Learning Beyond Limits: Multitask Learning and Synthetic Data for Low-Resource Canonical Morpheme Segmentation

Anonymous ACL submission

Abstract

We introduce a transformer-based morpheme segmentation system that augments a low-resource training signal through multitask learning and LLM-generated synthetic data. Our framework jointly predicts morphological segments and glosses from orthographic input, leveraging shared linguistic representations obtained through a common documentary process to enhance model generalization. To further address data scarcity, we integrate synthetic training data generated by large language models (LLMs) using in-context learning. Experimental results on the SIGMORPHON 2023 dataset show that our approach significantly improves word-level segmentation accuracy and morpheme-level F1-score across multiple low-resource languages.

1 Introduction

Morphological segmentation—the process of breaking words into their smallest meaningful units—is a fundamental task in linguistic analysis. This process has two goals: first, to identify morpheme boundaries, and second, to restore phonological changes between canonical and surface forms. For example, the word *happiness* is composed of two surface morphemes: *happi* + *-ness*. Underlyingly, the root *happy* undergoes an orthographic modification when it combines with *-ness*. Canonical segmentation produces the normalized *happy-ness*.

Canonical segmentation is particularly critical for analyzing low-resource and morphologically-complex languages. Linguistic documentation relies on language experts creating Interlinear Glossed Texts (IGT). An IGT entry consists of four tiers: 1. orthographic text, the original sentence; 2. morpheme segmentation, decomposing words into canonical morphemes; 3. glossing, assigning linguistic labels to each morpheme; and 4. translation, providing an equivalent sentence in a high-resource matrix language like English. An example from Gitksan follows.

Orthography:	Ii hahla'lsdi'y goohl IBM
Segmentation:	ii hahla'lst-'y goo-hl IBM
Gloss:	CCNJ work-1SG.II LOC-CN IBM
Translation:	And I worked for IBM.

The construction of IGTs is a process that requires significant linguistic expertise. For languages with few speakers, the segmentation step alone can be a complex and time-consuming task. Previous research has begun to automate this process using neural models (Kann et al., 2016; Ruzsics and Samardžić, 2017; Wang et al., 2019; Rice et al., 2024), but performance remains limited by scarce annotated training data. Most approaches focus exclusively on segmenting the orthographic tier (Kann et al., 2016; Ruzsics and Samardžić, 2017; Wang et al., 2019). Rice et al. (2024), however explore augmenting the segmentation signal with an additional encoder tied to the translation tier. This method depends on manual word alignment between source and translated text, and does not ease the need for linguistic expertise. We instead propose two methods for leveraging existing signals to improve canonical segmentation in low-resource language documentation:

Multitask learning Multitask learning encourages generalization across complementary objectives (Caruana, 1997), and can enhance robustness in low-resource scenarios (Lin et al., 2018; Johnson et al., 2017). In our framework, the model is trained to jointly predict the segmentation and glossing tiers of an IGT, with only the orthographic tier as input. Incorporating glossing as a parallel objective in multitask learning can exploit beneficial information without necessitating further data curation, as glossing is already a component of IGT. By learning these related tasks simultaneously, the model gains access to rich linguistic information —morpheme boundaries from the segmentation tier,

and labels from the glossing tier.

LLM synthetic data The scarcity of annotated datasets for low-resource languages often causes neural models to overfit frequent character sequences rather than generalizing to true morphological structures, a phenomenon known as label bias (Wiseman and Rush, 2016). To address this, we supplement the training data with synthetic examples created by large language models (LLMs) with in-context learning. Since canonical segmentation involves resolving phonological alternations (e.g., mapping hahla'lsdi to -hahla'lst-), LLMs excel at this task by learning and replicating these alternations directly from interlinear glossed text (IGT) examples—without requiring explicit rule encoding. By systematically varying the proportion of synthetic data, we assess its role in mitigating data scarcity while maintaining segmentation consistency.

