Multilingual Sequence-to-Sequence Models for Hebrew NLP

Matan Eyal Hila Noga Roee Aharoni
Idan Szpektor Reut Tsarfaty

Google Research
{matane,hilanoga,roeeaharoni,szpektor,reutt}@google.com

Abstract

Recent work attributes progress in NLP to large language models (LMs) with increased model size and large quantities of pretraining data. Despite this, current state-of-the-art LMs for Hebrew are both under-parameterized and under-trained compared to LMs in other languages. Additionally, previous work on pretrained Hebrew LMs focused on encoder-only models. While the encoder-only architecture is beneficial for classification tasks, it does not cater well for sub-word prediction tasks, such as Named Entity Recognition, when considering the morphologically rich nature of Hebrew. In this paper we argue that sequence-to-sequence generative architectures are more suitable for large LMs in *morphologically rich languages* (MRLs) such as Hebrew. We demonstrate this by casting tasks in the Hebrew NLP pipeline as text-to-text tasks, for which we can leverage powerful multilingual, pretrained sequence-to-sequence models as mT5, eliminating the need for a separate, specialized, morpheme-based, decoder. Using this approach, our experiments show substantial improvements over previously published results on all existing Hebrew NLP benchmarks. These results suggest that multilingual sequence-to-sequence models present a promising building block for NLP for MRLs.

1 Introduction

In recent years, large pretrained language models showed impressive results in a variety of NLP tasks (Devlin et al., 2019; Raffel et al., 2020), domains (Beltagy et al., 2019; Araci, 2019), and languages (Antoun et al., 2020; Chan et al., 2020). These models were trained in a self-supervised fashion on large corpora with language modeling objectives. By doing this, models are taking advantage of information available in the training data without any access to explicit labels (Petroni et al., 2019). Recent works (Kaplan et al., 2020; Hoffmann et al., 2022) argue that language models capabilities scale as a power-law with model size, dataset size, and the amount of compute used for training.

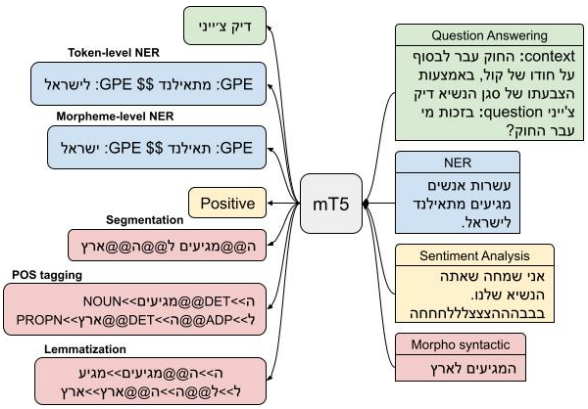

Figure 1: Finetuning mT5 on Hebrew tasks in text-to-text settings. See Fig. 2 for an English translated version.

Following the success such models achieved on English benchmarks, analogous language-specific models were developed to improve benchmark results in a variety of languages such as Arabic and French to name a few (Antoun et al., 2020; Martin et al., 2020). Hebrew NLP was no different, with a number of encoder-based BERT variations proposed, specifically HeBERT (Chriqui and Yahav, 2022), AlephBERT (Seker et al., 2022), and recently AlephBERTGimmel (ABG), a variation of AlephBERT with a much larger vocabulary (Guetta et al., 2022).

Despite the scaling laws proposed in Kaplan et al. (2020), all of the Hebrew variations of BERT are trained with a relatively small set of pretraining data, and are under-parameterized. In terms of training data, HeBERT and AlephBERT (and similarly ABG) were trained on 10.5GB and 16GB of Hebrew pretraining data respectively. mT5 (Xue et al., 2021) in comparison, was trained on mC4, where its public replica (Dodge et al., 2021) is a 27TB collection of natural text in 101 languages drawn from the public Common Crawl, 66GB of

Findings of the Association for Computational Linguistics: ACL 2023, pages 7700–7708
July 9-14, 2023

which is Hebrew (4.125x more Hebrew training data compared to AlephBERT). In terms of parameterization and model size, ABG, the largest Hebrew LM, has 60x and 945x fewer parameters compared to English T5 XXL (Raffel et al., 2020) and GPT3 (Brown et al., 2020), respectively, language models that were released two years earlier. Crucially for downstream use, although English T5 is a much larger model than available Hebrew language models, it can still be fine-tuned on common GPUs. See Appendix A for more details.

