No Spaces? No Problem! Segmenting Complex Scripts with Machine Learning

Unicode Technology Workshop 2025

Hosted at Microsoft Silicon Valley Campus

Shane Carr, Chair of ICU4X TC

November 11-13

UNICODE

Shane Carr

Convener, ECMA TC39-TG2

Chair, Unicode ICU4X-TC

Internationalization Engineering @ Google

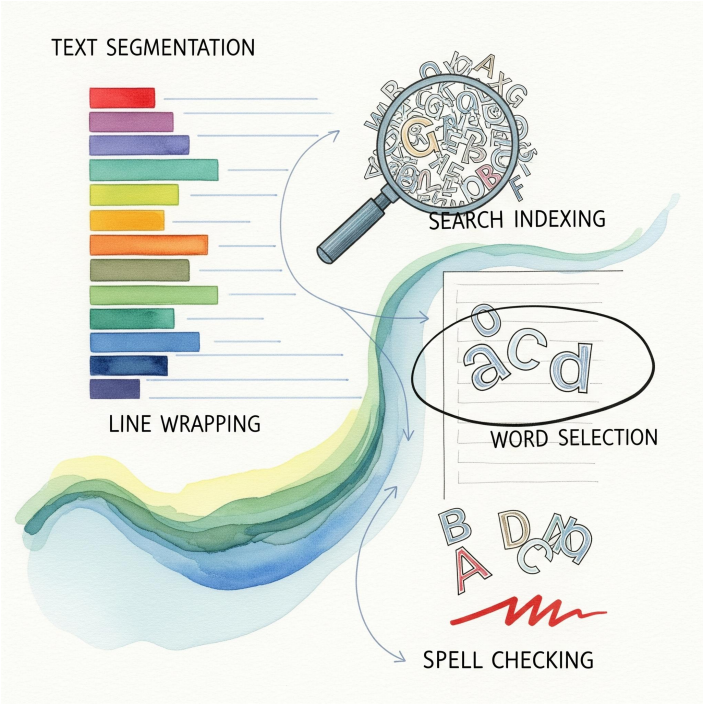
(this talk is not on behalf of Google)

What is text segmentation?

- ➔ Dividing a continuous sequence of text into meaningful units
- ➔ Four units Unicode supports:
  - ⇢ Grapheme Clusters
  - ⇢ Words
  - ⇢ Sentences
  - ⇢ Lines
- ➔ Not currently supported, but maybe in the future:
  - ⇢ Hyphenation
  - ⇢ Phonemes
  - ⇢ Syllables

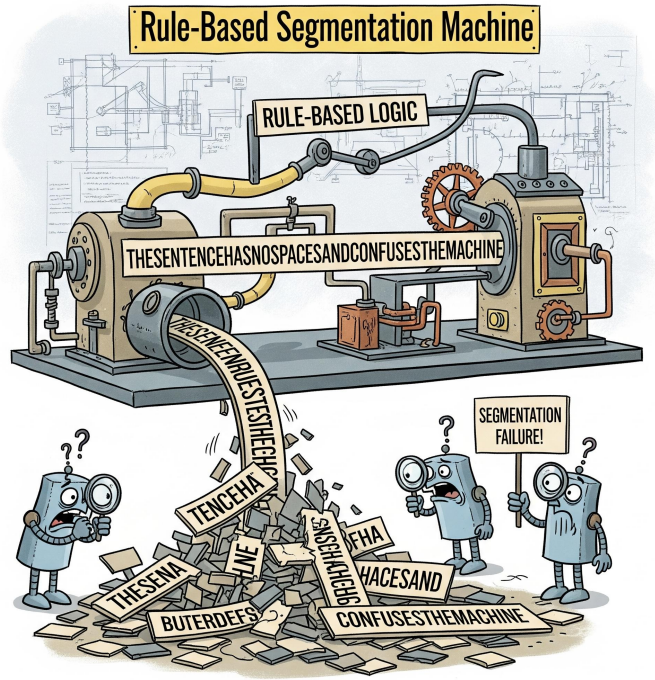
Why do you need text segmentation?