Our contributions are as follows:

- We introduce a multitask learning framework that jointly learns to segment and gloss, improving segmentation performance across multiple low-resource languages.
- We synthesize data to augment sparse training data for segmentation and evaluate its effectiveness at different saturation levels.
- We combine the two strategies, demonstrating that multitask learning and synthetic data complement each other to enhance segmentation quality.

2 Experiment Setup and Methodology

Following the work of (Rice et al., 2024), we conduct experiments in the languages of the SIGMORPHON 2023 Shared Task dataset (Ginn et al., 2023)[1]. The TAMS system proposed by Rice et al. (2024) requires a manual alignment between source and matrix language, and therefore, linguistic expertise, limiting their results to a subset of the dataset's languages (Arapaho, Lezgi, and Tsez), for which we use the same data splits. We expand our experiments to the remaining languages in the data, including Gitksan, Natügu, Nyangbo, and Uspanteko. Data is split by identifying all unique words in each language dataset, and re-split using the same 6:2:2 split in the TAMS paper[2]. Specifics for each language are in Table 1.

[1]The dataset is licensed under CC BY-NC 4.0.

[2]Our splits will be made available after publication.

Language	Train	Dev	Test	Matrix lang.
Arapaho (arp)	16666	10760	9849	(eng)
Gitksan (git)	323	107	109	(eng)
Lezgi (lez)	1236	412	412	(eng)
Natügu (ntu)	1953	651	652	(eng)
Tsez (ddo)	3,558	445	445	(eng)
Uspanteko (usp)	7033	2345	2344	(spa)
Nyangbo (nyb)	1499	499	501	-

Table 1: 2023 SIGMORPHON Shared Task Dataset (Ginn et al., 2023)

2.1 Multitask Model for Canonical Segmentation

We treat canonical segmentation as a sequence-to-sequence task and conduct our experiments with a modified version of Fairseq's (Ott et al., 2019) implementation of transformers (Vaswani et al., 2017). We modify the transformer architecture with a multitask objective [3] . Our model consists of a shared encoder that processes the input word from the orthographic tier, generating a latent representation. This representation then serves as input to a pair of decoders: the first learns to produce a canonical segmentation and the other generates the corresponding gloss[4]. We define a joint loss function as the weighted sum of segmentation loss and glossing loss:

$$\mathcal{L}_{total} = \lambda \mathcal{L}_{seg} + (1 - \lambda)\mathcal{L}_{gloss} \quad (1)$$

where the segmentation loss weight λ is tuned within the range of 0.8 to 1[5], while the weight of the glossing objective is complemetary, ensuring that the model prioritizes segmentation accuracy while still leveraging glossing information as auxiliary supervision. Hyper-parameters and model details are in the Appendix A.1.

2.2 Generating Synthetic Examples

To address data scarcity, we generate synthetic segmentation data using GPT-4o with in-context learning to supplement the limited training data.

First, we extract all words from the training data which have a disjunction between their underlying

[3]Our code is available: `https://link/to/our/repo`. Our implementation is modified based on Zhou et al. (2019)'s work: `https://github.com/shuyanzhou/multi-task_transformer`.

[4]If the word gloss is "work-1SG.II", the gloss decoder will generate it as "w-o-r-k-1SG.II"

[5]Appendix A.2 illustrates the impact of λ on Lezgi model performance.

and surface morphemes. These forms will serve as in-context examples for the LLM.

Next, we construct a structured prompt that includes: 1. A word stem and its meaning. 2. Example words from the training data that share this stem, along with their canonical segmentations and glosses. 3. A list of grammatical morphemes[6] and their corresponding glosses, extracted directly from the training data.

The LLM then generates new words by combining the stem with grammatical morphemes, applying morphophonological alternations based on the examples provided.The resulting triples—surface form, canonical segmentation, and gloss—approximate IGT text and expand the model's morphological coverage. An example prompt for Natügu is in Appendix A.4.

