Linguistic Units as Tokens: Intrinsic and Extrinsic Evaluation with BabyLM

Achille Fusco [1,2] Maria Letizia Piccini Bianchessi [2]
Tommaso Sgrizzi [2] Asya Zanollo [2] Cristiano Chesi [2]

[1]University of Florence [2]NeTS Lab, IUSS Pavia

Motivation and contributions

- Tokenization defines the model's basic representational units and critically shapes downstream behavior.
- We compare BPE with three **linguistically motivated tokenizers**: MorPiece and ParadigmFinder (inducing morpheme segmentation) and SylliTok (targeting syllables).
- Two evaluations:
  1. **Intrinsic** on SIGMORPHON 2022 morphological segmentation benchmark;
  2. **Extrinsic** on BabyLM 2025 benchmark (with a GPT-2 model baseline).
- Contribution: link **morphological faithfulness** to **downstream performance** under strict budgets.

Data and Model setup

- **BabyLM strict-small**: ~10M words, mixed conversational and written sources.
- **Preprocessing**: mild normalization; keep apostrophes.
- **GPT-2 base**: 12 layers, 12 heads, 768 hidden; context 1024.
- **Training**: seq len 512, batch 16, AdamW 5e-5, warmup, weight decay, 10 epochs.

Tokenization, Compression and Morphology

Tokenization sits at the intersection of practical engineering choices, information-theoretic principles, and linguistic theory.

- It defines the model's **basic vocabulary**, determining out-of-vocabulary coverage, representational granularity, and **effective sequence length**.
- It's a form of **data compression**: BPE originated as general-purpose compression algorithm (Gage, 1994), while **Goldman et al. (2024)** show that higher compression capacity correlates with increased log-likelihood and better model performance.
- From a **linguistic** standpoint, tokenization parallels morphology in uncovering **sub-word structure**. Goldsmith (2001) also linked morphology to compression, proposing an unsupervised morphological learner optimizing a Minimum Description Length (MDL) metric.
- Human learners show an exceptional efficiency: children acquire morphological rules from sparse input (Berko 1958; Lignos & Yang 2016), generalizing patterns only when irregularities remain below a certain threshold.
- This threshold, formalized by the **Tolerance–Sufficiency Principle** (Yang 2016), defines when a rule R over N items is productive, given e exceptions:

$$e \leq \theta_N \quad \text{where } \theta_N = \frac{N}{\ln N} \tag{1}$$

Tokenizers at a glance

BPE

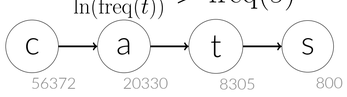
- Frequency-based merges (Gage, 1994; Sennrich et al., 2016).
- Favors compression over morphological accuracy.
- With enough merges or large vocab, *dog* and *dogs* become separate tokens.

MorPiece

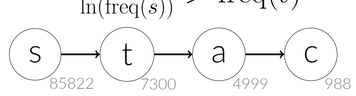
Split-based segmentation guided by the Tolerance–Sufficiency Principle.

1. Two tries are used: one obtained from a left-to-right pass of each space-separated token (root trie), and one from a right-to-left traversal of each token.
2. During a single pass over the corpus, each token is traversed in both directions. Each pendant, if present, is updated for frequency, or created if it is new.
3. When the (following) Sufficiency Principle holds between a mother node and a daughter node, a split is postulated:

split *iff* in *root-trie*: $\frac{\text{freq}(t)}{\ln(\text{freq}(t))} > \text{freq}(s)$



and in *infl-trie*: $\frac{\text{freq}(s)}{\ln(\text{freq}(s))} > \text{freq}(t)$



ParadigmFinder

- Inspired by Goldsmith (2001) and Xu et al. (2018), it infers morphological paradigms from the word vocabulary and uses them to segment words.