- ➔ Improves text display with proper **line wrapping**
- ➔ Essential for **search indexing** and accurate results
- ➔ Enables accurate **selection** of whole words in text
- ➔ Supports linguistic analysis and **spell checking**



What makes text segmentation tricky across scripts and languages?

- ➔ Many languages do not use spaces between words
- ➔ Scripts can be used for many different languages
- ➔ Rule-based methods cannot understand a whole language's vocabulary to find word boundaries



What languages require custom models for text segmentation?

For line and word breaks:

- ➔ Thai
- ➔ Khmer
- ➔ Shan
- ➔ …

For word breaks:

- ➔ Chinese
- ➔ Japanese
- ➔ Javanese
- ➔ …

UNICODE

~2 Billion people use languages that require custom models for text segmentation.

What is dictionary-based segmentation?

- ➔ Segmentation uses a predefined list of words
- ➔ System finds the longest possible match in the dictionary
- ➔ The process repeats from the end of the found word

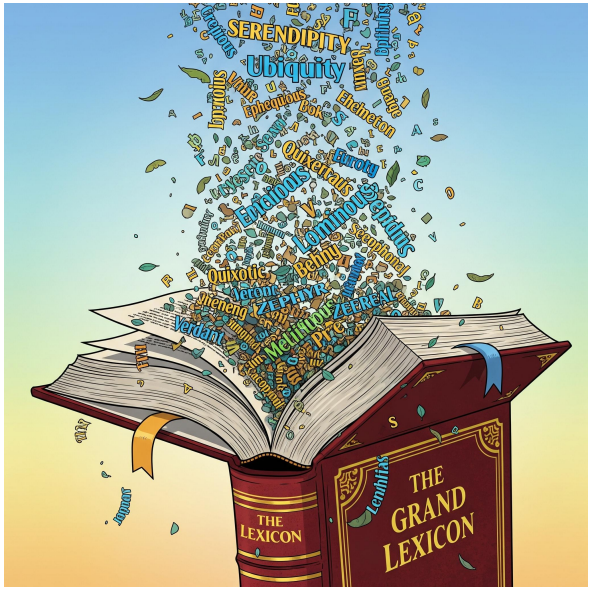
🚚 Available in all ICUs

UNICODE

Why does dictionary-based segmentation fall short?

🌟 Key Point! 🌟

- ➔ Dictionaries are too large to ship on low-resource devices
- ➔ New or specialized words are not easily recognized
- ➔ Longest match can fail by missing correct shorter words



Two broad cases needing different solutions

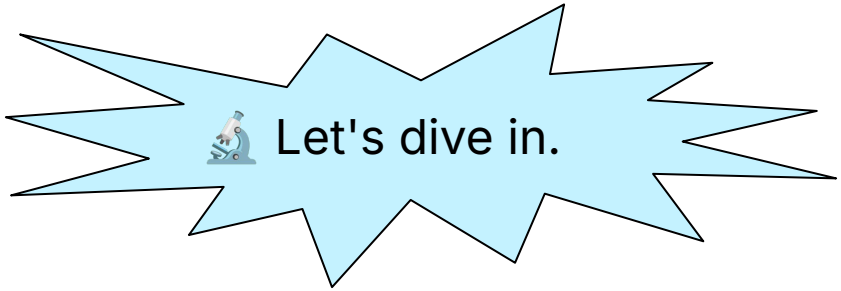
🌟 Key Point! 🌟

South-East Asian (SEA)

- A hundredish code points per language/script
- Words are often 3-7 code points in length

East Asian (CJK)

- Tens of thousands of unique code points
- Words are often 1-3 code points in length



East Asian (CJK) Text Segmentation

Approach 1: BudouX AdaBoost

A tiny open-source library from Google that segments CJK text by:

1. Looking around each possible cut point in a sentence
2. Collects simple “n-gram” patterns (single chars, pairs, triples).