3 Results and Findings

We now discuss the findings of our experiments. Following TAMS (Rice et al., 2024), we evaluate across 3 metrics: word-level accuracy, morpheme-level F1, and the sum of edit-distances across all test instances. We evaluate against reported results from the TAMS paper, as well as a Fairseq baseline with a single decoder devoted to segmentation.

3.1 Multitask Learning Performance

Model	Metric	lez	ddo	arp	git	ntu	nyb	usp	ave
Baseline	ACC↑	44.66	**82.6**	67.08	47.71	63.04	**80.48**	55.05	62.95
	F1↑	60.75	90.44	81.11	65.5	80.3	90.24	75.66	77.71
	ED↓	568	652	10495	117	458	154	1799	2034.71
TAMS	ACC↑	46.84	80.78	67.72	–	–	–	–	–
	F1↑	62.48	89.52	81.62	–	–	–	–	–
	ED↓	532	701	9899	–	–	–	–	–
TAMS-CLS	ACC↑	47.09	81.96	67.4	–	–	–	–	–
	F1↑	62.48	90.08	81.45	–	–	–	–	–
	ED↓	537	**643**	9970	–	–	–	–	–
Multitask	ACC↑	**48.54**	82.51	**78.01**	**52.29**	**68.87**	79.84	**56.12**	**66.59**
	F1↑	**68.84**	**92.12**	**84.14**	**71.64**	**84.09**	**91.43**	**77.18**	**81.35**
	ED↓	**519**	698	**6543**	**112**	**373**	**149**	**1623**	**1431**

Table 2: Comparison of canonical segmentation models across multiple languages. Each model includes three sub-rows for ACC, F1, and ED, with the last column showing average metrics. **Bolded** values indicate language bests for each metric. ↓ indicates that lower is better.

Table 2 demonstrates that the multitask model achieves superior overall performance. Most languages see improvements over the best alternative. Furthermore, attaching a multitask objective improves over the single-task objective for each metric, on average. Languages which already have higher performance, such as Nyangbo and Tsez, still see improvements at the morpheme level, although Nyangbo demonstrates that improvements in F1 are not always accompanied by a similar improvement in accuracy. It is possible that the benefits of multitask learning may be more significant at the morpheme level than at the word level.

Training data size seems to have little impact on the benefits of multitask learning. Languages such as Arapaho, with significantly more data than the sparsest languages, observes large improvements, while Gitksan and Natügu, which have much less training data, also improve when a multitask objective is introduced.

A qualitative analysis suggests that multitask learning improves the overall accuracy of morpheme segmentation by reducing unnecessary modifications. That is, the baseline model is too aggressive in employing textual normalization, making changes where they are not appropriate. In languages with numerous morphological alternations, such as Arapaho and Lezgi, multitask learning significantly reduces edit distances by removing alternations that the baseline deems necessary. In contrast, in languages with already high segmentation accuracy, such as Tsez, decreases in edit distance are less pronounced - the glossing information may not add much extra signal.

Overall, these findings indicate that integrating glossing information as an extra predictive task improves model quality, without the need for extra annotation. The improvements are particularly noticeable in languages with complex segmentation patterns, demonstrating the effectiveness of this approach in improving canonical segmentation in low-resource settings.

3.2 Learning Curve of multitask Learning

After observing in our previous experiments that data size had less of an impact than linguistic constraints, we conducted experiments aimed at further investigating the role that data size plays on multitask learning. For each language, we create artificially small training sets by limiting the data to 25, 50, 75, and 100% of the original training set. The comparison of the average learning curves is presented in Figure 1.[7]

We observe that in general, the improvements

[6] Grammatical morphemes are functional elements in language that indicate grammatical relationships such as tense, number, case, or person, rather than carrying lexical meaning, as seen in markers like 1SG.II (first-person singular) and LOC (locative) in the IGT example.

[7] For individual language curves, please see Appendix A.3.

