\mathbf{w} \mid \mathcal{C}_j)\, p(\mathcal{C}_j)} \qquad (5)$$

where, $p_\theta(\mathbf{w} \mid \mathcal{C}_k)$ is the likelihood function parametrized by θ, and $p(\mathcal{C}_k)$ is the class prior. In generative classifiers, the likelihood function is assumed to have a known parametric form (e.g. Gaussian, Multinomial). For Gaussian linear classifier (GLC), the likelihood function is $p_\theta(\boldsymbol{w} \mid \mathcal{C}_k) = \mathcal{N}(\boldsymbol{w} \mid \boldsymbol{\mu}_k, \mathbf{S}^{-1})$, where $\boldsymbol{w}$ is the input feature (point estimate of the embedding), $\boldsymbol{\mu}_k$ is the mean of class $\mathcal{C}_k$, and $\mathbf{S}$ is the precision matrix shared across all the classes.

Given that the input features come in the form of Gaussian distributions, i.e., $q(\mathbf{w}) = \mathcal{N}(\mathbf{w} \mid \boldsymbol{\nu}, \operatorname{diag}(\boldsymbol{\gamma})^{-1})$, we can integrate out (exploit) the uncertainty in the input while evaluating the likelihood function. In the case of GLC, where the likelihood function is also Gaussian, the expected likelihood has an analytical form:

$$\mathbb{E}_q[p_\theta(\mathbf{w} \mid \mathcal{C}_k)] = \mathcal{N}(\boldsymbol{\nu} \mid \boldsymbol{\mu}_k, \mathbf{S}^{-1} + \operatorname{diag}(\boldsymbol{\gamma})^{-1}). \qquad (6)$$

GLC with likelihood function replaced by (6) is called GLCU. Both are essentially the same classifiers, i.e., they have the same assumptions about the underlying data and hence the same model parameters. The only difference lies in the evaluation of likelihood function.

3.2 Discriminative classifier

For discriminative classifier such as multi-class logistic regression (LR), the posterior probability of a class ($\mathcal{C}_k$) given an input feature vector $\mathbf{w}$ is

$$p(\mathcal{C}_k \mid \mathbf{w}) = \frac{\exp\{\mathbf{h}_k^{\mathsf{T}}\mathbf{w} + b_k\}}{\sum_j \exp\{\mathbf{h}_j^{\mathsf{T}}\mathbf{w} + b_j\}}, \qquad (7)$$

where $\{b_k,\, \mathbf{h}_k\}\, \forall\, k$ are the parameters of the classifier. Unlike in GLC, we cannot analytically compute the expectation over (7) with-respect-to the input embeddings (Gaussian distributions). Instead we approximate the expectation using Monte Carlo samples (Xiao and Wang, 2019):

$$p(\mathcal{C}_k \mid \mathbf{w}) \approx \frac{1}{M} \sum_{m=1}^{M} \frac{\exp\{\mathbf{h}_k^{\mathsf{T}}\boldsymbol{\varepsilon}_m + b_k\}}{\sum_j \exp\{\mathbf{h}_j^{\mathsf{T}}\boldsymbol{\varepsilon}_m + b_j\}}, \quad (8)$$

$\boldsymbol{\varepsilon}_m \sim q(\mathbf{w})\, \forall\, m$. Eq. (8) represents the posterior probability computation for logistic regression with uncertainty (LRU).

Theoretically, given the true uncertainties in the training examples, GLCU and LRU can better estimate the model parameters of the classifier. Similarly, it can also exploit the uncertainties in the test examples during prediction. However, in our case, the uncertainties are estimated using the Bayesian multilingual document model as described in Section 2.3. The underlying assumption here is that uncertainties extracted using the model are close enough to the true uncertainties as expected by the classifiers, which is empirically supported through our experimental results presented in Section 5.

4 Experimental setup

This section presents the details on data for multilingual training of MBay model and dataset preparation for downstream classification (topic ID) task. We also discuss the details of various pre-trained multilingual models and downstream classifiers that are used in our experiments.

4.1 Data for multilingual training

The following datasets were used for training the proposed MBay model. Europarl(v7) (Koehn, 2005), UNPC(v1) (Ziemski et al., 2016), MultiUN(v1) (Eisele and Chen, 2010), GlobalVoices(v2018q4) (Tiedemann, 2012), News-Commentary(v16) (Akhbardeh et al., 2021), CVIT(PIBv1.3, MKB) (Siripragada et al., 2020), Samanantar(indic2indic) (Ramesh et al., 2022), Japanese-English Wikipedia, and CCAligned(EN-JA) (El-Kishky et al., 2020). The total number of sentences used are 17.89M covering 17 languages. All the words were lower-cased and punctuation was stripped. Further, words that do not occur in at least two sentences were removed. We used scikit-learn (Pedregosa et al., 2011) for pre-processing. More details are given in Appendix A.

4.2 Dataset preparation for topic ID

The original MLDoc corpus was prepared (Schwenk and Li, 2018) in order to have a standard training, development (dev) and test sets across 8 languages[2]. The usual setup contains 1000 samples each or training and dev, and 4000 for test, across 4 classes (topics). The aim was to create a class balanced sets (uniform class prior), which gives us 250 samples per topic in both training and dev, and 1000 samples per topic in the test. However, not every language in the original Reuters Multilingual Corpus (RCV) has enough examples, hence the class prior is not uniform (Schwenk and Li, 2018). Moreover, it only covers a small subset (6000 samples in total) of the actual RCV corpus, and results from such as smaller subset tend be less certain. To address this, we use the MLDoc data preparation scripts, and create 5 different splits of the data, where each split contains the same aforementioned number of training, dev and test samples. This is analogous to a 5-fold cross-validation scheme. The mean and standard deviations across 5 splits are reported during evaluation. The experimental results show that such a robust evaluation is needed as the standard deviation across 5 splits is noticeable (see Section 5 and Appendix E).

IndicNLP-suite (Kakwani et al., 2020) contains several resources for NLP in Indian languages. From this suite, we take the IndicNLP news articles (INA for short) classification dataset, and prepare a cross-lingual setup similar to that of MLDoc. The INA comprises of 9 languages[3] covering 7 classes (topics). However, not all the 7 topics are present in the news articles across all the 9 languages. In order to make cross-lingual experiments across multiple languages, we consider two setups: A two-class setup covering all 9 languages, and a three-class setup covering 5 languages. We keep at most of 250 samples per topic in both training and dev, and 1000 samples per topic in the test. Finally, we create 5 such splits, which allows us to report mean and standard deviations.

As we re-processed both MLDoc and INA datasets, we call the newer versions as MLDoc5x and INA5x respectively, where 5 represents the five different splits. Details in Appendix B.

ω	EN	DE	FR	IT	ES	RU	Avg. (s.d.)
5e-02	85.34	88.82	89.28	78.74	88.32	77.38	84.65 (4.84)
5e-03	85.88	90.72	**89.70**	**80.78**	**89.36**	**79.78**	**86.04** (4.34)
5e-04	**86.50**	**90.88**	89.68	79.88	88.62	79.34	85.82 (4.58)

Table 1: In language classification accuracy (in %) on the dev sets of MLDoc5x for various hyper-parameters of MBay-6L seed model. The embedding dimension is fixed to 256 and the classifier is GLCU.