Training:

1. Enumerate all binary splits (incl. $-\varnothing$) and group roots by identical suffix sets.
2. Expand by Tolerance–Sufficiency: given two paradigms P_i and P_j with root sets R_i and R_j and corresponding suffix sets S_i and S_j, where $|S_i| < |S_j|$, P_i is merged into P_j if and only if a majority of roots in R_j occur in R_i, that is, if

$$|R_j| - |R_i \cap R_j| \geq \theta_{|R_j|} \quad \text{where} \quad \theta_{|R_j|} = \frac{|R_j|}{\ln |R_j|} \tag{2}$$

3. Rank paradigms by $\text{Score}_P = \log_2 |R| \times \log_2 |S|$; set vocab size to **30K**.

Tokenization:

1. Try paradigm matches (best first); else longest known suffix;
2. Segment accordingly.

SylliTok

- Deterministic English syllabification rules; cognitively motivated units.
- Vocabulary of **~20K tokens**.

Intrinsic evaluation: SIGMORPHON 2022 (adapted)

- The original benchmark used *deep* segmentation (restoring lemmas), while our version uses *surface* segmentation (preserving character sequences):

deep segmentation	surface segmentation
collision → *collide+ion*	*collision* → *collis+ion*

- ParadigmFinder and MorPiece best track morpheme boundaries, as expected.

Tokenizer	Avg. Lev. Dist.	Prec.	Rec.	F1
BPE	2.08	21.03	26.62	23.50
MorPiece	1.96	24.54	29.52	26.80
SylliTok	2.77	12.45	18.81	14.98
ParadigmFinder	**1.24**	**38.99**	**30.12**	**33.99**

Extrinsic evaluation: GPT-2 on BabyLM 2025

Task	BPE	MoP	SylliTok	ParFind
BLiMP	**66.4**	63.5	63.1	65.2
BLiMP-Suppl.	57.1	52.6	**58.8**	**58.8**
COMPS	51.7	55.8	55.3	**56.6**
EWoK	49.9	**50.6**	49.9	49.4
Eye Track.	**8.7**	1.2	0.9	0.1
SPR	**4.3**	0.7	0.1	0.3
Entity Tracking	13.9	**64.4**	33.9	21.0
WUG_ADJ	**66.1**	37.6	33.1	-43.1
WUG_PAST	-5.0	**12.1**	-29.4	-2.6
GLUE	55.9	57.7	**58.1**	57.8
AoA	11.7	-25.6	-31.7	**16.3**

MoP = MorPiece; ParFind = ParadigmFinder; SPR = Self-Paced Reading.

Takeaways

- Under the 10M-token constraint, the top SIGMORPHON performer (**ParadigmFinder**) matched but did not surpass **BPE** on BLiMP benchmarks (**−1 BLiMP**, **+1 BLiMP-Suppl.**).
- **ParadigmFinder** yields clear gains on **COMPS** and **Age of Acquisition**, while **MorPiece** shows a striking advantage on the **Entity-Tracking** task.
- **SylliTok** provides no consistent gains across tasks.
- **BPE** best correlates with human processing measures (**eye-tracking**, **self-paced reading**), yet overall values remain low.

Selected References

Batsuren et al. (2022). The SIGMORPHON 2022 shared task on morpheme segmentation. * **Berko (1958).** The child's learning of English morphology. * **Fusco et al. (2024).** Recurrent networks are (linguistically) better? * **Gage (1994).** A new algorithm for data compression. * **Goldman et al. (2024).** Unpacking tokenization: Evaluating text compression and its correlation with model performance. * **Goldsmith (2001).** Unsupervised learning of the morphology of a natural language. * **Lignos & Yang (2016).** Morphology and language acquisition. * **Sennrich et al. (2016).** Neural machine translation of rare words with subword units. * **Warstadt et al. (2020).** BLiMP: The benchmark of linguistic minimal pairs for English. * **Xu et al. (2018).** Unsupervised morphology learning with statistical paradigms. * **Yang (2016).** *The price of linguistic productivity.*