HalleluBERT: Let Every Token That Has Meaning Bear Its Weight

Raphael Schmitt
TUM School of Computation, Information and Technology, Technical University of Munich, Germany
Institute of General Practice, Faculty of Medicine and Medical Center, University of Freiburg, Germany
raphael.schmitt@uniklinik-freiburg.de

Abstract

Transformer-based models have advanced NLP, yet Hebrew still lacks a RoBERTa encoder that is trained at scale and released in both base and large variants. We present HalleluBERT, a RoBERTa-based encoder family trained from scratch on 49.1 GB of deduplicated Hebrew web text and Wikipedia using a Hebrew-specific byte-level BPE vocabulary. On native Hebrew benchmarks for named entity recognition (BMC, NEMO) and sentiment classification (SMCD), HalleluBERT outperforms monolingual and multilingual baselines, and yields the highest unweighted mean score across the three benchmarks. We release model weights and tokenizer under the MIT license to support reproducible Hebrew NLP research.

Keywords: BERT, pre-training, language resources, evaluation, benchmarking

1. Introduction

The emergence of transformer-based language models such as BERT (Devlin et al., 2019) and RoBERTa (Liu et al., 2019) has transformed natural language processing (NLP), enabling contextualized word representations that generalize across diverse downstream tasks. While multilingual encoders such as mBERT, mmBERT (Marone et al., 2025), and XLM-RoBERTa (Chan, 2020) offer broad language coverage, monolingual models trained on large, high-quality corpora consistently outperform their multilingual counterparts on language-specific benchmarks (Chan et al., 2020; Martin et al., 2020; Delobelle et al., 2020; Scheible et al., 2024; Scheible-Schmitt and Frei, 2025).

For Hebrew, several BERT-style models have been introduced, but most are limited by corpus size, training budget, or absence of large-variant models. In particular, HeRo (Shalumov and Haskey, 2023) brought RoBERTa-style pre-training to Hebrew, but appears to have been trained under comparatively limited compute, and does not fully report the effective batch size or total update steps, leaving open how much additional performance could be unlocked under a more extensive pre-training setup. Further, there is no large-scale Hebrew model available.

To address this gap, we introduce HalleluBERT, a RoBERTa-based encoder family trained from scratch on large-scale, cleaned Hebrew web corpora. We release both base and large variants under the MIT open-source license and benchmark them against existing Hebrew-specific and multilingual models. Our contributions are as follows:

- We trained and released HalleluBERT$_{\text{base}}$ and HalleluBERT$_{\text{large}}$, RoBERTa-style encoders for Hebrew.
- We benchmarked HalleluBERT on NER and sentiment classification and provide a combined performance analysis against previous state-of-the-art models.

2. Related Work

Hebrew NLP has seen a steady progression of transformer encoders, from initial Hebrew-specific BERT models to larger-scale pre-training and RoBERTa-style setups.

HeBERT (Chriqui and Yahav, 2022) was the first transformer-based language model tailored for Modern Hebrew. Trained on Hebrew Wikipedia and OSCAR (Abadji et al., 2022), HeBERT improved over multilingual baselines on supervised tasks such as NER, PoS tagging, and sentiment analysis, and formed the basis for HebEMO, a tool for polarity and emotion recognition. However, its training corpus and vocabulary size were comparatively modest, limiting coverage of Hebrew's morphology and domain diversity.

AlephBERT (Seker et al., 2021) scaled pre-training to a larger and more diverse corpus (17.9 GB) by combining OSCAR, Wikipedia, and Twitter data. With a 52k vocabulary, AlephBERT established strong results on the Hebrew NLP pipeline and became a widely used baseline.

AlephBERT-Gimmel (Gueta et al., 2022) explores vocabulary scaling for Hebrew by increasing the WordPiece inventory to 128k items and additionally provides a smaller-capacity variant with fewer layers. Their analysis links larger vocabularies to fewer subword splits and reports gains across several Hebrew benchmarks; in our study, we consider only the base model.