4.3 MBay configurations

The proposed Bayesian multilingual document model has two important hyper-parameters, i.e., latent (embedding) dimension K and ℓ_2 regularization weight ω corresponding to the model parameters $\{\mathbf{T}^{(\ell)}\}\,\forall\,\ell$. We fixed the embedding dimension to 256 and explored $\omega \in \{5e-02, 5e-03, 5e-04\}$. The prior distribution (1) was set

[2] DE, EN, ES FR, IT, JA, RU, ZH

[3] BN, GU, KN, ML, MR, OR, PA, TE, TA

to $\mathcal{N}(\mathbf{w} \mid \mathbf{0}, (0.1)\mathbf{I})$ and the variational distribution $q(\mathbf{w})$ was initialized to be the same as prior. This enabled us to use same learning rate for both mean and variance parameters. The number of Monte Carlo samples R for approximating the objective function (4) was set to 8, which we found to be a reasonable trade-off between computation complexity and convergence speed. A maximum batch size of 4096 was used during training. A constant learning rate of $5e-02$ was used both during training and inference. The model is trained for a maximum of 100 epochs and inference is done fora maximum of 50 iterations. Our models are implemented using PyTorch (Paszke et al., 2017) and will be made public.

4.4 Topic ID systems for MBay

In total we trained 4 different linear classifiers on the embeddings extracted from MBay model. The first two linear classifiers, GLC and LR are trained using only the point estimates of the embeddings, i.e., using only the mean parameter ($\boldsymbol{\nu}$). The next two classifiers, GLCU and LRU are trained with the full posterior distributions of embeddings, $q(\mathbf{w}) = \mathcal{N}(\mathbf{w} \mid \boldsymbol{\nu}, \mathrm{diag}(\boldsymbol{\gamma})^{-1})$, as described in Section 3. To better illustrate the importance of uncertainties during the test (prediction) time, we used the trained GLC and LR models, but during the prediction, we evaluate likelihood using the full posterior distributions (along with uncertainties) of the test document embeddings. This is valid because both GLC and GLCU have exactly the same model parameters (Section 3.1). Similarly LR and LRU have exactly the same model parameters (Section 3.2). We represent these two classifiers as GLCU-P and LRU-P, where -P denotes *uncertainty exploited only during prediction*.

The generative classifiers (GLC, GLCU) have no hyper-parameters to tune. We added ℓ_2 regularization term with weight $\alpha \in \{1e-4, \ldots 5e+1\}$ for the parameters of LR, LRU. This classifier was trained for a maximum 100 epochs using ADAM with a constant learning rate of $5e-2$. For LRU, we used $M = 32$ for the empirical approximation (8). $M > 32$ did not affect the classification performance significantly but, lower values degraded the performance about 5%.

Initially three MBay models were trained on 6 languages (DE, EN, ES, FR, IT, RU) with different hyper-parameters. We performed in-language classification on MLDoc5x using GLCU on these 6 languages and picked the MBay model configuration that gave the best performance on dev set. These results are presented in Table 1. We denote this seed model as MBay-6L. This model with the same hyper-parameter ($\omega = 5e-03$) is then extended independently to {JA, ZH}, and to 9 Indian languages using EN as pivot (bridge). More details are in Appendix C.

4.5 Pre-trained multilingual models

There are numerous pre-trained multilingual models from which we picked the following[4] based on their diversity in architecture, training criterion and overall performance.

LASER (Artetxe and Schwenk, 2019) is based on seq2seq BiLSTM trained in 223M parallel sentence covering 93 languages, sharing a common subword vocabulary. The language-agnostic embeddings are obtained by forward propagating through the encoder followed by a pooling layer.

XLM-R-stsb (Reimers and Gurevych, 2020) is based on sentence transformers (Reimers and Gurevych, 2019) and XLM-R (Conneau et al., 2020), where knowledge distillation is used to adapt the the multilingual student model XLM-R to align the representations from BERT.

LaBSE (Feng et al., 2020) is based on dual-encoder architecture and is trained on 17B monolingual sentences for MLM, and on 6B translation pairs for translation ranking task, covering 109 languages. The pre-trained model is available for public, whereas the exact training data is not.

Distill-mUSE is multilingual knowledge distilled version of mUSE (Yang et al., 2020). While the original was trained on 15 languages, this version supports 50 languages (Reimers and Gurevych, 2020).

We trained two different classifiers on the embeddings extracted pre-trained multilingual language models. The first one is a two layer perceptron (MLP) widely used in prior works (Artetxe and Schwenk, 2019). The second one is the LR.

5 Results and discussion

Here we present only the main zero-shot transfer results, while the detailed results are given in the Appendix E (Tables 9, 10, 11). The mean and std. deviation across 5 splits for MLDoc and INA are only presented in the Appendix. For LASER + MLP system, we observed around 14 points of std.

[4] More details are given in Table 8 from Appendix.

Model	Classifier	Zero-shot transfer (source language to the rest) EN	DE	FR	IT	ES	RU	JA	ZH	ZS* Avg.	IL* Avg.
LASER^{ag}	MLP	73.28	73.47	71.98	70.84	68.13	69.08	66.29	72.53	70.70	88.46
LASER^{a}	MLP	71.43	72.57	74.73	70.02	71.25	68.27	54.82	68.35	68.93	**88.91**
LASER^{a}	LR	70.52	73.12	75.80	70.56	74.99	66.27	48.37	68.46	65.81	88.65
LASER^{p}	MLP	74.76	75.02	75.93	69.55	69.41	69.32	60.95	68.00	70.37	87.81
LASER^{p}	LR	73.97	75.19	75.75	70.22	73.93	68.68	61.70	69.34	**71.10**	87.87
XLM-R-stsb^{t}	MLP	74.29	72.48	74.02	70.84	70.24	69.08	72.06	70.28	**71.66**	**87.09**
XLM-R-stsb^{t}	LR	72.87	70.87	72.61	68.49	68.31	65.76	70.75	69.47	69.89	85.78
XLM-R-stsb^{m}	MLP	68.11	68.10	69.80	66.22	65.92	66.98	64.03	63.94	66.64	85.80
XLM-R-stsb^{m}	LR	67.18	67.79	68.10	64.47	64.17	64.35	63.17	62.17	65.17	84.63
Distil-mUSE^{t}	MLP	75.92	74.86	75.90	72.51	74.01	69.84	69.77	71.40	73.03	88.14
Distil-mUSE^{t}	LR	77.02	76.41	76.98	76.04	74.80	71.28	72.02	74.08	**74.83**	**88.33**
Distil-mUSE^{m}	MLP	73.34	73.33	73.81	71.72	74.66	69.85	68.05	71.29	72.01	87.92
Distil-mUSE^{m}	LR	74.55	75.30	75.57	74.17	74.57	71.07	68.51	73.75	73.44	88.14
LaBSE^{t}	MLP	80.02	79.29	79.11	78.70	79.93	77.16	78.42	76.90	78.69	**89.93**
LaBSE^{t}	LR	80.48	79.91	80.00	79.08	80.02	76.71	78.60	78.04	**79.13**	89.85
LaBSE^{m}	MLP	79.29	80.07	80.36	78.76	79.07	76.05	78.49	76.17	78.53	89.84
LaBSE^{m}	LR	80.27	80.22	79.81	78.72	79.66	75.92	78.95	77.47	78.88	89.77
MBay	GLC	65.04	64.71	65.39	61.65	62.28	57.05	54.02	59.99	61.27	83.70
MBay	GLCU	74.14	70.07	72.40	73.20	72.64	67.57	64.48	66.03	70.06	85.30
MBay	GLCU-P	74.08	70.46	72.59	73.07	72.62	67.66	64.35	66.49	**70.16**	86.05
MBay	LR	70.33	70.43	71.18	67.30	68.45	62.35	59.81	65.69	66.94	**86.35**
MBay	LRU	72.59	71.02	71.87	72.04	71.31	65.11	63.21	65.17	69.04	86.08
MBay	LRU-P	72.92	70.83	71.67	71.89	69.57	64.30	63.42	65.59	68.77	86.02

Table 2: Results on MLDoc5x. a: Averaging sentence embeddings. g: Results taken from official GitHub repository. p: Max-pooling over encoder outputs. t: Input trimmed to 128 tokens. m : Input trimmed to maximum sequence length. ZS*: Zero-shot. IL*: In-language.

deviation across 5 splits in MLDoc5x when transferring from IT→DE (Table 9). Higher (> 5) std. deviations are also observed for other pre-trained models in different transfer directions. This suggests that one needs to have a robust evaluation scheme in order to study and compare the performance of multilingual models across various languages and tasks. Further, when reporting average results, care should be taken to separate them into in-language vs (zero-shot) transfer directions. A simple way to summarize the results is to compute average only across transfer directions for every language (excluding the source language). This gives us an idea of how well the model can transfer to other languages on an average. The in-language classification accuracy across various languages should be reported separately.