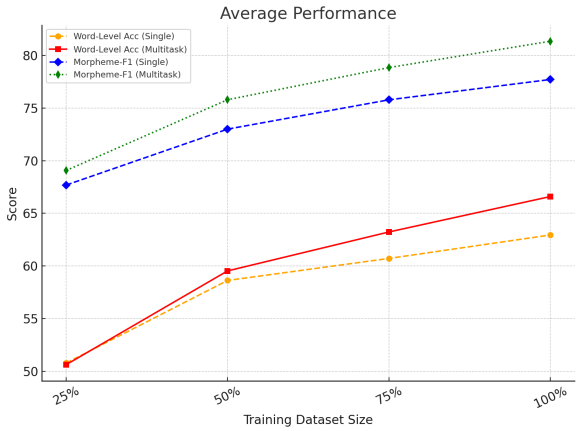

Figure 1: The average learning curves for the F1 (top) and Accuracy (bottom) metrics.

Model	Metric	lez	ddo	arp	git	ntu	nyb	usp	ave
M	ACC↑	48.54	**82.51**	78.01	52.29	68.87	79.84	56.12	66.59
	F1↑	68.84	**92.12**	84.14	71.64	84.09	**91.43**	77.18	81.35
	ED↓	519	698	6543	112	373	149	1623	1431
M+ LLM (0.25)	ACC↑	49.27	80.41	78.14	52.29	**69.02**	80.21	57.10	66.63
	F1↑	**69.6**	91.03	84.49	72.78	**84.47**	91.30	77.86	81.65
	ED↓	500	779	6632	118	350	136	1538	1436.14
M + LLM (0.5)	ACC↑	**49.51**	81.64	78.41	52.29	67.02	80.84	56.89	66.66
	F1↑	67.44	91.87	84.91	70.84	82.84	90.45	76.97	80.76
	ED↓	529	**687**	6483	117	367	164	1557	1414.86
M+ LLM (0.75)	ACC↑	48.82	81.32	**79.5**	**56.88**	68.71	**81.24**	**58.29**	**67.82**
	F1↑	67.69	91.51	**85.65**	**74.32**	84.18	91.34	**79.05**	**81.96**
	ED↓	**491**	723	**6502**	**96**	**333**	**127**	**1507**	**1397**

Table 3: Comparison of segmentation models across languages. Each model includes three sub-rows for ACC, F1, and ED, with the last column showing average metrics. M denotes multitask learning, with synthetic data added at 25%, 50%, and 75% of training size.

obtained from multitask learning increase as more training data is available, although there is still an observed benefit in extremely low-data settings. This is promising, as it suggests that improvements obtained in aiding the documentary process at the beginning will eventually feed a virtuous cycle, with increasing gains as further data is created.

3.3 Addressing Data Scarcity with LLM-Generated Data

After observing that the model benefits from extra training data, we seek to augment the training data with synthetic examples. In our final experiment, we supplement our multitask model with training examples generated by an LLM. We control the percentage of added synthetic examples - increasing in increments of 25% of the gold training data. We report the results in Table 5.

We observe continued, if modest, improvements when supplementing multitask learning with synthetic data. Some languages, like Gitksan, only start to improve when the percentage of synthetic examples approaches the number of natural ones. Other languages, like Arapaho, which already contains much larger data stores, see regular improvements as more data is added. There do seem to be some limitations to the idea that more data is always better, however; Lezgi sees an improvement only with moderate levels of extra data, and high-performing languages like Tsez and Nyangbo are difficult to improveme any further. On average, we see similar trends to multitask learning on its own - with most of the benefit coming at the morpheme level.

LLM-generated synthetic data can be highly beneficial in addressing the data scarcity problem for canonical segmentation. By providing diverse and linguistically plausible training examples, LLMs help compensate for the lack of annotated data while preserving the structural integrity of morphological patterns. The improvements observed in both accuracy and consistency demonstrate the value of incorporating LLMs into segmentation models, particularly for languages with limited annotated resources. We have constrained our presented experiments to the multitask setting, but an ablation study on the single-task objective (Appendix A.5) demonstrates similar trends.

