Phonemizing User-Generated Text: A Benchmark, Taxonomy, and Compositional Approach

MinJu Jeon[1,2]*, Younghan Park[3], Han Sung Park[4], Jong-Hwan Kim[1], Dong-Jin Kim[2]†, Hoyeon Lee[1]†

[1]NAVER Cloud, [2]Hanyang University,
[3]Carnegie Mellon University, [4]Georgia Institute of Technology

Abstract

Text-to-speech systems increasingly process user-generated text (UGT) such as *ppl* and *imo*, whose pronunciation must be inferred from the canonical rather than surface form. We introduce UGTPHON, the first grapheme-to-phoneme (G2P) benchmark for UGT in English, Vietnamese, and Korean, together with an inference-grounded taxonomy for fine-grained diagnosis. Existing G2P models and frontier LLMs exhibit a systematic canonical-to-non-canonical performance gap, reaching up to 66.8 PER points. As a benchmark baseline, we propose a simple compositional G2P approach that incorporates canonical-form evidence through exact-match lookup and staged decoding. Across matched ByT5 and Qwen2.5-0.5B backbones, explicit canonical-form modeling consistently reduces non-canonical G2P errors. The 0.5B variant also performs competitively with much larger few-shot frontier LLMs, highlighting the benefit of explicitly modeling canonical-form inference for UGT phonemization.

1 Introduction

Grapheme-to-phoneme (G2P) conversion (Bisani and Ney, 2008; Novak et al., 2012; Rao et al., 2015) maps a word's written form to its phonemic transcription. It serves as the essential front-end of text-to-speech (TTS) systems (Ren et al., 2019; Ploujnikov and Ravanelli, 2022), where synthesis quality depends directly on the pronunciations it produces (Taylor, 2005; Van Esch and Sproat, 2017). Existing G2P approaches, including joint-sequence models (Bisani and Ney, 2008), transformer architectures (Yolchuyeva et al., 2019), and pretrained byte-level models (Zhu et al., 2022), achieve robust performance on canonical lexicons such as IPA-Dict (Doherty, 2016) and WikiPron (Lee et al., 2020). TTS pipelines, therefore, treat phonemization as a reliable upstream module (Kim et al., 2021; Shen et al., 2018), assuming input that follows standard orthography.

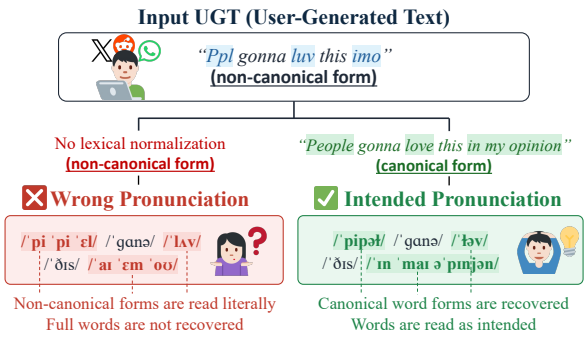

Figure 1: Conventional G2P models (red, left) read non-canonical words (*ppl*, *luv*, *imo*) letter by letter, producing letter names. Accurate phonemization (green, right) therefore requires recovering the canonical form (*people*, *love*, *in my opinion*) before generating phonemes.

However, this assumption breaks down in practice. As TTS systems increasingly process conversational and web-based text (Lee and Ashok, 2022), input is dominated by *user-generated text* (UGT) (Plank, 2016; Han and Baldwin, 2011): a mixture of canonical words (e.g., *love*, *people*) and non-canonical forms like *luv* and *ppl*. These non-canonical forms follow systematic patterns recurring across languages such as English (EN), Vietnamese (VI), and Korean (KO): abbreviations (EN: *lmao*, VI: *vcl*, KO: ㅇㄱㄹㅇ), orthographic lengthening (*cooool*, *điiiii*, ㅋㅋㅋㅋㅋㅋ), phonetic substitutions (*gr8*, *wá*, 머해), and slang (*lit*, *gato*, 킹받다). Existing G2P models therefore read these forms literally and yield letter-name strings rather than the intended pronunciations (Figure 1 (A)). Prior work has framed UGT as noise (Zhao et al., 2022), but UGT is fundamentally different: *imo* is a deliberate substitution whose pronunciation must be inferred from its canonical expansion

*Work performed during an internship at NAVER Cloud.
†Corresponding authors.

Findings of the Association for Computational Linguistics: EMNLP 2026, pages 21564–21581
October 24-29, 2026 ©2026 Association for Computational Linguistics

(*in my opinion*), not its surface form. G2P robustness under UGT remains unexplored.

To address this gap, we introduce UGTPHON[1], the first G2P benchmark for UGT, spanning three typologically diverse languages (English, Vietnamese, and Korean) with the intended canonical form and gold phoneme sequence for every non-canonical word.

To support fine-grained diagnosis, we further group non-canonical words by the inference a G2P model must perform: **(1) Reconstruct** (recover dropped characters, e.g., *idk* for *I don't know*), **(2) Repetition** (collapse prosodic repetitions, e.g., *soooo* for *so*), and **(3) Pass-through** (the surface already encodes the pronunciation, e.g., *luv* for *love*). This inference-grounded design lets a model's failure pattern indicate which kind of inference is missing, not which inputs it fails on.

Non-canonical G2P composes two stages, inferring the canonical form and phonemizing it. Standard end-to-end training collapses these into one, while a naive normalize-then-G2P pipeline overcorrects forms that need no rewriting. We therefore use a **simple compositional G2P baseline** with three components: *Contextualized input encoding* marks the target word within its sentence, *Retrieval* supplies a canonical-form hint from an external datastore, and *Staged decoding* emits the canonical form as an intermediate target before its phonemes (<Target> G_c <Phoneme> P_c).

On UGTPHON, all baselines suffer degradation on non-canonical words, with the canonical–non-canonical gap reaching up to 66.8 PER points. Further training on UGTPHON substantially lowers non-canonical error rates on English and Vietnamese but leaves a large gap. Our compositional baseline consistently reduces non-canonical G2P error relative to matched ByT5 and Qwen2.5-0.5B backbones across all three languages, while the 0.5B variant is also competitive with much larger few-shot frontier LLMs.

We summarize our contributions as follows:

- **UGTPHON**, the first G2P benchmark for user-generated text across English, Vietnamese, and Korean, with an inference-grounded taxonomy for fine-grained diagnosis.
- **A systematic evaluation** showing how canonical and non-canonical G2P errors vary across inference types, model families, and languages.
- **A simple compositional baseline** that injects canonical-form evidence through exact-match lookup and staged decoding, consistently improving matched supervised backbones on UGTPHON.

[1]The UGTPHON is publicly available at https://github.com/naver-ai/UGTPHON.

2 UGTPHON

We propose UGTPHON, the first G2P benchmark for UGT, comprising 18,373 sentences across three typologically diverse languages (Table 2): English (5,029), Vietnamese (10,847), and Korean (2,497). We formalize G2P on UGT (Section 2.1), group non-canonical words by required inference into a diagnostic taxonomy (Section 2.2), and annotate $(\mathrm{G}_{nc}, \mathrm{G}_c, \mathrm{P}_c)$ at the target level within full sentential context (Section 2.3). This supports aggregate evaluation and category-level diagnosis of G2P failures.

2.1 Task Formulation

Notation. Let S denote a sentence and w_i an annotated target token or span in S, which is either canonical or non-canonical. In Vietnamese, a single annotated target may span multiple whitespace-delimited units. A canonical word has a spelling listed in the standard dictionary (e.g., *love*, *please*, *không*). A non-canonical word G_{nc} is a conventionalized variant of a canonical form arising systematically in UGT (e.g., *luv*, *pls*, *k*). We denote the canonical form of G_{nc} as G_c and the corresponding gold IPA phoneme sequence as P_c. For each annotated target, we represent the G2P task as a quadruplet $(S, \mathrm{G}_{nc}, \mathrm{G}_c, \mathrm{P}_c)$, where the non-canonical form G_{nc} appears as a token or span in sentence S.

Canonical form annotation. We adopt the standard formulation of lexical normalization (Van Der Goot et al., 2018, 2021; Nguyen et al., 2024): the mapping $\mathrm{G}_{nc} \rightarrow \mathrm{G}_c$ admits 1-to-1 (*pls* $\rightarrow$ *please*), 1-to-N (*idk* $\rightarrow$ *I don't know*), and N-to-1 (*ch ó* $\rightarrow$ *chó*) mappings. We obtain G_c from LexNorm2015 (Baldwin et al., 2015), via MultiLexNorm (Van Der Goot et al., 2021), for English and from ViLexNorm (Nguyen et al., 2024) for Vietnamese. For Korean, we draw G_{nc} from MultiLexNorm++ (Buaphet et al., 2026) and KoMultiText (Choi et al., 2023), and annotate G_c from scratch with native-speaker experts. (procedure in Appendix A.2).

Type	ID	Category	English		Vietnamese		Korean	
			$G_{nc} \rightarrow G_c$	tr+dv / te	$G_{nc} \rightarrow G_c$	tr+dv / te	$G_{nc} \rightarrow G_c$	tr+dv / te
A *Reconstruct*	A1	Shortening-V	pls → please	595/166	đc → được	3,112/327	ㅅㄱ → 수고	451/67
	A2	Shortening-O	bc → because	313/82	k → không	4,609/496	–	–
	A3	Phrasal (Unp)	idk → I don't know	504/117	mn → mọi người	2,020/225	–	–
B *Repetition*	B1	Lengthening	soooo → so	175/75	điii → đi	475/43	구우웃 → 굿	166/66
	B2	Iteration	waitwait → wait wait	113/44	xem xem → xem xem	157/38	ㅋㅋㅋㅋ → 크크크크, 키키키키	322/136
C *Pass-through*	C1	Eye Direct	luv → love	515/140	zô → vô	2,385/368	어케 → 어케	509/119
	C2	Regular	droppin → dropping	1,559/405	iu → yêu	2,164/247	걍 → 걍	175/44
	C3	Slang/Teen	tho → though	903/236	hong → không	2,897/369	막짤 → 막짤	1,205/234
	C4	Phrasal (Pro)	rofl → rofl	166/62	GATO → GATO	129/39	윌클 → 윌클	89/56

Table 1: Diagnostic taxonomy of non-canonical UGT in UGTPHON, regrouping prior orthographic taxonomies by the inference G2P must perform: three types ordered by inference demand, with nine fine-grained categories. **(A) Reconstruct**: recover dropped characters. **(B) Repetition**: collapse repeats. **(C) Pass-through**: surface already encodes pronunciation. Each cell: example $\mathrm{G}_{nc} \rightarrow \mathrm{G}_c$ and counts (train+dev / test); "–": absent.