The first row from Table 2 show the results with LASER on the original single MLDoc split. We tried to replicate the results, but observed significant variance for JA and ZH (see Table 7 in Appendix D). All the subsequent rows are the average results on MLDoc5x for systems based on various pre-trained models and the proposed MBay model. For most of the pre-trained models we can see that LR performs slightly better than MLP in zero-shot transfer setting. The results from MBay are comparable to LASER and XLM-R-stsb, while LaBSE outperforms all the other systems. Moreover, in case of MBay, we can see that generative classifier exploiting uncertainty outperforms the discriminative classifiers. This suggests that in the common embedding space, our classifiers are able to exploit the estimated uncertainty from the MBay model.

The Tables 3 and 4 show average results in INA5x under 2-class and 3-class settings respectively. Here we can see that MBay outperforms other pre-trained models except LaBSE; and XLM-R-stsb in 3-class setting. The poor performance of LASER and XLM-R could be attributed to less and low-quality training data for these (mid-resource) Indian languages. The objective function of mUSE and LaBSE are similar where as the quality and quantity of the training data is much different. LaBSE was trained on large amounts of high-quality (manually verified, and filtered) data, which could explain its superior performance. Unfortunately, the exact training data used for LaBSE is not available for public.

6 Conclusions

In this paper, we revisited zero-shot cross-lingual topic identification. We identified shortcomings in the evaluation protocol of MLDoc corpus. We

Model	CLF.	Zero-shot transfer (from language)									ZS*	IL*
		BN	GU	KN	ML	MR	OR	PA	TA	TE	Avg.	Avg.
LASER^p	MLP	76.41	-	-	77.26	77.19	-	-	74.52	77.46	**76.57**	92.77
LASER^p	LR	76.70	-	-	77.86	74.85	-	-	74.22	77.15	76.16	90.90
XLM-R-stsb^t	MLP	90.77	89.38	94.13	91.92	91.78	92.29	89.88	91.95	92.72	91.65	95.52
XLM-R-stsb^t	LR	90.84	88.82	93.32	92.40	90.78	91.43	89.00	91.88	91.03	**91.08**	96.09
XLM-R-stsb^m	MLP	85.60	88.57	91.95	89.42	87.58	91.39	88.62	86.75	90.37	88.92	95.66
XLM-R-stsb^m	LR	87.69	87.68	92.74	90.98	87.61	91.83	88.61	85.41	90.99	89.28	95.48
Distil-mUSE^t	MLP	-	84.80	-	-	82.14	-	-	-	-	83.47	93.65
Distil-mUSE^t	LR	-	83.24	-	-	86.39	-	-	-	-	**84.81**	93.50
Distil-mUSE^m	MLP	-	77.65	-	-	72.53	-	-	-	-	75.09	92.98
Distil-mUSE^m	LR	-	83.79	-	-	76.51	-	-	-	-	80.15	92.55
LaBSE^t	MLP	96.41	96.91	97.18	97.31	97.43	96.83	96.41	97.43	97.13	97.00	98.03
LaBSE^t	LR	96.62	96.71	97.78	97.21	97.31	97.37	96.40	97.34	97.63	97.15	98.06
LaBSE^m	MLP	95.92	96.98	97.48	97.37	97.54	96.46	96.90	96.93	97.19	96.97	97.95
LaBSE^m	LR	96.60	96.69	97.56	97.23	97.56	97.47	96.87	97.38	97.75	**97.23**	97.98
MBay	GLC	67.22	50.86	82.83	50.47	50.71	82.20	49.54	83.64	85.38	66.98	73.48
MBay	GLCU	91.89	93.54	94.69	93.72	94.01	94.67	93.93	93.47	94.43	93.82	96.67
MBay	GLCU-P	92.67	93.24	94.96	94.38	95.48	94.70	93.65	93.86	94.66	**94.18**	97.03
MBay	LR	91.59	90.76	93.41	93.14	93.34	92.91	91.54	90.96	92.91	92.28	96.44
MBay	LRU	92.50	91.95	94.49	93.56	94.67	94.17	92.27	92.16	94.24	93.34	96.80
MBay	LRU-P	92.36	91.97	94.45	93.37	94.58	93.26	91.34	92.28	94.09	93.08	96.67

Table 3: Results on INA5x 2-class setup.

Model	CLF.	Zero-shot transfer (from language)					ZS*	IL*
		GU	ML	OR	PA	TE	Avg.	Avg.
LASER^p	MLP	-	72.83	-	-	83.90	78.37	93.51
LASER^p	LR	-	73.97	-	-	83.39	**78.68**	93.38
XLM-R-stsb^t	MLP	90.57	91.91	91.79	89.43	93.05	**91.35**	95.22
XLM-R-stsb^t	LR	88.94	90.99	90.68	86.41	91.38	89.68	93.79
XLM-R-stsb^m	MLP	86.99	87.86	90.16	87.23	90.09	88.47	95.39
XLM-R-stsb^m	LR	82.62	87.95	88.77	83.78	90.12	86.65	93.78
LaBSE^t	MLP	97.43	97.41	97.11	96.52	97.14	97.12	98.21
LaBSE^t	LR	95.91	96.86	96.69	95.28	97.01	96.35	97.64
LaBSE^m	MLP	97.45	97.47	96.65	96.92	97.18	**97.13**	98.09
LaBSE^m	LR	95.62	96.79	96.59	95.25	97.00	96.25	97.47
MBay	GLC	32.40	33.87	82.01	32.68	84.89	53.17	57.82
MBay	GLCU	89.21	89.92	91.12	88.63	90.58	89.89	95.29
MBay	GLCU-P	89.42	90.89	91.19	88.97	90.97	**90.29**	95.78
MBay	LR	87.16	87.18	89.18	85.84	88.84	87.64	94.95
MBay	LRU	87.87	89.91	89.84	86.64	90.22	88.90	95.26
MBay	LRU-P	87.00	90.09	89.92	86.76	90.22	88.80	95.16

Table 4: Results on INA5x 3-class setup.

proposed a simple robust alternative by creating 5 different splits and reporting the mean and std.dev. of the results. The same protocol was extended to Indic news articles dataset covering 9 languages. We benchmarked some of the diverse and popular pre-trained models on the new evaluation protocol covering 17 languages (128 transfer directions). We also presented a Bayesian multilingual document model, which learns language-independent document embeddings along with their uncertainties. We propagated the uncertainties into a generative and discriminative linear classifier for zero-shot cross-lingual topic ID. Our proposed system in budget friendly in terms of computation, while at the same time performs competitively to other large scale pre-trained models such as LASER, and XLM-R. We belive our MBay model can act as a strong baseline for future research works in the direction of cross-lingual topic ID. We observe that there is a need for creating a larger and diverse dataset covering several topics and languages.

7 Limitations

While we aimed to cover 17 languages, the number of topics in classification experiments are at most 4. There is a need to benchmark these systems on a diverse and large multi-label cross-lingual dataset. The proposed MBay model is build on bag-of-words simplification and may not be a suitable choice for fine-grained semantic similarity tasks.

