Natural Language Processing

Tokenisation fairness

Marco Kuhlmann
Department of Computer and Information Science

LINKÖPING UNIVERSITY

Fairness in AI

- **Fairness** in AI concerns whether model behaviours or outcomes systematically advantage or disadvantage certain users or groups.

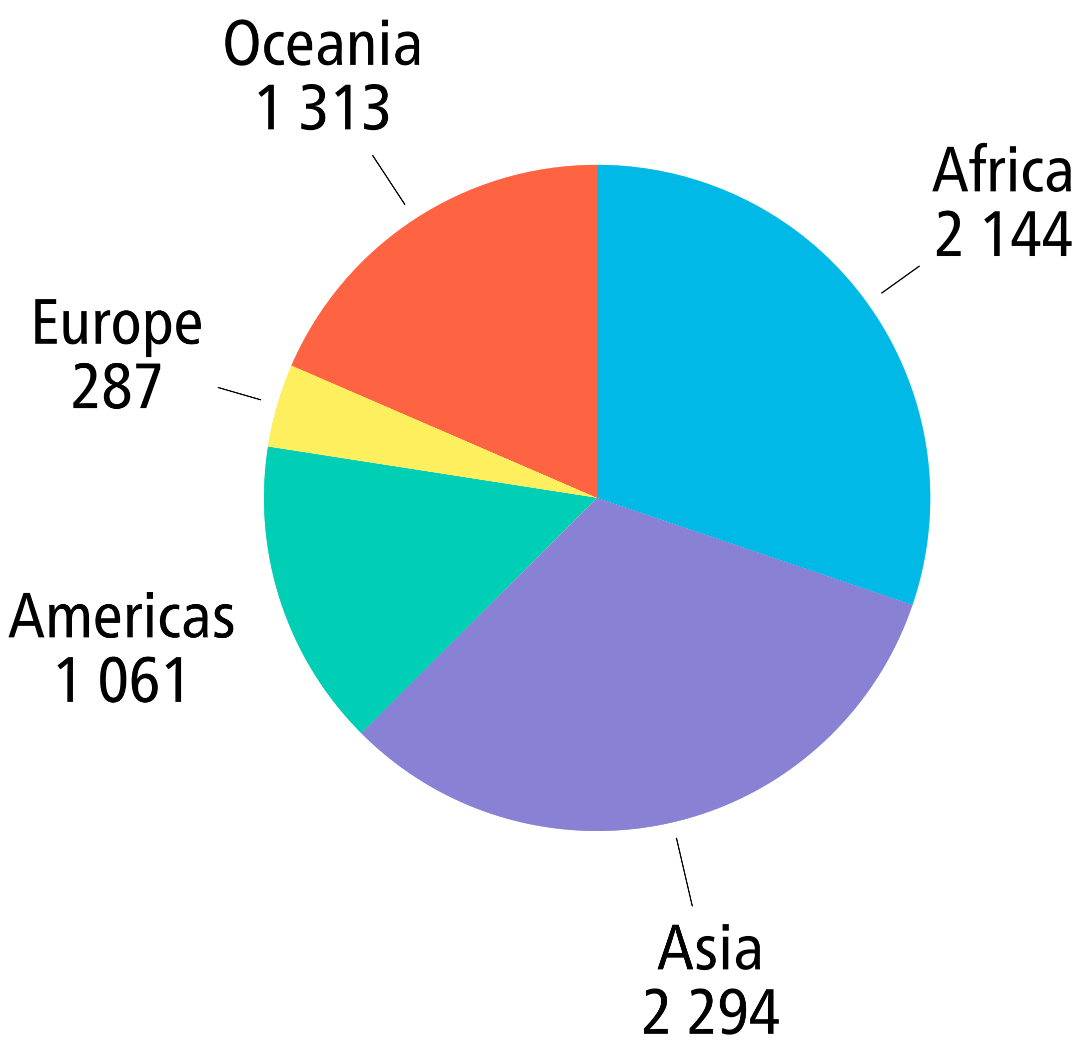
  gender, ethnicity, socioeconomic status, language, …

- Bias can enter AI systems at multiple stages, making fairness a system-level property rather than a single fix.

