To token or not to token: A Comparative Study of Text Representations for Cross-Lingual Transfer

Md Mushfiqur Rahman, Fardin Ahsan Sakib, Fahim Faisal, Antonios Anastasopoulos
Department of Computer Science, George Mason University
{mrahma45,fsakib,ffaisal,antonis}@gmu.edu

Abstract

Choosing an appropriate tokenization scheme is often a bottleneck in low-resource cross-lingual transfer. To understand the downstream implications of text representation choices, we perform a comparative analysis on language models having diverse text representation modalities including 2 segmentation-based models (BERT, mBERT), 1 image-based model (PIXEL), and 1 character-level model (CANINE). First, we propose a scoring Language Quotient (LQ) metric capable of providing a weighted representation of both zero-shot and few-shot evaluation combined. Utilizing this metric, we perform experiments comprising 19 source languages and 133 target languages on three tasks (POS tagging, Dependency parsing, and NER). Our analysis reveals that image-based models excel in cross-lingual transfer when languages are closely related and share visually similar scripts. However, for tasks biased toward word meaning (POS, NER), segmentation-based models prove to be superior. Furthermore, in dependency parsing tasks where word relationships play a crucial role, models with their character-level focus, outperform others. Finally, we propose a recommendation scheme based on our findings to guide model selection according to task and language requirements. [1]

1 Introduction

The performance of multilingual language models varies substantially across languages, with low-resource languages demonstrating particularly sub-optimal results compared to their high-resource counterparts. This disparity poses a global challenge for deploying effective NLP applications, given the diverse linguistic landscape worldwide (Blasi et al., 2022).

To address this challenge, cross-lingual transfer has emerged as a promising solution. By leveraging knowledge from high-resource languages, cross-lingual transfer aims to enhance the performance of low-resource ones. However, the effectiveness of cross-lingual knowledge transfer is not uniformly observed across all language pairs. It is influenced by various factors, including language style, structure, origin, dataset quality (Yu et al., 2022; Kreutzer et al., 2022), and the specific relationship between the source and target languages (Ahmad et al., 2019; He et al., 2019). On top of that, the selection of an appropriate language model becomes crucial to achieve successful cross-lingual knowledge transfer. While most state-of-the-art models rely on tokenization (Schuster and Nakajima, 2012; Gage, 1994), yielding high scores for various linguistic downstream tasks, their performance in terms of cross-lingual transfer has room for further investigation. Considering that word formation can significantly vary across different languages, differences in tokenization techniques can hinder the transfer of linguistic capabilities between languages (Hofmann et al., 2022). Hence, the exploration of tokenization-free models is also imperative.

This study thoroughly investigates the role and effectiveness of both tokenization-based (Devlin et al., 2019a) and tokenization-free models (Rust et al., 2022) in cross-lingual knowledge transfer. Our selection of models encompasses BERT and mBERT (Devlin et al., 2019a), which uses tradiional subword-based segmentation. In addition, we delve into tokenization-free models such as CANINE (Clark et al., 2022) and PIXEL (Rust et al., 2022). CANINE leverages character-level information to accommodate the diverse word formations and structures found in different languages. On the other hand, PIXEL represents texts using visual elements, introducing new possibilities for script-based transfer in visually similar languages.

In this study, we perform standard syntactic task evaluation in both zero-shot and few-shot manner

---

[1]The code for reproducing our results is available here https://github.com/mushfiqur11/tokenfreetransfer.

to evaluate the cross-lingual transfer capabilities of these models. While accuracy, F1 score, Labeled Attachment Score (LAS), etc. are all effective evaluation indicators of the goodness of a model, they are not particularly representative of how much a model has learned in a short span of training. We utilize these common metrics over zero-shot and few-shot steps and propose the Learning Quotient (LQ) metric, a novel scoring metric that depends on the relation between the zero-shot and few-shot scores. The metric evaluates the linguistic characteristics of the languages with the model's performance on the tasks. This metric enables a comprehensive evaluation of cross-lingual transfer capabilities, offering valuable insights into the strengths and weaknesses of the models. Our findings suggest contrastive downstream performance that relates to the model architecture. Furthermore, we present a decision tree framework, based on this extensive analysis providing practical guidance for selecting appropriate models based on specific task requirements and language relationships. This framework serves as a tool for researchers and practitioners seeking to harness the potential of NLP applications across diverse languages.

2 Methodology

Problem formulation   In this work, we use pre-trained language models and fine-tune them on source languages followed by few-shot training on the target languages. Consider the sets of target $T = \{t_1, t_2, \ldots, t_m\}$ and source languages $S = \{s_1, s_2, \ldots, s_n\}$. We assume source languages $s \in S$ have adequate resources for effective language model training. Conversely, target languages $t \in T$ are low-resource languages with limited data. For any language pair (s, t), we aim to quantify how efficiently a language model can learn the target language t using knowledge transferred from the source language s. Given the scarcity of data for t, our focus lies on the model's performance in the early stages of fine-tuning it, denoted by the evaluation score E.

Let $(M)_s^\infty$ represents a language model M fully finetuned on the language s and $(M)_t^c$ represents the model finetuned up to c steps. We investigate how fast can a model learn the language t in the early steps if it was previously finetuned on s. Essentially, we measure the performance of the model $((M)_s^\infty)_t^c$ where c is a small positive integer. It's important, however, to acknowledge that the effi-

ciency of this method can be influenced by factors such as the similarities between the source and target languages, as well as the quality and quantity of data available for both.

Our methodology can be broadly divided into two steps:

Fine-tuning on Sources   Following the pre-trained model selection, each system is fine-tuned using the selected source languages. This fine-tuning stage allows each system to adjust and optimize its parameters based on specific requirements. Once fine-tuned, the systems are prepared for the evaluation phase in a cross-lingual transfer scenario.

Evaluation and Scoring   The last step involves evaluating each system's performance on target language tasks after undergoing a certain amount of fine-tuning. Two scores are measured at this point: zero-shot and few-shot scores. To measure the final score, we calculate the LQ-score (§2). This score allows us to determine the speed and efficiency at which each system learns a new language based on the knowledge transferred from the source language.

Learning Quotient(LQ) metric   Let us denote $E_s^{(t_c)}$ as the score achieved by the model $(M)s^\infty$ on the language t after c steps of training on t. For different tasks, E can be different. We use accuracy for POS tagging and NER, and Labeled Attachment Score (LAS) for dependency parsing. $E_s^{(t_0)}$ stands for the zero-shot score of the model on t. Using the same logic, $\frac{1}{n} \sum_{i=0}^n E_i^{(t_0)}$ is the average zero-shot score across all source languages, denoted as Z_A.