References

Farhad Akhbardeh, Arkady Arkhangorodsky, Magdalena Biesialska, Ondřej Bojar, Rajen Chatterjee, Vishrav Chaudhary, Marta R. Costa-jussa, Cristina España-Bonet, Angela Fan, Christian Federmann, Markus Freitag, Yvette Graham, Roman Grundkiewicz, Barry Haddow, Leonie Harter, Kenneth Heafield, Christopher Homan, Matthias Huck, Kwabena Amponsah-Kaakyire, Jungo Kasai, Daniel Khashabi, Kevin Knight, Tom Kocmi, Philipp Koehn, Nicholas Lourie, Christof Monz, Makoto Morishita, Masaaki Nagata, Ajay Nagesh, Toshiaki Nakazawa, Matteo Negri, Santanu Pal, Allahsera Auguste Tapo, Marco Turchi, Valentin Vydrin, and Marcos Zampieri. 2021. Findings of the 2021 conference on machine translation (WMT21). In *Proceedings of the Sixth Conference on Machine Translation*, pages 1–88, Online. Association for Computational Linguistics.

Waleed Ammar, George Mulcaire, Yulia Tsvetkov, Guillaume Lample, Chris Dyer, and Noah A. Smith. 2016. Massively multilingual word embeddings. *CoRR*, abs/1602.01925.

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

Christopher M. Bishop. 2006. *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer-Verlag New York, Inc., Secaucus, NJ, USA.

David M. Blei. 2012. Probabilistic topic models. *Commun. ACM*, 55(4):77–84.

Alexandre Bérard. 2021. Continual Learning in Multilingual NMT via Language-Specific Embeddings. In *Proc. of the Sixth Conference on Machine Translation (WMT)*, pages 542–565. ACL.

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.

Sumanth Doddapaneni, Gowtham Ramesh, Anoop Kunchukuttan, Pratyush Kumar, and Mitesh M. Khapra. 2021. A primer on pretrained multilingual language models. *CoRR*, abs/2107.00676.

Andreas Eisele and Yu Chen. 2010. Multiun: A multilingual corpus from united nation documents. In *Proceedings of the International Conference on Language Resources and Evaluation, LREC 2010, 17-23 May 2010, Valletta, Malta*. European Language Resources Association.

Ahmed El-Kishky, Vishrav Chaudhary, Francisco Guzmán, and Philipp Koehn. 2020. CCAligned: A massive collection of cross-lingual web-document pairs. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 5960–5969, Online. Association for Computational Linguistics.

Fangxiaoyu Feng, Yinfei Yang, Daniel Cer, Naveen Arivazhagan, and Wei Wang. 2020. Language-agnostic BERT sentence embedding. *CoRR*, abs/2007.01852.

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. In *Proceedings of the 37th International Conference on Machine Learning (ICML)*.

Haoyang Huang, Yaobo Liang, Nan Duan, Ming Gong, Linjun Shou, Daxin Jiang, and Ming Zhou. 2019. Unicoder: A Universal Language Encoder by Pre-training with Multiple Cross-lingual Tasks. In *Proceedings of the 2019 Conference on EMNLP 9th IJCNLP, 2019, Hong Kong, China, November 3-7, 2019*, pages 2485–2494. Association for Computational Linguistics.

Divyanshu Kakwani, Anoop Kunchukuttan, Satish Golla, Gokul N. C., Avik Bhattacharyya, Mitesh M. Khapra, and Pratyush Kumar. 2020. Indicnlpsuite: Monolingual corpora, evaluation benchmarks and pre-trained multilingual language models for indian languages. In *Findings of the Association for Computational Linguistics: EMNLP 2020, Online Event, 16-20 November 2020*, volume EMNLP 2020 of *Findings of ACL*, pages 4948–4961. Association for Computational Linguistics.

Alex Kendall and Yarin Gal. 2017. What Uncertainties Do We Need in Bayesian Deep Learning for Computer Vision? In *Advances in Neural Information Processing Systems 30*, pages 5574–5584. Curran Associates, Inc.

Santosh Kesiraju, Oldřich Plchot, Lukaš Burget, and Suryakanth V. Gangashetty. 2020. Learning Document Embeddings Along With Their Uncertainties. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 28:2319–2332.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.

Diederik P Kingma and Max Welling. 2014. Auto-Encoding Variational Bayes. In *2nd International Conference on Learning Representations, ICLR Conference Track Proceedings*, Banff, AB, Canada.

Philipp Koehn. 2005. Europarl: A Parallel Corpus for Statistical Machine Translation. In *Conference*

Proceedings: the tenth Machine Translation Summit, pages 79–86, Phuket, Thailand. AAMT, AAMT.

Yishu Miao, Lei Yu, and Phil Blunsom. 2016. Neural variational inference for text processing. In *Proceedings of the 33rd International Conference on International Conference on Machine Learning*, ICML'16, pages 1727–1736, New York, NY, USA. JMLR.org.

David Mimno, Hanna M. Wallach, Jason Naradowsky, David A. Smith, and Andrew McCallum. 2009. Polylingual topic models. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing*, pages 880–889, Singapore. Association for Computational Linguistics.

Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. 2017. Automatic differentiation in PyTorch. In *NIPS Workshop*.

F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. 2011. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.

Gowtham Ramesh, Sumanth Doddapaneni, Aravinth Bheemaraj, Mayank Jobanputra, Raghavan AK, Ajitesh Sharma, Sujit Sahoo, Harshita Diddee, Mahalakshmi J, Divyanshu Kakwani, Navneet Kumar, Aswin Pradeep, Srihari Nagaraj, Kumar Deepak, Vivek Raghavan, Anoop Kunchukuttan, Pratyush Kumar, and Mitesh Shantadevi Khapra. 2022. Samanantar: The Largest Publicly Available Parallel Corpora Collection for 11 Indic Languages. *Transactions of the Association for Computational Linguistics*, 10:145–162.

Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3982–3992, Hong Kong, China. Association for Computational Linguistics.

Nils Reimers and Iryna Gurevych. 2020. Making monolingual sentence embeddings multilingual using knowledge distillation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4512–4525, Online. Association for Computational Linguistics.

Danilo Jimenez Rezende, Shakir Mohamed, and Daan Wierstra. 2014. Stochastic backpropagation and approximate inference in deep generative models. In *Proceedings of the 31st International Conference on Machine Learning*, volume 32 of *Proceedings of Machine Learning Research*, pages 1278–1286, Bejing, China. PMLR.

Sebastian Ruder, Ivan Vulić, and Anders Søgaard. 2019. A survey of cross-lingual word embedding models. *J. Artif. Int. Res.*, 65(1):569–630.

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, Rep4NLP@ACL 2017, Vancouver, Canada, August 3, 2017*, pages 157–167.

Holger Schwenk and Xian Li. 2018. A Corpus for Multilingual Document Classification in Eight Languages. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation, LREC 2018, Miyazaki, Japan, May 7-12, 2018*.

Aditya Siddhant, Melvin Johnson, Henry Tsai, Naveen Ari, Jason Riesa, Ankur Bapna, Orhan Firat, and Karthik Raman. 2020. Evaluating the Cross-Lingual Effectiveness of Massively Multilingual Neural Machine Translation. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI NY, USA, February 7-12, 2020*, pages 8854–8861. AAAI Press.

Shashank Siripragada, Jerin Philip, Vinay P. Namboodiri, and C V Jawahar. 2020. A multilingual parallel corpora collection effort for Indian languages. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 3743–3751, Marseille, France. European Language Resources Association.

Emma Strubell, Ananya Ganesh, and Andrew McCallum. 2019. Energy and Policy Considerations for Deep Learning in NLP. In *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, pages 3645–3650. Association for Computational Linguistics.