In addition to scale differences, all previous Hebrew LMs use an encoder-only architecture, even though the morphological complexity of Hebrew and other *morphologically rich languages* (MRLs)[1] pose challenges for the efficacy of this model. Consider, for instance, the task of POS tagging. Assigning POS tags for the phrase "babayit halavan"[2] requires to initially segment the phrase to its morphemes and only then assign each morpheme its matching POS tag. Since the number of input tokens does not match the number of output tags (2 input words and 5 output tags, one for each morpheme), a one-to-one token-to-tag classification head, as commonly employed in encoder-only models, is not feasible. The same problem appears in semantic tasks like Question-Answering (QA) and Named Entity Recognition (NER). For example, the named entity for "**ba**bayit halavan" is "**ha**bayit halavan". This goes beyond what encoder-only models can do by requiring the model to label a string that is not part of the input text.

To overcome this architectural obstacle in encoder-only models, the authors of AlephBERT and ABG (Seker et al., 2022; Guetta et al., 2022) used Brusilovsky and Tsarfaty (2022)'s three-step segmentation and tagging approach: contextualize the input, pass resulting embeddings to an LSTM decoder, which then generates the segmentation separated by a space symbol, in a char-by-char fashion. Then they pass the whitespace representation to a classification head. While effective for morpho-syntactic tasks, these additional components do not enable full generative capabilities, and are not pretrained, therefore the representation of morphemes cannot enjoy the pretrained LMs advantages.

The departure point of this work is that, in contrast to pre-trained encoders, sequence-to-sequence models can simply take the raw text as input and for any sequence labeling task, generate the morphemes and tags in a sequence. In POS tagging, for example, the generated output can be:

```
be»ADP@@ha»DET@@bayit»NOUN
ha»DET@@lavan»ADJ
```

where "@@" acts as a morpheme delimiter within a word and "»" is the morpheme-tag delimiter. See Sec. 3 for more details. For tasks such as Question-Answering, we can simply generate the target word forms without explicitly going through a segmentation phase. This change in approach to using sequence-to-sequence modeling is relevant for all MRLs, and in this paper we demonstrate its efficacy and effectiveness specifically for Hebrew.

This work thus identifies the challenge of current Hebrew LLMs as a *three-faceted* problem: Underparameterization, limited training data, and the use of a suboptimal pre-training architecture.[3] To address these three challenges at once, we propose using mT5 (Xue et al., 2021), a large multilingual sequence-to-sequence model that was pretrained on a vast amount of multilingual data including a significant amount of Hebrew data.[4] To adapt classification, span prediction, and token/morpheme classification tasks to mT5's text-to-text paradigm, we propose the text-only formulations illustrated in Figure 1. Subsequently, we report here that this paradigm change produces empirical improvements on all tasks evaluated compared to previous state-of-the-art, some of which are dramatic, as a 27.9 F1 increase in Hebrew Question-Answering.

2 Modeling

We use mT5 (Xue et al., 2021), a multilingual generative text-to-text version of T5 (Raffel et al., 2020), trained simultaneously on 101 languages. We evaluate mT5 on all its available sizes — Small, Base, Large, XL and XXL — ranging from 300M to 13B parameters. Subsequently, we propose casting all Hebrew NLP tasks for which evaluation benchmarks exist as text-to-text tasks: the input text is fed into the model, and targets are produced in a generative manner.