Now, let's introduce our proposed scoring metric, applicable for any pair of languages $t \in T$ and $s \in S$:

$$LQ(t, s) = \frac{\left(E_s^{(t_c)} - Z_A\right)\left(E_s^{(t_c)} + E_s^{(t_0)}\right)}{Z_A + \epsilon} \quad (1)$$

$LQ(t, s)$ is comprised of two primary terms, along with a normalization factor. The first term measures the performance of the model after few-shot training on language t, relative to the average zero-shot scores for that target language. The second term simply sums the zero-shot and the few-shot scores. To normalize the metric value, we employ the average zero-shot score, Z_A. A minute value ϵ is added to the denominator to avoid division by zero cases.
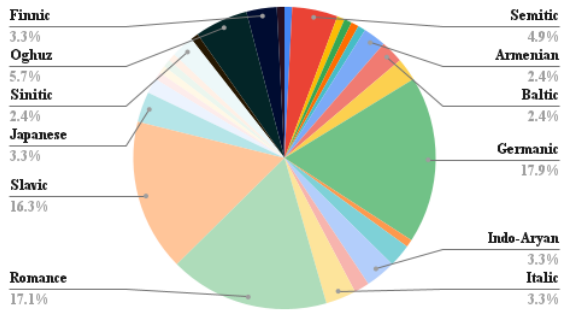
Figure 1: Distribution of the languages according to their sub-families. The majority of these are of Indo-European origin. The languages belong to 28 sub-families spanning 13 different families
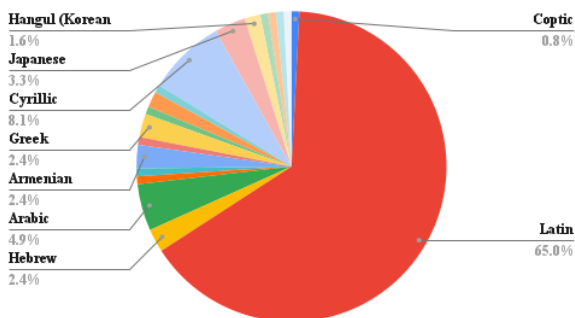

Figure 2: Distribution of the languages according to their scripts. The majority of these use Latin script. The languages use 19 different scripts

The *LQ* score provides positive reinforcement for both zero-shot and few-shot scores. Any few-shot score that falls below the zero-shot average incurs a substantial penalty. This metric proves effective in quantifying the *pace* at which a model adapts to a new language.[2]

3 Experimentation

Task Selection We perform the evaluation on three downstream tasks that heavily depend on fundamental linguistic capabilities and syntactic structure: Dependency Parsing, Part-of-Speech (POS) tagging and Named Entity Recognition (NER). These tasks can work as indicators of a model's understanding of language dynamics and its ability to comprehend and interpret linguistic information (Chen and Manning, 2014; Manning, 2011; Lample et al., 2016)

Language and Dataset Selection For the execution of POS tagging and Dependency Parsing, we utilized the Universal Dependencies (UD) Dataset

(Nivre et al., 2017, 2020). To maintain focus and ensure a meaningful study, we selected 9 languages (as listed in Figure 3(a)) as our source languages and 123 languages as our target languages for the experiments[3]. All the models were comprehensively fine-tuned on the selected source languages, thereby establishing a baseline for performance comparison[4]. For NER, we utilized the MashakhaNER dataset (Adelani et al., 2021) and all its associated languages as sources and targets (as described in Figure 3(b)). MasakhaNER mainly focuses on a few African languages. These languages are quite low-resource. Hence, these were perfect for this research.

Model Selection To ensure a fair comparison, we use `BERT`, `mBERT`, `CANINE`, and `PIXEL` as our choice of pre-trained models. `BERT` and `mBERT` use sub-word segmentation whereas `CANINE` is a character-based model. Unlike these, `PIXEL` represents text using visual elements rather than traditional tokens. We selected `BERT`, as it is the most well-established tokenization-based model that aligns with `PIXEL`'s pre-training dataset. On the other hand, character-level models provide another perspective for understanding and processing languages, capturing the distinct attributes of word formations. `CANINE`, with its pre-training on 104 languages, emerged as a strong candidate. As a counterpart, we chose `mBERT`, which shares a similar scope of pre-training languages.

Experimental Setup Our experiments involved two major training phases followed by a result extraction step. In the first training phase, each language model was fully fine-tuned on each of the source languages for each task. The experimental setup maintained a high computational standard to ensure robust training and evaluation. All experiments were conducted on a remote server equipped with an A100 GPU. The analysis was conducted over 4 (models) x 9 (source languages) x 123 (target languages) data points for Dependency Parsing and POS tagging. For NER, the analysis was conducted over all 4 (models) x 12 (source languages) x 12 (target languages) data points. We used 10 fine-tuning steps (for §1, set $c = 10$) for the target languages for all tasks.

For reproducing the results, the language models can be fully fine-tuned on the source languages (our

---

[2]The proof can be found in Appendix A.2

[3]A detailed list is provided in appendix A.5

[4]All fine-tuned models are available on HuggingFace for further research and investigation

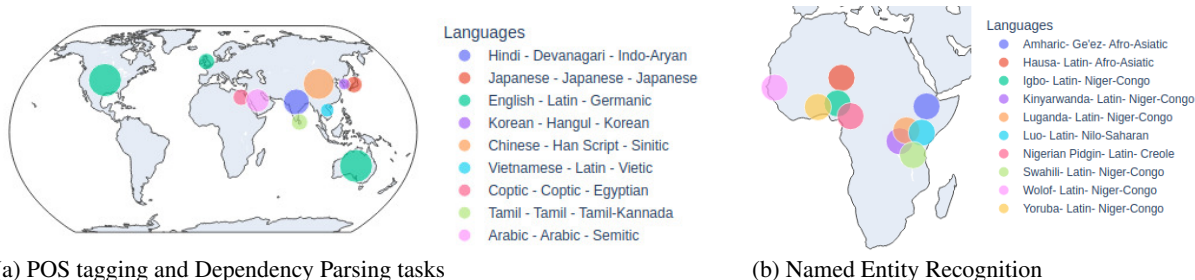(a) POS tagging and Dependency Parsing tasks   (b) Named Entity Recognition

Figure 3: Geographic distribution of source languages (with script and family) used in the analysis across tasks.

finetuned versions can be used directly from HuggingFace) to get the zero-shot results. These models can then be finetuned on the target languages for 10 steps to get the few-shot score.

4 Results and Discussion

First, we break down the results by several key variables including the visual similarity of languages, their lexical correspondence, and the type of language task. Then, we discuss the performance of these models in light of these variables, revealing patterns regarding model characteristics.

4.1 Visual similarity is all you need

Case1 (English → European) Both of PIXEL and BERT are pre-trained in English. Therefore, for a fair comparison with other models, we perform a comparison where English is the only source language. For evaluation, we consider various European languages, taking into account both lexical similarity and the LQ score on the POS tagging task. Figure 4 represent the LQ scores of PIXEL and CANINE when English is used as the source language and various other languages as the targets. Here, in Figure 4(a) we observe the proficiency of PIXEL in handling tasks between languages sharing a similar script. For example, English shares similar degrees of lexical similarity with French (0.27) and Russian (0.24) (§A.5 and §A.6). However, when considering the LQ scores, French significantly outperforms Russian for PIXEL. Moreover, despite Spanish and Portuguese exhibiting low lexical similarity coefficients with English, they both have achieved high LQ scores. A key factor contributing to these scores is the usage of the Latin script. French, Spanish, and Portuguese, which have all garnered high scores, also use the Latin script. Russian employs a different (Cyrillic) script, which likely explains its relatively lower score. Finnish, despite its use of the Latin script, belongs to a different language family compared to

English, which may account for the less impressive performances. Moreover, when the script is non-Latin as presented in Figure 4(b), CANINE has an edge over PIXEL. The lexical similarities between different European languages are outlined in Table 8 in the appendix.

| | POS Tagging | | | |
Model	Hindi→Urdu Score	Rank	Hindi→Marathi Score	Rank
PIXEL	-0.4	94	17.9	5
CANINE	96.1	3	14.6	15
mBERT	102.2	2	7.3	112

Table 1: Comparison between different language models on Hindi as the source and Urdu and Marathi as target shows CANINE and mBERT massively favor linguistically similar languages. PIXEL favors visual similarity

Case2 (Hindi → Urdu | Marathi) Despite the high mutual intelligibility and substantial grammatical and linguistic similarities between Hindi and Urdu, as acknowledged in the literature (Bhatt, 2005), the LQ score on the POS tagging task attained by PIXEL for this language pairing is not as high as one would anticipate (ranked 94th). The relatively low performance can be attributed to their disparate scripts, underscoring the importance of visual similarity when using image-based language models such as PIXEL. However, for the other three models, with Hindi as the source, Urdu ranked in the top 3 target languages. Table 1 represents this phenomenon.