HeRo (Shalumov and Haskey, 2023) introduced RoBERTa-style pre-training for Hebrew and re-

Proceedings of the Fifteenth Language Resources and Evaluation Conference (LREC 2026), pages 3022–3030
11–16 May 2026. ©ELRA Language Resources Association (ELRA), 2026

leased the HeDC4 corpus (47.5 GB), a deduplicated and cleaned dataset derived from OSCAR22 and mC4. Building on this resource, they trained HeRo and proposed LongHeRo, a Hebrew Longformer initialized from an intermediate HeRo checkpoint and continued for two additional epochs. The paper does not fully specify key training details such as the effective batch size or the total number of update steps, which can complicate direct comparisons across training budgets.

An overview of existing Hebrew transformer-based models is provided in Table 1.

3. Methods

3.1. Training Data

For pre-training HalleluBERT, we used the HeRo dataset (HeDC4) (Shalumov and Haskey, 2023) together with a Hebrew Wikipedia dump. HeDC4 is a deduplicated, language-identified, and quality-filtered Hebrew web corpus constructed from mC4 and OSCAR22. In our setup, the HeDC4 portion amounted to approximately 47.5 GB, and Wikipedia contributed an additional 1.6 GB, yielding a total of about 49.1 GB of pre-training text. To mitigate ordering effects from source-specific crawls, we shuffled the documents before pre-training.

3.2. Pre-processing

In line with RoBERTa, HalleluBERT employs byte-pair encoding (BPE) (Radford et al., 2019) for subword segmentation, operating directly on raw text without requiring pre-tokenization or external tools such as Moses (Koehn et al., 2007). Since the GPT-2 tokenizer is optimized for English, we constructed a Hebrew-specific tokenizer instead. Following the strategy applied in GottBERT (Scheible et al., 2024), we trained a byte-level BPE vocabulary on 20 GB of shuffled Hebrew text, drawn from both HeDC4 and Wikipedia. This resulted in a 52k subword inventory adapted to Hebrew's orthographic and morphological properties. Although we did not separately quantify its impact on compression or downstream performance, prior work in Dutch (Delobelle et al., 2020) and German (Scheible et al., 2024) indicates that language-specific tokenizers can yield improvements in both efficiency and accuracy. In practice, we found that sampling about 20 GB was sufficient for subword statistics to stabilize, while scaling vocabulary training to the entire corpus would primarily increase computational cost without offering substantial gains.

3.3. Pre-training

Following the setup of GottBERT, we pre-trained HalleluBERT$_{\text{base}}$ and HalleluBERT$_{\text{large}}$ using the fairseq framework (Ott et al., 2019) on a 128-core TPUv4 pod (Jouppi et al., 2023). Due to limitations of our fairseq setup, we did not employ mixed precision; both models were trained in full precision.

HalleluBERT$_{\text{base}}$ completed training in approximately 30.2 hours, while HalleluBERT$_{\text{large}}$ required around 6.0 days. We followed the standard RoBERTa pre-training schedule with 100k update steps, a global batch size of 8k, a 10k-step warmup, and polynomial learning rate decay. The base model used a peak learning rate of 0.0004, and the large model 0.00015. Although training was configured for 100k steps, the dataset permitted roughly 61 epochs.

3.4. Evaluation

Our evaluation design follows the HeRo benchmark suite (Shalumov and Haskey, 2023), which established NER (BMC and NeMO) and sentiment classification (SMCD) as the primary benchmarks for Modern Hebrew, complemented by a Hebrew QA task. In our work, we restrict the focus to NER and sentiment classification, building on the NNI- and Huggingface `transformers` (Wolf et al., 2020) based grid-search pipeline[1] of He et al. (2025), which we customized for our experiments. For each model and task, we performed a small grid search over batch size and learning rate. We selected the best configuration based on validation performance and report the corresponding score on the fixed test set; we used a single fine-tuning run per configuration with a fixed random seed (default value of 42). To stay efficient, we restricted our grid search to batch sizes {16, 32} and learning rates {5e-6, 7e-6, 1e-5, 2e-5, 5e-5}, based on the most frequent best-performing values in prior experiments (GottBERT, GeistBERT). Training was capped at a maximum of 30 epochs for NER and classification tasks, with early stopping applied using a patience of three epochs. All models used a linear learning rate schedule with a warmup phase of 10% of the total training steps. We computed all experiments on two NVIDIA RTX 3090 GPUs.