[1] MRLs, in contrast configurational languages (e.g. English), express grammatical functions at the word level via phenomena such as inflectional affixes, pronominal clitics, etc.

[2] Hebrew transcribed to English. Translated as "In the White House". The phrase is made of the morphemes be-ha-bayit ha-lavan (in-the-house the-white), but written and pronounced as "babayit halavan" without explicit boundaries.

[3] So far we mentioned encoder-only models as an example for suboptimal modeling choices for MRLs, but this is also the case when using poor tokenization, small vocabularies etc.

[4] It is beyond the scope of this paper to examine the factors that contributed to its improved performance, see Sec. 6.

Model	Param Count	ParaShoot		NEMO		BMC	Sentiment Analysis
				Token	Morph.		
		EM	F1	F1	F1	F1	F1
mBERT	178M	32	56.1	79.11	69.78	87.77	84.21
HeBERT	110M	18.2	36.7	81.13	77.29	89.41	87.13
AlephBERT	126M	26	49.6	83.62	77.55	91.12	89.02
ABG	185M	-	-	86.26	80.39	-	**89.51**
mT5 - Small	300M	24.52	48.71	66.74	62.08	77.7	87.55
mT5 - Base	580M	36.65	62.97	74.7	69.12	85.5	87.25
mT5 - Large	1.2B	42.6	70.13	84.33	81.85	90.77	88.73
mT5 - XL	3.7B	46.15	73.27	88.65	84.43	93.01	88.9
mT5 - XXL	13B	**50.37**	**77.5**	**89.86**	**88.65**	**93.29**	**89.61**

Table 1: mT5 outperforms previous encoder-only LMs on a variety of semantic Hebrew downstream tasks.

In contrast to text-to-text formulations of classification and span prediction, token classification is not as common in the literature, and specifically when the tokens consist of multiple morphemes, as is the case in MRLs. For example, in POS tagging for MRLs, each morpheme is assigned a POS tag, therefore multiple tags are assigned per word. As a result, a generative model cannot simply generate tag predictions one after the other, but it requires to first segment the text and only then label it accordingly. E.g., An unsatisfactory generation for "habayit" is *DET, NOUN* as we cannot recover which morpheme belongs to which tag. An acceptable model output, on the other hand, is *ha-DET, lavan-ADJ* as we can recover that *ha* was tagged with a *DET* and *lavan* with a *ADJ*. Throughout our experiments we tested a number of different text-to-text formulations. The best formulations for the tasks at hand are depicted in Fig. 1.

3 Experiments

Goal The goal of this study is to assess the performance of a sequence-to-sequence large language model, specifically mT5, that was trained on a large quantity of multilingual data, compared to existing Hebrew language models.

Models We fine-tuned different sizes of mT5 (Small to XXL) on all Hebrew tasks in a single-task fashion for 4096 steps, with a constant learning rate of 1e-3. For test set evaluation, we used the best-performing checkpoint from the development set, as tasks usually converge earlier. We compared the mT5 models against YAP (More et al., 2019), [5] mBERT (Devlin et al., 2019), HeBERT (Chriqui and Yahav, 2022), AlephBERT (Seker et al., 2022) and ABG (Guetta et al., 2022).

[5]YAP is the only available model trained for Hebrew lemmatization. YAP's scores are produced by us.

3.1 Tasks

We assembled an evaluation suite of Hebrew benchmarks composed of the following tasks: QA (Keren and Levy, 2021), NER (Bareket and Tsarfaty, 2021; Mordecai and Elhadad, 2005), Sentiment Analysis (Amram et al., 2018), and the morpho-syntactic tasks of segmentation, POS tagging and lemmatization from Sade et al. (2018), where we used the the latest dataset version, compatible with the ABG experiments (Guetta et al., 2022).