4 Conclusions

In this work, we have demonstrated that low-resource canonical morpheme segmentation is improved through the use of multitask learning and synthetic data. Using glossing as an auxiliary task and LLMs to strengthen the training signal, we provide a new benchmark for canonical morpheme segmentation in low-resource languages, aiding in the development of effective computational tools for linguistic documentation and preservation. Future research should refine data augmentation techniques, explore active learning strategies, and investigate multilingual training frameworks to improve cross-linguistic generalization, while also working with documentary linguists to evaluate the value of automation in the field.

5 Limitations

Despite the improvements demonstrated in our experiments, our approach has several limitations that should be addressed in future research. One key limitation is our reliance on synthetic data generated by large language models (LLMs). While we observe performance gains when augmenting training with synthetic examples, the quality and lin-

guistic validity of these examples remain uncertain. LLMs may introduce hallucinations, generating segmentation patterns that do not fully align with the true morphological structure of the target language. Since our study does not include a detailed qualitative error analysis, it is difficult to determine whether the improvements stem from genuinely better morphological generalization or simply from increased exposure to frequent patterns. A more thorough investigation of the impact of synthetic data on segmentation quality, particularly in low-resource settings, is necessary.

One potential risk of LLM-generated synthetic data lies in the misuse of these data for deceptive or unethical purposes. Since we propose using LLMs to generate structured linguistic data, this technique could be exploited to fabricate linguistic evidence in historical or sociolinguistic studies. In particular, if synthetic morphological data is presented as authentic, it could be used to falsely attribute linguistic features to certain languages or communities, potentially leading to misrepresentation or erasure of genuine linguistic diversity.

A second limitation is that because our synthetic data generation process relies on patterns observed in the training set, it is inherently limited to existing vocabulary. The LLM-generated data cannot create new stems or morphological categories that have not appeared in the training data, restricting its ability to model truly novel linguistic forms. This limitation means that the model may still struggle with out-of-vocabulary (OOV) words or rare morphological constructions that were not adequately represented in the original dataset. Future research could explore alternative methods, such as leveraging morphological rule induction or few-shot learning with human-in-the-loop guidance, to generate more diverse and linguistically valid synthetic data that extends beyond what has been seen in the training set.

6 Ethical Concerns

As with any work involving language data, but particularly data from underserved and historically marginalized communities, steps should be taken that language corpora are collected and stewarded with respect and the support of the communities. These data represent the linguistic and cultural heritage of communities of people, and we thank the people of these communities for allowing us to work with their languages.

References

Rich Caruana. 1997. Multitask learning. *Machine learning*, 28:41–75.

Michael Ginn, Sarah Moeller, Alexis Palmer, Anna Stacey, Garrett Nicolai, Mans Hulden, and Miikka Silfverberg. 2023. Findings of the SIGMORPHON 2023 shared task on interlinear glossing. In *Proceedings of the 20th SIGMORPHON workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 186–201.

Melvin Johnson, Mike Schuster, Quoc V. Le, Maxim Krikun, Yonghui Wu, Zhifeng Chen, Nikhil Thorat, Fernanda Viégas, Martin Wattenberg, Greg Corrado, Macduff Hughes, and Jeffrey Dean. 2017. Google's multilingual neural machine translation system: Enabling zero-shot translation. *Transactions of the Association for Computational Linguistics*, 5:339–351.

Katharina Kann, Ryan Cotterell, and Hinrich Schütze. 2016. Neural morphological analysis: Encoding-decoding canonical segments. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 961–967, Austin, Texas. Association for Computational Linguistics.

Ying Lin, Shengqi Yang, Veselin Stoyanov, and Heng Ji. 2018. A multi-lingual multi-task architecture for low-resource sequence labeling. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 799–809, Melbourne, Australia. Association for Computational Linguistics.