3.4.1. Sentiment Classification

For sentiment analysis in Modern Hebrew, we adopt the benchmark introduced by Amram et al. (2018). The dataset, released by OmiLab, consists of approximately 12,800 user-generated social media comments annotated for sentiment polarity (*positive*, *neutral*, *negative*). It is provided in two variants: a token-based and a morpheme-based representation, reflecting the morphological richness of Hebrew. In line with prior work, we focus on

[1] https://github.com/microsoft/nni

Model	Architecture	Pre-training Data	Corpus Size
HeBERT	BERT-base	Wikipedia, OSCAR, Emotion UGC	∼10.5 GB
AlephBERT AlephBERT-Gimmel	BERT-base	OSCAR, Wikipedia, Twitter	∼17.9 GB
HeRo	RoBERTa-base	HeDC4 (mC4 + OSCAR22)	∼47.5 GB
LongHeRo	Longformer	HeRo (continued, 2 epochs)	∼47.5 GB
HalleluBERT	RoBERTa-base / large	HeDC4 + Wikipedia	∼49.1 GB

Table 1: Overview of Hebrew transformer-based language models, including HalleluBERT.

the token-based variant, which has become the standard for model comparison.

As noted by Seker et al. (2021), the original dataset suffered from data leakage between splits due to duplicate samples. To address this, we follow the deduplicated version[2], which contains 8,465 unique samples after removing duplicates. Following HeRo (Shalumov and Haskey, 2023), we refer to this deduplicated corpus as SMCD (Social Media Comments Deduplicated).

The corpus is distributed with an official train/test split, where the test portion covers roughly 20% of the data. Following common practice, we retain the official test set unchanged for evaluation and extract a validation set comprising 10% of the training portion to support model selection. This results in a final distribution of approximately 72% train, 8% validation, and 20% test data. We report all performance figures on the official test set to ensure comparability with previous studies, using macro-averaged F_1 as our primary evaluation metric.

3.4.2. Named Entity Recognition

Named Entity Recognition (NER) is a core NLP task that involves identifying and classifying spans of text referring to entities such as persons, organizations, or locations. Early work on Hebrew NER was pioneered by Ben-Mordecai (2005).

A key difficulty for Hebrew is its rich morphology, where surface tokens often bundle multiple morphemes. To address this, Bareket and Tsarfaty introduced the NEMO^2 benchmark (Bareket and Tsarfaty, 2021), which provides both token- and morpheme-level annotations and has become the standard resource for Hebrew NER.

In our experiments, we evaluate HalleluBERT on the BMC dataset (split 1) (Ben-Mordecai, 2005) as well as on the token-based variant of NEMO^2 (Bareket and Tsarfaty, 2021), enabling comparison with earlier baselines while aligning with the current state of the art. Following standard practice, we report results in terms of the micro-averaged F1 score.

[2] https://github.com/omilab/Neural-Sentiment-Analyzer-for-Modern-Hebrew

4. Results

Table 2 summarizes the performance of all evaluated models across NER and sentiment classification benchmarks, including a combined average score.

Named Entity Recognition. Among the base models, HalleluBERT$_{\text{base}}$ achieves the highest micro-F_1 on both BMC (93.33) and NEMO (87.06), resulting in the strongest NER average (90.20). AlephBERT-Gimmel follows closely, slightly outperforming HalleluBERT$_{\text{base}}$ on classification but falling behind on NER. When comparing base to large models, it is noteworthy that HalleluBERT$_{\text{base}}$ marginally outperforms HalleluBERT$_{\text{large}}$ on BMC (93.33 vs. 93.23), suggesting that the BMC benchmark may not fully reward larger models, possibly due to its limited size or domain coverage. However, across NEMO and the NER AVG column, large models retain a small but consistent advantage, confirming that scaling generally benefits NER performance.

Sentiment Classification. For classification, AlephBERT obtains the highest F_1 among the base models (83.66), slightly ahead of mmBERT$_{\text{base}}$. HalleluBERT$_{\text{base}}$ remains competitive but does not lead this task. In the large-model category, HalleluBERT$_{\text{large}}$ achieves the highest classification score (84.91), surpassing XLM-RoBERTa$_{\text{large}}$ by over one point, further confirming its robustness across tasks.

Overall Average. Considering the final AVG column (BMC+NEMO+SMCD), HalleluBERT$_{\text{large}}$ attains the highest overall score (88.95), with HalleluBERT$_{\text{base}}$ ranking second (87.83). This highlights that HalleluBERT scales gracefully and delivers strong, balanced performance across both NER and classification, outperforming other baselines even when averaged across tasks.