	English	Vietnamese	Korean	Total
Train	3,329	8,597	1,976	13,902
Dev	600	1,129	246	1,975
Test	1,100	1,121	275	2,496
Total	5,029	10,847	2,497	18,373

Table 2: UGTPHON composition by language and split.

Task definition. Existing G2P benchmarks evaluate word-level conversion in isolation (Bisani and Ney, 2008; Novak et al., 2012; Rao et al., 2015). This is insufficient for UGT: a non-canonical surface like *ig* can normalize to *I guess* or *Instagram* depending on the sentence. UGTPHON therefore situates every non-canonical word within its original sentence S. Given G_{nc} and S, the task is to predict the gold phoneme sequence P_c derived from the intended canonical form G_c, which remains latent during evaluation. Evaluation is at the target level for precise error attribution, while conditioning on the full sentence.

2.2 Diagnostic Taxonomy of Non-Canonical Words

Non-canonical surfaces in UGT vary in the inference required for G2P: some can be read directly, others require recovering dropped characters, and others require collapsing repeated units. Existing UGT taxonomies (Van Der Goot et al., 2018) group surfaces by orthographic transformation, which is suited to lexical normalization but not phonemization. UGTPHON instead regroups these surfaces by the inference required for phonemization (Table 1), so that failure patterns localize *which kind of inference* a model lacks.

Coarse-grained types. We define three coarse types, ordered from the most to the least demanding inference.

(A) Reconstruct. Surface content required for pronunciation has been omitted and must be recovered, possibly as a multi-word expansion (e.g., *idk* → *I don't know*).

(B) Repetition. Characters or units have been repeated for prosodic emphasis (Brody and Diakopoulos, 2011) and must be collapsed (e.g., *soooo* → *so*).

(C) Pass-through. The surface already preserves the intended pronunciation and can be read directly (e.g., *luv* → *love*).

Fine-grained categories. Within each coarse type, we distinguish 2–4 fine-grained categories reorganized from prior UGT taxonomies (Van Der Goot et al., 2018). **Reconstruct** splits by what is omitted (vowels A1, other characters A2, full phrases A3). **Repetition** splits by what is repeated (characters B1, units B2). **Pass-through** splits by how pronunciation is preserved in the surface (phonetic respelling C1, productive variation C2, lexicalized slang C3, pronounceable acronyms C4). Examples and per-language counts are shown in Table 1. Korean Reconstruct collapses to A1, since it consists almost entirely of initial-consonant abbreviations (e.g., ㄱㅅ → 감사).

2.3 Data Construction

We construct UGTPHON in three stages: source corpora selection, phoneme annotation, and category labeling. We preserve the raw UGT sentence in the released text field. During phoneme construction, non-lexical elements such as punctuation and numbers are dropped or verbalized as appropriate, while unphonemizable items may remain as placeholders. Full details, licenses, and annotation procedures are provided in Appendix A.

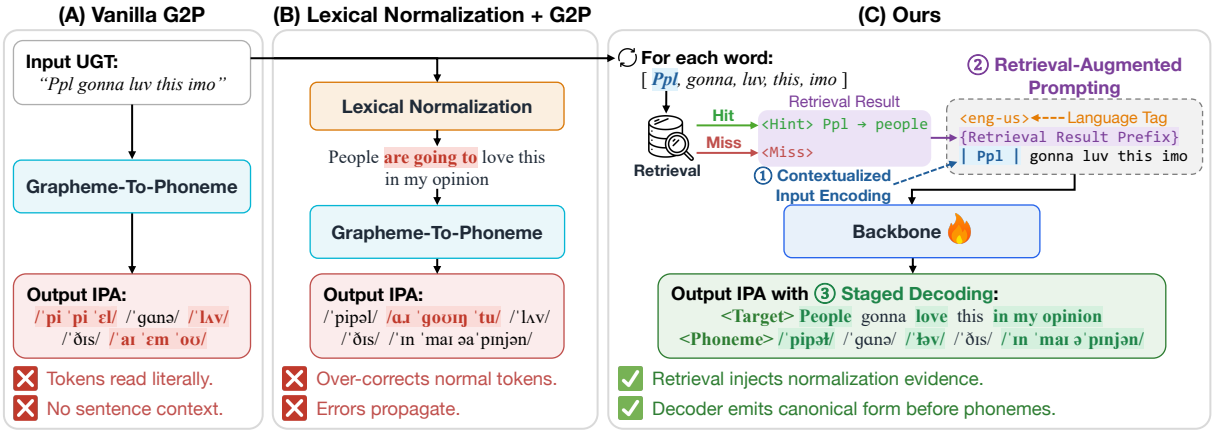

Figure 2: Three approaches to non-canonical G2P. (A) Vanilla G2P reads words literally. (B) A lexical normalization model rewrites the input but overcorrects canonical words, propagating errors downstream. (C) Our retrieval-augmented compositional G2P baseline injects a $G_{nc} \rightarrow G_c$ canonical form hint and emits G_c as an intermediate before the phonemes.

Source corpora. For English, G_{nc}–G_c pairs are derived from LexNorm (Baldwin et al., 2015), via MultiLexNorm (Van Der Goot et al., 2021). For Vietnamese, they are derived from ViLexNorm (Nguyen et al., 2024). For Korean, G_{nc} is drawn from MultiLexNorm++ (Buaphet et al., 2026) and KoMultiText (Choi et al., 2023), and G_c is annotated from scratch (Appendix A.2). In addition, the English and Vietnamese subsets include 175 and 385 human-curated augmented UGT sentences, respectively, generated from existing UGT patterns. We preserve the natural resource asymmetry across languages rather than down-sampling, ensuring sufficient samples per category in every split. Since the source corpora include UGT from social media, UGTPHON inherits some toxic content; we report a sentence-level moderation analysis in Appendix B.

Phoneme annotation. Target phonemes P_c are derived from G_c via IPA-Dict (Doherty, 2016), an expert-curated grapheme-to-IPA lexicon, and verified by native-speaker experts for language-specific phenomena (English heteronyms, Vietnamese tones, Korean phonological rules). For canonical forms G_c not covered by the lexicon (e.g., recent loanwords or proper nouns), experts provide P_c from scratch and cross-verify. Inter-annotator agreement reaches Cohen's $\kappa = 0.86$ (English) and 0.87 (Vietnamese), both in the almost-perfect range (Landis and Koch, 1977), with Korean at 0.75 (substantial) reflecting variability in IPA transcription rather than disagreement on the underlying pronunciation. Per-category agreement and OOV statistics are reported in Appendix A.4.

Category labeling. Each non-canonical word is assigned to one of the nine categories (Table 1) through manual annotation by native speakers, with second-pass verification and disagreements reconciled through discussion. Operational definitions and full guidelines are in Appendix A.3 (Table 7).

3 Method

Two conventional approaches to non-canonical G2P lack joint supervision of the normalization-phonemization composition (Figure 2). **Vanilla G2P** (Figure 2 (A)) phonemizes the surface directly, leaving $G_{nc} \rightarrow G_c$ unsupervised. **Lexical normalization + G2P** (Figure 2 (B)) decouples the stages but over-corrects canonical-recoverable surfaces. We use a retrieval-augmented compositional baseline (Figure 2 (C)) that jointly supervises both stages within a single architecture.

Our approach comprises three components. First, *Contextualized input encoding* (Section 3.1) marks the target within its sentential context. Then, *Retrieval-augmented canonical-form prompting* (Section 3.2) supplies a $G_{nc} \rightarrow G_c$ hint through an exact-match lookup when the surface form occurs in a training-derived datastore. Finally, *Staged decoding* (Section 3.3) emits G_c before P_c, binding the retrieved hint to prediction.

3.1 Contextualized Input Encoding

For each annotated target token or span w_i in sentence S, we construct the input string by prepending a language tag and surrounding w_i with pipe

markers:

$$\texttt{<lang>}\ w_1\, w_2 \cdots \texttt{|}\ w_i\ \texttt{|} \cdots w_n.$$

This format treats w_i as a single annotated target span, even when its surface or canonical form spans multiple whitespace-delimited units (e.g., *idk* → *I don't know*). Each target is processed as an independent forward pass, so multiple non-canonical words do not accumulate cross-word errors during decoding (Yoon et al., 2023).

This input format applies uniformly across training and inference, and across different backbone families: ByT5-small (Xue et al., 2022) and Qwen 2.5-0.5B (Qwen et al., 2025). Since the | symbol is a regular character in both vocabularies, no special tokens are added. Our approach is therefore backbone-agnostic and does not require backbone-specific tokenization. We use ByT5-small as the default backbone for its byte-level input, which handles UGTPHON's diverse scripts without language-specific tokenizers.

3.2 Retrieval-Augmented Canonical-Form Inference

Training a model to map G_{nc} directly to P_c provides no explicit supervision for the $\mathrm{G}_{nc} \rightarrow \mathrm{G}_c$ normalization stage: the surface form bears little phonemic correspondence to the target (*idk* vs. /aɪ doʊnt noʊ/). We augment the input with an optional canonical-form hint retrieved through a simple exact-match key–value lookup over a training-derived datastore.

Retrieval datastore. We construct the datastore $\mathcal{D}$ as a key–value map from training-set $(\mathrm{G}_{nc}, \mathrm{G}_c)$ pairs, with G_{nc} as the key and G_c as the value. For each target w_i in S, we retrieve its canonical hint from $\mathcal{D}$:

$$\text{prefix}(w_i) = \begin{cases} \texttt{<Hint>}\ w_i \rightarrow \mathcal{D}[w_i] & \text{if } w_i \in \mathcal{D}, \\ \texttt{<Miss>} & \text{otherwise.} \end{cases} \tag{1}$$

On a miss, the <Miss> marker explicitly marks retrieval failure as a learnable condition, prompting the model to rely on sentence context for normalization.