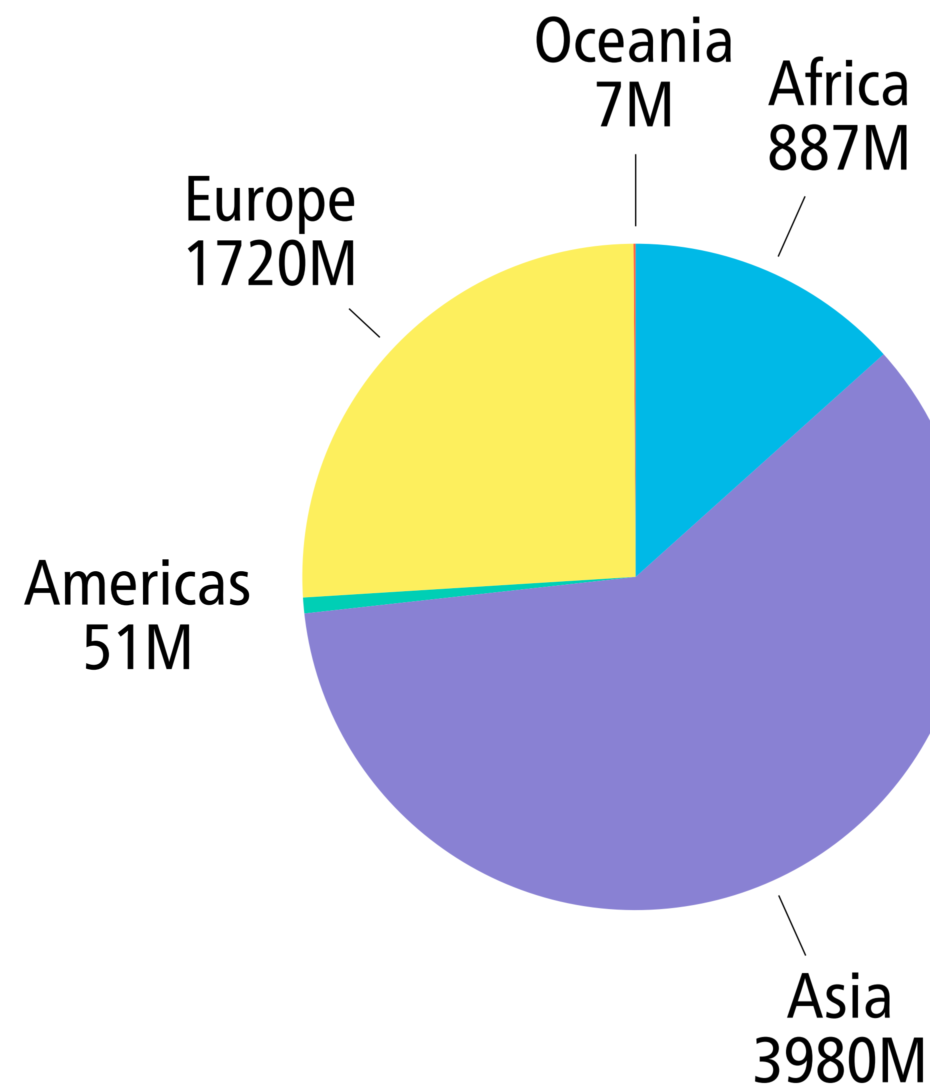
  data collection, model architecture, optimisation objectives, deployment

- Fairness is contextual and normative. It requires explicit choices rather than a one-size-fits-all technical definition.

Languages of the world



Languages by region of origin



Population by region of origin

Data taken from Ethnologue

Tokenisation premiums

Select language:
English

Sentence:

These websites have gotten a lot of attention, especially in the education setting.

