Universal Byte-Level Encoding: UTF-8/UTF-16 Routing to Reduce Cross-Script Token-Budget Disparities

Lower cross-script disparity for BPE tokenizers, and up to one-third shorter inputs for byte-level models.

Hyunsik Kim · **Youngmoon Jung**
Samsung Research

Project page

1 UTF-8 charges some scripts more

Byte-level BPE (BBPE) [2] starts from UTF-8 bytes. Where no learned merge covers a character, it costs one base symbol per byte: the **encoding floor**.

	ASCII	Cyrillic, Arabic, Hebrew	Korean, Chinese, Japanese, Devanagari, Thai	Non-BMP
BBPE	1	2	3	4
BBPE16	2	2	2	4
UBE	**1**	2	**2**	4

Base symbols per character before BPE merges.

BBPE tokens per FLORES-200 sentence, 101-lang/16K

308.5
LC-BMP (low-coverage BMP): Tibetan, Myanmar, Ol Chiki, Tifinagh scripts

34.9
English

A fixed token budget fits about **one-ninth** as many parallel sentences as English.

2 UTF-8 is less compact for 3-byte BMP

`1` format bit `0` leading zero `x` code-point bit ▢ UBE route

1-byte U+0000–007F
UTF-8 `0xxxxxxx` — **1 byte**
UTF-16 `00000000 0xxxxxxx` — 2 bytes

2-byte U+0080–07FF
UTF-8 `110xxxxx 10xxxxxx` — **2 bytes**
UTF-16 `00000xxx xxxxxxxx` — 2 bytes

3-byte U+0800–FFFF
UTF-8 `1110xxxx 10xxxxxx 10xxxxxx` — 3 bytes
UTF-16 `xxxxxxxx xxxxxxxx` — **2 bytes**

4-byte U+10000–10FFFF
UTF-8 `11110xxx 10xxxxxx 10xxxxxx 10xxxxxx` — 4 bytes
UTF-16 `110110xx xxxxxxxx 110111xx xxxxxxxx` — **4 bytes**

A 3-byte UTF-8 character spends **8 of its 24 bits** on length and continuation markers, the price of self-synchronization; its **16 code-point bits** fit in 2 bytes of UTF-16 [1]. UTF-16 saves a byte only for 3-byte BMP characters and adds one for ASCII.

Ranges exclude surrogate code points. In surrogate pairs, x bits encode the code point minus 0x10000. UTF-16 code units are drawn most significant bit first; UBE writes them little-endian.

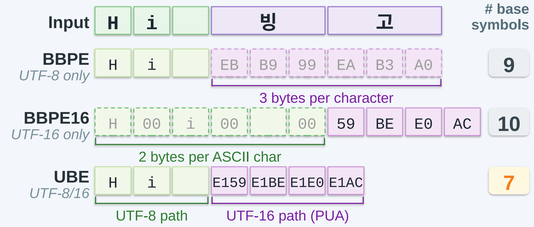
3 UBE: route each character

One deterministic rule on each character's UTF-8 length:

1–2 bytes	UTF-8 bytes	GPT-2 byte symbols
3–4 bytes	UTF-16LE bytes	PUA U+E100–E1FF

Two disjoint 256-symbol alphabets tell the decoder which path each symbol took: **no marker tokens, no language ID, no script table.**

Example: Hi 빙고 before BPE merges



Each box is one base symbol, the input unit of a byte-level model.

Standard BPE, same model
Only the initial symbols change: same BPE trainer, merge rule and Transformer.

Exact decoding
Round-trips every Unicode 17 scalar value and the official normalization, grapheme-break and emoji test inputs. Of UBE, SCRIPT-BPE and MYTE, only UBE recovers every raw audit input.

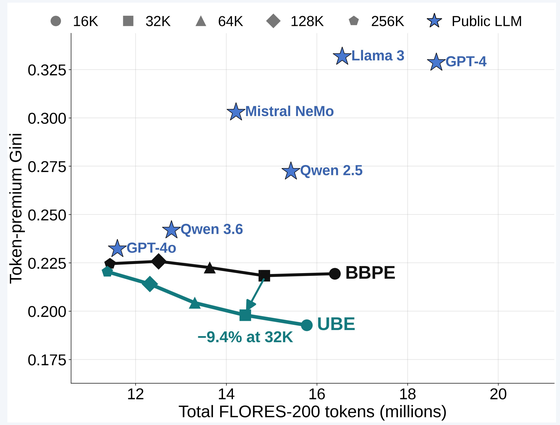
Small overhead
256 extra base symbols: <1% of a vocabulary of 32K or more.