Optional LLM-based datastore expansion. New non-canonical forms appear continuously, and the training split cannot exhaust the space of variants seen at test time. To extend coverage, we prompt GPT-5.4-mini (Singh et al., 2025) with seed pairs from $\mathcal{D}$ to generate additional $(\mathrm{G}_{nc}, \mathrm{G}_c)$ candidates following the same UGT category patterns. Seeds and all filtering are restricted to the training split. We discard candidates whose G_{nc} matches a standard word in the train or dev split, to avoid spuriously rewriting canonical inputs, or where $\mathrm{G}_c = \mathrm{G}_{nc}$, which carries no normalization signal. Prompt details and coverage statistics are provided in Appendix C.

Hit and Miss prefixes. The two special tokens introduced in Equation (1), <Hint> and <Miss>, are added to the vocabulary. Prepending the retrieval prefix to the contextualized input from Section 3.1 produces the following inputs:

Hit: $\texttt{<lang>}\ \texttt{<Hint>}\ w_i \rightarrow G_{c,i}\ \ w_1\, w_2 \cdots \texttt{|}w_i\texttt{|} \cdots w_n$

Miss: $\texttt{<lang>}\ \texttt{<Miss>}\ \ w_1\, w_2 \cdots \texttt{|}w_i\texttt{|} \cdots w_n$

The <Miss> marker also distinguishes a non-canonical word absent from $\mathcal{D}$ from a canonical input that requires no retrieval.

3.3 Staged Decoding

To bind the retrieved hint to prediction, we make the canonical form an explicit intermediate target during decoding. The model produces the normalization output G_c before the phoneme sequence P_c in a single autoregressive trajectory, with the target sequence structured as follows:

$$\text{Non-canonical:}\quad \underbrace{\texttt{<Target>}\,\mathrm{G}_c}_{\text{normalization}}\ \underbrace{\texttt{<Phoneme>}\,\mathrm{P}_c}_{\text{phonemization}},$$

$$\text{Canonical:}\quad \texttt{<Phoneme>}\,\mathrm{P}_c.$$

For canonical words, the model bypasses the normalization stage and emits P_c directly, aligning with conventional G2P training. For non-canonical words, it generates G_c as a learned intermediate before P_c, providing explicit supervision for both the normalization stage (via G_c) and the phonemization stage (via P_c). The model is trained end-to-end with cross-entropy loss over the full target sequence, without auxiliary objectives or additional loss terms.

Together, the three components make canonical-form modeling explicit within a single G2P architecture, supervising normalization and phonemization without sacrificing canonical-token accuracy.

4 Experiments

Using the taxonomy of Section 2.2, we quantify the canonical-to-non-canonical gap on UGTPHON, decompose it by category, and compare our

Model	English			Vietnamese			Korean		
	$\text{PER}_{\text{ipa}}\downarrow$	$\text{PER}_c\downarrow$	$\text{PER}_{nc}\ \langle\Delta\rangle\downarrow$	$\text{PER}_{\text{ipa}}\downarrow$	$\text{PER}_c\downarrow$	$\text{PER}_{nc}\ \langle\Delta\rangle\downarrow$	$\text{PER}_{\text{ipa}}\downarrow$	$\text{PER}_c\downarrow$	$\text{PER}_{nc}\ \langle\Delta\rangle\downarrow$
Rule-based G2P									
Epitran (–)	33.4	13.4	45.1 ⟨+31.7⟩	32.7	34.7	75.1 ⟨+40.4⟩	47.0	37.5	50.7 ⟨+13.2⟩
viG2P (–)	–	–	–	25.4	27.3	78.4 ⟨+51.1⟩	–	–	–
Supervised G2P (trained on IPA-Dict canonical lexicon)									
Transformer (2019) (2.0M)	17.4	12.9	44.8 ⟨+31.9⟩	15.3	5.8	68.4 ⟨+62.6⟩	7.5	9.5	54.8 ⟨+45.3⟩
ByT5-tiny (2022) (7.3M)	17.0	18.8	46.4 ⟨+27.6⟩	**15.2**	6.0	71.9 ⟨+65.9⟩	7.2	6.8	51.1 ⟨+44.3⟩
GBERT (2022) (8.0M)	21.5	17.6	45.0 ⟨+27.4⟩	24.0	30.8	93.0 ⟨+62.2⟩	11.7	16.8	45.9 ⟨+29.1⟩
ByT5 (2022) (300M)	**13.5**	10.5	46.6 ⟨+36.1⟩	**15.2**	5.9	72.7 ⟨+66.8⟩	**6.7**	4.2	46.2 ⟨+42.0⟩
Supervised G2P (further trained on UGTPHON)									
Transformer (2.0M)	18.8	10.3	32.7 ⟨+22.4⟩	17.8	5.9	33.7 ⟨+27.8⟩	7.5	5.8	38.7 ⟨+32.9⟩
ByT5-tiny (7.3M)	30.1	9.7	30.8 ⟨+21.1⟩	17.0	7.7	43.6 ⟨+35.9⟩	8.1	**4.0**	39.0 ⟨+35.0⟩
GBERT (8.0M)	25.6	7.8	50.5 ⟨+42.7⟩	50.3	5.6	60.4 ⟨+54.8⟩	8.5	5.8	**31.6** ⟨+25.8⟩
ByT5 (300M)	15.9	5.2	29.8 ⟨+24.6⟩	17.3	7.7	43.2 ⟨+35.5⟩	7.2	4.3	47.5 ⟨+43.2⟩
Ours (ByT5) (300M)	17.8	**4.7**	22.5 ⟨+17.8⟩	15.9	5.9	26.7 ⟨+20.8⟩	9.0	5.1	32.0 ⟨+26.9⟩
Ours (Qwen 2.5-0.5B) (494M)	19.5	10.2	**14.3** ⟨+4.1⟩	15.5	**4.0**	**20.0** ⟨+16.0⟩	8.2	6.0	32.4 ⟨+26.4⟩

Table 3: Comparison with existing G2P models on UGTPHON. PER (%) on canonical (PER_c) and non-canonical (PER_{nc}) words, with the within-input gap $\Delta = \text{PER}_{nc} - \text{PER}_c$. Color encodes Δ: ≥ 30 **pp** (large gap), 10–30 **pp** (medium), < 10 **pp** (small). Bold: best per column.

model against two alternatives: a normalize-then-phonemize pipeline and frontier LLMs.

4.1 Experimental Setup

Evaluation protocol. Models take the full sentence as input and predict the marked target word. We report Phoneme Error Rate (PER, %) on UGTPHON's test split as PER_c (canonical) and PER_{nc} (non-canonical), with $\Delta = \text{PER}_{nc} - \text{PER}_c$. For lexicon-only supervised baselines, we additionally report PER_{ipa} on the IPA-dict (Doherty, 2016) lexicon to track canonical coverage.

Baselines. We compare our model against task-relevant G2P, normalization, and frontier-LLM baselines. *Rule-based G2P*: Epitran (Mortensen et al., 2018) and viG2P (Tri et al., 2020). *Supervised G2P*: Transformer (Yolchuyeva et al., 2019), ByT5/ByT5-tiny (Zhu et al., 2022), and GBERT (Dong et al., 2022), each evaluated in two training regimes: *lexicon-only* (trained on canonical IPA-Dict) and *fine-tuned* (further trained on UGTPHON). Frontier LLMs: GPT-4o, GPT-5.4-mini (Singh et al., 2025), Gemini-2.5-flash, Gemini-3-flash, and Claude-Sonnet-4.6, evaluated under the nominal $K = 9$ in-context setting described in Appendix E, following prior G2P-with-LLM work (Han et al., 2024). *Task-relevant baselines*: ICKR (Han et al., 2024), R-G2P (Zhao et al., 2022), and lexical-normalization + G2P pipelines using MoNoise (Van Der Goot, 2019) or PolyNorm (Wong et al., 2025). The full cross-lingual comparison is reported in Table 10.

Implementation. We use ByT5-small as the default backbone, with Qwen2.5-0.5B for backbone ablation. Training uses AdamW (lr $= 10^{-4}$, batch size=32, 10 epochs) on 4 NVIDIA V100 GPUs. The retrieval datastore $\mathcal{D}$ uses the exact-match lookup described in Section 3.2. For the frontier LLMs, we query official APIs under the nominal $K = 9$ in-context setting described in Appendix E. Model identifiers and access dates are listed in Appendix D; the prompt template is provided in Appendix E.

4.2 Main Results

All baselines exhibit large canonical-to-non-canonical gaps. Table 3 reports PER for rule-based and supervised baselines on UGTPHON. The within-input gap Δ ranges from $+27.4$ to $+36.1$ on English, $+40.4$ to $+66.8$ on Vietnamese, and $+13.2$ to $+45.3$ on Korean. The pattern persists across rule-based (Mortensen et al., 2018), transformer (Yolchuyeva et al., 2019), and byte-level (Zhu et al., 2022) architectures, and is most severe on Vietnamese, where supervised baselines average $\Delta \approx +64$ despite PER_c below 8%.