AdaBoost: many tiny rules each “vote” on whether a break is good; combined votes decide word boundaries

Approach 2: RAdaBoost

Radicals (偏旁部首) are the **core components** of Han characters and often indicate a character's meaning or category.

For example,「 氵」refers to water, as in 河 river or 海 sea.

Since certain radicals frequently appear together, they provide useful cues for **word segmentation**. AdaBoost is well suited for this task, as it assigns different weights to radical pairs during training, emphasizing the most informative ones.

🧠 Thanks, Shenghong Liu!
🚧 Experimental code in ICU4X

AdaBoost Learners (taken into account)

Learners \ Model	BudouX	RAdaBoost
UW (A, F)	✔️	❌
UW (B, C, D, E)	✔️	✔️
BW (BC, ED)	✔️	❌
BW (CD)	✔️	✔️
TW (ABC, BCD, CDE, DEF)	✔️	❌
RAD (C'D', C'D, CD')	❌	✔️

At boundary between C and D

* UW, BW and TW refers to single, two and three characters respectively
* C' and D' are the radicals of C and D respectively.

CJK Model Comparison: Data Size

ICU Dictionary	2.0 MB
BudouX zh-hant	64 KB
BudouX zh-hans	63 KB
Radical (all zh variants)	60 KB

CJK Model Comparison: Accuracy Scenarios

Model	Cantonese	Traditional Chinese	Traditional Chinese No Punctuation	Simplified Chinese	Simplified Chinese No Punctuation
ICU Dict	79.2	90.9	89.5	91.5	91.2
Canton Dict	94.9	91.7	90.4	-	-
BudouX	73.4	81.4	86.3	81.4	87.3
RAdaBoost	92.5	90.1	86.5	91.1	87.7

- All cells show the F1 score
- The Cantonese Dictionary comes from PyCantonese
- BudouX models: Hant for columns 1-3, Hans for column 4-5

UNICODE

CJK Model Comparison: Performance

zh-hant (CITYU Test Dataset)	F1-Score / %	Latency (Relative)
ICU Dictionary	90.9	1x
BudouX zh-hant	81.4	10.6x
RAdaBoost	90.1	6.1x

zh-hans (PKU Test Dataset)	F1-Score / %	Latency (Relative)
ICU Dictionary	91.5	1x
BudouX zh-hans	91.1	10.6x
RAdaBoost	81.4	6.1x

"Latency" refers to inference time (lower is better)

CJK Model Comparison: Text Sample

English: My friends and I had dim sum together at a teahouse in Guangzhou.
zh-hant: 我和朋友在廣州茶樓一起喝早茶。

Dictionary:	我|和	|朋友|在|廣州	|茶樓|一起|喝	|早|茶|。|
Radical:	我|和	|朋友|在|廣州	|茶樓|一起|喝	|早茶|。|
BudouX:	我|和	|朋友|在|廣州	|茶樓|一起|喝	|早茶。|

➔ BudouX and dictionary can be affected by unseen vocabulary

CJK Model Comparison: Text Sample

English: My friends and I had dim sum together at a teahouse in Guangzhou.
zh-hant-yue: 我同朋友喺廣州茶樓一齊飲早茶。

Dictionary:	我|同	|朋友|喺|廣州	|茶樓|一齊|飲	|早|茶|。|
Radical:	我|同	|朋友|喺|廣州	|茶樓|一|齊|飲	|早茶|。|
BudouX:	我|同	|朋|友喺廣州	|茶樓|一|齊|飲	|早茶。|

➔ BudouX and dictionary can be affected by unseen vocabulary
➔ This can be further shown using Hong Kong (Cantonese) text