Jörg Tiedemann. 2012. Parallel data, tools and interfaces in OPUS. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation, LREC 2012, Istanbul, Turkey, May 23-25, 2012*, pages 2214–2218. European Language Resources Association (ELRA).

Shijie Wu and Mark Dredze. 2019. Beto, Bentz, Becas: The Surprising Cross-Lingual Effectiveness of BERT. In *Proceedings of the 2019 Conference on EMNLP and the 9th IJCNLP*, pages 833–844, Hong Kong, China. Association for Computational Linguistics.

Yijun Xiao and William Yang Wang. 2019. Quantifying Uncertainties in Natural Language Processing Tasks. In *The Thirty-Third AAAI Conference on Artificial Intelligence, AAAI, Honolulu, Hawaii, USA, January 27 - February 1, 2019*, pages 7322–7329.

Weiwei Yang, Jordan Boyd-Graber, and Philip Resnik. 2019. A multilingual topic model for learning weighted topic links across corpora with low comparability. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 1243–1248, Hong Kong, China. Association for Computational Linguistics.

Yinfei Yang, Daniel Cer, Amin Ahmad, Mandy Guo, Jax Law, Noah Constant, Gustavo Hernandez Abrego, Steve Yuan, Chris Tar, Yun-hsuan Sung, Brian Strope, and Ray Kurzweil. 2020. Multilingual universal sentence encoder for semantic retrieval. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 87–94, Online. Association for Computational Linguistics.

Michal Ziemski, Marcin Junczys-Dowmunt, and Bruno Pouliquen. 2016. The united nations parallel corpus v1. 0. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation LREC 2016*.

A Data for multilingual training

- We considered only top 450k sentence from EN-JA pair from CCAligned corpus, which was further filtered based on heuristics, resulting in 185k parallel sentences.

- From UNPC(v1), we considered only top 2 million sentences.

- The initial seed model (MBay-6L) was trained on 6 languages (DE, EN, ES, FR, IT, RU) using the data from Europarl, UNPC, MultiUN, Global-Voices, and News-Commentary. From these datasets, we considered only those sentences that are at least 30 words long.

- The seed model (MBay-6L) is extended to JA and ZH languages with the help of parallel data from UNPC, MultiUN, Wikipedia (EN-JA)[5], filtered CCAligned (EN-JA), Global-Voices and News-Commentary.

- The seed model (MBay-6L) is extended to 9 Indian languages (BN, GU, ML, MR, KN, OR, PA, TA, TE) with the help of parallel data from CVIT (PIB, MKB), Samanantar (indic2indic), Global-Voices and News-Commentary datasets. From these datasets, we considered only those sentences that are at least 10 words long.

- The Table 5 shows the detailed statistics of the number of sentences and their parallel languages across all the 17 languages.

B Data for Topic ID

This section presents the statistics of MLDoc5x and INA5x topic ID datasets created for the experiments reported in this paper. We attempted to keep about 250 examples per topic in each training and development sets, and 1000 examples per topic in the test set. We created 5 such splits and the average number of examples per language-set-topic are illustrated in Tables 6. The original data for languages GU, ML, PA, were smaller, hence they have smaller number of examples per set.

C MBay models

- The initial (seed) MBay models were trained on 6 languages (DE, EN, ES, FR, IT, RU) using the parallel data (7.48M sentences) described in Appendix A. The training took about 25 hrs on a single NVIDIA RTX A6000 with 48 GB of memory. The trained model has 154M parameters. This model trained on 6 languages is referred as MBay-6L.

- The MBay-6L seed model was extended to JA, ZH using EN as pivot. It was trained on 3.3M parallel sentences, and took about 11 hrs on a similar GPU. This extended training added 51.4M additional parameters for JA and ZH. During training, the parameters of EN were frozen and the parameters of other languages (DE, ES, FR, IT, RU) were not loaded as they are not required.

- The MBay-6L seed model was extended to 9 Indian languages using EN as pivot. It was trained on 7.29M parallel sentences, and took about 21hrs to train on a similar GPU. This added 96.2M additional parameters to represent 9 Indian languages. As the vocabulary sizes for these languages is not as big as other high-resources languages (Table 5, the number of additional parameters were also relatively less.

[5] https://alaginrc.nict.go.jp/WikiCorpus/index_E.html

Group	Language	ISO code	Parallel pairs	Sentences (M)	Tokens (M)	Vocabulary size
$\mathcal{E}$	English	EN	$\mathcal{E} \cup \mathcal{U} \cup \mathcal{I} \setminus \{\text{KN}\}$	3.89	154.46	100k
$\mathcal{E}$	French	FR	$\mathcal{E}$	1.62	73.38	100k
$\mathcal{E}$	German	DE	$\mathcal{E}$	0.85	33.02	100k
$\mathcal{E}$	Italian	IT	$\mathcal{E}$	0.67	25.18	100k
$\mathcal{E}$	Russian	RU	$\mathcal{E}$	1.11	37.01	100k
$\mathcal{E}$	Spanish	ES	$\mathcal{E}$	1.64	74.03	100k
$\mathcal{U}$	Chinese	ZH	{EN, JA}	1.19	54.84	100k
$\mathcal{U}$	Japanese	JA	{EN, ZH}	0.37	21.15	100k
$\mathcal{I}$	Kannada	KN	$\mathcal{I}$	0.36	7.92	25521
$\mathcal{I}$	Bengali	BN	$\mathcal{I} \cup$ {EN}	0.95	20.05	36925
$\mathcal{I}$	Gujarati	GU	$\mathcal{I} \cup$ {EN}	0.75	15.92	28268
$\mathcal{I}$	Malayalam	ML	$\mathcal{I} \cup$ {EN}	0.57	13.89	36877
$\mathcal{I}$	Marathi	MR	$\mathcal{I} \cup$ {EN}	0.86	18.43	30557
$\mathcal{I}$	Odia	OR	$\mathcal{I} \cup$ {EN}	0.50	11.36	25450
$\mathcal{I}$	Punjabi	PA	$\mathcal{I} \cup$ {EN}	0.95	15.34	24209
$\mathcal{I}$	Tamil	TA	$\mathcal{I} \cup$ {EN}	0.93	21.21	33960
$\mathcal{I}$	Telugu	TE	$\mathcal{I} \cup$ {EN}	0.68	11.96	32548
	Total			17.89	609.16	

Table 5: Statistics of the data used in training and extending the MBay model. Sentences and tokens are in millions (M).

Lang.	**MLDoc5x Topics**											
	CCAT			ECAT			GCAT			MCAT		
DE	257	270	1019	245	259	936	251	259	1022	247	237	1023
EN	257	270	1019	245	259	936	251	259	1022	247	237	1023
ES	305	310	1186	198	197	782	205	214	816	292	279	1216
FR	257	270	1019	245	259	936	251	259	1022	247	237	1023
IT	257	270	1019	245	259	936	251	259	1022	247	237	1023
JA	257	270	1019	245	259	936	251	259	1022	247	237	1023
RU	274	283	1081	255	265	1023	204	256	819	267	256	1077
ZH	306	313	1193	282	312	1187	118	93	401	294	282	1219
	INA5x Topics											
	Entertainment			Sports			Business					
BN	250	250	1001	250	249	999	-	-	-			
GU	34	37	149	37	36	147	40	38	160			
KN	233	237	931	235	231	938	-	-	-			
ML	78	81	319	83	80	328	75	76	302			
MR	121	122	494	120	123	494	-	-	-			
OR	237	232	928	236	233	948	237	237	959			
PA	40	41	163	40	39	165	43	44	170			
TA	236	232	928	231	235	935	-	-	-			
TE	248	249	988	252	253	997	250	247	1015			

Table 6: Number of examples in each *topic* for every *language* in RCV (MLDoc5x) and IndicNLP news articles (INA5x) datasets. Under each topic, the three columns represent training, development and test *sets* respectively. Each number represents the average number (rounded to nearest integer) of examples across 5 splits for the respective *language-set-topic*.