On the flip side, Hindi and Marathi are not mutually intelligible. But both of these languages use the Devanagari script. Sorting the LQ scores for Hindi as the source language, Marathi comes out as one of the top-performing target languages (4th).

Case3 (Arabic → X) In the case of Arabic as the source language, PIXEL received the highest scores for Persian (ranked 2nd) and Urdu (ranked 3rd) as
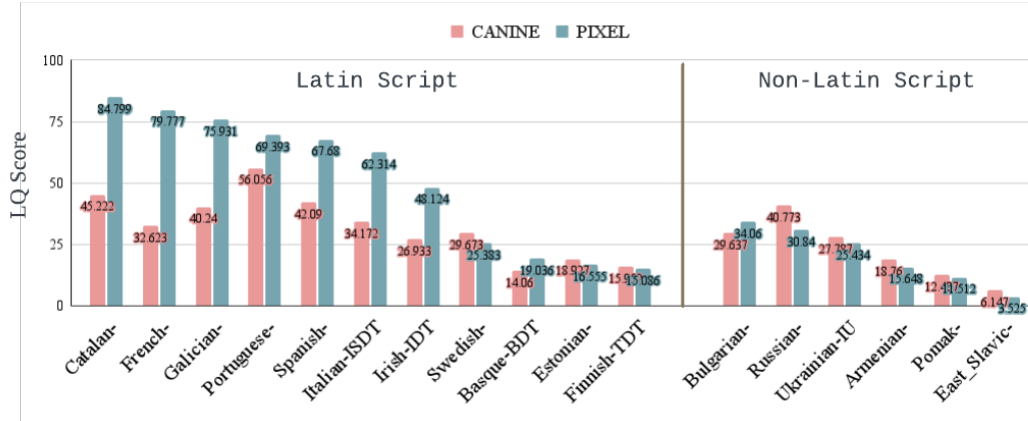
Figure 4: LQ score obtained by PIXEL and CANINE on Latin and non-Latin scripts on POS tagging. PIXEL outperforms CANINE on the POS tagging task when both source and target use the same script (on the left portion of the graph). Conversely, PIXEL does not outperform CANINE when the scripts are dissimilar (on the right portion of the graph)

	Arabic→X (POS Tagging)			
Lang. (X)	CANINE LQ Score, (Rank)	PIXEL LQ Score, (Rank)	Script Similarity	Linguistics Similarity
Maltese	5.9 (24)	1.5 (80)	Dissimilar	Very Close
Persian	15.7 (6)	42.8 (2)	Same	Dissimilar
Hebrew	43.1 (3)	36.9 (3)	Close	Related
Urdu	0.3 (74)	24.1 (6)	Same	Dissimilar

Table 2: LQ score and rank of PIXEL with Arabic as the source language shows PIXEL receives a high score when scripts are visually similar rather than when languages are only linguistically similar.

respective source languages. Persian and Urdu are both Indo-European languages and are not at all lexically similar to Arabic. However, these are both written using Arabic script. On the contrary, like Arabic, Maltese is an Afro-Asiatic language with Semitic origin. But PIXEL performed extremely poorly in the case of Maltese (ranked 81st). This, we suspect, is due to the use of Latin script in Maltese, which further emphasizes the effect of visual similarity for PIXEL.

In the case of mBERT and CANINE, these patterns of favoring similar-looking scripts were absent. Rather, we saw an average score for the languages irrespective of the script.

Case4 (African → African) We've compared all four models using 10 African languages from the MasakhaNER dataset for the Named Entity Recognition (NER) task. Aside from Amharic, which uses the Ge'ez script, all other languages use the Latin script. Figure 5 shows the average LQ score obtained by PIXEL and CANINE models for each lan-

guage as sources. The Table shows Amharic as an unfit choice for the source language when the target languages are in Latin script. Comparing PIXEL and CANINE, we notice CANINE outperforms PIXEL. Since PIXEL was only pre-trained on English, it is comparatively difficult for PIXEL to perform well on African languages. Conversely, CANINE was pre-trained on Yoruba (an African language) which has strong linguistic similarities with other African languages.

Observation Clearly, the above findings highlight the positive correlation between the performance of PIXEL, an image-based language model, and the visual similarity between languages. It is logical to expect that visually similar language would demonstrate better performance in cross-lingual transfer when utilizing PIXEL. The findings in the CANINE and mBERT comparison further reinforce the notion that language models that do not rely on visual representations do not exhibit a strong correlation between their scores and the visual similarity of the source and target languages.

4.2 Task Specific Performance

POS tagging In general, mBERT learns quickly compared to other models. This can be attributed to several reasons. First of all, mBERT operates on token-level representations and manifests heavy reliance on word-level semantics. So it is easier to associate the word or subword tokens with their respective POS tags, compared to character-level models like CANINE. Moreover, mBERT's predefined vocabulary, which includes commonly used subwords can potentially expedite the learning process

71

Figure 5: Average LQ scores with each language as sources for NER task (for PIXEL and CANINE) shows Amharic (only non-Latin script) pairs significantly worse with other languages that use Latin script

as the model can leverage semantic associations between these known tokens and their POS tags. On the contrary, character-level models have larger input sequence lengths and may require more examples to adequately learn the pattern in data which can lead to slower learning as compared to the tokenization-based models.

In addition, mBERT is trained on multilingual data. So it is more efficient than BERT at transferring knowledge from a high-resource language to a low-resource language, enhancing its few-shot learning capabilities for POS tagging tasks across different languages.

Dependency Parsing Interestingly, CANINE performs better than mBERT or BERT. This may be partly attributed to the nature of the task. Parsing is centered more on understanding the syntactic relationships between words in a sentence rather than on the meanings of individual words. As CANINE works on character level, it is more equipped to capture finer-grained patterns in these relationships, outperforming mBERT, exactly because the necessary information is marked with affixal morphemes in many languages. Moreover, CANINE operates without a predefined vocabulary, and its language independence might be advantageous when parsing sentences in a low-resource language or multilingual context. As a result, it can transfer knowledge across languages more fluidly. On top of that, the occurrence of out-of-vocabulary words or rare words can impact the parsing accuracy. As a character-level model, CANINE is better equipped in handling out-of-vocabulary words, which might be the reason for its improved performance in parsing in few-shot scenarios.

Coptic→X (POS tagging)			
Lang. (X)	**mBERT**	**CANINE**	**BERT**
Telegu	38.84	37.45	55.76
French	20.73	26.93	50.59
Italian	22.63	26.07	47.12
Russian	33.48	27.15	43.55
Persian Seraji	23.21	21.26	43.53

Table 3: Few-shot accuracy for POS tagging task with Coptic as the source language highlighting the performance of BERT (monolingually pre-trained) over mBERT and CANINE. Coptic is the only source language (in our analysis) that is not part of the pre-training languages of mBERT and CANINE and the only language where BERT significantly outperforms mBERT and CANINE

Named Entity Recognition NER, like POS tagging, leans heavily on understanding the meanings of individual words in order to accurately identify and classify named entities. This semantic nature of the task presents an advantage for segmentation-based models such as mBERT over character-level models like CANINE. Despite the multilingual strength of CANINE, its focus on character-level patterns may not sufficiently capture the semantic nuances needed for effective NER. Conversely, mBERT, with its token-based approach, can better handle the word meanings central to NER tasks. Therefore, in our analysis, mBERT demonstrates slightly superior performance in NER compared to CANINE. This suggests that while character-level models may excel in tasks centered on syntactic relationships, segmentation-based models may still hold the edge in tasks with a strong semantic dependency.