Further training on UGTPHON reduces PER_{nc} but does not close Δ. Training the supervised baselines on UGTPHON's training split substantially lowers PER_{nc} on English ($46.6 \rightarrow 29.8$) and Vietnamese ($72.7 \rightarrow 43.2$), but the strongest fine-tuned model (ByT5) still shows $\Delta = +24.6$ on English and $+35.5$ on Vietnamese. Korean even slightly worsens ($46.2 \rightarrow 47.5$), reflecting its Pass-through–dominated composition in Table 1, where

Input	Retrieval Augmentation	Staged Decoding	English			Vietnamese			Korean		
			$\text{PER}_{all}\downarrow$	$\text{PER}_{c}\downarrow$	$\text{PER}_{nc}\downarrow$	$\text{PER}_{all}\downarrow$	$\text{PER}_{c}\downarrow$	$\text{PER}_{nc}\downarrow$	$\text{PER}_{all}\downarrow$	$\text{PER}_{c}\downarrow$	$\text{PER}_{nc}\downarrow$
Word→Word	✗	✗	7.9	5.2	29.8	14.1	7.7	43.2	17.8	4.3	47.5
Sent→Sent	✗	✗	19.7 (+11.8)	18.0 (+12.8)	30.5 (+0.7)	29.7 (+15.6)	19.2 (+11.5)	70.7 (+27.5)	33.5 (+15.7)	41.6 (+37.3)	31.3 (−16.2)
Sent→Word	✗	✗	7.6 (−0.3)	5.4 (+0.2)	25.5 (−4.3)	15.0 (+0.9)	9.1 (+1.4)	41.9 (−1.3)	**12.0** (−5.8)	4.7 (+0.4)	28.1 (−19.4)
Sent→Word	✗	✓	8.9 (+1.0)	4.9 (−0.3)	30.1 (+0.3)	10.1 (−4.0)	5.9 (−1.8)	31.8 (−11.4)	13.1 (−4.7)	**4.0** (−0.3)	33.1 (−14.4)
Sent→Word	✓	✗	7.0 (−0.9)	5.0 (−0.2)	22.5 (−7.3)	9.6 (−4.5)	5.8 (−1.9)	29.1 (−14.1)	12.4 (−5.4)	5.5 (+1.2)	**27.7** (−19.8)
Sent→Word	✓	✓	**6.8** (−1.1)	**4.7** (−0.5)	**22.5** (−7.3)	**9.3** (−4.8)	**5.9** (−1.8)	**26.7** (−16.5)	13.5 (−4.3)	5.1 (+0.8)	32.0 (−15.5)

Table 4: Component ablation of our retrieval-augmented G2P model on ByT5-small. Green/Red: gain/regression vs. the *Word→Word* baseline. **Retrieval Aug**: $G_{\text{nc}}\rightarrow G_c$ hints from the datastore. **Staged Decoding**: <Target> → <Phoneme> emission. Bold: column-best per language (PER, %).

the surface already encodes pronunciation and further fine-tuning provides little signal. The remaining Δ shows that further training alone does not eliminate the non-canonical gap, motivating explicit surface-to-canonical modeling.

Explicit canonical-form modeling improves matched backbones. Relative to matched ByT5 fine-tuning, our formulation lowers PER_{nc} from 29.8 to 22.5 on English, 43.2 to 26.7 on Vietnamese, and 47.5 to 32.0 on Korean. The same pattern holds for Qwen2.5-0.5B: 14.8 → 14.3, 23.1 → 20.0, and 77.3 → 32.4, respectively. Three-seed runs show stable English/Vietnamese results, while Korean has higher variance; paired bootstrap tests on the reported checkpoints give confidence intervals excluding zero for all six matched-backbone PER_{nc} comparisons (Tables 11 and 12). The largest matched-backbone reductions occur on Vietnamese and Korean, while English Qwen improves from 14.8 to 14.3 PER_{nc}.

4.3 Analysis

Sentence context helps as evidence, not as a generation target. Table 4 shows that generating the full sentence (Sent→Sent) hurts PER_c across all three languages, reflecting exposure bias from long autoregressive outputs (Yoon et al., 2023) (Korean PER_c rises by 37.3 points). Sent→Word avoids this trade-off and lowers Korean PER_{nc} (47.5 → 28.1), since Korean is dominated by pass-through forms that sentence context alone can disambiguate. Vietnamese, by contrast, improves little under Sent→Word, motivating the explicit canonical-form evidence that the lookup provides.

Exact-match lookup supplies canonical-form evidence; staged decoding encourages the model to use it. On top of Sent→Word, retrieval alone reduces PER_{nc} from 41.9 to 29.1 on Vietnamese and 25.5 to 22.5 on English, supplying the surface-to-canonical evidence the surface lacks. Staged decoding alone improves Vietnamese (41.9 → 31.8) but worsens English (25.5 → 30.1) and Korean (28.1 → 33.1): without a retrieval hint, forcing a canonical intermediate invites hallucinated reconstructions. The two together attain the lowest PER_{nc} on English (22.5) and Vietnamese (26.7) while keeping PER_c at or below the no-retrieval level. Korean shows no further gain, since its bottleneck has shifted from normalization to phonemization.

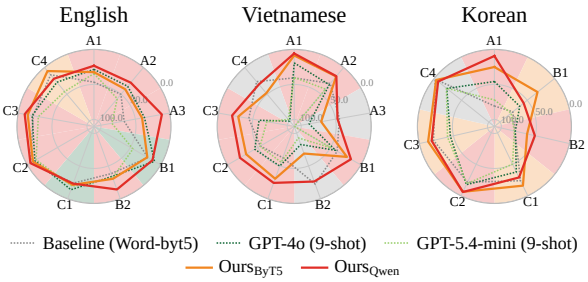

Figure 3: PER_{nc} (%) by fine-grained category. Gains concentrate on *Reconstruct* (A), with smaller margins on *Repetition* (B) and *Pass-through* (C).

Model	English	
	PER_c	PER_{nc}
word-ByT5 (no norm.)	5.2	29.8
word-ByT5 + MoNoise (2019)	14.2	18.0
word-ByT5 + GPT-5.4-mini	13.6	18.6
Ours (ByT5)	**4.7**	22.5
Ours (Qwen 2.5-0.5B)	10.2	**14.3**

Table 5: Comparison with two-stage LexNorm + G2P pipelines on English (PER, %).

Improvements track the inference demand of each category. Figure 3 decomposes PER_{nc} along the taxonomy of Table 1, which predicts where canonical-form modeling should help and where it should add little. The results match.

Reconstruct (A) is where the surface and canonical form share the least phonemic evidence (e.g.,

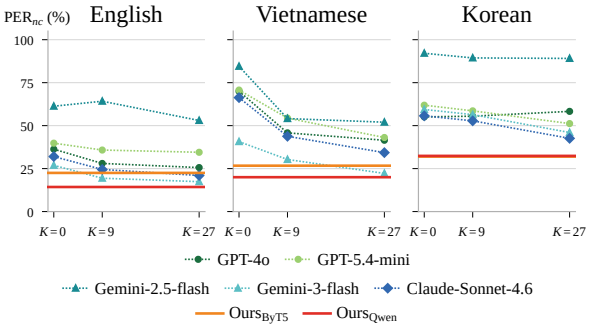

Figure 4: Shot scaling for frontier LLMs on UGTPHON. PER$_{nc}$ (%) under the nominal $K \in \{0, 9, 27\}$ shot settings; Korean uses 0, 7, and 21 actual in-context examples because A2 and A3 are absent.

idk → *I don't know*), so our model's gain is largest: with the Qwen 2.5 (0.5B) backbone, A-average PER$_{nc}$ drops by 33.4 points on English (54.4 → 21.0) and 15.2 on Vietnamese (38.3 → 23.1). On English A3 specifically, our model reaches 11.5, compared with 66.3 for the word-level baseline and 36.8 for GPT-4o, yielding the largest separation on a category that requires phrasal canonical-form reconstruction.

Repetition (B) is intermediate: the surface encodes the pronunciation but with redundant content that must be collapsed. Our model attains the best PER$_{nc}$ on English B2 and sharply reduces Vietnamese B2 against frontier LLMs, though the word-level ByT5 baseline remains marginally stronger on Vietnamese B2 (22.5 vs. 26.3), since the redundant surface still preserves the underlying phonemes.

Pass-through (C) is mixed: the surface already encodes the pronunciation, so the canonical-form intermediate is no longer strictly necessary. Where the baseline is near-zero (Korean C4, 1.9), our model offers no benefit; where the surface remains non-trivial to phonemize, the gain returns. Our model attains the best PER$_{nc}$ on nine of twelve Pass-through subcategories, so the cost-benefit depends on residual phonemization difficulty rather than the coarse category label.

Our approach outperforms two-stage pipelines without trading canonical for non-canonical accuracy. Table 5 compares our model against two-stage English pipelines as shown in Figure 2 (B) that pair a lexical normalizer with word-level ByT5: MoNoise (Van Der Goot, 2019) and GPT-5.4-mini (Singh et al., 2025). English is the only language where this comparison is possible, since dedicated lexical normalization models exist for English but not for the lower-resource languages we evaluate. The two pipelines reduce PER$_{nc}$ from 29.8 to roughly 18, but PER$_c$ degrades from 5.2 to over 13. Our model avoids this trade-off: with the ByT5 backbone, PER$_c$ stays at 4.7 and PER$_{nc}$ drops to 22.5; with Qwen 2.5-0.5B, PER$_{nc}$ reaches the lowest value across all systems (14.3) while PER$_c$ (10.2) remains well below the pipelines.

Our approach remains competitive with frontier LLMs. Figure 4 compares our approach against five frontier LLMs under the nominal $K \in \{0, 9, 27\}$ shot settings. Across all three languages, our 0.5B model achieves performance competitive with or better than the evaluated frontier LLMs. On English, the strongest frontier model (Gemini-3-flash) reaches PER$_{nc}$ = 17.4 under $K = 27$, while our Qwen2.5-0.5B model achieves 14.3. On Vietnamese, frontier LLMs exhibit substantially larger errors on several UGT categories, including GPT-5.4-mini on Reconstruct A3 (PER$_{nc}$ = 109.6; Figure 3), whereas our model achieves the lowest overall PER$_{nc}$ of 20.0. On Korean, our ByT5 and Qwen2.5-0.5B variants achieve PER$_{nc}$ of 32.0 and 32.4, respectively, compared with 42.6 for the strongest evaluated frontier baseline (Claude-Sonnet-4.6 under $K = 27$). These results further highlight the difficulty of canonical-form inference for UGT, particularly in Vietnamese and Korean.

The training-set datastore alone closes most of the gap to oracle retrieval. Table 6 compares four datastores: no retrieval, training-set entries only, training-set with LLM-generated augmentation, and an oracle that uses gold canonical forms at test time. The training-set datastore alone reaches PER$_{nc}$ = 22.7 on English, within 2.1 points of the oracle (20.6), and 26.7 on Vietnamese, within 0.8 points of the oracle (25.9). Adding LLM-generated (G_{nc}, G_c) pairs reduces English PER$_{nc}$ by only 0.2 point and leaves Vietnamese unchanged. The deployable datastore approaches oracle performance without test-time canonical-form access.

Downstream TTS-ASR validation. On downstream TTS-ASR evaluations with eSpeak NG and Whisper large-v3 (Radford et al., 2023) (Appendix F), our ByT5 variant attains the lowest WER on English (0.359), while our Qwen2.5-0.5B variant attains the lowest observed WER among predicted-input systems on Vietnamese (0.640). These results are consistent with the token-level PER trends.