Myle Ott, Sergey Edunov, Alexei Baevski, Angela Fan, Sam Gross, Nathan Ng, David Grangier, and Michael Auli. 2019. fairseq: A fast, extensible toolkit for sequence modeling. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics (Demonstrations)*, pages 48–53, Minneapolis, Minnesota. Association for Computational Linguistics.

Enora Rice, Ali Marashian, Luke Gessler, Alexis Palmer, and Katharina von der Wense. 2024. TAMS: Translation-assisted morphological segmentation. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 6752–6765, Bangkok, Thailand. Association for Computational Linguistics.

Tatyana Ruzsics and Tanja Samardžić. 2017. Neural sequence-to-sequence learning of internal word structure. In *Proceedings of the 21st Conference on Computational Natural Language Learning (CoNLL 2017)*, pages 184–194, Vancouver, Canada. Association for Computational Linguistics.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems*, 30.

Weihua Wang, Rashel Fam, Feilong Bao, Yves Lepage, and Guanglai Gao. 2019. Neural morphological segmentation model for mongolian. In *2019 International Joint Conference on Neural Networks (IJCNN)*, pages 1–7. IEEE.

Sam Wiseman and Alexander M Rush. 2016. Sequence-to-sequence learning as beam-search optimization. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1296–1306.

Shijie Wu, Ryan Cotterell, and Mans Hulden. 2021. Applying the transformer to character-level transduction. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 1901–1907, Online. Association for Computational Linguistics.

Shuyan Zhou, Xiangkai Zeng, Yingqi Zhou, Antonios Anastasopoulos, and Graham Neubig. 2019. Improving robustness of neural machine translation with multi-task learning. In *Proceedings of the Fourth Conference on Machine Translation (Volume 2: Shared Task Papers, Day 1)*, pages 565–571, Florence, Italy. Association for Computational Linguistics.

A Appendix

A.1 Model Hyperparameters

We train our models with 4 layers in each encoder and 2 or 4 layers in each decoder, each containing 4 attention heads. The embedding size is 256 and the hidden layer size is 1024. These hyper-parameter settings roughly correspond to the values used by (Wu et al., 2021) for character-level tasks. We use the Adam optimizer with an initial learning rate of 0.001,with both dropout and attention dropout set to 0.1, and batch size 400. We train the model for 150-300 epochs and the prediction is performed with the best checkpoint model, according to the development accuracy, using a beam of width 5.

A.2 Effect of λ Weighting on Multitask Performance

Table 4 illustrates the impact of adjusting the segmentation-glossing weight (λ) on Lezgi model performance. As λ increases, placing greater emphasis on segmentation loss, both accuracy and morpheme-level F1-score improve consistently.

These results suggest that balancing segmentation and glossing loss is crucial for multitask learning effectiveness. While high values of λ are generally beneficial, completely discarding glossing supervision could lead to the loss of valuable linguistic information. Thus, fine-tuning λ is essential to achieve the best trade-off between segmentation precision and linguistic generalization.

Model	Accuracy (%)	F1-score (%)
Single-task Baseline	44.66	60.75
Multitask ($\lambda = 0.5$)	40.78	62.12
Multitask ($\lambda = 0.6$)	42.20	63.31
Multitask ($\lambda = 0.7$)	43.23	65.59
Multitask ($\lambda = 0.8$)	46.23	66.59
Multitask ($\lambda = 0.9$)	48.54	68.84
Multitask ($\lambda = 1$)	48.04	68.12

Table 4: Impact of λ weighting on Lezgi model performance.

A.3 Learning Curves among All Languages

Figure 2 presents learning curves across different training dataset sizes (25%, 50%, 75%, and 100%). Each subplot corresponds to a different language, with the final panel showing the average trends across all languages.