3.1.1 Question-Answering

Keren and Levy (2021) introduced ParaShoot, a Hebrew Question-Answering dataset which was created using the format and crowdsourcing methodology of SQuAD (Rajpurkar et al., 2016). We report token-level F1 and Exact Match scores as no morpheme boundaries are available. ParaShoot scores are from Keren and Levy (2021). The input is constructed by concatenating the context and question, with the output being the answer. We also conducted manual evaluation of different mT5 models on this dataset to evaluate the impact of model sizes, see details in Appendix B.

3.1.2 Named Entity Recognition

Bareket and Tsarfaty (2021) created NEMO, a NER add-on annotation for the Hebrew UD corpus (Sade et al., 2018). The authors proposed two dataset versions: token-level, where entities correspond to whitespace boundaries, similarly to BMC (Mordecai and Elhadad, 2005), and morpheme-level, with morpheme-based boundaries. The authors additionally revised the common NER evaluation procedure by comparing predicted and target entities on the surface form, boundaries and entity types, but not char positions. Thus, we train the seq-to-seq model to simply generate all of the sentence entities and their labels one after the other.

3.1.3 Sentiment Analysis

Correspondingly with previous work, we report F1 scores for Amram et al. (2018), a sentiment analysis dataset curated by annotating Facebook user comments with positive/negative/neutral tags.[6] In our sequence-to-sequence formulation the encoder receives raw text with the decoder generating one of three labels that correspond to the positive, negative and neutral tags. We use special tokens to ensure that generation only requires a single token.

[6]We use Seker et al. (2022) refined version which does not include leaks between split sets.

Model	Segmentation	POS Tagging	Lemmatization
YAP	93.64	90.13	78.6
mBERT	96.07	93.14	-
HeBERT	97.90	95.80	-
AlephBERT	97.88	95.81	-
ABG	98.09	96.22	-
mT5 - Small	94.83	94.55	89.96
mT5 - Base	96.34	95.9	92.09
mT5 - Large	96.76	95.58	92.21
mT5 - XL	98.32	96.91	95.13
mT5 - XXL	**98.67**	**97.46**	**95.53**

Table 2: Morpheme-Based Aligned MultiSet (mset) Results on the UD Corpus

3.1.4 Word Segmentation, POS Tagging and Lemmatization

Sade et al. (2018) manually validated the UDv2 version of the Hebrew treebank resulting in a set of morpho-syntactic tasks. Aligned to previous work we report word segmentation and POS tagging. We also evaluate our model on the lemmatization task and compare it to YAP (More et al., 2019), an open-source Hebrew parser. In accordance with previous work in Hebrew, we report aligned MultiSet (mset) scores. To produce the output for all these tasks we use two additional tokens: "`@@`" is the morpheme delimiter within a word and "`»`" is the morpheme-tag delimiter. E.g., segmentation and POS tagging of "habayit halavan" should result in the following sequences, `be@@ha@@bayit ha@@lavan` and `be»ADP@@ha»DET@@bayit»NOUN ha»DET@@lavan»ADJ`, respectively.

4 Results

Tables 1,2 summarize our empirical findings. Our results demonstrate a marked improvement over previously published results on existing Hebrew benchmarks. mT5 produces the biggest performance boost for the QA task of ParaShoot, with mT5-base already surpassing baseline models and mT5-XXL outperforming AlephBERT by 27.9 F1 points. For NER, mT5 produces better results than evaluated baselines on both of the dataset annotation levels. The largest performance boost comes in NEMO's morpheme-level version where mT5 learns to segment and label entities in an end-to-end fashion.

For sentiment analysis, mT5 outperforms the baseline models by a small fraction, however, manual error analysis we performed shows that 34% of its errors are annotation errors and for further 30% our annotators were not able to decide on the correct label. We conclude that work towards a cleaner, more challenging sentiment analysis dataset in Hebrew is needed. For segmentation and POS tagging we report error reduction of 30.3% and 32.8% compared to previous state-of-the-art. For the lemmatization task we report an increase of 16.93 mset F1 points compared to YAP. All of these are an important step towards closing the gap in morpho-syntactic tasks compared with other languages.