CJK Model Comparison: Text Sample

English: My friends and I had dim sum together at a teahouse in Guangzhou.
zh-hans-yue: 我同朋友喺广州茶楼一齐饮早茶。

Dictionary: 我|同 |朋友|喺|广州 |茶楼|一齐|饮 |早|茶|。|
Radical: 我|同 |朋友|喺|广州 |茶楼|一|齐|饮 |早茶|。|
BudouX: 我|同 |朋友喺 |广州 |茶|楼|一|齐饮 |早|茶。|

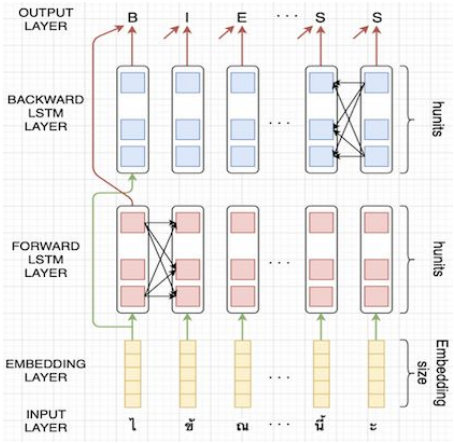
➔ BudouX and dictionary can be affected by unseen vocabulary
➔ N-grams inconsistencies can be seen using Simplified Cantonese text

South-East Asian (SEA) Text Segmentation

Approach 1: LSTM

Bi-directional ML model that use both past and future context before making a decision to segment words.

Pros: learns patterns that fixed rules often miss
Cons: inherently sequential, so bottlenecks both throughput and latency on large texts

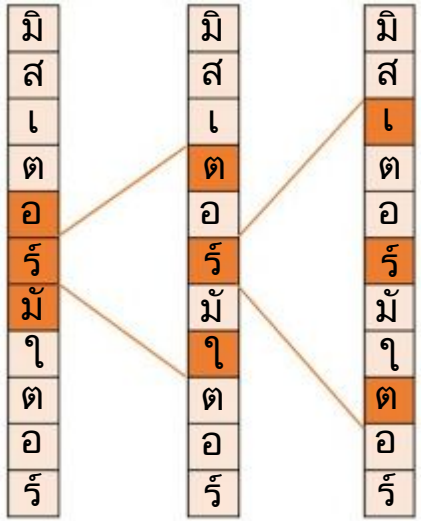


🧠 Thanks, Sahand Farhoodi!
🚚 Available in ICU4X, and ICU via build option
🦊 Currently in production in Firefox!

Approach 2: CNN

The new architecture uses two parallel convolutional layers. Having one standard and another dilated balances **detail** and **broader context** without adding extra computation.

This enabled us to **lower latency** and model size while boosting performance.



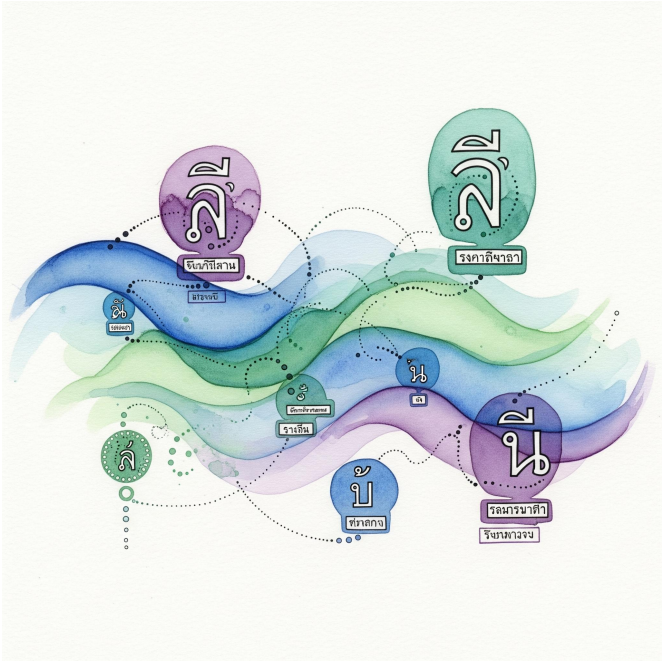
🧠 Thanks, Shenghong Liu!
🚧 Experimental code in ICU4X

Approach 3: BudouX

AdaBoost models were designed for CJK, but we can see how they work with Thai.