5. Discussion

We were unable to reproduce HeRo as the reported state-of-the-art under our standardized fine-tuning

Model	Named Entity Recognition			Classification	AVG
	BMC	NEMO	AVG	SMCD	
HeBERT	89.33	76.16	82.74	82.64	82.71
AlephBERT	91.36	81.52	86.44	**83.66**	85.51
HeRo	92.0	83.35	87.68	80.95	85.43
HalleluBERT$_{\text{base}}$	**93.33**	**87.06**	**90.20**	83.09	**87.83**
mmBERT$_{\text{small}}$	83.96	71.95	77.96	81.89	79.27
AlephBERT-Gimmel	92.46	85.86	89.16	82.66	86.99
XLM-RoBERTa$_{\text{base}}$	86.32	79.37	82.84	82.07	82.59
mmBERT$_{\text{base}}$	84.61	77.97	81.29	83.55	82.04
HalleluBERT$_{\text{large}}$	**93.23**	**88.70**	**90.96**	**84.91**	**88.95**
XLM-RoBERTa$_{\text{large}}$	92.31	86.41	89.36	83.74	87.49

Table 2: All results are reported as percentages, based on the official test set and the best score out of 10 hyperparameter configurations (selected by validation performance). NER performance is measured by micro-F_1 on the BMC and NEMO corpora, with the AVG (NER) column showing their unweighted mean. SMCD refers to sentiment classification and is measured by macro-F_1. The rightmost AVG column reports the overall unweighted mean across BMC, NEMO, and SMCD, providing a single combined performance metric. Best scores are in bold, second-best are underlined, for base and large models respectively.

pipeline and hyperparameter search space. In our experiments, AlephBERT-Gimmel$_{\text{base}}$ ranked second among the base models and outperformed HeRo in both NER and overall averages, which may reflect differences in fine-tuning hyperparameters, preprocessing, or evaluation details. We hypothesize that differences in pre-training compute and training setup contributed to the gap. The HeRo paper reports two 35-day training stages on four NVIDIA GTX 1080 Ti GPUs with an initial learning rate of $1e^{-4}$ and linear decay, but key information needed for a like-for-like comparison (e.g., effective batch size and total update steps) is not fully specified (Shalumov and Haskey, 2023). Moreover, prior work suggests that downstream performance can be substantially influenced not only by corpus design but also by tokenizer and vocabulary choices; for instance, large-vocabulary encoder variants can be competitive across model families (Scheible-Schmitt and Schweter, 2025). Related work has also explored BPE vocabulary-size ablations in Turkish (Toraman et al., 2023); however, reported trends are not always directly comparable, as they may depend on the underlying pre-training budget and training depth (e.g., data scale, batch size, and number of update steps).

HalleluBERT$_{\text{base}}$ achieved the best NER results, even slightly surpassing its large variant on BMC (93.33 vs. 93.23), likely due to BMC's limited size and domain coverage. Classification was led by AlephBERT, with mmBERT$_{\text{base}}$ also performing well, possibly benefitting from multilingual pre-training. Despite this, HalleluBERT$_{\text{base}}$ remained competitive on classification and achieved the highest combined AVG (87.83). The large variant further improved to 88.95, clearly surpassing XLM-RoBERTa$_{\text{large}}$ (87.49).

Among multilingual models, XLM-RoBERTa$_{\text{base}}$ outperformed mmBERT$_{\text{base}}$ despite having fewer parameters, confirming its efficiency as a strong baseline. HalleluBERT, however, outperformed both multilingual models across all tasks, underscoring the value of monolingual pre-training for Hebrew. This contrasts with Scheible et al. (2024), where conservative pre-training learning rates and TPU-based pre-training, lacking dynamic memory allocation and mixed-precision optimization, likely reduced training efficiency and limited the benefits of scaling (Scheible et al., 2024). Under the same fairseq setup and TPU hardware, HalleluBERT achieved markedly better overall results.