5 Related work

HeBERT (Chriqui and Yahav, 2022) is the first pretrained transformer-based language model trained on Hebrew Wikipedia and OSCAR (Ortiz Suárez et al., 2020) for the task of user-generated sentiment analysis. AlephBERT (Seker et al., 2022) was pretrained on the same copora in addition to a very large number of Hebrew tweets.

Guetta et al. (2022) tackled the extreme data sparseness in MRLs lexica (Tsarfaty et al., 2020) by pretraining with roughly 2.5x of AlephBERT vocabulary size, leading to performance improvements. Orthogonally, Keren et al. (2022) proposed using char-level LMs to mitigate the same sparseness problem, however results were inconclusive.

Xue et al. (2021) showed that mT5 outperforms baseline models on a number of multilingual datasets but did not directly evaluate on Hebrew. Alternatively, monolingual Hebrew LM papers only compared against mBERT (Devlin et al., 2019) as the sole multilingual baseline.

6 Limitations

mT5, compared with previous Hebrew LMs, is bigger, pretrained on more multiligual data, and learning to segment and tag in an end-to-end manner. While it was beyond the scope of this paper to pretrain new LMs and study which factors contributed to the improved performance, identifying these factors will be useful for determining the most effective approach for future work.

While larger mT5 models perform better than available LMs, they require more powerful hardware accelerators and take longer to train and infer. However, this is a reasonable trade-off from pretraining designated monolingual models from scratch, a more expensive task by itself. Additionally, the inclusion of data from 101 languages in the training of mT5 may have negatively impacted its performance on Hebrew, as some of the data may not have been relevant or beneficial to this particular language. Future work will need to address

this issue by training a monolingual Hebrew LM in order to further improve performance for Hebrew.

An inherent risk in sequence-to-sequence models is that they can generate inconsistent text with respect to the input text (Lee et al., 2018; Rohrbach et al., 2018). While potentially sensitive in different applications, a number of evaluation frameworks have been suggested to reduce the number of such "hallucinations" (Honovich et al., 2021, 2022). Another limitation of our evaluation framework is that, for lack of available datasets, we did not evaluate mT5 on purely generative tasks such as summarization and paraphrasing.

7 Conclusions

All of the Hebrew LMs to date are encoder-only models, which could not directly generate morpheme sequences, and thus necessitate a specialized monolingual decoder. In this work we propose to take advantage of mT5, a publicly available multilingual large language model that was trained on a considerable amount of multilingual and Hebrew data. Additionally the generative approach of text-to-text modeling is more aligned with the morphological challenges inherent in Hebrew and by that dispense with the need for specially-tuned decoders. We fine-tuned and evaluated mT5 on a set of Hebrew downstream tasks and report dramatic improvements. Subsequently, we propose that multilingual sequence-to-sequence models provide a more suitable pretraining alternative for MRLs, compared with the smaller, monolingual, encoder-only models.

8 Acknowledgements

We thank Dan Bareket and Eylon Guetta from Bar Ilan University for their help in sharing the UD and NEMO data.

References

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.

Wissam Antoun, Fady Baly, and Hazem Hajj. 2020. AraBERT: Transformer-based model for Arabic language understanding. In *Proceedings of the 4th Workshop on Open-Source Arabic Corpora and Processing Tools, with a Shared Task on Offensive Language Detection*, pages 9–15, Marseille, France. European Language Resource Association.

Dogu Araci. 2019. Finbert: Financial sentiment analysis with pre-trained language models. *arXiv preprint arXiv:1908.10063*.

Dan Bareket and Reut Tsarfaty. 2021. Neural modeling for named entities and morphology (NEMO2). *Transactions of the Association for Computational Linguistics*, 9:909–928.

Iz Beltagy, Kyle Lo, and Arman Cohan. 2019. SciBERT: A pretrained language model for scientific text. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3615–3620, Hong Kong, China. Association for Computational Linguistics.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Idan Brusilovsky and Reut Tsarfaty. 2022. Neural token segmentation for high token-internal complexity. *arXiv preprint arXiv:2203.10845*.