4.3 Unseen Languages

BERT performs better than mBERT and CANINE on some languages that these multilingual models
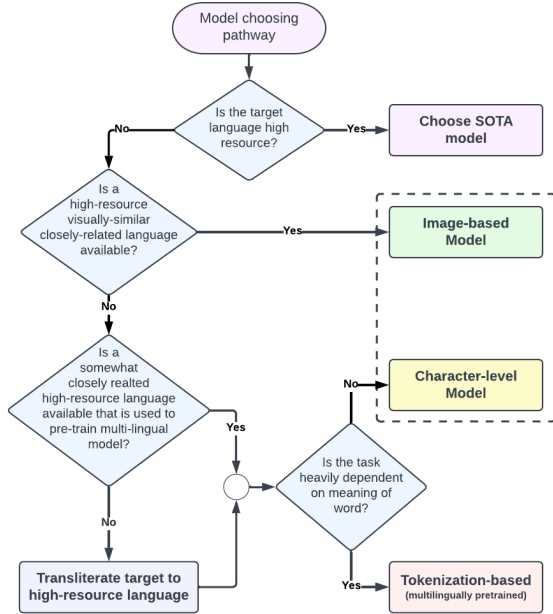
Figure 6: Model Recommendation Tree

were not pre-trained on. For example, consider the case study of Coptic. In comparison to CANINE and mBERT, BERT has better scores for POS tagging when Coptic is used as the source language (Table 3). Multilingual models like CANINE and mBERT underperform in this case. Among all the source languages used in our analysis, Coptic is the only source that is not part of the pre-training languages of CANINE and mBERT. It is also the only language where BERT has consistently outperformed the multi-lingually pre-trained models.

This inability to effectively adapt to a new unseen language could be attributed to the influence of the scripts of those languages. In these cases, transliterating the target to a high-resource language has been shown to improve performance on downstream tasks (Muller et al., 2021).

5 Model Recommendation Tree

Based on our findings, we propose a model selection pathway predicated on three primary considerations: resource availability for the target language, the presence of a visually similar high-resource language, and the task's semantic dependency.

High Resource Languages In the context of high-resource languages, we recommend employing the most advanced models. Our research indicates that both character-based models like CANINE and tokenization-based models like mBERT ex-

hibit superior performances in this setting. Generally, multilingual pre-training grants these models a notable edge over their monolingually trained counterparts, making them well-suited for tasks involving high-resource languages and ensuring efficient performance.

Visual Similarity In cases where the target language is resource-poor but visually resembles a high-resource language, our suggestion is to undertake a cross-lingual transfer from the high-resource language using a tokenization-free model like the PIXEL. PIXEL is explicitly designed to discern and capitalize on visual correspondences between languages, which makes it an optimal choice in instances where such resemblances can be exploited.

Semantic Dependency If a high-resource language somewhat closely related to the target language has been used in pre-training a multilingual model, the choice between different models should be guided by the task's semantic content requirements. If the task depends heavily on semantic understanding, models like mBERT or similar tokenization-based models are advisable. These models excel in scenarios where deep semantic comprehension is key. Conversely, if the task doesn't require a strong understanding of semantics, character-based models like CANINE may be a more efficient choice. These models typically perform well in scenarios where semantic dependence is lower.

Special Cases For scenarios that do not fall within the purview of the above-mentioned conditions, a multitude of factors come into play. For instance, when the source language was not part of the pre-training set for the multilingual model, we suggest transliterating the target language to a high-resource language. Transliterating those languages substantially enhances the performance of these multilingual models on downstream tasks.

6 Related Work

Cross-lingual transfer Cross-lingual transfer has emerged as a valuable approach to enhance model performance in low-resource languages without requiring extensive amounts of target language data (Conneau et al., 2020). XLM-R, proposed by Conneau et al., demonstrates the effectiveness of pre-training on a large-scale masked language model trained on 100 languages from CommonCrawl data. It outperforms multilingual

BERT (mBERT) on various cross-lingual benchmarks. Similarly, Devlin et al. and Xue et al. propose finetuning approaches for existing pre-trained language models (PLMs). Recently, another approach by Lee et al. employs adapters for cross-lingual transfer in low-resource languages. Fusing Multiple Adapters for Cross-Lingual Transfer (FAD-X) utilizes language adapters and task adapters to address the imbalance in lower-resource languages. MAD-X (Pfeiffer et al., 2020) is another adapter-based method that employs language, task, and invertible adapters. Moreover, this similar setting coupled with language phylogeny information proved to be useful for low-resource cross-lingual transfer (Faisal and Anastasopoulos, 2022).

Tokenization-free models Tokenization-based models such as BERT (Devlin et al., 2019b), RoBERTa (Liu et al., 2019), GPT-3 (Brown et al., 2020), ALBERT (Lan et al., 2020), T5 (Raffel et al., 2020) and ELECTRA (Clark et al., 2020b) are leading the field when it comes to performance across a broad range of natural language processing tasks. However, tokenization-based models like BERT demonstrate poor performance in unexplored domains (Boukkouri et al., 2020) and lack resilience to noisy data such as typos and missed clicks (Sun et al., 2020).

Studies have shown that models using visual text representations are more robust (Salesky et al., 2021). PIXEL (Rust et al., 2022) proposes the use of visual embeddings for language modeling, eliminating the need for a fixed vocabulary. Research suggests that models utilizing visual text representations exhibit greater resilience to noisy texts and enable rapid adaptation to new languages while maintaining performance.

CANINE (Clark et al., 2022), a character-based model, provides an alternative approach that eliminates the reliance on predefined vocabularies. CANINE surpasses vanilla BERT on the TyDiQA benchmark (Clark et al., 2020a) by downsampling input sequences to achieve similar speeds.

ByT5 (Xue et al., 2021a) introduces a modified version of the standard transformer that processes byte sequences, addressing the limitations of a finite vocabulary. Similarly, CHARFORMER (Tay et al., 2021) proposes a gradient-based sub-word tokenization method that operates directly on a byte level. It performs on par with tokenizer-based approaches and outperforms most byte-level methods.

Language Similarity Metrics Several researchers have proposed different methodologies to quantify similarity among languages. For instance, (Petroni and Serva, 2010) introduced a measure of lexical distance, which quantifies the difference between languages based on their vocabulary. On the other hand, (Chiswick and Miller, 2005) suggests a metric of linguistic distance that represents how challenging it is for English speakers to learn other languages. However, this method relies on English speakers' learning difficulty, making it language-biased and not generalizable for speakers of other languages.

A different approach is presented by Ciobanu and Dinu, who propose an automated method for identifying pairs of cognates (words with a common etymology) across languages. But this cognate identification method requires a known list of cognates, limiting its usefulness for less-studied languages, and it may overlook non-lexical aspects of language similarity.

Another common tool is the Automated Similarity Judgment Program (Automated Similarity Judgment Program, 2023) which uses a comprehensive database of vocabulary to analyze linguistic relationships but has been criticized for its simplified standard orthography and its reliance on a limited vocabulary list.

7 Conclusion

This study provides pivotal insights into the practical application of tokenization-based as well as tokenization-free models in cross-lingual transfer tasks, accentuating the importance of context and task-based model selection. However, there's an abundance of uncharted territory awaiting exploration. The gaps in our understanding of tokenization-free models such as PIXEL and CANINE present a significant opportunity for further research. These models, though promising, are still in their early stages of development. This paves the way for studies aiming to enhance their performance, potentially through the integration of advanced learning algorithms or novel feature extraction techniques.

Additionally, investigating the role of tokenization in handling different language families could provide profound insights. For instance, how do these models perform with agglutinative languages like Turkish or Finnish, or with logographic languages like Chinese? Exploring such linguistic

diversity could further clarify the strengths and weaknesses of different model types. An iterative inclusion of extinct or less commonly spoken languages is also essential at this point.

In summary, this study marks a significant step in understanding the capabilities and limitations of different models in cross-lingual transfer tasks. It opens several doors for future research, promising an exciting trajectory for the evolution of language modeling and translation tasks. The journey ahead, albeit challenging, presents a wealth of opportunities for innovation and discovery.