D MLDoc results with LASER

We tried to replicate the MLDoc results using LASER, however we found significant differences is few language directions. The Table 7 shows the absolute differences in the results we obtained as compared the ones reported in the official github repository: `https://github.com/facebookresearch/LASER/tree/main/tasks/mldoc`.

In Table 7, a positive value indicates that we obtained a better result, while a negative value indicates the opposite.

	Test language							
	EN	DE	FR	IT	ES	RU	JA	ZH
EN	-0.42	-1.70	-4.72	-0.23	3.05	0.13	-1.67	-3.30
DE	0.93	-0.68	0.65	-2.18	-1.83	0.30	**-7.17**	-0.78
FR	-1.60	-2.40	-0.58	-2.22	-1.98	-0.17	**-6.87**	**-13.31**
IT	-2.38	-2.59	-2.25	2.78	3.20	**5.86**	**-6.47**	**-10.72**
ES	-0.24	-2.70	-2.85	-1.58	-4.93	**7.80**	-4.05	**8.95**
RU	0.53	-2.76	0.35	2.55	2.25	-1.00	-2.05	2.17
JA	**10.70**	**14.37**	**10.28**	**6.23**	**11.87**	**9.53**	-0.07	**15.32**
ZH	2.02	0.20	1.01	0.38	**5.85**	0.17	4.15	0.51

Table 7: Discrepancy in replicating the results of LASER + MLP system.

Model	URL
LASER	`https://github.com/facebookresearch/LASER`
XLM-R-stsb	`https://huggingface.co/sentence-transformers/stsb-xlm-r-multilingual`
LaBSE	`https://huggingface.co/sentence-transformers/LaBSE`
Distil-mUSE	`https://huggingface.co/sentence-transformers/distiluse-base-multilingual-cased-v2`

Table 8: Pre-trained models and their download URL.

E Detailed results

Here we present the detailed results i.e., mean and std.dev. across 5-splits in all the transfer directions. For each pre-trained multilingual model, we only show the results of the system that yielded best downstream performance. Notice in Tables 9, 10 and 11 that the high std.dev. indicates that the choosing a different training / dev / test split could result in different performance of the system. The original MLDoc is sampled from RCV multilingual corpus and had only one such split and hence couldn't capture the variance in the results.

The first two parts in Table 9 show LASER^p + MLP and LASER^p + LR. Notice that for LR the variance across 5 splits is much lower as compared to MLP.

	Test language							
	EN	DE	FR	IT	ES	RU	JA	ZH
	LASER^p + MLP							
EN	87.0 (0.7)	85.5 (1.4)	83.0 (3.8)	68.4 (4.6)	77.8 (2.2)	67.4 (2.7)	67.1 (3.8)	74.1 (0.7)
DE	73.7 (4.9)	92.1 (0.1)	83.9 (0.7)	73.4 (1.4)	81.0 (1.9)	66.9 (5.7)	71.6 (4.9)	**74.6 (9.5)**
FR	76.2 (0.5)	88.0 (0.4)	90.5 (0.5)	72.3 (1.2)	79.9 (2.0)	68.0 (0.9)	69.8 (0.5)	77.2 (1.0)
IT	**61.3 (12.5)**	**78.8 (14.1)**	**77.1 (8.3)**	84.2 (0.7)	76.0 (2.6)	63.4 (4.6)	**61.2 (7.6)**	**69.1 (6.5)**
ES	64.4 (0.3)	81.9 (1.9)	78.8 (3.1)	74.4 (2.5)	92.5 (0.1)	57.7 (3.3)	**64.1 (7.6)**	**64.5 (12.2)**
RU	64.9 (2.3)	78.1 (5.5)	**70.2 (7.3)**	66.7 (3.1)	70.2 (5.4)	83.4 (0.3)	**67.5 (6.3)**	**67.5 (11.3)**
JA	58.6 (1.1)	70.7 (3.5)	62.8 (2.7)	57.3 (1.3)	59.8 (1.9)	51.7 (2.9)	85.9 (0.1)	**65.8 (8.1)**
ZH	**63.7 (7.9)**	76.3 (5.2)	**70.5 (7.9)**	**64.8 (6.7)**	68.5 (3.9)	61.0 (1.9)	71.2 (2.5)	86.9 (0.7)
	LASER^p + LR							
EN	87.3 (0.5)	86.2 (1.1)	81.9 (1.2)	67.2 (2.1)	76.8 (2.0)	66.0 (2.0)	66.3 (2.3)	73.5 (2.1)
DE	73.0 (1.4)	92.4 (0.3)	83.0 (0.8)	73.5 (1.1)	81.1 (1.2)	68.0 (0.8)	71.3 (0.9)	76.4 (2.0)
FR	75.1 (0.3)	88.2 (1.0)	90.3 (0.5)	72.8 (1.1)	79.3 (1.0)	67.9 (2.0)	69.3 (1.9)	77.6 (2.4)
IT	61.1 (1.8)	80.4 (2.6)	77.1 (2.0)	84.5 (0.7)	76.6 (1.7)	64.4 (1.0)	**61.9 (3.6)**	**70.1 (3.5)**