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 & hebemo: a hebrew bert model and a tool for polarity analysis and emotion recognition. *INFORMS Journal on Data Science*.

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.

Jesse Dodge, Maarten Sap, Ana Marasović, William Agnew, Gabriel Ilharco, Dirk Groeneveld, Margaret Mitchell, and Matt Gardner. 2021. Documenting large webtext corpora: A case study on the colossal clean crawled corpus. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 1286–1305, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Eylon Guetta, 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.

Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de Las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, et al. 2022. Training compute-optimal large language models. *arXiv preprint arXiv:2203.15556*.

Or Honovich, Roee Aharoni, Jonathan Herzig, Hagai Taitelbaum, Doron Kukliansy, Vered Cohen, Thomas Scialom, Idan Szpektor, Avinatan Hassidim, and Yossi Matias. 2022. TRUE: Re-evaluating factual consistency evaluation. In *Proceedings of the Second DialDoc Workshop on Document-grounded Dialogue and Conversational Question Answering*, pages 161–175, Dublin, Ireland. Association for Computational Linguistics.

Or Honovich, Leshem Choshen, Roee Aharoni, Ella Neeman, Idan Szpektor, and Omri Abend. 2021. q^2: Evaluating factual consistency in knowledge-grounded dialogues via question generation and question answering. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 7856–7870, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. 2020. Scaling laws for neural language models. *arXiv preprint arXiv:2001.08361*.

Omri Keren, Tal Avinari, Reut Tsarfaty, and Omer Levy. 2022. Breaking character: Are subwords good enough for mrls after all? *arXiv preprint arXiv:2204.04748*.

Omri Keren and Omer Levy. 2021. ParaShoot: A Hebrew question answering dataset. In *Proceedings of the 3rd Workshop on Machine Reading for Question Answering*, pages 106–112, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Katherine Lee, Orhan Firat, Ashish Agarwal, Clara Fannjiang, and David Sussillo. 2018. Hallucinations in neural machine translation. *NeurIPS 2018 Workshop on Interpretability and Robustness for Audio, Speech, and Language*.

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.

Naama Ben Mordecai and Michael Elhadad. 2005. Hebrew named entity recognition. *MONEY*, 81(83.93):82–49.

Amir More, Amit Seker, Victoria Basmova, and Reut Tsarfaty. 2019. Joint transition-based models for morpho-syntactic parsing: Parsing strategies for MRLs and a case study from Modern Hebrew. *Transactions of the Association for Computational Linguistics*, 7:33–48.

Pedro Javier Ortiz Suárez, Laurent Romary, and Benoît Sagot. 2020. A monolingual approach to contextualized word embeddings for mid-resource languages. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1703–1714, Online. Association for Computational Linguistics.

Fabio Petroni, Tim Rocktäschel, Sebastian Riedel, Patrick Lewis, Anton Bakhtin, Yuxiang Wu, and Alexander Miller. 2019. Language models as knowledge bases? In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2463–2473, Hong Kong, China. Association for Computational Linguistics.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, Peter J Liu, et al. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *J. Mach. Learn. Res.*, 21(140):1–67.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. SQuAD: 100,000+ questions for machine comprehension of text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2383–2392, Austin, Texas. Association for Computational Linguistics.

Anna Rohrbach, Lisa Anne Hendricks, Kaylee Burns, Trevor Darrell, and Kate Saenko. 2018. Object hallucination in image captioning. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4035–4045, Brussels, Belgium. Association for Computational Linguistics.

Shoval Sade, Amit Seker, and Reut Tsarfaty. 2018. The Hebrew Universal Dependency treebank: Past present and future. In *Proceedings of the Second Workshop on Universal Dependencies (UDW 2018)*, pages 133–143, Brussels, Belgium. Association for Computational Linguistics.