Across all languages, the multitask model (solid lines) consistently outperforms the single-task model (dashed lines), particularly at lower training data sizes. This trend is most pronounced in Lezgi, Gitksan, and Arapaho, where multitask learning significantly boosts both word-level accuracy (red squares vs. orange circles) and morpheme F1-score (green diamonds vs. blue diamonds).

For languages like Nyangbo and Tsez, the difference between single-task and multitask learning diminishes as dataset size increases. Additionally, while morpheme F1-score improves steadily with more training data, word-level accuracy plateaus earlier in some languages (e.g., Uspanteko, Natügu), suggesting that segmentation benefits more from additional data than word-level reconstruction does.

A.4 LLM Prompt

You are a linguistics expert of Natügu. Your job is to generate new words based on the examples you learned. You are given this stem "pr", its meaning is "go". Here are several word examples of this stems:

Example 1:

surface form: prtrp, canonical segmentation: pr-tr-mq, gloss: go-GDIR.IN-PDIR.HITHER

.....

You are also given a list of grammatical morphemes and their corresponding gloss:

Grammatical gloss "3AUG", its morpheme is "nz"

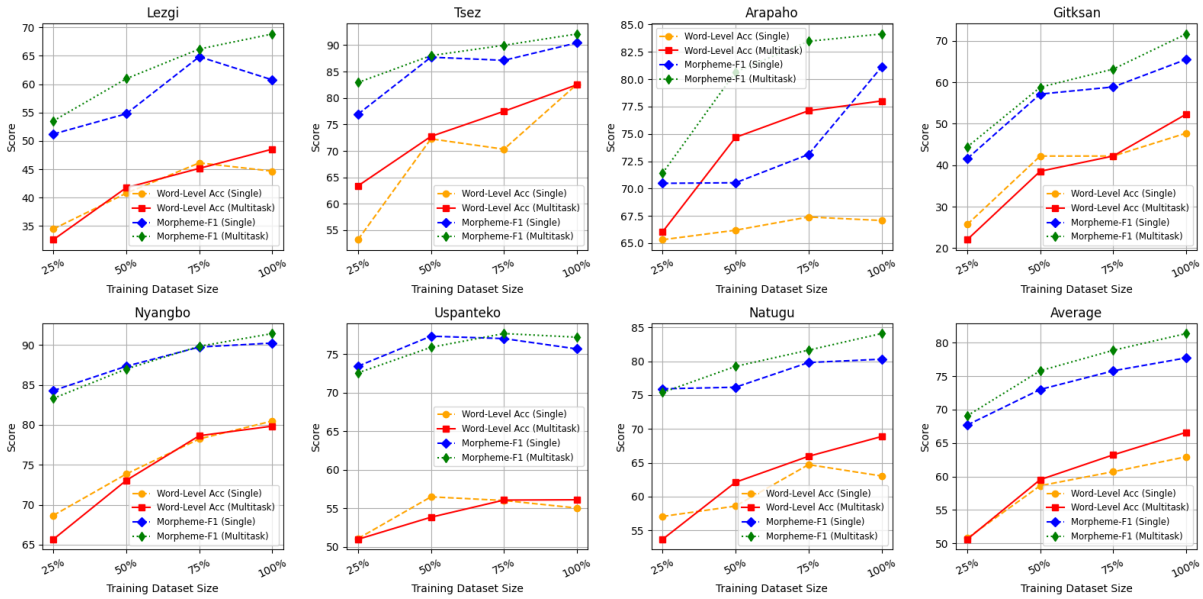

Figure 2: The learning curves for the F1 (top) and Accuracy (bottom) metrics among all languages.

Grammatical gloss "COS", its morpheme is "pe".
.....

Can you generate 3 new words using the stem and randomly use 2-5 grammatical morphemes. You need to return the result in the same format as the examples (word, canonical segmentation, and gloss). Please note that canonical segmentation will have character change.