Future work includes experimenting with Whole Word Masking (WWM), broader ablations, and throughput analyses (cf. PortBERT (Scheible-Schmitt et al., 2025)). In particular, ablating the byte-level BPE vocabulary size (e.g., 32k vs. 52k vs. 128k) could clarify the performance–efficiency trade-off in terms of sequence length and throughput. Finally, extending evaluation to additional native Hebrew benchmarks would provide a more comprehensive view of model capabilities.

6. Conclusion

We presented HalleluBERT, a RoBERTa-style language model for Modern Hebrew, trained extensively on large-scale Hebrew web and Wikipedia text using high batch sizes and long training schedules. Both base and large variants were released and evaluated on native NER and sentiment classification benchmarks, where HalleluBERT achieved state-of-the-art results, surpassing all monolingual and multilingual baselines and delivering the highest overall average across tasks.

These findings highlight the value of large-batch monolingual pre-training for Hebrew and demonstrate that substantial gains are achievable even with conservative hyperparameter choices.

7. Limitations

This work has several limitations. First, we used a fixed random seed and do not report confidence intervals or multi-seed averages. Our evaluation targets core encoder-style benchmarks for Modern Hebrew: NER (BMC, NEMO) and sentiment classification (SMCD). We did not include QA or the LongHeRo long-context variant, nor did we extend evaluation to broader suites such as NLI or paraphrase-style tasks. Although translated GLUE-style datasets exist for Hebrew,[3] their non-native nature limits comparability with established Hebrew benchmarks.

Second, although we used deduplicated corpora (HeDC4 and Wikipedia), we did not apply additional filtering or cross-source deduplication. As a result, residual noise, low-quality text, and potential biases may remain in the training data and influence the learned representations.

Third, HalleluBERT was trained exclusively on web-based Hebrew text, without explicit control for dialectal or register variation (e.g., Modern Hebrew vs. Biblical Hebrew, formal vs. colloquial text). This may limit the model's performance on underrepresented varieties or domain-specific language such as legal or medical text, unless additional fine-tuning is performed.

Fourth, we trained HalleluBERT$_{\text{large}}$ with a conservative peak learning rate of 0.00015 and did not explore extensive hyperparameter tuning. For both model variants we did not apply Whole Word Masking (WWM), which could potentially yield further gains. Pre-training was also performed without mixed precision, which increased computational cost and limited the feasibility of exploring longer training schedules or larger model configurations.

Fifth, we did not perform a detailed error analysis of the model outputs. Such an analysis could provide valuable insights into systematic weaknesses (e.g., common NER boundary errors, sentiment misclassifications) and guide future model and dataset improvements.

8. Ethical Considerations

Like all large-scale language models, HalleluBERT may inherit biases from its training data, which can influence downstream tasks such as classification or decision-making. While deduplication reduces redundancy and noise, it does not remove deeper societal or representational biases. Furthermore, training on large web-based corpora raises privacy concerns, as models may inadvertently retain sensitive information. Responsible deployment is especially important in high-stakes domains like legal, medical, or financial NLP.

Despite optimizations for efficiency, pre-training and evaluating transformer models remain computationally demanding, contributing to energy use and carbon emissions. These environmental costs highlight the need for balancing model performance with sustainable development goals.

[3]QQP: https://huggingface.co/datasets/imvladikon/qqp_he, STSB: https://huggingface.co/datasets/imvladikon/stsb_he

9. Acknowledgments

First and foremost, the author gives all honor and glory to his Lord and Savior, Jesus Christ, whose grace, strength, and guidance made this work possible.

Further, the author gratefully acknowledges the support of Google's TensorFlow Research Cloud (TFRC) for providing access to Cloud TPUs, which enabled efficient pre-training of HalleluBERT. The author also thanks Nora Limbourg, the assigned Google Cloud Customer Engineer, for her valuable technical assistance and coordination throughout the project.

10. Bibliographical References

Julien Abadji, Pedro Ortiz Suarez, Laurent Romary, and Benoît Sagot. 2022. Towards a Cleaner Document-Oriented Multilingual Crawled Corpus. *arXiv e-prints*, page arXiv:2201.06642.

Adam Amram, Anat Ben David, and Reut Tsarfaty. 2018. Representations and architectures in neural sentiment analysis for morphologically rich languages: A case study from Modern Hebrew. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 2242–2252, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

Dan Bareket and Reut Tsarfaty. 2021. Neural Modeling for Named Entities and Morphology (NEMO2). *Transactions of the Association for Computational Linguistics*, 9:909–928.