Amit Seker, Elron Bandel, Dan Bareket, Idan Brusilovsky, Refael Greenfeld, and Reut Tsarfaty. 2022. AlephBERT: Language model pre-training and evaluation from sub-word to sentence level. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 46–56, Dublin, Ireland. Association for Computational Linguistics.

Reut Tsarfaty, Dan Bareket, Stav Klein, and Amit Seker. 2020. From SPMRL to NMRL: What did we learn (and unlearn) in a decade of parsing morphologically-rich languages (MRLs)? In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7396–7408, Online. Association for Computational Linguistics.

Linting Xue, Noah Constant, Adam Roberts, Mihir Kale, Rami Al-Rfou, Aditya Siddhant, Aditya Barua, and Colin Raffel. 2021. 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.

A Running T5 on common GPUs

AlephBertGimmel (Guetta et al., 2022), the largest Hebrew LM to date, is roughly the same size as BERT base (Devlin et al., 2019), and even though T5 is 60 times larger than AlephBertGimmel, we do not need to horizontally scale our hardware accelerators by 60 to accommodate it.

As models have grown in size, hardware accelerators have also become more advanced. T5 Small, Base and Large can all be fine-tuned on a 2016 Nvidia `P100` or 2017 Nvidia `V100` GPU accelerators. T5 XL and XXL can be fine-tuned on the 2020 Nvidia `A100` GPU, the same accelerator used for pretraining AlephBERTGimmel.

Given the widespread availability of these GPU accelerators, we argue that the T5 models we evaluate in this work can be easily fine-tuned and deployed nowadays.

B Qualitative Evaluation of mT5 on the Question-Answering Task

The mT5-small model performs similarly to previous state-of-the-art models on the Question-Answering task of ParaShoot (Keren and Levy, 2021). We conducted a qualitative analysis of mT5-XXL compared with mT5-small, as a way to analyse the impact of model size while holding other factors constant, and in order to compare to the performance of previous state-of-the-art models.

We ran our mT5 experiments using 3 seeds with the best performing model, mT5-XXL, achieving 77.99 F1 and 50.63 EM scores. Our worst performing model, mT5-small, reached 47.67 F1 and 24.39 EM scores. From the 519 exact match prediction mT5-XXL model made, 167 of which mT5-small received F1 scored of 0. Based on a manual evaluation of the errors made by mT5-small, it can be concluded that the model often struggled with comprehending the fundamental meaning of the question in many instances. As an illustrative example, here the model mixes *when* and *where*:

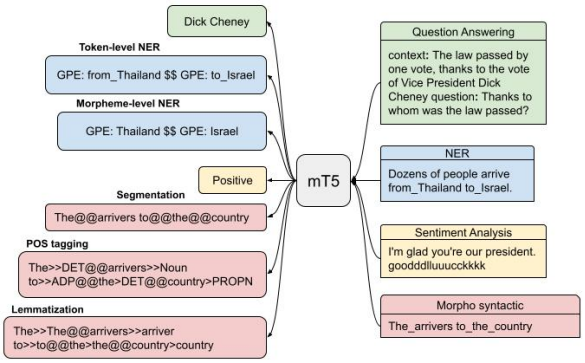

Figure 2: Translation of Fig. 1 from Hebrew to English. For consistently with the original Figure, Right-to-left presentation is kept. Words with multiple morphemes are merged in this Figure with the _ sign.

Context:[7]

לאחר תבוסת צרפת במסגרת המערכה על צרפת החליט מפקד הצבא הצרפתי במושבה פול לואי לזאנטיום להמשיך בלחימה לצד צרפת החופשית

Question:[8]

מתי החליט מפקד הצבא להמשיך בלחימה לצד צרפת החופשית

The gold and mT5-XXL prediction is:[9]

לאחר תבוסת צרפת במסגרת המערכה על צרפת

mT5 small's model predicted:[10]

במושבה פול לואי לזאנטיום.