Datastore $\mathcal{D}$	PER$_{nc}$ (%)		
	EN	VI	KO
No retrieval ($\emptyset$)	30.1	31.8	33.1
Train-only	22.7	26.7	32.4
+ LLM Aug	**22.5**	**26.7**	**32.0**
Oracle (not deployable)	20.6	25.9	18.6

Table 6: Retrieval datastore $\mathcal{D}$ sensitivity (ByT5). *Oracle* uses gold canonical forms (not deployable).

5 Conclusion

We diagnose user-generated text as a setting where G2P must compose surface-to-canonical inference with phonemization, and introduce UGTPHON, the first G2P benchmark for UGT with an inference-grounded taxonomy across English, Vietnamese, and Korean. Our retrieval-augmented compositional baseline makes canonical-form inference explicit through exact-match hints and staged decoding, reducing non-canonical PER relative to matched ByT5 and Qwen2.5-0.5B backbones across all three languages. The gains are largest on Reconstruct cases, while the 0.5B variant remains competitive with the evaluated frontier LLMs. Together, UGTPHON provides a multilingual benchmark and diagnostic framework for studying canonical-form inference in UGT phonemization.

Limitations

We include English and Vietnamese TTS-ASR evaluations (Appendix F) showing that lower PER is accompanied by lower word error rate for several systems, but we do not report subjective MOS ratings or evaluate across multiple TTS systems; characterizing the full relationship between PER and TTS quality on UGT is left to future work. UGTPHON spans three typologically diverse languages (alphabetic English, tonal-diacritic Vietnamese, syllabic-block Korean) but does not cover logographic, abjad, or abugida writing systems, where UGT phenomena and the relevant normalization signals may differ. Whether the compositional decomposition transfers to these scripts is an open question. Our retrieval datastore is built from training-set pairs and an LLM-based expansion. UGT vocabulary drifts continuously, so maintaining coverage on emerging non-canonical forms will require periodic datastore refresh; we leave the dynamics of such refresh, and zero-shot extensions of the retrieval mechanism, to future work. Beyond these scope limitations, more accurate UGT phonemization is dual-use: it improves TTS intelligibility on harmful UGT as well as benign UGT, and UGTPHON inherits toxic content from its source corpora, whose moderation profile we report in Appendix B.

Acknowledgments

This work was partly supported by NAVER Cloud Corporation, the Institute of Information & Communications Technology Planning & Evaluation (IITP) grant funded by the Korean government (MSIT) (No. RS-2020-II201373, Artificial Intelligence Graduate School Program, Hanyang University), and the Institute of Information & Communications Technology Planning & Evaluation (IITP) grant funded by the Korean government (MSIT) (No. RS-2025-02215122, Development and Demonstration of Lightweight AI Model for Smart Homes).

References

Timothy Baldwin, Marie-Catherine De Marneffe, Bo Han, Young-Bum Kim, Alan Ritter, and Wei Xu. 2015. Shared tasks of the 2015 workshop on noisy user-generated text: Twitter lexical normalization and named entity recognition. In *Proceedings of the workshop on noisy user-generated text*, pages 126–135.

Maximilian Bisani and Hermann Ney. 2008. Joint-sequence models for grapheme-to-phoneme conversion. *Speech communication*, 50(5):434–451.

Samuel Brody and Nicholas Diakopoulos. 2011. Cooooooooooooooollllllllllllll!!!!!!!!!!!!!! using word lengthening to detect sentiment in microblogs. In *Proceedings of the 2011 conference on empirical methods in natural language processing*, pages 562–570.

Weerayut Buaphet, Thanh-Nhi Nguyen, Risa Kondo, Tomoyuki Kajiwara, Yumin Kim, Jimin Lee, Hwanhee Lee, Holy Lovenia, Peerat Limkonchotiwat, Sarana Nutanong, and Rob Van der Goot. 2026. Multilexnorm++: A unified benchmark and a generative model for lexical normalization for asian languages. *ACM Transactions on Asian and Low-Resource Language Information Processing*.

Dasol Choi, Jooyoung Song, Eunsun Lee, Jinwoo Seo, Heejune Park, and Dongbin Na. 2023. Komultitext: Large-scale korean text dataset for classifying biased speech in real-world online services. *arXiv preprint arXiv:2310.04313*.

Liam Doherty. 2016. Ipa dict: Monolingual wordlists with pronunciation information in ipa. https://github.com/open-dict-data/ipa-dict. Accessed 2026-05-22.

Lu Dong, Zhi-Qiang Guo, Chao-Hong Tan, Ya-Jun Hu, Yuan Jiang, and Zhen-Hua Ling. 2022. Neural grapheme-to-phoneme conversion with pre-trained grapheme models. In *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6202–6206. IEEE.

Bo Han and Timothy Baldwin. 2011. Lexical normalisation of short text messages: Makn sens a# twitter. In *Proceedings of the 49th annual meeting of the association for computational linguistics: Human language technologies*, pages 368–378.

Dongrui Han, Mingyu Cui, Jiawen Kang, Xixin Wu, Xunying Liu, and Helen Meng. 2024. Improving grapheme-to-phoneme conversion through in-context knowledge retrieval with large language models. In *2024 IEEE 14th International Symposium on Chinese Spoken Language Processing (ISCSLP)*, pages 631–635. IEEE.

Jaehyeon Kim, Jungil Kong, and Juhee Son. 2021. Conditional variational autoencoder with adversarial learning for end-to-end text-to-speech. In *International conference on machine learning*, pages 5530–5540. PMLR.

J Richard Landis and Gary G Koch. 1977. The measurement of observer agreement for categorical data. *biometrics*, pages 159–174.

Hae-Na Lee and Vikas Ashok. 2022. Impact of out-of-vocabulary words on the twitter experience of blind users. In *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems*, pages 1–20.

Jackson L Lee, Lucas FE Ashby, M Elizabeth Garza, Yeonju Lee-Sikka, Sean Miller, Alan Wong, Arya D McCarthy, and Kyle Gorman. 2020. Massively multilingual pronunciation modeling with wikipron. In *Proceedings of the twelfth language resources and evaluation conference*, pages 4223–4228.

David R Mortensen, Siddharth Dalmia, and Patrick Littell. 2018. Epitran: Precision g2p for many languages. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*.

Thanh-Nhi Nguyen, Thanh-Phong Le, and Kiet Van Nguyen. 2024. Vilexnorm: a lexical normalization corpus for vietnamese social media text. In *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1421–1437.

Josef R Novak, Nobuaki Minematsu, and Keikichi Hirose. 2012. Wfst-based grapheme-to-phoneme conversion: Open source tools for alignment, model-building and decoding. In *Proceedings of the 10th International Workshop on Finite State Methods and Natural Language Processing*, pages 45–49.

Barbara Plank. 2016. What to do about non-standard (or non-canonical) language in nlp. In *Conference on Natural Language Processing (KONVENS)*.

Artem Ploujnikov and Mirco Ravanelli. 2022. Soundchoice: Grapheme-to-phoneme models with semantic disambiguation. In *Interspeech*.

Qwen, :, An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, and 25 others. 2025. Qwen2.5 technical report. *arXiv preprint arXiv:2412.15115*.

Alec Radford, Jong Wook Kim, Tao Xu, Greg Brockman, Christine McLeavey, and Ilya Sutskever. 2023. Robust speech recognition via large-scale weak supervision. In *International conference on machine learning*, pages 28492–28518. PMLR.

Kanishka Rao, Fuchun Peng, Haşim Sak, and Françoise Beaufays. 2015. Grapheme-to-phoneme conversion using long short-term memory recurrent neural networks. In *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4225–4229. IEEE.

Yi Ren, Yangjun Ruan, Xu Tan, Tao Qin, Sheng Zhao, Zhou Zhao, and Tie-Yan Liu. 2019. Fastspeech: Fast, robust and controllable text to speech. *Advances in neural information processing systems*, 32.

Jonathan Shen, Ruoming Pang, Ron J Weiss, Mike Schuster, Navdeep Jaitly, Zongheng Yang, Zhifeng Chen, Yu Zhang, Yuxuan Wang, Rj Skerrv-Ryan, Rif A. Saurous, Yannis Agiomyrgiannakis, and Yonghui Wu. 2018. Natural tts synthesis by conditioning wavenet on mel spectrogram predictions. In *2018 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, pages 4779–4783. IEEE.

Aaditya Singh, Adam Fry, Adam Perelman, Adam Tart, Adi Ganesh, Ahmed El-Kishky, Aidan McLaughlin, Aiden Low, AJ Ostrow, Akhila Ananthram, Akshay Nathan, Alan Luo, Alec Helyar, Aleksander Madry, Aleksandr Efremov, Aleksandra Spyra, Alex Baker-Whitcomb, Alex Beutel, Alex Karpenko, and 466 others. 2025. Openai gpt-5 system card. *arXiv preprint arXiv:2601.03267*.

Paul Taylor. 2005. Hidden markov models for grapheme to phoneme conversion. In *Interspeech*, pages 1973–1976.

Nguyen Minh Tri, Cao Xuan Nam, and 1 others. 2020. Vietnamese speech synthesis with end-to-end model and text normalization. In *2020 7th NAFOSTED Conference on Information and Computer Science (NICS)*, pages 179–184. IEEE.

Rob Van Der Goot. 2019. Monoise: A multi-lingual and easy-to-use lexical normalization tool. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 201–206.

Rob Van Der Goot, Alan Ramponi, Arkaitz Zubiaga, Barbara Plank, Benjamin Muller, Iñaki San Vicente Roncal, Nikola Ljubešić, Özlem Çetinoğlu, Rahmad Mahendra, Talha Çolakoğlu, Timothy Baldwin, Tommaso Caselli, and Wladimir Sidorenko. 2021. Multilexnorm: A shared task on multilingual lexical normalization. In *Proceedings of the Seventh Workshop on Noisy User-generated Text (W-NUT 2021)*, pages 493–509.

Rob Van Der Goot, Rik Van Noord, and Gertjan van Noord. 2018. A taxonomy for in-depth evaluation of normalization for user generated content. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*.