ES	68.5 (0.5)	84.1 (0.8)	81.6 (1.3)	76.0 (1.3)	92.7 (0.4)	64.9 (1.2)	69.5 (1.5)	72.9 (2.3)
RU	64.8 (0.8)	76.9 (1.4)	69.3 (2.4)	66.6 (1.3)	69.1 (1.8)	83.1 (0.4)	67.0 (1.2)	**66.9 (3.6)**
JA	60.3 (1.4)	72.9 (1.1)	64.1 (1.9)	56.4 (1.5)	60.4 (1.6)	50.1 (1.2)	85.7 (0.4)	67.6 (1.7)
ZH	64.2 (2.4)	77.5 (1.2)	71.6 (2.2)	66.8 (0.9)	68.4 (1.5)	64.0 (1.7)	73.1 (0.4)	87.0 (0.7)
	XLM-R-stsb^t + MLP							
EN	88.0 (0.7)	85.1 (1.2)	79.4 (1.7)	69.4 (1.0)	78.8 (0.8)	**66.3 (3.7)**	68.8 (1.6)	72.2 (2.7)
DE	75.1 (0.7)	92.5 (0.4)	83.0 (0.5)	71.5 (2.4)	77.9 (0.9)	61.7 (0.5)	69.5 (2.6)	**68.7 (4.1)**
FR	77.3 (1.2)	87.8 (1.4)	89.7 (0.8)	72.9 (2.0)	78.6 (2.2)	63.7 (2.7)	69.0 (2.9)	**68.9 (3.3)**
IT	69.0 (1.2)	82.6 (1.2)	79.1 (1.6)	83.0 (0.6)	77.9 (1.7)	**56.5 (3.1)**	67.6 (1.8)	63.2 (1.7)
ES	71.3 (1.5)	78.7 (2.6)	78.4 (1.8)	72.4 (2.6)	92.3 (0.3)	**57.8 (6.1)**	68.4 (1.5)	64.7 (2.1)
RU	69.3 (0.7)	76.9 (2.8)	75.3 (1.2)	65.0 (2.8)	70.9 (2.1)	82.9 (0.2)	63.6 (1.6)	**62.4 (3.8)**
JA	71.6 (1.4)	83.0 (0.7)	77.1 (1.2)	66.4 (1.6)	73.7 (1.0)	**61.7 (4.1)**	83.7 (0.7)	70.9 (1.2)
ZH	70.1 (2.8)	79.4 (2.7)	75.1 (2.5)	64.3 (2.5)	**69.3 (3.1)**	62.6 (2.2)	71.2 (1.3)	84.7 (0.4)
	Distil-mUSE^t + LR							
EN	89.3 (0.3)	85.9 (1.4)	82.4 (1.3)	69.1 (1.9)	77.7 (1.5)	62.3 (2.4)	**65.2 (3.3)**	79.3 (1.3)
DE	77.7 (1.0)	93.1 (0.4)	84.9 (0.3)	73.0 (0.6)	79.6 (1.5)	66.4 (1.0)	**65.5 (4.2)**	80.1 (1.0)
FR	78.7 (0.6)	89.4 (0.4)	90.7 (0.4)	73.1 (1.0)	80.0 (1.7)	64.4 (1.7)	63.3 (1.1)	80.1 (1.3)
IT	72.1 (2.2)	83.4 (1.2)	81.4 (0.9)	83.7 (0.6)	79.8 (1.9)	64.5 (1.8)	**60.7 (3.7)**	77.2 (1.1)
ES	77.7 (1.1)	85.7 (1.3)	82.5 (1.1)	75.4 (0.7)	92.7 (0.4)	**63.9 (3.1)**	60.8 (1.8)	76.1 (1.8)
RU	70.8 (2.6)	82.0 (1.9)	74.3 (2.3)	66.5 (1.0)	68.8 (2.4)	83.2 (0.4)	64.1 (0.8)	71.1 (2.1)
JA	70.4 (2.1)	76.6 (1.9)	71.9 (1.9)	62.7 (1.3)	66.3 (2.7)	56.0 (2.4)	85.0 (0.6)	75.6 (1.3)
ZH	76.1 (2.2)	83.6 (2.1)	77.9 (2.1)	71.2 (2.0)	75.5 (1.6)	67.0 (2.2)	65.0 (2.2)	87.4 (0.4)
	LaBSE^t + LR							
EN	90.6 (0.4)	89.0 (0.9)	87.8 (0.6)	76.2 (1.3)	82.7 (1.0)	**69.6 (3.4)**	75.8 (1.1)	82.2 (0.6)
DE	77.8 (1.5)	93.8 (0.3)	88.2 (0.7)	76.2 (0.8)	84.9 (2.2)	72.0 (2.9)	76.5 (1.5)	83.7 (0.7)
FR	81.4 (0.4)	91.1 (0.6)	92.1 (0.4)	76.3 (1.2)	83.6 (1.4)	**71.0 (3.2)**	73.7 (1.1)	82.9 (1.3)
IT	73.3 (1.2)	87.0 (0.6)	84.2 (1.1)	86.6 (0.2)	85.2 (0.7)	71.6 (1.1)	71.7 (1.2)	80.6 (1.3)
ES	77.9 (1.5)	89.9 (0.6)	86.9 (1.3)	81.0 (0.9)	93.9 (0.3)	**68.9 (3.0)**	75.4 (1.5)	81.6 (1.0)
RU	73.3 (1.1)	86.1 (1.8)	80.7 (2.2)	74.1 (1.3)	**72.8 (3.4)**	86.0 (0.5)	71.2 (1.8)	78.8 (1.5)
JA	76.8 (0.5)	87.1 (1.3)	83.7 (1.4)	72.8 (0.5)	79.5 (1.1)	67.6 (2.7)	86.2 (0.5)	82.8 (1.8)
ZH	76.8 (0.6)	86.6 (2.4)	83.4 (2.7)	74.2 (2.5)	79.2 (1.5)	**69.4 (3.8)**	76.7 (1.9)	89.6 (0.6)
	MBay + GLCU-P							
EN	86.8 (0.3)	85.6 (0.3)	82.4 (1.0)	70.3 (1.1)	78.5 (0.6)	65.0 (1.7)	66.8 (1.6)	70.1 (1.5)
DE	75.2 (0.9)	91.1 (0.5)	85.6 (0.5)	70.6 (1.0)	79.8 (1.1)	54.9 (1.7)	57.9 (2.7)	69.3 (1.3)
FR	75.3 (0.6)	87.0 (0.7)	89.8 (0.4)	74.1 (0.4)	80.9 (0.6)	65.7 (1.3)	51.9 (1.1)	73.2 (1.3)
IT	73.8 (1.0)	84.6 (0.9)	83.9 (0.5)	82.1 (0.7)	82.9 (0.9)	58.3 (2.4)	59.5 (1.8)	68.4 (0.9)
ES	74.1 (0.6)	84.6 (1.1)	82.5 (0.7)	75.3 (0.7)	89.1 (0.2)	58.5 (1.8)	64.7 (1.6)	68.6 (1.2)
RU	66.8 (1.7)	74.9 (1.6)	75.3 (1.7)	67.6 (1.3)	73.6 (1.2)	81.2 (0.5)	55.3 (1.5)	60.1 (2.8)
JA	67.5 (0.9)	76.6 (1.2)	68.1 (1.5)	59.9 (0.8)	67.4 (0.8)	51.6 (1.5)	84.7 (0.7)	59.4 (1.2)
ZH	69.1 (0.9)	77.9 (1.8)	73.5 (1.9)	63.5 (1.2)	66.9 (1.5)	50.8 (1.9)	63.7 (1.1)	83.7 (0.6)