A.5 Single-Task Ablation Results

Table 5 presents an ablation study evaluating the impact of LLM-generated synthetic data on both single-task and multitask models for canonical segmentation. Across all languages, adding synthetic data consistently improves segmentation performance, particularly at the morpheme level (F1-score). Notably, for single-task models, synthetic data provides incremental improvements, but these gains are more pronounced in the multitask setting, where segmentation and glossing are jointly learned.

When comparing S+LLM (0.5) vs. M+LLM (0.5) we observe that multitask learning consistently outperforms single-task learning across all metrics. The average F1-score for the multitask model (80.76%) is higher than the single-task model (80.02%), and the edit distance (ED) is also reduced more effectively (1414.86 vs. 1480.57). This suggests that multitask learning better integrates synthetic data, leveraging glossing as an auxiliary task to reduce segmentation errors and improve consistency.

Interestingly, in lower-resource languages like Gitksan, LLM augmentation provides the largest gains, particularly at higher proportions (75%), reinforcing that synthetic data is most beneficial in extreme data-scarce conditions. However, for languages with richer training data like Tsez and Nyangbo, improvements plateau.

Model	Metric	lez	ddo	arp	git	ntu	nyb	usp	ave
Baseline (S)	ACC↑	44.66	**82.6**	67.08	47.71	63.04	80.48	55.05	62.95
	F1↑	60.75	90.44	81.11	65.5	80.3	90.24	75.66	77.71
	ED↓	568	**652**	10495	117	458	154	1799	2034.71
M	ACC↑	48.54	82.51	78.01	52.29	68.87	79.84	56.12	66.59
	F1↑	68.84	**92.12**	84.14	71.64	84.09	**91.43**	77.18	81.35
	ED↓	519	698	6543	112	373	149	1623	1431
S+ LLM (0.25)	ACC↑	48.79	80.28	78.17	53.96	66.10	80.04	56.50	66.26
	F1↑	68.17	90.82	83.96	70.95	80.55	90.73	76.59	80.25
	ED↓	475	852	6534	**92**	357	137	1544	1427.29
M+ LLM (0.25)	ACC↑	49.27	80.41	78.14	52.29	69.02	80.21	57.10	66.63
	F1↑	**69.6**	91.03	84.49	72.78	**84.47**	91.30	77.86	81.65
	ED↓	500	779	6632	118	350	136	1538	1436.14
S+ LLM (0.5)	ACC↑	48.54	80.64	76.77	52.29	67.02	81.44	58.98	66.52
	F1↑	67.86	89.81	82.61	67.43	82.84	90.84	78.76	80.02
	ED↓	518	873	7037	101	367	**127**	1441	1480.57
M + LLM (0.5)	ACC↑	**49.51**	81.64	78.41	52.29	67.02	80.84	56.89	66.66
	F1↑	67.44	91.87	84.91	70.84	82.84	90.45	76.97	80.76
	ED↓	529	687	6483	117	367	164	1557	1414.86
S + LLM (0.75)	ACC↑	48.57	80.59	78.27	55.05	**72.47**	80.24	**59.87**	**67.87**
	F1↑	67.71	86.02	83.98	70.72	84.40	90.01	78.97	80.26
	ED↓	536	863	6635	99	**287**	147	**1432**	1428.43
M+ LLM (0.75)	ACC↑	48.82	81.32	**79.5**	**56.88**	68.71	**81.24**	58.29	67.82
	F1↑	67.69	91.51	**85.65**	**74.32**	84.18	91.34	**79.05**	**81.96**
	ED↓	**491**	723	**6502**	96	333	**127**	1507	**1397**

Table 5: Comparison of segmentation models across multiple languages. Each model has three sub-rows representing Word-Level Accuracy (ACC), Morpheme F1-Score (F1), and Edit Distance (ED). The last column provides the average of each metric across languages. M denotes multitask learning, and S denotes single-task learning, with synthetic data added at 25%, 50%, and 75% of training size.