Daan Van Esch and Richard Sproat. 2017. An expanded taxonomy of semiotic classes for text normalization. In *Interspeech*, pages 4016–4020.

Michel Wong, Ali Alshehri, Sophia Kao, and Haotian He. 2025. Polynorm: Few-shot llm-based text normalization for text-to-speech. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing: Industry Track*, pages 77–85.

Linting Xue, Aditya Barua, Noah Constant, Rami Al-Rfou, Sharan Narang, Mihir Kale, Adam Roberts, and Colin Raffel. 2022. Byt5: Towards a token-free future with pre-trained byte-to-byte models. *Transactions of the Association for Computational Linguistics*, 10:291–306.

Sevinj Yolchuyeva, Géza Németh, and Bálint Gyires-Tóth. 2019. Transformer based grapheme-to-phoneme conversion. In *Interspeech*.

Eunseop Yoon, Hee Suk Yoon, Dhananjaya Gowda, SooHwan Eom, Daehyeok Kim, John Harvill, Heting Gao, Mark Hasegawa-Johnson, Chanwoo Kim, and Chang D Yoo. 2023. Mitigating the exposure bias in sentence-level grapheme-to-phoneme (g2p) transduction. In *Interspeech*.

Chendong Zhao, Jianzong Wang, Xiaoyang Qu, Haoqian Wang, and Jing Xiao. 2022. r-g2p: Evaluating and enhancing robustness of grapheme to phoneme conversion by controlled noise introducing and contextual information incorporation. In *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6197–6201. IEEE.

Jian Zhu, Cong Zhang, and David Jurgens. 2022. Byt5 model for massively multilingual grapheme-to-phoneme conversion. In *Interspeech*.

A Creating UGTPHON

This appendix details the dataset construction procedure summarized in Section 2.3, including the procedures for source corpus collection and preprocessing, as well as the annotation pipelines for canonical forms, categories, and phonemes.

A.1 Source Corpora and Preprocessing

Source corpora. Non-canonical tokens G_{nc} in UGTPHON are drawn from existing user-generated text resources, with small human-curated augmented subsets for English and Vietnamese. For English, we use MultiLexNorm (Van Der Goot et al., 2021), whose English data builds on LexNorm2015 (Baldwin et al., 2015). For Vietnamese, we use ViLexNorm (Nguyen et al., 2024), a lexical normalization corpus of Vietnamese social media text whose annotations align with our $G_{\text{nc}} \rightarrow G_{\text{c}}$ formulation. For Korean, we source G_{nc} from MultiLexNorm++ (Buaphet et al., 2026), supplemented by the Korean UGT corpus of Choi et al. (2023), and re-annotate G_{c} from scratch using the protocol in Appendix A.2. In addition, the English and Vietnamese subsets include 175 and 385 human-curated augmented UGT samples, respectively, generated from existing UGT patterns.

Licenses. Source corpora are used under their original licensing terms. MultiLexNorm's English data builds on LexNorm2015 (Baldwin et al., 2015), which has no explicit license; following the convention adopted by MultiLexNorm (Van Der Goot et al., 2021), attribution is provided through citation. The English subset of UGTPHON is provided strictly for academic and research purposes, and commercial use is prohibited. ViLexNorm (Nguyen et al., 2024) is released under CC BY-NC-SA 4.0. The Korean subset is derived from KoMultiText (Choi et al., 2023), partly via MultiLexNorm++ (Buaphet et al., 2026), and is released under Apache 2.0. UGTPHON is released with per-language licenses reflecting these upstream terms.

Data provenance and consent. UGTPHON is constructed from previously released UGT datasets rather than through direct collection from the original social-media users. We therefore do not separately recruit or obtain consent from the original text authors, and instead follow the data-use conditions and licensing terms of the respective source datasets.

Preprocessing. We preserve the original sentence boundaries, casing, and surrounding context of each G_{nc} token in the UGT sentence. Non-lexical elements are handled during phoneme construction rather than uniformly removed from the raw text: punctuation may be dropped, numbers may be verbalized, and elements that cannot be phonemized may remain as placeholders.

A.2 Canonical Form Annotation

For English and Vietnamese, canonical forms G_c are obtained directly from the source lexical normalization corpora (Baldwin et al., 2015; Nguyen et al., 2024), which provide surface-to-canonical mappings.

Manual annotation for Korean. Unlike English and Vietnamese, no comparably large-scale lexical normalization corpus exists for Korean. We therefore annotate G_c from scratch with two native-speaker experts following a category-specific protocol that reflects the structure of Korean UGT; the full annotation guide is provided below:

- **A1 (Shortening-V, 초성체)**: Consonant abbreviations are expanded to their full Hangul form (e.g., ㄱㅅ → 감사, ㅇㄱㄹㅇ → 이거레알).
- **B1 (Lengthening)**: The elongation of vowel or syllable is collapsed to their base form (e.g., 꺄아악 → 꺅).
- **B2 (Repetition)**: For consonant-only repetition (e.g., ㅋㅋㅋㅋ), G_c is the Hangul realization preserving the intended count (e.g., 크크크크, 키키키키, 킥킥킥킥; recorded as comma-separated alternatives when more than one is acceptable). For syllable- or word-level repetition (e.g., 빨리빨리, 감사감사), the surface itself is the intended pronunciation, so G_c retains the repetition.
- **C1–C4 (Pass-through)**: The surface form already encodes the intended pronunciation, so $G_c = \mathrm{G}_{nc}$ (e.g., 머해, 킹받다, 갑분싸).

Tokens from the source corpora that turn out to be standard Korean rather than UGT are removed from the benchmark.

A.3 Category Annotation

Category definitions. We use the nine fine-grained categories listed in Table 1, grouped into the three macro-types, *Reconstruct* (A1–A3), *Repetition* (B1–B2), and *Pass-through* (C1–C4). The

Type	ID	Category	Description	Examples
A *Reconstruct*	A1	Shortening-V (Vowel Drop)	Canonical-form consonants are preserved, vowels are dropped.	*pls* → *please* *đc* → *được*
	A2	Shortening-O (Other Drop)	One or more characters dropped from a single canonical word, beyond vowel-only deletion.	*bc* → *because* *k* → *không*
	A3	Phrasal (Unpronounceable)	A multi-word canonical expansion whose pronunciation is not directly recoverable from the surface form.	*idk* → *I don't know* *mn* → *mọi người*
B *Repetition*	B1	Lengthening	Repeated characters convey prosodic emphasis; the canonical form collapses the repetition.	*soooo* → *so*
	B2	Iteration	Repeated tokens or words; the canonical form retains the repetition for phonemization.	*wait wait*
C *Pass-through*	C1	Eye Direct	A phonetic respelling whose surface form already encodes the intended pronunciation.	*luv* → *love* *zô* → *vô*
	C2	Regular	Standard productive variation of a canonical word.	*droppin* → *dropping*
	C3	Slang/Teen	Lexicalized non-standard form whose pronunciation is well established.	*tho* → *though* *hong* → *không*
	C4	Phrasal (Pronounceable)	Multi-letter abbreviation pronounced as a word in its own right.	*rofl* *GATO*

Table 7: Operational definitions and representative examples of the fine-grained categories for non-canonical words.

macro-grouping reflects the type of inference required to phonemize a token, rather than its surface transformation. Table 7 presents the operational definitions and representative examples used during annotation.

Category annotation pipeline. Category labels are assigned manually by native-speaker annotators following the operational definitions in Table 7. For each $\mathrm{G}_{nc} \rightarrow \mathrm{G}_c$ pair, an annotator inspects the surrounding sentence and selects the single best-fitting category among A1–C4. Each pair is then independently verified by a second native-speaker annotator, with disagreements resolved through discussion. All annotators have prior training in NLP or linguistics and are instructed to flag examples where (i) the canonical form is genuinely ambiguous given the context, or (ii) the category assignment is unclear; cases of type (i) are resolved using sentence context as described in Section 2.1, and cases of type (ii) default to the more specific category (e.g., A3 rather than A1 when both apply).

A.4 Phoneme Annotation

Target phonemes P_c are derived from the canonical form G_c using IPA-Dict (Doherty, 2016). For English heteronyms (e.g., *lead* as /liˇd/ or /lɛd/), the correct pronunciation depends on sentence-level context and IPA-Dict alone is insufficient. Therefore, an annotator with linguistic expertise manually selects the appropriate phoneme sequence based on the surrounding sentence.

Out-of-vocabulary handling. Canonical forms not covered by IPA-Dict are manually transcribed following the same IPA conventions. We measure OOV over unique canonical-form types across all splits, using the corresponding IPA-Dict lexicons (`en_US`, `vi_N`, and `ko`); non-phonemizable and purely symbolic forms are excluded. Across all canonical types, the OOV rates are 39.4% (EN), 23.1% (VI), and 77.6% (KO). Restricted to non-canonical UGT targets, they are 4.3% (48/1,110), 7.4% (175/2,378), and 92.8% (1,991/2,146), respectively. The high Korean rate largely reflects the mismatch between IPA-Dict entries and the inflected canonical forms preserved in UGTPHON.

Inter-annotator agreement. To assess phoneme annotation reliability, two annotators independently perform the procedure above, and we compute agreement between the resulting transcriptions. As shown in Table 8, Cohen's κ over IPA phoneme sequences reaches 0.86 for English and 0.87 for Vietnamese, with Korean at 0.75. The Korean score, while lower, remains within the substantial band (Landis and Koch, 1977). This mainly reflects divergent IPA transcription conventions, particularly differing choices of consonant allophone symbols.

We report the detailed agreement rates by UGT type in Table 8. For English, agreement remains consistently high across most categories, although relatively lower agreement was observed for A3 and C4. Vietnamese shows a similar overall pattern, with lower agreement for B2 and especially

Language	Tokens	Cohen's κ	Raw Agreement by UGT Type (%)									
			Overall	A1	A2	A3	B1	B2	C1	C2	C3	C4
English	1,327	0.860	86.3	98.2	87.8	51.3	100.0	86.4	92.1	88.1	88.6	67.7
Vietnamese	2,152	0.869	87.3	84.1	92.7	83.6	97.7	48.6	89.4	92.3	91.6	0.0
Korean	722	0.754	75.6	87.9	—	—	78.8	68.1	86.4	90.9	70.9	62.5

Table 8: Inter-annotator agreement for phoneme annotations across languages and UGT types. While English and Vietnamese show near-perfect agreement overall, Korean exhibits lower agreement due to greater variation in IPA transcription conventions.