15 tokens, 0% characters mapped to the UNK token:

These websites have gotten a lot of attention, especially in the education setting.

Token IDs:

9673133356171745426427633156666115423304279687363761313

Select language:
Swedish

Sentence:

Dessa webbplatser har fått mycket uppmärksamhet, särskilt inom utbildningsområdet.

29 tokens, 0% characters mapped to the UNK token:

Dessa webbplatser har fått mycket uppmärksamhet, särskilt inom utbildningsområdet.

Token IDs:

3526577356665434398054960397055683808617095298143049274714122112741430449913036304316879161465395693169802119842 13

In GPT-4, the tokenisation length for Swedish is 1.58 times that of English.

Petrov et al. (2023)

Tokenisation premiums

For Shan, the factor is 15.05.

"Tokenisation premium" relative to English (Petrov et al., 2023) – Link

Language	GPT-4	GPT-2	BLOOM
Spanish	1.55	1.99	1.21
Swedish	1.58	1.95	1.65
German	1.58	2.14	1.68
Chinese (Simplified)	1.91	3.21	0.95
Icelandic	2.15	2.43	1.99
Standard Arabic	3.04	4.40	1.14
Hindi	4.79	7.46	1.28
Shan	15.05	18.76	12.06

Tokenisation (un)fairness

Petrov et al. (2023)

- **Higher latency:** Users of disadvantaged languages have to wait longer for the same content to be processed.
- **Higher cost:** Commercial LLM services charge per token. Users of disadvantaged languages pay more for the same task.

  GPT-5.2: $1.75/1M tokens (input), $14/1M tokens (output)
- **Lower quality:** Current models have a fixed-size context window. Users of disadvantaged languages get less quality.

Unfairness due to variable-length encoding

- BPE tokenisation is usually applied to byte sequences coming from the UTF-8 encoding of Unicode characters.
- UTF-8 uses a variable-length encoding, where a Unicode character is represented by one or several bytes.
- This encoding scheme penalises languages written in scripts with high codepoints in the Unicode standard.

  English → 1 byte per character, Shan → 3 bytes per character

Block-structured encoding

Land and Arnett (2025)

Unicode codepoints:

a	Ω	书	🤗

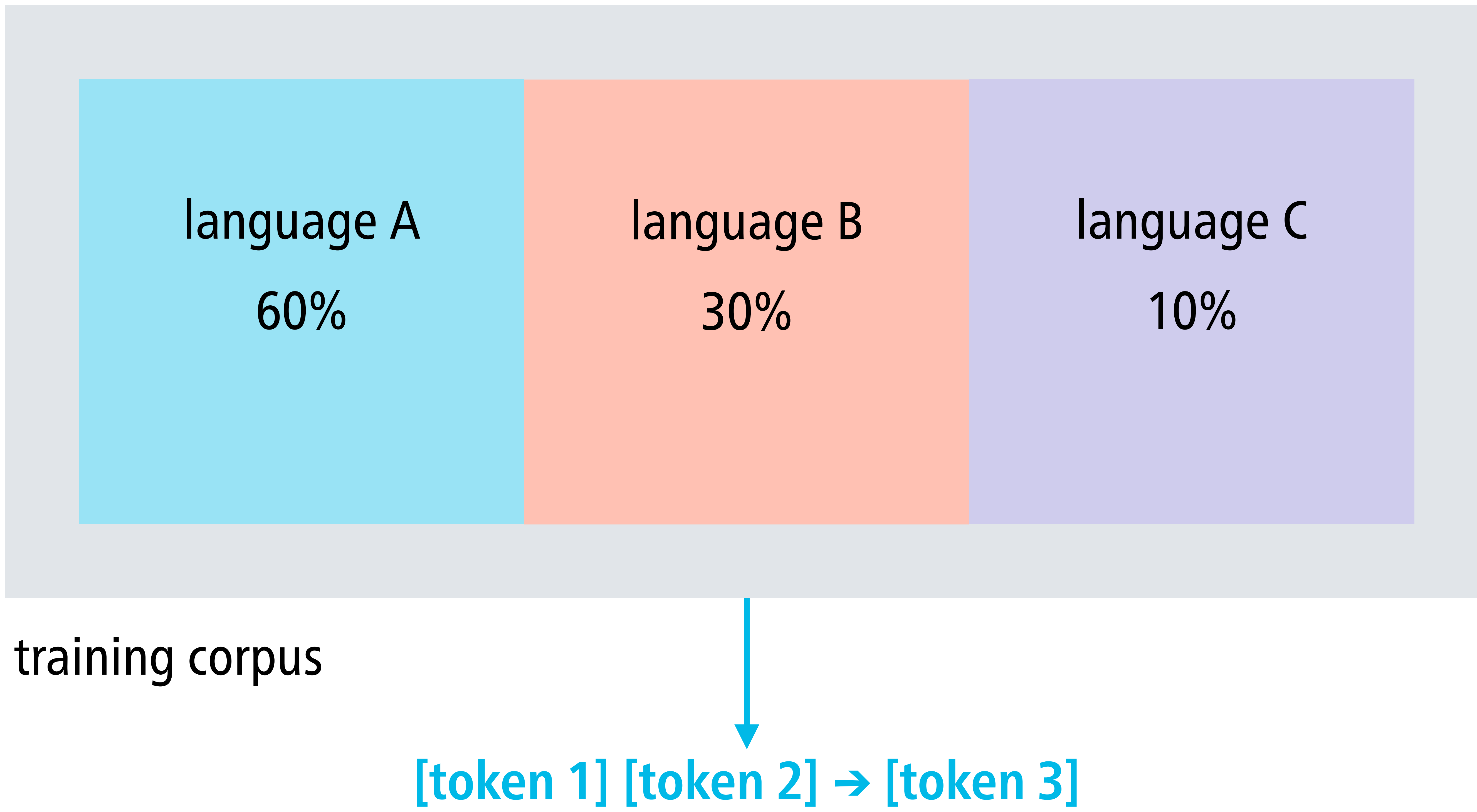
UTF-8 bytes:

61	CE A9	E4 B9 A6	F0 9F A4 97

The proposed SCRIPT encoding maps each codepoint to a block token and an index token:

L 7	EL 45	HS 912	EH 429

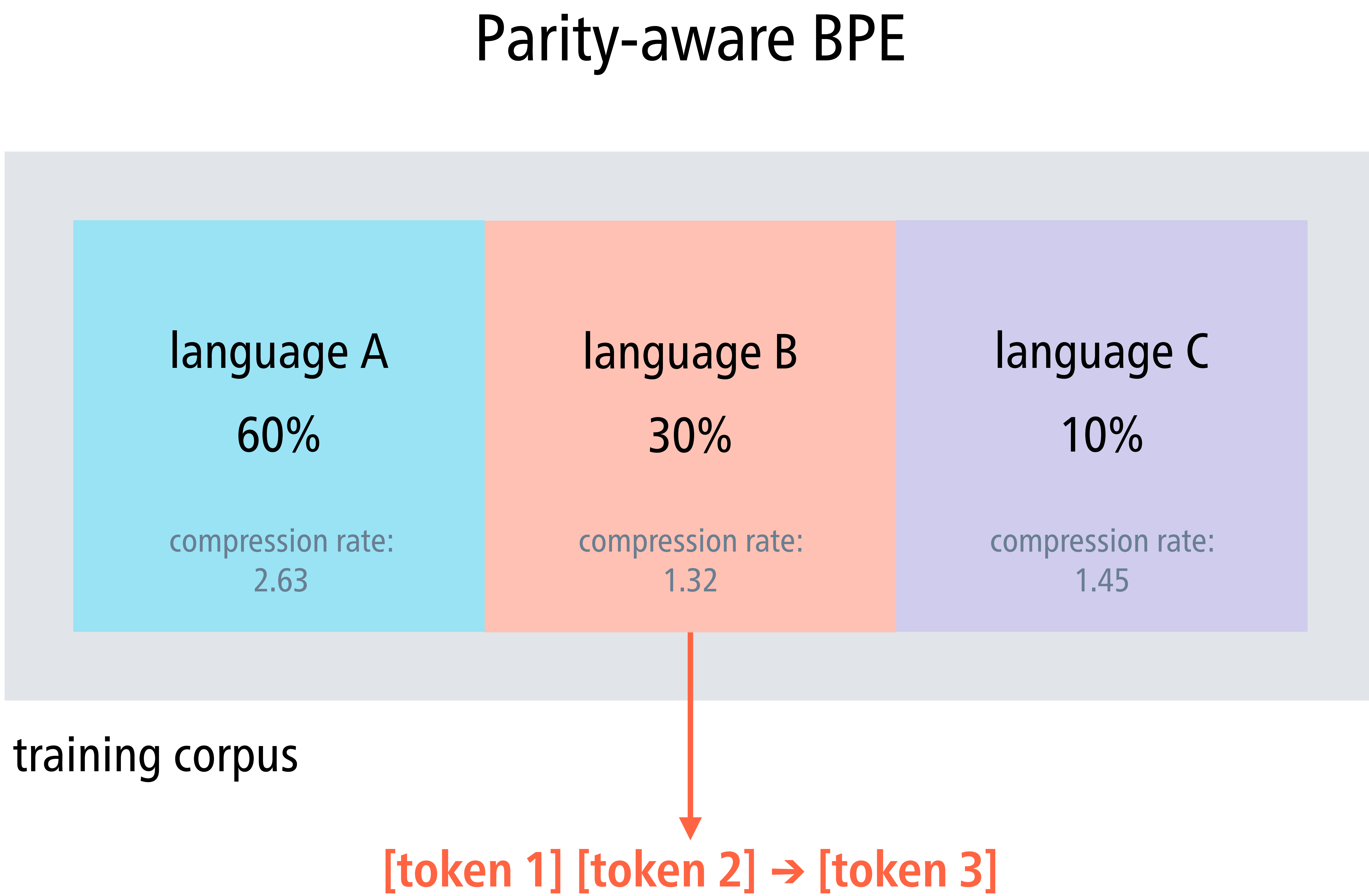
BPE penalises low-frequency languages

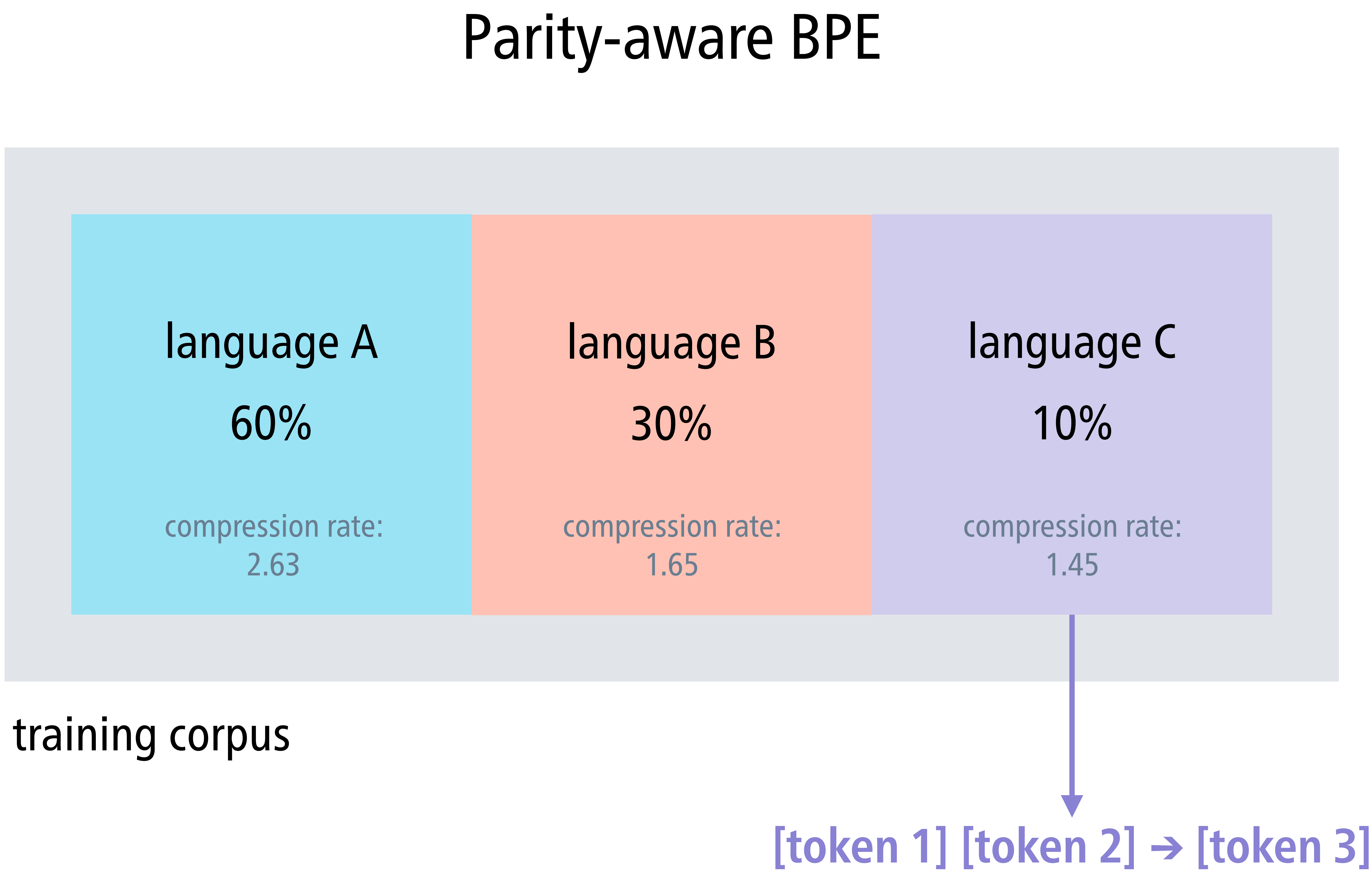


Parity-aware BPE

Foroutan et al. (2025)

- BPE learns merge rules based on the most frequent pairs in the complete corpus, implicitly favoring well-represented languages.
- **Parity-aware BPE** adds the merge rule that most improves the language with the currently worst tokenisation efficiency.
- Concretely, we find the next merge rule in the sub-corpus for the language with the currently lowest **compression rate**.
- This rule is then applied to the full corpus, as in standard BPE.


Parity-aware BPE
language A
60%
compression rate:
2.63
language B
30%
compression rate:
1.32
language C
10%
compression rate:
1.45
training corpus
[token 1] [token 2] ➔ [token 3]


Parity-aware BPE
language A
60%
compression rate:
2.63
language B
30%
compression rate:
1.65
language C
10%
compression rate:
1.45
training corpus
[token 1] [token 2] → [token 3]

Compression rate

Foroutan et al. (2025)

- Text can be decomposed at many granularities. Here we assume that tokeniser inputs are represented as **byte-strings**.
- The **compression rate** of a byte-string b is the ratio between the lengths of the original and the tokenised version of b:

$$\mathrm{CR}(b) = \frac{|b|}{|\mathrm{tokenize}(b)|}$$

- We are generally interested in a tokeniser's average compression rate, which can be estimated on a text corpus.

Summary

- Tokenisers encode explicit or implicit decisions about which languages get technology that is fast, cheap, and expressive.
- Fairness metrics are not just post-hoc evaluations, but can be optimised during training.
- While there are several proposals for how to address tokenisation unfairness, the problem is not "solved".

  new trade-offs; many layers of linguistic inequality