Limitations

This research, while extensive, presents certain limitations. Our study focuses primarily on syntactic tasks, leaving semantic tasks unexplored. While our work delves into the performance of specific models like BERT, mBERT, PIXEL, and CANINE, other models, especially emerging ones like decoder-based language models, remain unexamined in this context. The research also predominantly concerns low-resource languages, potentially limiting the applicability of our findings to high-resource contexts. Moreover, the consideration of different language families, such as agglutinative or logographic languages, is lacking in this analysis. Looking ahead, we plan to address these limitations by incorporating a broader range of language tasks, investigating a wider array of language models, and expanding our research to include high-resource languages and different language families. This will allow us to present a more holistic understanding of cross-lingual transfer in future studies.

Acknowledgements

References

David Ifeoluwa Adelani, Jade Abbott, Graham Neubig, Daniel D'souza, Julia Kreutzer, Constantine Lignos, Chester Palen-Michel, Happy Buzaaba, Shruti Rijhwani, Sebastian Ruder, Stephen Mayhew, Israel Abebe Azime, Shamsuddeen H. Muhammad, Chris Chinenye Emezue, Joyce Nakatumba-Nabende, Perez Ogayo, Aremu Anuoluwapo, Catherine Gitau, Derguene Mbaye, Jesujoba Alabi, Seid Muhie Yimam, Tajuddeen Rabiu Gwadabe, Ignatius Ezeani, Rubungo Andre Niyongabo, Jonathan Mukiibi, Verrah Otiende, Iroro Orife, Davis David, Samba Ngom, Tosin Adewumi, Paul Rayson, Mofetoluwa Adeyemi, Gerald Muriuki, Emmanuel Anebi, Chiamaka Chukwuneke, Nkiruka Odu, Eric Peter Wairagala, Samuel Oyerinde, Clemencia Siro, Tobius Saul Bateesa, Temilola Oloyede, Yvonne Wambui, Victor Akinode, Deborah Nabagereka, Maurice Katusiime, Ayodele Awokoya, Mouhamadane MBOUP, Dibora Gebreyohannes, Henok Tilaye, Kelechi Nwaike, Degaga Wolde, Abdoulaye Faye, Blessing Sibanda, Orevaoghene Ahia, Bonaventure F. P. Dossou, Kelechi Ogueji, Thierno Ibrahima DIOP, Abdoulaye Diallo, Adewale Akinfaderin, Tendai Marengereke, and Salomey Osei. 2021. MasakhaNER: Named entity recognition for African languages. *Transactions of the Association for Computational Linguistics*, 9:1116–1131.

Wasi Uddin Ahmad, Zhisong Zhang, Xuezhe Ma, Eduard Hovy, Kai-Wei Chang, and Nanyun Peng. 2019. On difficulties of cross-lingual transfer with order differences: A case study on dependency parsing. In *Proceedings of NAACL-HLT*, pages 2440–2452.

Automated Similarity Judgment Program. 2023. Automated similarity judgment program — Wikipedia, the free encyclopedia. [Online; accessed 18-June-2023].

Rajesh Bhatt. 2005. Long distance agreement in hindi-urdu. *Natural Language & Linguistic Theory*, 23(4):757–807.

Damian Blasi, Antonios Anastasopoulos, and Graham Neubig. 2022. Systematic inequalities in language technology performance across the world's languages. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5486–5505, Dublin, Ireland. Association for Computational Linguistics.

Hicham El Boukkouri, Olivier Ferret, Thomas Lavergne, Hiroshi Noji, Pierre Zweigenbaum, and Junichi Tsujii. 2020. Characterbert: Reconciling elmo and bert for word-level open-vocabulary representations from characters.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners.

Danqi Chen and Christopher D Manning. 2014. A fast and accurate dependency parser using neural networks. In *Proceedings of the 2014 conference on*

empirical methods in natural language processing (EMNLP), pages 740–750.

Barry R. Chiswick and Paul W. Miller. 2005. Linguistic distance: A quantitative measure of the distance between english and other languages. *Journal of Multilingual and Multicultural Development*, 26(1):1–11.

Alina Maria Ciobanu and Liviu P. Dinu. 2014. Automatic detection of cognates using orthographic alignment. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 99–105, Baltimore, Maryland. Association for Computational Linguistics.

Jonathan H. Clark, Eunsol Choi, Michael Collins, Dan Garrette, Tom Kwiatkowski, Vitaly Nikolaev, and Jennimaria Palomaki. 2020a. Tydi qa: A benchmark for information-seeking question answering in typologically diverse languages.

Jonathan H. Clark, Dan Garrette, Iulia Turc, and John Wieting. 2022. Canine: Pre-training an efficient tokenization-free encoder for language representation. *Transactions of the Association for Computational Linguistics*, 10:73–91.

Kevin Clark, Minh-Thang Luong, Quoc V. Le, and Christopher D. Manning. 2020b. Electra: Pre-training text encoders as discriminators rather than generators.

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.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019a. 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.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019b. Bert: Pre-training of deep bidirectional transformers for language understanding.

Ethnologue. 2023. Ethnologue. [Online; accessed 18-June-2023].

Fahim Faisal and Antonios Anastasopoulos. 2022. Phylogeny-inspired adaptation of multilingual models to new languages. In *Proceedings of the 2nd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 12th*

International Joint Conference on Natural Language Processing (Volume 1: Long Papers), pages 434–452, Online only. Association for Computational Linguistics.

Yimin Fan, Yaobo Liang, Alexandre Muzio, Hany Hassan, Houqiang Li, Ming Zhou, and Nan Duan. 2021. Discovering representation sprachbund for multilingual pre-training.

Philip Gage. 1994. A new algorithm for data compression. *C Users J.*, 12(2):23–38.

Junxian He, Zhisong Zhang, Taylor Berg-Kirkpatrick, and Graham Neubig. 2019. Cross-lingual syntactic transfer through unsupervised adaptation of invertible projections. *arXiv preprint arXiv:1906.02656*.

Valentin Hofmann, Hinrich Schuetze, and Janet Pierrehumbert. 2022. An embarrassingly simple method to mitigate undesirable properties of pretrained language model tokenizers. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 385–393, Dublin, Ireland. Association for Computational Linguistics.

Julia Kreutzer, Isaac Caswell, Lisa Wang, Ahsan Wahab, Daan van Esch, Nasanbayar Ulzii-Orshikh, Allahsera Tapo, Nishant Subramani, Artem Sokolov, Claytone Sikasote, Monang Setyawan, Supheakmungkol Sarin, Sokhar Samb, Benoî t Sagot, Clara Rivera, Annette Rios, Isabel Papadimitriou, Salomey Osei, Pedro Ortiz Suarez, Iroro Orife, Kelechi Ogueji, Andre Niyongabo Rubungo, Toan Q. Nguyen, Mathias Müller, André Müller, Shamsuddeen Hassan Muhammad, Nanda Muhammad, Ayanda Mnyakeni, Jamshidbek Mirzakhalov, Tapiwanashe Matangira, Colin Leong, Nze Lawson, Sneha Kudugunta, Yacine Jernite, Mathias Jenny, Orhan Firat, Bonaventure F. P. Dossou, Sakhile Dlamini, Nisansa de Silva, Sakine Çabuk Ballı, Stella Biderman, Alessia Battisti, Ahmed Baruwa, Ankur Bapna, Pallavi Baljekar, Israel Abebe Azime, Ayodele Awokoya, Duygu Ataman, Orevaoghene Ahia, Oghenefego Ahia, Sweta Agrawal, and Mofetoluwa Adeyemi. 2022. Quality at a glance: An audit of web-crawled multilingual datasets. *Transactions of the Association for Computational Linguistics*, 10:50–72.