Language	Sentences	Flagged (raw)	Flagged (norm)
English	5,029	568 (11.29%)	606 (12.05%)
Vietnamese	10,847	401 (3.70%)	473 (4.36%)
Korean	2,497	321 (12.86%)	322 (12.90%)
Total	18,373	1,290 (7.02%)	1,401 (7.63%)

Table 9: Sentence-level moderation flagged rates on UGTPHON, evaluated with OpenAI `omni-moderation-2024-09-26` on both raw UGT forms and canonical-form normalizations. A sentence is flagged if any of the 13 moderation categories triggers.

C4. Notably, the zero agreement for Vietnamese C4 primarily reflects genuine ambiguity in letter-by-letter IPA renderings of acronyms rather than annotation errors. Overall, these results indicate that the annotation process produces reliable phoneme transcriptions across languages while also revealing categories where pronunciation conventions are inherently more variable. For the released dataset, we resolve all inter-annotator conflicts through discussion between the two annotators to produce a single adjudicated version.

B Content Moderation Analysis

User-generated text often contains harmful or offensive content, and UGTPHON inherits this property from its source corpora. To characterize the toxicity profile of UGTPHON and to measure how canonical-form normalization affects moderation signal, we apply OpenAI's `omni-moderation-2024-09-26` model to every sentence in the corpus, scoring both the raw UGT form and its canonical-form normalization across 13 moderation categories.

B.1 Flagged Rates

Table 9 reports sentence-level flagged rates. Korean shows the highest flagged rate (12.9%), comparable to English (11.3%) and roughly three times higher than Vietnamese (3.7%). The dominant category is *harassment* in English and Korean, and *violence* in Vietnamese, reflecting source-corpus content rather than language-intrinsic properties.

B.2 Effect of Normalization

In general, canonical-form normalization marginally increases the overall flagged rate. The binary flag state changes for 385 of 18,373 sentences (2.10%), while the aggregate flagged rate increases by only 0.60 percentage points (7.02% $\rightarrow$ 7.63%). This indicates that most sentences retain the same moderation outcome under normalization. The increase is most pronounced in Vietnamese (3.70% $\rightarrow$ 4.36%), while English (11.29% $\rightarrow$ 12.05%) and Korean (12.86% $\rightarrow$ 12.90%) show smaller changes.

C LLM-Based Datastore Expansion

This appendix details the LLM-based expansion of the retrieval datastore $\mathcal{D}$ summarized in Section 3.2.

C.1 Prompt Template

We prompt GPT-5.4-mini (Singh et al., 2025) with seed examples from the training set to generate additional plausible (G_{nc}, G_c) pairs. The prompt is structured as follows: (i) a short description of UGT and the desired output format, (ii) k seed pairs sampled from the training datastore, and (iii) an instruction to generate n additional pairs that are similar in style but not identical to the seeds. We use k seed pairs sampled from all training entries of the given UGT type (ranging from a few dozen to over 200 per type depending on training-set density) and set $n \in \{150, 100, 200\}$ for types A1, A2, and A3, respectively. The remaining categories (B1, B2, C1–C4) are populated from the training-set entries only, without LLM-generated augmentation.

The full prompt template is given below: `{nc_type}`, `{type_desc}`, `{lang_name}`, `{seed_lines}`, and `{target}` are filled per type. The system message instructs the model to

act as an expert in the target language's internet slang.

```
Here are real examples of {nc_type} type
UGT tokens from {lang_name} social media
data.

{nc_type} definition: {type_desc}

Existing examples (do NOT repeat these):
{seed_lines}

Generate {target} MORE {nc_type}
tokens commonly used in {lang_name}
Twitter/Instagram/Discord/Reddit/texting.
Requirements:
- Follow the same pattern as {nc_type}
definition above.
- Each expansion must be the standard
{lang_name} form.
- Do NOT include any of the examples
above.
- Do NOT include standard dictionary
words as the raw form.
- Do NOT include readable acronyms that
are pronounced letter by letter.
- Cover a wide variety: different topics,
registers, formality levels.
- Return ONLY a JSON object:
{"raw_token": "expanded form", ...}
```

C.2 Filtering Criteria

Each generated (G_{nc}, G_c) candidate is filtered before insertion into $\mathcal{D}$. We discard candidates that satisfy any of the following conditions:

- **Trivial identity**: $G_c = G_{\text{nc}}$, meaning no normalization is needed and inserting the pair would provide no useful retrieval signal.
- **Standard token conflict**: G_{nc} matches a canonical token already present in the training or development set, which would risk over-correcting canonical inputs at inference.
- **Test set leakage**: The test set is never inspected during generation or filtering. Candidates whose G_{nc} happens to coincide with a test non-canonical token are retained only if the same surface form already appears in the training set with a consistent G_c.

C.3 Coverage Statistics

The final datastore D used in our main results combines training-set entries with LLM-augmented pairs. The training-only datastore contains $1{,}413/3{,}674/1{,}607$ unique non-canonical surface-form keys for English, Vietnamese, and Korean, respectively. The LLM-based expansion adds $491/220/246$ entries, respectively.

We define test-set coverage as the proportion of non-canonical target-token occurrences whose surface form G_{nc} has an exact-match key in the datastore. Under this definition, the training-only datastore covers 70.9% $(941/1{,}327)$, 82.9% $(1{,}785/2{,}152)$, and 25.3% $(183/722)$ of the English, Vietnamese, and Korean test targets, respectively.

D Training and Implementation Details

Backbones. We test our approach with two transformer backbones: ByT5-small (Xue et al., 2022) (300M parameters, byte-level encoder-decoder, default) and Qwen2.5-0.5B (Qwen et al., 2025) (500M parameters, decoder-only causal LM).

Optimization. All backbones are trained with AdamW (learning rate 1×10^{-4}, batch size 32, 10 epochs) on 4 NVIDIA V100 GPUs. We select the checkpoint with the lowest development set PER_{nc}.

Compute budget. Each of our method's training runs takes approximately 6 hours on 4 NVIDIA V100 GPUs. Including baseline retraining, backbone ablation, and frontier-LLM inference, the total compute used for all results reported in the paper is approximately 400 V100-GPU hours. Frontier-LLM API calls are not included in this figure.

Decoding. During inference, the model decodes greedily. For non-canonical tokens, the model first emits G_c following the `<Target>` marker, then the phoneme sequence P_c following `<Phoneme>`; for canonical tokens, P_c is emitted directly without the `<Target>` step.

Baselines. The rule-based baselines are Epitran (Mortensen et al., 2018) and viG2P (Tri et al., 2020), both used out-of-the-box with default settings. For viG2P, we use the Viphoneme implementation.[2] The supervised neural baselines comprise Transformer (Yolchuyeva et al., 2019), ByT5 and ByT5-tiny (Xue et al., 2022), and GBERT (Dong et al., 2022); each is trained on the IPA-Dict training split with the same hyperparameters as in its original paper. The frontier-LLM baselines are GPT-4o, GPT-5.4-mini (Singh et al., 2025), Gemini-2.5-flash, Gemini-3-flash, and Claude-Sonnet-4.6, queried via their official APIs (model identifiers: `gpt-4o`, `gpt-5.4-mini`, `gemini-2.5-flash`, `gemini-3-flash`, and `claude-sonnet-4-6`; all accessed May 2026). For the $K = 9$ setting, we use one in-context example per available UGT category, yielding

[2] https://github.com/v-nhandt21/Viphoneme

nine examples for English and Vietnamese and seven for Korean, where A2 and A3 are absent. For the $K \in \{0, 9, 27\}$ shot-scaling study, this corresponds to 37,440 frontier-LLM requests (2,496 test sentences $\times$ 5 models $\times$ 3 shot settings). The task-relevant baselines are reproduced or adapted following their published descriptions and evaluated with the same UGTPHON splits and PER pipeline: ICKR (Han et al., 2024) uses GPT-4o with in-context knowledge retrieval; R-G2P (Zhao et al., 2022) uses its controlled-noise robustness training; MoNoise (Van Der Goot, 2019) (English only) and PolyNorm (Wong et al., 2025) perform lexical normalization before the same word-level ByT5 G2P backend.

Multi-seed stability. We rerun both proposed variants with three random seeds and report mean $\pm$ sample standard deviation in Table 11. English and Vietnamese are stable; Korean varies more, consistent with its smaller test split and larger unseen-form rate.

Paired statistical tests. We additionally run two-sided paired bootstrap tests with 10,000 sentence-level resamples on the checkpoints reported in the main tables. Table 12 reports Δ = Ours $-$ matched baseline for PER_{nc}. All intervals exclude zero, although the English Qwen effect is small.

E Frontier LLM Prompt Template

This section describes the in-context learning setup used to query the frontier-LLM baselines (Appendix D). We use the same chat-message structure across GPT-4o, GPT-5.4-mini, Gemini-2.5-flash, Gemini-3-flash, and Claude-Sonnet-4.6, with decoding parameters `temperature=0` and `max_completion_tokens=512`.

Message structure. For each input sentence S, we issue a single chat request consisting of (i) a language-specific system message, (ii) a set of few-shot exchanges, each as a (*user*, *assistant*) pair, and (iii) the query sentence as the final *user* turn:

```
[system]      language-specific   G2P
instructions
[user]    1. tok₁ \n 2. tok₂ \n . . .
[assistant]    1. ipa₁ \n 2. ipa₂ \n
. . .
. . . (few-shot pairs)
[user]    1. w₁ \n 2. w₂ \n . . .
```

Both input and output use a numbered one-token-per-line format, and responses are parsed with the regex `^(\d+)[.):\s]\s*(.*)$`. Each token in a sentence is presented to the model as one numbered line, and the model returns the IPA transcription on the same line number.

Shot scaling. The nominal $K \in \{0, 9, 27\}$ settings in Figure 4 correspond to: K=0, with no few-shot pairs; K=9, with one pair per available UGT category; and K=27, with three pairs per available UGT category. Thus, the actual numbers of in-context examples are $0/9/27$ for English and Vietnamese and $0/7/21$ for Korean, since A2 and A3 do not appear in Korean. Few-shot example pairs are sampled from the training split, drawn once per UGT category, and reused across all queries.