Naama Ben-Mordecai. 2005. Hebrew named entity recognition. Master's thesis, Department of Computer Science, Ben-Gurion University.

Branden Chan. 2020. XLM-RoBERTa: The multilingual alternative for non-english NLP. Library Catalog: towardsdatascience.com.

Branden Chan, Stefan Schweter, and Timo Möller. 2020. German's next language model. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 6788–6796, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Avihay Chriqui and Inbal Yahav. 2022. Hebert and hebemo: A hebrew bert model and a tool for polarity analysis and emotion recognition. *INFORMS Journal on Data Science*, 1(1):81–95.

Pieter Delobelle, Thomas Winters, and Bettina Berendt. 2020. RobBERT: a Dutch RoBERTa-based Language Model. *arXiv:2001.06286 [cs]*. ArXiv: 2001.06286.

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

Eylon Gueta, Avi Shmidman, Shaltiel Shmidman, Cheyn Shmuel Shmidman, Joshua Guedalia, Moshe Koppel, Dan Bareket, Amit Seker, and Reut Tsarfaty. 2022. Large pre-trained models with extra-large vocabularies: A contrastive analysis of hebrew bert models and a new one to outperform them all.

Henry He, Johann Frei, and Raphael Scheible-Schmitt. 2025. The Word and the Way: Strategies for Domain-Specific BERT Pre-Training in German Medical NLP. ISSN: 2693-5015.

Norman P. Jouppi, George Kurian, Sheng Li, Peter Ma, Rahul Nagarajan, Lifeng Nai, Nishant Patil, Suvinay Subramanian, Andy Swing, Brian Towles, Cliff Young, Xiang Zhou, Zongwei Zhou, and David Patterson. 2023. TPU v4: An optically reconfigurable supercomputer for machine learning with hardware support for embeddings.

Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondřej Bojar, Alexandra Constantin, and Evan Herbst. 2007. Moses: Open Source Toolkit for Statistical Machine Translation. In *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics Companion Volume Proceedings of the Demo and Poster Sessions*, pages 177–180, Prague, Czech Republic. 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. *arXiv:1907.11692 [cs]*. ArXiv: 1907.11692.

Marc Marone, Orion Weller, William Fleshman, Eugene Yang, Dawn Lawrie, and Benjamin Van Durme. 2025. mmbert: A modern multilingual encoder with annealed language learning.

Louis Martin, Benjamin Muller, Pedro Javier Ortiz Suárez, Yoann Dupont, Laurent Romary, Éric de la Clergerie, Djamé Seddah, and Benoît Sagot. 2020. CamemBERT: a Tasty French Language Model. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7203–7219, Online. 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. *arXiv:1904.01038 [cs]*. ArXiv: 1904.01038.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners. *OpenAI Blog*, 1(8):9.

Raphael Scheible, Johann Frei, Fabian Thomczyk, Henry He, Patric Tippmann, Jochen Knaus, Victor Jaravine, Frank Kramer, and Martin Boeker. 2024. GottBERT: a pure German language model. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 21237–21250, Miami, Florida, USA. Association for Computational Linguistics.

Raphael Scheible-Schmitt and Johann Frei. 2025. GeistBERT: Breathing life into German NLP. In *Proceedings of the Workshop on Beyond English: Natural Language Processing for all Languages in an Era of Large Language Models*, pages 42–50, Varna, Bulgaria. INCOMA Ltd., Shoumen, BULGARIA.

Raphael Scheible-Schmitt, Henry He, and Armando B. Mendes. 2025. PortBERT: Navigating the depths of Portuguese language models. In *Proceedings of the Workshop on Beyond English: Natural Language Processing for all Languages*

in an Era of Large Language Models, pages 59–71, Varna, Bulgaria. INCOMA Ltd., Shoumen, BULGARIA.

Raphael Scheible-Schmitt and Stefan Schweter. 2025. Sindbert, the sailor: Charting the seas of turkish nlp.

Amit Seker, Elron Bandel, Dan Bareket, Idan Brusilovsky, Refael Shaked Greenfeld, and Reut Tsarfaty. 2021. Alephbert:a hebrew large pre-trained language model to start-off your hebrew nlp application with.

Vitaly Shalumov and Harel Haskey. 2023. Hero: Roberta and longformer hebrew language models.