Guillaume Lample, Miguel Ballesteros, Sandeep Subramanian, Kazuya Kawakami, and Chris Dyer. 2016. Neural architectures for named entity recognition. *arXiv preprint arXiv:1603.01360*.

Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. 2020. Albert: A lite bert for self-supervised learning of language representations.

Jaeseong Lee, Seung-won Hwang, and Taesup Kim. 2022. FAD-X: Fusing adapters for cross-lingual transfer to low-resource languages. In *Proceedings of the 2nd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the*

12th International Joint Conference on Natural Language Processing (Volume 2: Short Papers), pages 57–64, Online only. Association for Computational Linguistics.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach.

Christopher D Manning. 2011. Part-of-speech tagging from 97% to 100%: is it time for some linguistics? In *International conference on intelligent text processing and computational linguistics*, pages 171–189. Springer.

Benjamin Muller, Antonios Anastasopoulos, Benoît Sagot, and Djamé Seddah. 2021. When being unseen from mBERT is just the beginning: Handling new languages with multilingual language models. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 448–462, Online. Association for Computational Linguistics.

Joakim Nivre, Marie-Catherine de Marneffe, Filip Ginter, Jan Hajič, Christopher D. Manning, Sampo Pyysalo, Sebastian Schuster, Francis Tyers, and Daniel Zeman. 2020. Universal Dependencies v2: An evergrowing multilingual treebank collection. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 4034–4043, Marseille, France. European Language Resources Association.

Joakim Nivre, Daniel Zeman, Filip Ginter, and Francis Tyers. 2017. Universal Dependencies. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Tutorial Abstracts*, Valencia, Spain. Association for Computational Linguistics.

Filippo Petroni and Maurizio Serva. 2010. Measures of lexical distance between languages. *Physica A: Statistical Mechanics and its Applications*, 389(11):2280–2283.

Jonas Pfeiffer, Ivan Vulic, Iryna Gurevych, and Sebastian Ruder. 2020. MAD-X: an adapter-based framework for multi-task cross-lingual transfer. *CoRR*, abs/2005.00052.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer.

Phillip Rust, Jonas F. Lotz, Emanuele Bugliarello, Elizabeth Salesky, Miryam de Lhoneux, and Desmond Elliott. 2022. Language modelling with pixels.

Elizabeth Salesky, David Etter, and Matt Post. 2021. Robust open-vocabulary translation from visual text representations.

Mike Schuster and Kaisuke Nakajima. 2012. Japanese and korean voice search. In *2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5149–5152.

Lichao Sun, Kazuma Hashimoto, Wenpeng Yin, Akari Asai, Jia Li, Philip Yu, and Caiming Xiong. 2020. Adv-bert: Bert is not robust on misspellings! generating nature adversarial samples on bert.

Yi Tay, Vinh Q. Tran, Sebastian Ruder, Jai Gupta, Hyung Won Chung, Dara Bahri, Zhen Qin, Simon Baumgartner, Cong Yu, and Donald Metzler. 2021. Charformer: Fast character transformers via gradient-based subword tokenization.

Linting Xue, Aditya Barua, Noah Constant, Rami Al-Rfou, Sharan Narang, Mihir Kale, Adam Roberts, and Colin Raffel. 2021a. Byt5: Towards a token-free future with pre-trained byte-to-byte models.

Linting Xue, Noah Constant, Adam Roberts, Mihir Kale, Rami Al-Rfou, Aditya Siddhant, Aditya Barua, and Colin Raffel. 2021b. mT5: A massively multilingual pre-trained text-to-text transformer. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 483–498, Online. Association for Computational Linguistics.

Xinyan Velocity Yu, Akari Asai, Trina Chatterjee, Junjie Hu, and Eunsol Choi. 2022. Beyond counting datasets: A survey of multilingual dataset construction and necessary resources.

+

A Appendix

A.1 Frequently Asked Questions

1. Q: What did the authors mean by 'few-shot' and 'zero-shot'?
   A: The term 'few-shot' is quite loosely used in this paper. Each model is at first fully trained on a source language and then evaluated on some target language. In the evaluation phase, the model is either (i) directly evaluated on the target language (termed as zero-shot), or (ii) fine-tuned for a few steps on the target language (termed as few-shot).

2. Q: How can LQ score be negative and what does it imply?
   A: The LQ score does not have strong bounds. So it can have negative scores. Since it is a relative metric rather than an absolute one, having a negative score does not create any issue. It implies that the model is performing worse for the source-target pair compared to other sources in the system.

3. Q: Can LQ metric be used to compare different models?

A: Yes, LQ metric can be used to compare different models if the same pair of source and target languages are considered.

A.2 LQ Score

Proof of Effectiveness of LQ Score Let $E_s^{(t_c)} = F$, $E_s^{(t_0)} = Z_0$, and $Z_A = \frac{1}{n}\sum_{i=1}^{n} E_i^{(t_0)}$. We can rewrite the LQ score as:

$$LQ(x,k) = \frac{(F - Z_A)(F + Z_0))}{Z_A + \epsilon} \qquad (2)$$

We assume that a score would effectively measure the cross-lingual transfer capabilities if it gets positively rewarded for a higher score after a few shots of training in comparison to other language pairs and in comparison to the state before few-shot training. That means the growth of F from Z_0 and the difference of F with Z_A should play a high impact on the score.

Simplifying the right-hand-side of Eqn 1, we get,

$$\frac{F^2 - FZ_A + FZ_0 - Z_AZ_0}{Z_A + \epsilon} \qquad (3)$$

$$= F\frac{F}{Z_A} - F + F\frac{Z_0}{Z_A} - Z_0 \qquad (4)$$

$$= F\left(\frac{F + Z_0}{Z_A}\right) - F\left(1 + \frac{Z_0}{F}\right) \qquad (5)$$

In equation 5, the term $(F + Z_0)/Z_A$ will be greater than 1 when either F is very large or Z_0 is significantly larger than Z_A. That means a strong positive score can be obtained when the few-shot score is very high or the leap from zero-shot to few-shot is high. The remaining term $F\left(1 + \frac{Z_0}{F}\right)$ ensures the stability of the score. So, if a model learns quickly and gains good accuracy/las in the early steps of training, the LQ score will give out a strong score. If a model achieves a good score in zero-shot learning, it also receives a good LQ score.

Limitations of LQ Score The score utilizes a normalizing term that averages the zero-shot scores across all source languages. So, for any pair of languages, x and k, the LQ score will not always be the same. It will vastly depend on the list of source languages used in the experimentation. So, the numeric value of the LQ score does not have a

direct meaning. However, for a given source, the relation between the target languages is indicative of how compatible the source and target are. On the flip side, for a target language, the relation between the source languages is also meaningful.

A.3 Hyper-parameters

A.3.1 Dependency Parsing

Full Fine-tuning (on source)
- Train batch size: 32
- Max Training Steps: 15000
- Early Stopping: Yes
- Learning Rate: 5e-5
- Maximum Sequence Length: 256
- Eval metric: LAS

Few-shot Fine-tuning (on targets)
- Train batch size: 32
- Max Training Steps: 10
- Learning Rate: 5e-5
- Maximum Sequence Length: 256
- Eval metric: LAS

A.3.2 POS Tagging

Full Fine-tuning (on source)
- Train batch size: 32
- Max Training Steps: 15000
- Early Stopping: Yes
- Learning Rate: 5e-5
- Maximum Sequence Length: 256
- Eval metric: Accuracy

Few-shot Fine-tuning (on targets)
- Train batch size: 32
- Max Training Steps: 10
- Learning Rate: 5e-5
- Maximum Sequence Length: 256
- Eval metric: Accuracy

A.3.3 Named Entity Recognition

Full Fine-tuning (on source)
- Train batch size: 32
- Max Training Steps: 15000
- Early Stopping: Yes
- Learning Rate: 5e-5
- Maximum Sequence Length: 256
- Eval metric: Accuracy

Few-shot Fine-tuning (on targets)
- Train batch size: 32
- Max Training Steps: 10
- Learning Rate: 5e-5
- Maximum Sequence Length: 256

- Eval metric: Accuracy

A.4 Source languages as target languages

Table 4 provides a comprehensive analysis of the PIXEL model's performance in terms of accuracy in the POS-tagging task, evaluated in both zero-shot and few-shot scenarios. Here, the set of source languages also serves as the target languages, creating a self-referential evaluation method. This unique approach further allows for a deeper understanding of the model's strengths and weaknesses when dealing with identical sources and target languages.