System messages. The system message specifies the G2P task and target IPA conventions for each language. We reproduce the English system message below; the Vietnamese and Korean variants follow the same template, written in the target language and referencing language-specific phonological conventions (Vietnamese tone marks, Korean tense consonants and consonant assimilation).

```
You are a grapheme-to-phoneme (G2P)
expert for English. You will receive
a numbered list of tokens from a
sentence (possibly containing informal,
abbreviated, or slang forms). Output the
IPA transcription for each token in the
same numbered order.
Rules:
- Each line of input is ONE token,
identified by its number (e.g. '3.
idk').
- Output exactly one numbered IPA line
per token: '3. ˈaɪ dount noʊ'.
- A token may contain spaces (e.g. a
NeMo-expanded handle) — treat it as one
unit.
- For abbreviated/informal tokens
(e.g. pls→please, idk→i don't know,
sooo→so), first determine the standard
form, then transcribe.
- For repeated chars/words (e.g. frfr,
sooooo), transcribe the normalized form.
- Use standard IPA notation.
- Output ONLY numbered IPA lines, no
explanations or extra text.
```

Few-shot example pools. For each language and UGT category, we maintain a fixed pool of three (*token list*, *IPA list*) pairs and draw n pairs per available category in order, where $n = 1$ for the nominal K=9 setting and $n = 3$ for the nominal K=27 setting. Because Korean lacks A2 and A3, these settings contain 7 and 21 actual in-context examples for Korean, respectively. Each pair is presented at the full sentence level so the model also

Model	English		Vietnamese		Korean	
	PER_c	PER_{nc}	PER_c	PER_{nc}	PER_c	PER_{nc}
ByT5 (300M)	5.2	29.8	7.7	43.2	4.3	47.5
Qwen2.5-0.5B (naive FT)	11.2	14.8	4.4	23.1	14.2	77.3
ICKR (GPT-4o, reproduced)	7.7	14.9	7.9	30.8	8.2	47.9
R-G2P (reproduced)	18.2	53.2	6.9	59.3	4.1	42.3
PolyNorm (GPT-4) + ByT5	25.8	44.5	15.6	49.3	15.5	**28.5**
MoNoise + ByT5	14.2	18.0	–	–	–	–
Ours (Qwen2.5-0.5B)	10.2	**14.3**	4.0	**20.0**	6.0	32.4
Ours (ByT5, 300M)	**4.7**	22.5	**5.9**	26.7	5.1	32.0

Table 10: Task-relevant and matched-backbone comparisons on UGTPHON (PER, %). Bold marks the best value among the listed systems for each metric. Normalization-first systems can improve PER_{nc} while substantially increasing PER_c.

Backbone	Lang.	PER_c	PER_{nc}
Qwen2.5-0.5B	EN	10.07±0.15	14.03±0.31
	VI	3.93±0.06	20.03±0.06
	KO	4.47±1.39	29.57±2.60
ByT5	EN	4.70±0.10	20.60±2.07
	VI	5.83±0.06	27.07±0.32
	KO	4.73±0.32	30.17±1.59

Table 11: Three-seed mean ± sample SD PER (%).

Lang.	Backbone	Ours	Base	95% CI of Δ
EN	Qwen	14.3	14.8	$[-0.8, -0.1]$
EN	ByT5	22.5	29.8	$[-8.9, -3.4]$
VI	Qwen	20.0	23.1	$[-5.4, -1.3]$
VI	ByT5	26.7	43.2	$[-18.3, -13.8]$
KO	Qwen	32.4	77.3	$[-47.9, -43.1]$
KO	ByT5	32.0	47.5	$[-18.4, -10.3]$

Table 12: Paired-bootstrap PER_{nc} comparisons (10,000 sentence-level resamples).

observes that canonical words inside a sentence should be phonemized normally. We illustrate one (*user*, *assistant*) pair per category for English in Table 13.

F Downstream TTS-ASR Evaluation

To examine whether token-level PER gains transfer to synthesized speech, we conduct TTS-ASR evaluations in English and Vietnamese using eSpeak NG and Whisper large-v3.

Vietnamese evaluation. We additionally evaluate 297 Vietnamese examples. Table 16 reports sentence-level WER with 95% bootstrap confidence intervals from 10,000 resamples. Our Qwen2.5-0.5B variant attains the lowest observed WER among predicted-input systems (0.640), followed by our ByT5 variant (0.660).

Setup. We sample 100 sentences from UGTPHON's English test split, stratified by dominant UGT macro-type (30 Reconstruct, 20 Repetition, 50 Pass-through; mean length 11.7 tokens, 1,169 valid tokens total) under seed 42. For each system, the predicted IPA is synthesized with eSpeak NG 1.50 (`en-us` voice, direct IPA input) and transcribed by OpenAI Whisper large-v3 (Radford et al., 2023) (temperature 0, fp16). We compute WER between the transcript and the NeMo-normalised gold sentence. Five systems are compared: *Canonical* (gold IPA from G_c), *Word ByT5* (strongest non-retrieval baseline), *GPT-4o (9-shot)*, and our approach with the Qwen2.5-0.5B and ByT5-small backbones.

Caveats. For the English evaluation, $N = 100$ makes absolute WER differences below roughly ± 0.02 unreliable without a bootstrap interval, which we do not compute. *Canonical* is not a strict upper bound: gold IPA contains length and secondary-stress marks (ː, ˈ) that eSpeak NG does not always realize, so the gap to Canonical overestimates the achievable improvement.

Overall WER. Table 14 reports overall sentence-level WER. Ours (ByT5) achieves the lowest WER (0.359), below Word ByT5 (0.368, −0.009) and GPT-4o (0.382, −0.023). The downstream ranking matches the token-level PER ranking across all four trained systems, indicating that PER on UGTPHON is a faithful proxy for downstream intelligibility on this evaluation. Ours (Qwen) underperforms (0.436), consistent with its higher token-level PER (0.140).

Per-UGT-type WER. Table 15 breaks WER down by UGT category. Ours (ByT5) wins on $5/9$ categories (A1, A2, C1, C2, C3), with the largest

Cat.	User (token list)	Assistant (IPA list)
A1	*Some / guys / hv / a / bad / game*	ˈsəm / ˈgaɪz / ˈhæv / ə / ˈbæd / ˈgeɪm
A2	*bruh / y / u / not / playin*	bɹʌ / ˈwaɪ / ˈju / ˈnɑt / ˈpleɪɪŋ
A3	*naah / im / just / idk / sorry*	ˈnɑ / ˈaɪm / ˈdʒəst / ˈaɪ doʊnt noʊ / ˈsɑɹi
B1	*yeah / like / a / loooot*	ˈjæ/ ˈlaɪk / ˈeɪ / ˈlʊt
B2	*frfr / goodmorning*	fɹɝ fɹɝ / ˈɡʊdˌmɔɹnɪŋ
C1	*wat / u / kno / bout / hockey*	ˈwət / ˈju / ˈnoʊ / əˈbaʊt / ˈhɑki
C2	*Wats / goin / on*	ˈhwəts / ˈɡoʊɪŋ/ ˈɑn
C3	*goin / wit / da / crew*	ˈɡoʊɪŋ/ ˈwɪθ / ðə / ˈkɹu
C4	*i / thought / of / u / omg*	ˈaɪ / ˈθɔt / əv / ˈju / ˈoʊ ˈmaɪ ˈɡɑd

Table 13: English few-shot example pool: one representative (*user*, *assistant*) pair per UGT category, drawn first when $K = 9$. "/" separates the per-token entries that the prompt presents on numbered lines.

System	$\text{PER}_{\text{overall}} \downarrow$	WER ↓
Canonical (gold IPA)	—	0.410
GPT-4o (9-shot)	0.111	0.382
Word ByT5	0.078	0.368
Ours (Qwen 2.5-0.5B)	0.140	0.436
Ours (ByT5)	**0.066**	**0.359**

Table 14: Token-level overall PER and sentence-level WER on the 100-sentence English TTS-ASR evaluation. Lower is better. Bold: best per column.

Type	Can.	Word ByT5	GPT-4o (9-shot)	Ours Qwen	Ours ByT5
A1	0.541	0.438	0.481	0.492	**0.423**
A2	0.396	0.466	0.425	0.547	**0.379**
A3	**0.431**	0.485	0.462	0.559	0.459
B1	0.363	**0.271**	0.327	0.341	0.303
B2	0.369	0.301	**0.266**	0.372	0.369
C1	0.392	0.409	0.467	0.469	**0.378**
C2	0.352	0.324	0.356	0.385	**0.318**
C3	0.387	0.309	0.331	0.374	**0.313**
C4	0.456	0.371	**0.361**	0.448	0.429

Table 15: Sentence-level WER by UGT category on the 100-sentence English TTS-ASR evaluation. Bold: best per row. Ours (ByT5) wins on $5/9$ categories, with the largest gains on Reconstruct (A1–A2) and Pass-through phonetic respelling (C1–C3), where retrieved canonical-form evidence directly resolves the surface.

gains on recoverable shortenings: A2 (-0.087 vs. Word ByT5) and A1 (-0.015). Where our approach does not win, the failure mode is interpretable: A3 (phrasal abbreviation, e.g., *idk*, *nvm*) requires multi-word expansion that the single-token retrieval store does not cover; B-type repetition is already handled by the byte-level encoder of Word ByT5; and C4 (phrasal acronyms) benefits from GPT-4o's broader world knowledge. Categories that benefit most from retrieval at the token level (A1–A2, C1–C3) are also where our model wins downstream, corroborating that the taxonomy localizes the same difficulty axis at both levels.

System	WER ↓	95% CI	UGT PER ↓
Gold canonical input	0.634	$[0.607, 0.661]$	0.000
Word ByT5	0.667	$[0.640, 0.694]$	0.432
GPT-4o (few-shot)	0.712	$[0.687, 0.737]$	0.456
Ours (Qwen 2.5-0.5B)	**0.640**	$[0.613, 0.666]$	**0.200**
Ours (ByT5)	0.660	$[0.633, 0.686]$	0.272

Table 16: Vietnamese downstream TTS-ASR evaluation on 297 examples. WER confidence intervals use 10,000 bootstrap resamples. Lower is better; bold marks the best predicted-input system.