Cagri Toraman, Eyup Halit Yilmaz, Furkan Şahinu¸c, and Oguzhan Ozcelik. 2023. Impact of tokenization on language models: An analysis for turkish. *ACM Transactions on Asian and Low-Resource Language Information Processing*, 22(4):1–21.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander M. Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

A. Model Properties

Table 3 summarizes the vocabulary sizes and parameter counts of the Hebrew and multilingual models considered in our evaluation. HeBERT (109M) and AlephBERT (126M) represent the first generation of Hebrew BERT-style encoders. The HeRo model (125M) and its extended LongHeRo variant (149M) were trained on the HeDC4 corpus and provide strong baselines, particularly for long-sequence tasks. Our HalleluBERT$_{base}$ (126M) is comparable in size to AlephBERT and HeRo, while HalleluBERT$_{large}$ scales up to 357M parameters.

For multilingual points of reference, mBERT contains 178M parameters with a WordPiece vocabulary of 119k tokens, while XLM-R$_{base}$ and XLM-R$_{large}$ contain 278M and 560M parameters, respectively, with 250k-token SentencePiece vocabularies. All values were extracted using Huggingface's `transformers` library.

Table 3: Vocabulary size and total parameter count for Hebrew transformer-based models. Values were extracted using Huggingface's `transformers` library.

Model	Vocab Size	#Params
HeBERT	30522	109M
AlephBERT	52000	126M
HeRo	50265	125M
HalleluBERT$_{base}$	52009	126M
mmBERT$_{small}$	256000	140M
LongHeRo	50265	149M
AlephBERT-Gimmel	128000	184M
XLM-RoBERTa$_{base}$	250002	278M
mmBERT$_{base}$	256000	307M
HalleluBERT$_{large}$	52009	357M
XLM-RoBERTa$_{large}$	250002	560M

B. Perplexity

During pre-training, perplexity was tracked both on the training set (at each optimization step) and on the validation set (after each epoch; see Figure 1). Across both model variants, the curves exhibit a plateau phase: this phase is brief for the base models but more pronounced for the large ones. In some cases, short upward spikes occur, which might be misinterpreted as signs of divergence if considered in isolation. The base models typically stabilize after 20k–30k steps, while the large models require slightly longer but also converge by around 40k steps. This overall convergence pattern is mirrored in the validation perplexity, which was evaluated once per epoch.

C. Parameters

Table 5 lists the hyperparameters of the best models (selected by validation performance) for each benchmark, supporting reproducibility of our results. For transparency, Table 4 reports the total computation time per task, showing that all Hebrew downstream experiments together required roughly 65 hours of GPU time (about 2.7 days).

Task	Computation Time
BMC	15:43
NEMO	27:54
SMCD	22:10
Total	65:47

Table 4: Computation time in hours and minutes for the Hebrew downstream tasks, summing to 65 hours and 47 minutes (approximately 2.74 days).

Model	BMC		NEMO		SMCD	
	BS	LR	BS	LR	BS	LR
HalleluBERT$_{base}$	16	2 E-05	32	2 E-05	16	5 E-06
HeRo	16	7 E-06	32	2 E-05	16	2 E-05
AlephBERT	32	5 E-05	16	5 E-06	16	2 E-05
mmBERT$_{small}$	16	5 E-05	16	2 E-05	16	2 E-05
AlephBERT-Gimmel	16	5 E-05	16	2 E-05	32	5 E-05
HalleluBERT$_{large}$	32	7 E-06	16	5 E-06	32	7 E-06
XLM-RoBERTa$_{large}$	32	2 E-05	32	1 E-05	32	2 E-05
HeBERT	32	5 E-05	16	1 E-05	16	2 E-05
mmBERT$_{base}$	16	7 E-06	16	7 E-06	32	5 E-05
XLM-RoBERTa$_{base}$	16	2 E-05	16	7 E-06	32	5 E-05

Table 5: Hyperparameters of the best downstream task models for each task and pre-trained model. BS refers to batch size, and LR denotes the learning rate.