A.5 List of target languages

Tables 5, 6, and 7 give an elaborate list of languages and their scripts along with their respective families. The languages are spread across multiple scripts and multiple families.

A.6 Lexical Similarity

Lexical similarity is the percentage obtained by comparing standardized wordlists from two linguistic varieties and tallying words similar in form and meaning (Ethnologue, 2023). It ranges from 0 to 100, representing the vocabulary overlap between two languages. Values over 85% often suggest the speech variant may be a dialect of the compared language. The proportion of lexical similarity between two kinds of language is calculated by comparing standardized lists of words and tallying the forms that demonstrate similarity in both structure and meaning.

Table 8 gives the similarity scores between different European Language pairs (Ethnologue, 2023; Fan et al., 2021).

B Additional Materials

Target Language	English	Arabic	Korean	Vietnamese	Tamil	Chinese	Japanese	Coptic	Hindi	Average (Z_A)
English	**0.967**	0.238	0.297	0.284	0.255	0.149	0.297	0.289	0.219	0.33
Arabic	0.238	**0.958**	0.412	0.379	0.289	0.152	0.403	0.177	0.07	0.34
Korean	0.28	0.382	**0.944**	0.476	0.284	0.23	0.413	0.329	0.172	0.39
Vietnamese	0.286	0.341	0.47	**0.86**	0.3	0.234	0.458	0.321	0.233	0.39
Tamil	0.135	0.3	0.388	0.331	**0.817**	0.224	0.37	0.25	0.223	0.34
Chinese	0.336	0.32	0.428	0.412	0.3	**0.93**	0.525	0.3	0.274	0.43
Japanese	0.276	0.294	0.376	0.349	0.229	0.303	**0.973**	0.226	0.179	0.36
Coptic	0.103	0.144	0.189	0.188	0.154	0.056	0.162	**0.962**	0.093	0.23
Hindi	0.229	0.215	0.292	0.302	0.24	0.202	0.274	0.209	**0.964**	0.33

(a) Accuracy for POS task at zero-shot

	Arabic	Chinese	Coptic	English	Hindi	Japanese	Korean	Tamil	Vietnamese
Arabic	**0.958**	0.328	0.337	0.396	0.277	0.34	0.388	0.337	0.355
Chinese	0.371	**0.93**	0.339	0.366	0.395	0.531	0.414	0.328	0.391
Coptic	0.191	0.11	**0.962**	0.183	0.163	0.188	0.193	0.166	0.229
English	0.25	0.219	0.324	**0.968**	0.283	0.304	0.292	0.265	0.29
Hindi	0.311	0.288	0.331	0.319	**0.964**	0.264	0.261	0.257	0.349
Japanese	0.417	0.403	0.295	0.374	0.334	**0.973**	0.385	0.295	0.364
Korean	0.42	0.373	0.416	0.404	0.403	0.409	**0.943**	0.384	0.47
Tamil	0.328	0.303	0.298	0.33	0.298	0.302	0.39	**0.817**	0.337
Vietnamese	0.385	0.312	0.328	0.379	0.395	0.439	0.454	0.336	**0.859**

(b) Accuracy for POS task at few-shot

Table 4: Accuracy of PIXEL model (on POS-tagging task) of zero-shot evaluation and few-shot evaluation of 9 source languages on the same languages as targets

Language Name	Script	Language Family	Sub-family
Armenian-ArmTDP	Armenian	Indo-European	Armenian
Armenian-BSUT	Armenian	Indo-European	Armenian
Western_Armenian-ArmTDP	Armenian	Indo-European	Armenian
Latvian-LVTB	Latin	Indo-European	Baltic
Lithuanian-ALKSNIS	Latin	Indo-European	Baltic
Lithuanian-HSE	Latin	Indo-European	Baltic
Irish-IDT	Latin	Indo-European	Celtic
Scottish_Gaelic-ARCOSG	Latin	Indo-European	Celtic
Welsh-CCG	Latin	Indo-European	Celtic
Afrikaans-AfriBooms	Latin	Indo-European	Germanic
Danish-DDT	Latin	Indo-European	Germanic
Dutch-Alpino	Latin	Indo-European	Germanic
Dutch-LassySmall	Latin	Indo-European	Germanic
English-Atis	Latin	Indo-European	Germanic
English-ESL	Latin	Indo-European	Germanic
English-EWT	Latin	Indo-European	Germanic
English-GUM	Latin	Indo-European	Germanic
English-GUMReddit	Latin	Indo-European	Germanic
English-LinES	Latin	Indo-European	Germanic
English-ParTUT	Latin	Indo-European	Germanic
Faroese-FarPaHC	Latin	Indo-European	Germanic
German-GSD	Latin	Indo-European	Germanic
German-HDT	Latin	Indo-European	Germanic
Icelandic-IcePaHC	Latin	Indo-European	Germanic
Icelandic-Modern	Latin	Indo-European	Germanic
Norwegian-Bokmaal	Latin	Indo-European	Germanic
Norwegian-Nynorsk	Latin	Indo-European	Germanic
Norwegian-NynorskLIA	Latin	Indo-European	Germanic
Swedish-LinES	Latin	Indo-European	Germanic
Swedish-Talbanken	Latin	Indo-European	Germanic
Gothic-PROIEL	Gothic	Indo-European	Germanic
Turkish_German-SAGT	Latin	Indo-European	Germanic (German)
Ancient_Greek-Perseus	Greek	Indo-European	Hellenic
Ancient_Greek-PROIEL	Greek	Indo-European	Hellenic
Greek-GDT	Greek	Indo-European	Hellenic
Hindi_English-HIENCS	Devanagari and Latin	Indo-European	Indo-Aryan
Hindi-HDTB	Devanagari	Indo-European	Indo-Aryan
Marathi-UFAL	Devanagari	Indo-European	Indo-Aryan
Urdu-UDTB	Arabic	Indo-European	Indo-Aryan
Persian-PerDT	Arabic	Indo-European	Iranian
Persian-Seraji	Arabic	Indo-European	Iranian
Latin-ITTB	Latin	Indo-European	Italic
Latin-LLCT	Latin	Indo-European	Italic

Table 5: List of Target Languages (Part 1)