SEA Model Comparison: Data Size

Thai	Size
ICU Dictionary	126-225 KB
LSTM - Medium	36 KB
CNN - Medium	28 KB
BudouX	33 KB



SEA Model Comparison: Performance

BEST 2019 Dataset	F1-Score / %	Relative Latency	Relative Latency, Length x1000
ICU Dictionary	86.4	1x	1x
LSTM - Medium	90.1	762x *	4336x
CNN - Medium	90.4	254x	210x
BudouX	82.3	27x	45x

** In Rust, we've reduced this to ~300x*

➔ CNN scales better than LSTM to longer chunks of text
➔ AdaBoost is competitive; good area for future work

UNICODE

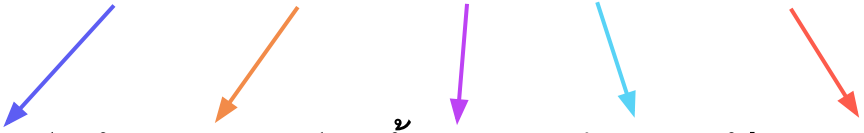
The dictionary remains substantially faster than the ML models.

ICU4X offers both so that clients can weigh size, speed, and accuracy.

Example factors: server or client, layout or indexing, throughput.

SEA Model Comparison: Text Sample

English: The three Mr. Mumbles bent forward and listened eagerly.
Thai: มิสเตอร์มัมเบิลส์ทั้งสามก้มลงฟังอย่างตั้งใจ



Dict:	มิสเตอร์ |มั|ม|เบิ|ล|ส์	|ทั้ง|สาม	|ก้ม|ลง	|ฟัง	|อย่าง	|ตั้งใจ
LSTM:	มิสเตอร์ |มัม|เบิลส์	|ทั้ง|สาม	|ก้ม|ลง	|ฟัง	|อย่าง	|ตั้งใจ
CNN:	มิสเตอร์ |มัม|เบิลส์	|ทั้งสาม	|ก้ม|ลง	|ฟัง	|อย่าง	|ตั้งใจ
Budou:	มิสเตอร์ มัมเบิลส์	|ทั้ง|สาม	|ก้ม|ลง	|ฟัง	|อย่าง	|ตั้งใจ

- ➔ Phonetic loan words are challenging for all of the models
- ➔ The ML models learned "มัม", or "mum" (mother) [citation]
- ➔ "ทั้งสาม" means "all three" as in "both" but for three things

Next Steps

Call to Action

Let's keep in touch!
shane@unicode.org

ML Segmentation is a high-impact project, still with room to grow.

What you can do:

1. Use the ML models and provide feedback
2. Data collection to improve quality
3. Explore hybrid ML/dictionary models
4. Training on multilingual data
5. Engineering work on implementations
6. Project management

Thank you to our Organizational Members

Thank you to our Industry and Media Partners

PLEASE READ FIRST

(1) PLEASE MAKE A COPY OF THIS DECK - and then name your file as follows:

YEAR_MO_Last NameFirstName_Event

Example: 2023_10_SmithJack_UTW

(2) To add slides:

- (a) Go to the submenu bar.
- (b) Right next to the magnifying glass is a plus sign (+)
- (c) Select the down arrow (▾) next to the plus sign
- (d) Insert the slide formats that are most useful for your presentation.

(3) Include thank you slide for Org Members and Industry and Media Partners (slide 18 - updated 08/08/25)