Table 9: Detailed classification results on the MLDoc5x test sets using various models with best downstream classification performance. Values in the parenthesis indicate the std.dev. across 5 splits. Bold values indicate the numbers with std.dev > 3. p: Max-pooling over encoder outputs. t: Input trimmed to 128 tokens. m : Input trimmed to maximum sequence length.

	Test language								
	BN	GU	KN	ML	MR	OR	PA	TA	TE
	LASER^t + MLP								
BN	95.4 (0.5)	-	-	90.8 (0.5)	85.4 (2.5)	-	-	**57.4 (8.1)**	**72.0 (5.5)**
ML	86.2 (1.6)	-	-	93.4 (1.8)	86.3 (2.8)	-	-	**63.7 (5.2)**	72.8 (2.2)
MR	**89.2 (3.0)**	-	-	88.8 (2.4)	93.4 (1.0)	-	-	**59.8 (7.0)**	70.9 (2.3)
TA	80.6 (2.3)	-	-	77.8 (2.2)	**76.9 (5.1)**	-	-	88.1 (0.7)	**62.8 (5.1)**
TE	**83.6 (3.9)**	-	-	**83.9 (3.3)**	86.0 (1.9)	-	-	56.3 (2.9)	93.7 (0.5)
	XLM-R-stsb^t + LR								
BN	95.0 (0.3)	92.4 (0.9)	87.4 (2.2)	93.6 (1.1)	93.6 (1.2)	91.3 (1.7)	**91.5 (3.0)**	93.7 (1.4)	**83.2 (3.1)**
GU	84.7 (1.9)	93.4 (2.0)	84.0 (1.7)	91.5 (1.0)	92.7 (1.2)	**90.6 (3.5)**	**89.3 (4.0)**	91.9 (1.4)	85.8 (1.9)
KN	**88.5 (3.3)**	94.3 (0.7)	93.7 (0.7)	95.6 (0.6)	95.0 (0.5)	93.0 (1.0)	94.1 (1.6)	96.6 (0.3)	89.5 (1.1)
ML	88.3 (2.4)	93.6 (1.2)	89.6 (2.1)	95.3 (0.9)	95.6 (0.8)	94.5 (0.5)	94.8 (0.7)	95.6 (1.0)	87.2 (0.5)
MR	86.7 (2.1)	94.8 (0.7)	83.9 (2.3)	94.4 (0.7)	96.4 (0.3)	94.5 (0.7)	93.8 (2.0)	94.1 (0.4)	84.1 (1.3)
OR	**90.1 (3.4)**	93.4 (1.8)	86.6 (1.4)	93.0 (2.9)	95.3 (1.0)	96.3 (0.4)	92.8 (2.8)	94.1 (1.5)	86.0 (2.5)
PA	87.3 (1.9)	91.9 (1.3)	84.8 (1.3)	90.7 (2.0)	91.7 (1.9)	91.7 (1.4)	94.5 (2.0)	92.7 (1.8)	81.3 (2.3)
TA	90.6 (2.0)	92.9 (2.0)	89.7 (1.4)	95.0 (1.0)	95.6 (0.6)	**90.0 (3.4)**	91.5 (2.4)	97.9 (0.4)	89.7 (0.3)
TE	87.2 (2.6)	91.9 (1.5)	89.0 (1.8)	92.0 (2.6)	93.5 (2.0)	**89.8 (4.4)**	**90.1 (3.7)**	94.8 (1.9)	94.7 (0.7)
	LaBSE^m + LR								
BN	97.0 (0.4)	95.5 (2.1)	95.0 (0.7)	97.3 (0.8)	96.1 (1.7)	97.6 (0.4)	97.6 (1.2)	97.5 (2.2)	96.2 (1.1)
GU	95.4 (0.7)	96.9 (1.2)	95.6 (0.5)	97.3 (0.3)	96.5 (0.6)	97.3 (0.5)	97.4 (1.0)	97.2 (1.0)	96.7 (0.4)
KN	95.7 (1.1)	96.7 (0.8)	96.9 (0.2)	98.8 (0.6)	97.9 (0.3)	97.3 (0.4)	98.1 (0.4)	98.6 (0.3)	97.5 (0.6)
ML	95.2 (1.1)	97.0 (1.1)	96.2 (0.3)	98.4 (0.3)	98.0 (0.3)	97.6 (0.6)	97.8 (0.7)	98.6 (0.4)	97.4 (0.4)
MR	96.5 (0.3)	97.2 (1.0)	96.5 (0.2)	98.4 (0.6)	98.1 (0.2)	98.1 (0.2)	98.4 (0.5)	98.6 (0.2)	97.0 (0.3)
OR	96.5 (0.6)	96.6 (0.7)	96.2 (0.2)	98.0 (0.6)	98.0 (0.2)	98.4 (0.2)	98.5 (0.8)	98.6 (0.4)	97.4 (0.3)
PA	96.4 (0.3)	95.6 (0.9)	95.5 (0.2)	97.7 (0.4)	97.5 (0.3)	97.9 (0.2)	98.5 (0.5)	98.1 (0.4)	96.3 (0.3)
TA	96.6 (0.4)	96.5 (1.2)	96.5 (0.3)	98.1 (0.2)	97.6 (0.6)	98.0 (0.2)	98.0 (0.9)	99.2 (0.1)	97.7 (0.2)
TE	96.7 (0.4)	97.4 (0.9)	96.9 (0.3)	98.3 (0.5)	97.9 (0.2)	97.4 (0.5)	98.3 (0.9)	99.1 (0.2)	98.5 (0.2)
	MBay + GLCU-P								
BN	96.2 (0.6)	91.8 (2.4)	91.5 (0.7)	90.0 (0.8)	94.5 (0.4)	93.6 (0.4)	91.8 (1.1)	96.4 (0.5)	92.0 (0.9)
GU	94.4 (0.9)	97.0 (0.9)	91.6 (0.7)	92.0 (0.3)	94.3 (1.0)	94.8 (0.5)	91.2 (0.9)	96.7 (0.6)	90.8 (1.6)
KN	91.4 (1.4)	94.4 (1.9)	95.6 (0.5)	94.6 (1.1)	95.9 (0.4)	95.8 (0.6)	95.1 (1.3)	98.0 (0.3)	94.5 (0.5)
ML	93.1 (1.9)	94.0 (2.3)	93.5 (0.5)	97.2 (0.6)	94.9 (1.2)	94.6 (1.6)	93.8 (2.3)	98.0 (0.4)	93.2 (0.3)
MR	94.4 (1.2)	95.0 (1.4)	94.3 (0.5)	96.0 (0.3)	96.7 (0.3)	96.5 (0.5)	96.1 (1.1)	98.2 (0.4)	93.4 (0.7)
OR	93.9 (0.7)	95.0 (0.9)	93.1 (0.6)	95.8 (0.7)	96.5 (0.6)	97.9 (0.2)	94.5 (1.4)	96.9 (0.3)	91.8 (0.5)
PA	**92.2 (3.6)**	93.4 (1.9)	92.2 (0.8)	93.9 (0.8)	95.1 (0.6)	94.0 (0.9)	97.2 (0.8)	97.1 (0.6)	91.2 (0.6)
TA	93.4 (0.4)	94.3 (1.4)	93.0 (0.8)	94.4 (1.4)	94.8 (0.9)	93.5 (0.4)	93.4 (0.7)	98.7 (0.1)	94.1 (0.5)
TE	91.9 (1.2)	95.3 (0.7)	94.2 (0.6)	94.2 (1.1)	95.4 (0.6)	94.7 (0.3)	93.4 (1.6)	98.1 (0.2)	96.7 (0.4)

Table 10: Detailed classification results on the 2-class setup from INA5x test sets using various models with best downstream classification performance. Values in the parenthesis indicate the std.dev. across 5 splits. Bold values indicate the numbers with std.dev > 3. p: Max-pooling over encoder outputs. t: Input trimmed to 128 tokens. m : Input trimmed to maximum sequence length.

	Test language				
	GU	ML	OR	PA	TE
	XLM-R-stsb^t + MLP				
GU	94.2 (1.5)	92.6 (1.1)	92.6 (1.0)	91.3 (1.8)	85.8 (2.9)
ML	93.2 (1.5)	95.2 (0.7)	93.8 (0.5)	94.0 (1.6)	86.7 (1.8)
OR	94.0 (1.1)	94.3 (0.9)	96.4 (0.4)	93.1 (2.6)	85.8 (2.0)
PA	91.6 (1.6)	91.5 (1.4)	92.3 (2.4)	95.3 (1.2)	82.3 (2.9)
TE	93.5 (0.9)	93.4 (1.0)	92.9 (1.8)	92.3 (1.0)	94.9 (0.7)
	LaBSE^m + MLP				
GU	97.0 (1.0)	97.7 (0.4)	97.6 (0.3)	97.6 (1.1)	97.0 (0.3)
ML	97.1 (0.8)	98.3 (0.4)	97.5 (0.8)	97.8 (0.6)	97.4 (0.4)
OR	95.4 (1.7)	97.5 (0.3)	98.3 (0.3)	97.4 (0.6)	96.3 (1.1)
PA	95.9 (0.9)	97.4 (0.3)	97.9 (0.3)	98.5 (0.5)	96.4 (0.2)
TE	97.0 (1.0)	97.8 (0.5)	96.8 (0.9)	97.1 (1.2)	98.4 (0.3)
	MBay + GLCU-P				
GU	96.0 (0.3)	90.6 (0.9)	88.4 (0.9)	89.2 (1.6)	89.5 (1.3)
ML	92.6 (1.6)	94.9 (0.7)	88.9 (1.3)	91.3 (1.4)	90.8 (0.6)
OR	91.6 (0.5)	92.3 (0.6)	96.5 (0.3)	91.0 (1.2)	89.8 (0.7)
PA	91.1 (1.3)	90.2 (1.1)	85.0 (2.0)	95.3 (1.1)	89.5 (0.7)
TE	93.2 (1.3)	90.9 (0.6)	87.5 (0.7)	92.3 (1.4)	96.1 (0.3)

Table 11: Detailed classification results on the 2-class setup from INA5x test sets using various models with best downstream classification performance. Values in the parenthesis indicate the std.dev. across 5 splits. Bold values indicate the numbers with std.dev > 3. p: Max-pooling over encoder outputs. t: Input trimmed to 128 tokens. m : Input trimmed to maximum sequence length.