Language Name	Script	Language Family	Sub-family
Latin-PROIEL	Latin	Indo-European	Italic
Latin-UDante	Latin	Indo-European	Italic
Catalan-AnCora	Latin	Indo-European	Romance
French-FTB	Latin	Indo-European	Romance
French-GSD	Latin	Indo-European	Romance
French-ParTUT	Latin	Indo-European	Romance
French-Rhapsodie	Latin	Indo-European	Romance
French-Sequoia	Latin	Indo-European	Romance
Galician-CTG	Latin	Indo-European	Romance
Italian-ISDT	Latin	Indo-European	Romance
Italian-MarkIT	Latin	Indo-European	Romance
Italian-ParTUT	Latin	Indo-European	Romance
Italian-PoSTWITA	Latin	Indo-European	Romance
Italian-TWITTIRO	Latin	Indo-European	Romance
Italian-VIT	Latin	Indo-European	Romance
Old_French-SRCMF	Latin	Indo-European	Romance
Portuguese-Bosque	Latin	Indo-European	Romance
Portuguese-GSD	Latin	Indo-European	Romance
Romanian-Nonstandard	Latin	Indo-European	Romance
Romanian-RRT	Latin	Indo-European	Romance
Romanian-SiMoNERo	Latin	Indo-European	Romance
Spanish-AnCora	Latin	Indo-European	Romance
Spanish-GSD	Latin	Indo-European	Romance
Croatian-SET	Latin	Indo-European	Slavic
Czech-CAC	Latin	Indo-European	Slavic
Czech-CLTT	Latin	Indo-European	Slavic
Czech-FicTree	Latin	Indo-European	Slavic
Czech-PDT	Latin	Indo-European	Slavic
Polish-LFG	Latin	Indo-European	Slavic
Polish-PDB	Latin	Indo-European	Slavic
Slovak-SNK	Latin	Indo-European	Slavic
Slovenian-SSJ	Latin	Indo-European	Slavic
Old_Church_Slavonic-PROIEL	Glagolitic and Cyrillic	Indo-European	Slavic
Belarusian-HSE	Cyrillic	Indo-European	Slavic
Bulgarian-BTB	Cyrillic	Indo-European	Slavic
Old_East_Slavic-Birchbark	Cyrillic	Indo-European	Slavic
Old_East_Slavic-TOROT	Cyrillic	Indo-European	Slavic
Pomak-Philotis	Cyrillic	Indo-European	Slavic
Russian-GSD	Cyrillic	Indo-European	Slavic
Russian-SynTagRus	Cyrillic	Indo-European	Slavic
Russian-Taiga	Cyrillic	Indo-European	Slavic
Serbian-SET	Cyrillic	Indo-European	Slavic
Ukrainian-IU	Cyrillic	Indo-European	Slavic

Table 6: List of Target Languages (Part 2)

Language Name	Script	Language Family	Sub-family
Coptic-Scriptorium	Coptic	Afro-Asiatic	Egyptian
Maltese-MUDT	Latin	Afro-Asiatic	Semitic
Ancient_Hebrew-PTNK	Hebrew	Afro-Asiatic	Semitic
Hebrew-HTB	Hebrew	Afro-Asiatic	Semitic
Hebrew-IAHLTwiki	Hebrew	Afro-Asiatic	Semitic
Arabic-NYUAD	Arabic	Afro-Asiatic	Semitic
Arabic-PADT	Arabic	Afro-Asiatic	Semitic
Vietnamese-VTB	Latin	Austroasiatic	Vietic
Indonesian-GSD	Latin	Austronesian	Malayo-Polynesian
Tamil-TTB	Tamil	Dravidian	Tamil-Kannada
Telugu-MTG	Telugu	Dravidian	Telugu-Kui
Japanese-BCCWJ	Japanese (Kanji, Hiragana, Katakana)	Japonic	Japanese
Japanese-BCCWJLUW	Japanese (Kanji, Hiragana, Katakana)	Japonic	Japanese
Japanese-GSD	Japanese (Kanji, Hiragana, Katakana)	Japonic	Japanese
Japanese-GSDLUW	Japanese (Kanji, Hiragana, Katakana)	Japonic	Japanese
Korean-GSD	Hangul and Hanja	Koreanic	Korean
Korean-Kaist	Hangul and Hanja	Koreanic	Korean
Basque-BDT	Latin	Language Isolate	Language Isolate
Naija-NSC	Latin	Niger-Congo	Benue-Congo
Wolof-WTB	Latin	Niger-Congo	Senegambian
Swedish_Sign_Language	Swedish Sign Language (SignWriting)	Sign Language	Sign Language
Chinese-GSDSimp	Simplified Chinese (Han script)	Sino-Tibetan	Sinitic
Classical_Chinese-Kyoto	Classical Chinese (Han script)	Sino-Tibetan	Sinitic
Chinese-GSD	Chinese (Han script)	Sino-Tibetan	Sinitic
Uyghur-UDT	Arabic	Turkic	Karluk
Turkish-Atis	Latin	Turkic	Oghuz
Turkish-BOUN	Latin	Turkic	Oghuz
Turkish-FrameNet	Latin	Turkic	Oghuz
Turkish-IMST	Latin	Turkic	Oghuz
Turkish-Kenet	Latin	Turkic	Oghuz
Turkish-Penn	Latin	Turkic	Oghuz
Turkish-Tourism	Latin	Turkic	Oghuz
Estonian-EDT	Latin	Uralic	Finnic
Estonian-EWT	Latin	Uralic	Finnic
Finnish-FTB	Latin	Uralic	Finnic
Finnish-TDT	Latin	Uralic	Finnic
Hungarian-Szeged	Latin	Uralic	Ugric

Table 7: List of Target Languages (Part 3)

	Catalan	English	French	German	Italian	Portuguese	Romanian	Russian	Spanish
Catalan	1	-	0.85	-	0.87	0.85	0.73	-	0.85
English	-	1	0.27	0.6	-	-	-	0.24	-
French	0.85	0.27	1	0.29	0.89	0.75	0.75	-	0.75
German	-	0.6	0.29	1	-	-	-	-	-
Italian	0.87	-	0.89	-	1	0.8	0.77	-	0.82
Portuguese	0.85	-	0.75	-	0.8	1	0.72	-	0.89
Romanian	0.73	-	0.75	-	0.77	0.72	1	-	0.71
Russian	-	0.24	-	-	-	-	-	1	-
Spanish	0.85	-	0.75	-	0.82	0.89	0.71	-	1

Table 8: Lexical similarity among European languages (Ethnologue, 2023; Fan et al., 2021)

	mBERT	CANINE	BERT
UD_Telugu-MTG	38.83	37.45	55.76
UD_French-ParTUT	20.37	26.93	50.52
UD_Italian-ParTUT	22.63	26.07	47.12
UD_French-Sequoia	22.57	27.72	46.64
UD_Spanish-AnCora	24.10	24.17	46.09
UD_French-GSD	22.94	28.09	46.03
UD_Galician-CTG	27.80	22.67	45.95
UD_Italian-ISDT	23.07	26.80	45.62
UD_Italian-VIT	24.43	27.54	44.61
UD_Spanish-GSD	22.55	23.2	43.80
UD_Russian-GSD	33.48	27.15	43.54
UD_Persian-Seraji	23.21	21.26	43.54
UD_Catalan-AnCora	22.42	23.93	43.41
UD_Turkish-Kenet	32.31	32.29	43.21
UD_Portuguese-Bosque	26.99	22.92	42.51
UD_Portuguese-GSD	26.36	22.36	41.95
UD_Italian-MarkIT	21.57	26.19	41.78
UD_Turkish-FrameNet	33.33	32.45	41.38
UD_Turkish-Penn	29.87	30.68	41.25
UD_French-Rhapsodie	27.63	32.16	40.88
UD_Hebrew-IAHLTwiki	26.53	19.43	40.13
UD_Russian-SynTagRus	33.16	27.29	40.09
UD_Polish-PDB	30.01	25.15	39.90
UD_Lithuanian-ALKSNIS	34.08	25.40	39.78
UD_Arabic-PADT	30.52	19.67	39.62
UD_Belarusian-HSE	30.87	23.30	38.41
UD_Polish-LFG	30.18	29.38	38.24
UD_Ukrainian-IU	30.56		37.60
UD_Hebrew-HTB	23.88	17.32	37.58
UD_Vietnamese-VTB	21.60	25.97	37.52
UD_Turkish-BOUN	30.42	25.66	37.35
UD_Greek-GDT	25.18	15.39	37.26
UD_Latvian-LVTB	32.35	24.42	37.24
UD_Romanian-SiMoNERo	34.12	21.87	37.23

Table 9: LQ scores of different models (using Coptic as source language)