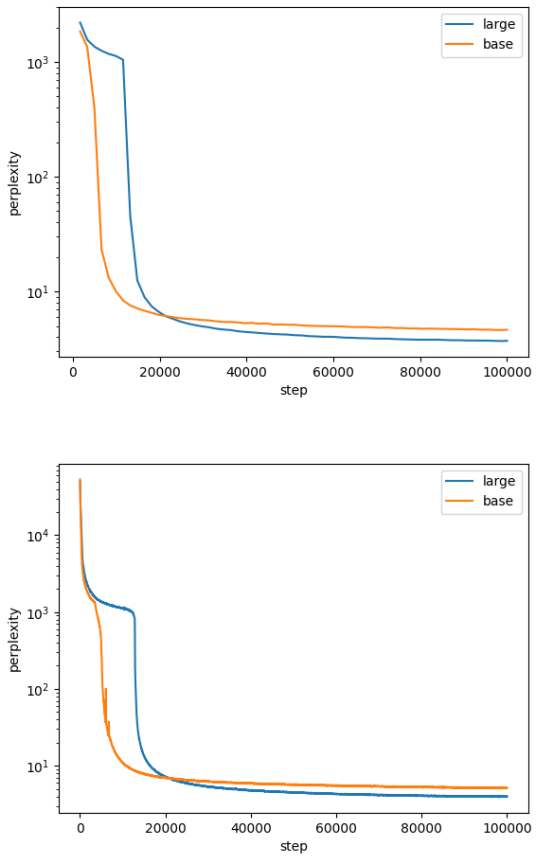

Figure 1: Perplexity of the HalleluBERT models. Top based on a validation at the checkpoints. Bottom based on the validation of each optimization cycle during the training.

D. Sequence Length

As shown in Table 6, we set maximum input lengths using the 95th percentile of the sequence length distribution for each dataset. To absorb small preprocessing variations, we added a safety margin and rounded up to the next power-of-two bucket (e.g., 64, 128, 192, 256). This choice balances efficiency and accuracy by avoiding excessive padding while minimizing truncation. For the NEMO corpus, although more than 95% of sequences are shorter than 128 tokens, the maximum observed length is 179. To cover these rare longer cases, we opted for a maximum length of 192 tokens. For the SMCD dataset, which exhibits a similar pattern, with a maximum of 1697 tokens but a 95th percentile far below 128, we likewise used 192 tokens as the maximum length. For the BMC dataset, although sequence lengths never exceed seven tokens, we fixed the maximum length to 64 tokens to match memory-efficient training configurations.

Task	Model	Max Len	Mean Len	95th Pctl.	Seq Len Used
SMCD	HalleluBERT$_{base}$	1697	30.79	96	192
	HalleluBERT$_{large}$	1697	30.79	96	
	XLM-RoBERTa$_{base}$	2028	40.74	131	
	XLM-RoBERTa$_{large}$	2028	40.74	131	
	mmBERT$_{small}$	2606	48.82	158	
	mmBERT$_{base}$	2606	48.82	158	
	HeBERT	1708	31.83	99	
	HeRo	1680	30.47	95	
	AlephBERT	1631	30.01	94	
	AlephBERT-Gimmel	1577	28.87	89	
BMC	HalleluBERT$_{base}$	5	3.92	4	64
	HalleluBERT$_{large}$	5	3.92	4	
	XLM-RoBERTa$_{base}$	4	3.78	4	
	XLM-RoBERTa$_{large}$	4	3.78	4	
	mmBERT$_{small}$	7	6.50	7	
	mmBERT$_{base}$	7	6.50	7	
	HeBERT	5	3.85	4	
	HeRo	5	3.90	4	
	AlephBERT	4	3.82	4	
	AlephBERT-Gimmel	4	3.46	4	
NEMO	HalleluBERT$_{base}$	106	29.02	53	192
	HalleluBERT$_{large}$	106	29.02	53	
	XLM-RoBERTa$_{base}$	151	39.94	76	
	XLM-RoBERTa$_{large}$	151	39.94	76	
	mmBERT$_{small}$	179	48.15	91	
	mmBERT$_{base}$	179	48.15	91	
	HeBERT	110	28.77	54	
	HeRo	108	28.64	52	
	AlephBERT	102	27.40	51	
	AlephBERT-Gimmel	100	26.03	48	

Table 6: Sequence length statistics for all benchmark datasets (SMCD, BMC, NEMO) across evaluated models. Reported are maximum observed length, mean length, 95th percentile, and the sequence length used during training.