As known to be a problem with generative models, both mT5 models made several hallucination errors, returning answers that were not part of the original context. Additionally, mT5-XXL failed to answer 49 questions correctly which mT5-small was able to provide accurate responses for them. However, for only three of these questions, mT5-XXL received an F1 score of 0. Upon manual evaluation of these errors, it was found that two of them are alternative correct answers.

[7] Context translated to English: After France's defeat in The Campaign for France, the commander of the French army in the colony, Paul Legentilhomme, decided to continue fighting with the Free French Forces

[8] Question translated to English: When did the army's commander decide to continue fighting with the Free French Forces?

[9] The gold and mT5-XXL prediction translated to English: After France's defeat in the campaign for France

[10] mT5 small's model predicted translated to English: In colony Paul Legentilhomme

ACL 2023 Responsible NLP Checklist

A For every submission:

☑ A1. Did you describe the limitations of your work?
7

☑ A2. Did you discuss any potential risks of your work?
7

☑ A3. Do the abstract and introduction summarize the paper's main claims?
1

☒ A4. Have you used AI writing assistants when working on this paper?
Left blank.

B ☒ Did you use or create scientific artifacts?

Left blank.

☐ B1. Did you cite the creators of artifacts you used?
No response.

☐ B2. Did you discuss the license or terms for use and / or distribution of any artifacts?
No response.

☐ B3. Did you discuss if your use of existing artifact(s) was consistent with their intended use, provided that it was specified? For the artifacts you create, do you specify intended use and whether that is compatible with the original access conditions (in particular, derivatives of data accessed for research purposes should not be used outside of research contexts)?
No response.

☐ B4. Did you discuss the steps taken to check whether the data that was collected / used contains any information that names or uniquely identifies individual people or offensive content, and the steps taken to protect / anonymize it?
No response.

☐ B5. Did you provide documentation of the artifacts, e.g., coverage of domains, languages, and linguistic phenomena, demographic groups represented, etc.?
No response.

☐ B6. Did you report relevant statistics like the number of examples, details of train / test / dev splits, etc. for the data that you used / created? Even for commonly-used benchmark datasets, include the number of examples in train / validation / test splits, as these provide necessary context for a reader to understand experimental results. For example, small differences in accuracy on large test sets may be significant, while on small test sets they may not be.
No response.

C ☑ Did you run computational experiments?

3,4

☑ C1. Did you report the number of parameters in the models used, the total computational budget (e.g., GPU hours), and computing infrastructure used?
3

The Responsible NLP Checklist used at ACL 2023 is adopted from NAACL 2022, with the addition of a question on AI writing assistance.

☑ C2. Did you discuss the experimental setup, including hyperparameter search and best-found hyperparameter values?
3

☑ C3. Did you report descriptive statistics about your results (e.g., error bars around results, summary statistics from sets of experiments), and is it transparent whether you are reporting the max, mean, etc. or just a single run?
4

☒ C4. If you used existing packages (e.g., for preprocessing, for normalization, or for evaluation), did you report the implementation, model, and parameter settings used (e.g., NLTK, Spacy, ROUGE, etc.)?
Left blank.

D ☒ Did you use human annotators (e.g., crowdworkers) or research with human participants?

Left blank.

☐ D1. Did you report the full text of instructions given to participants, including e.g., screenshots, disclaimers of any risks to participants or annotators, etc.?
No response.

☐ D2. Did you report information about how you recruited (e.g., crowdsourcing platform, students) and paid participants, and discuss if such payment is adequate given the participants' demographic (e.g., country of residence)?
No response.

☐ D3. Did you discuss whether and how consent was obtained from people whose data you're using/curating? For example, if you collected data via crowdsourcing, did your instructions to crowdworkers explain how the data would be used?
No response.

☐ D4. Was the data collection protocol approved (or determined exempt) by an ethics review board?
No response.

☐ D5. Did you report the basic demographic and geographic characteristics of the annotator population that is the source of the data?
No response.