4 LM quality and speed

- Matches BBPE's LM quality in multilingual LM experiments from 182M to 1.3B parameters and 32K to 128K vocabularies.
- Same-content prefill rate: **+45.0%** Amharic, +3.9% English (FLORES-200; 6-lang/32K, 182M, A100 80 GB, batch 1, FP32).
- Token-count estimate: **53%** more Santali content per fixed token budget (FLORES-200, 101-lang/32K).

5 Token level: lower disparity

FLORES-200 devtest, 204 languages.



UBE/BBPE: 101-language corpus, 16K–256K. Stars: public tokenizers with different training data. Lower-left is better.

6–14%
lower Gini in 9/9 core settings (3 corpora × 16K–64K)

−9.4%
Gini at 101-lang/32K: 0.2183 → 0.1979

189/204
languages with fewer tokens (101-lang/32K)

Mean tokens per sentence, 101-lang/32K

Language (script)	UBE	BBPE	Δ
Santali (Ol Chiki)	**222.944**	341.287	**−34.7%**
Central Atlas Tamazight (Tifinagh)	**183.305**	269.166	**−31.9%**
Tibetan	**286.737**	336.804	**−14.9%**
Arabic	**60.407**	60.600	**−0.3%**
English	**31.068**	31.160	**−0.3%**

- UBE also lowers English tokens at every tested vocabulary size (6-lang and 101-lang). **Why:** UBE writes 3-byte BMP characters as 2 symbols, so BPE spends fewer merges on them and more on Latin-script text such as English. At 101-lang/32K, Gini drops on UDHR (−11.1%) and SIB-200 (−8.5%) too.
- Fewer tokens and lower Gini with SuperBPE-style [3] (0.1922 → 0.1725) and SCRIPT-style [4] (0.2036 → 0.1824) boundaries (101-lang/32K).

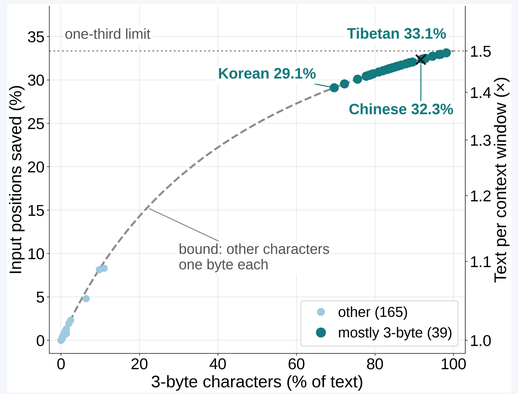
Emulated commercial mixtures. 30 GiB mC4 mixtures: inferred GPT-4o and Mistral NeMo shares [6], plus a Chinese-centric scenario. FLORES-200 results for all 10 UBE/BBPE pairs (32K–200K):

1.9–4.6% fewer tokens
12.5–25.3% fewer LC-BMP tokens
5.5–11.7% lower Gini

English tokens also fall in every pair.

6 Byte-level models gain more

Byte-level models such as ByT5 [7] read one input position per base symbol, with no merges.



UBE vs. UTF-8 bytes, one point per FLORES-200 language; mostly 3-byte: at least half of the characters; Chinese: mean of Chinese (Simplified), Chinese (Traditional), and Cantonese.

29–33%
shorter in all 39 languages with mostly 3-byte text

−27.8%
Gini at the byte level: 0.2077 → 0.1500

202/204
languages shorter, 2 tied, none longer

Mean base symbols per sentence, FLORES-200 (no merges)

Languages	UBE	UTF-8 bytes	Δ
Korean, Chinese, Japanese	**91.252**	133.665	**−31.7%**
LC-BMP	**277.056**	406.632	**−31.9%**
All 204	**176.088**	196.106	**−10.2%**
English	**130.481**	130.530	**−0.04%**

- English is also shorter: ASCII keeps 1 symbol; 3-byte characters such as em dashes and curly quotes (0.037%) go from **3 symbols to 2**.
- The same context window holds **1.41–1.50×** as much text in these 39 languages.
- With MYTE's fixed morphology maps [5]: **4.58%** fewer positions than MYTE and the same decoded output; 201/204 languages shorter, none longer.

UTF-8 where it is compact,
UTF-16 where it is not.
Lower disparity for BPE tokenizers; up to one-third shorter inputs for byte-level models.

[1] Kim et al., BBPE16, ICASSP 2026. [2] Wang et al., Neural machine translation with byte-level subwords, AAAI 2020. [3] Liu et al., SuperBPE, COLM 2025. [4] Land and Arnett, SCRIPT-BPE, Tokenization Workshop 2025. [5] Limisiewicz et al., MYTE, ACL 2024. [6] Hayase et al., Data mixture inference, NeurIPS 2024. [7] Xue et al., ByT5, TACL 2022.