STREAMING RUSSIAN-UZBEK TEXT TRANSLATION UNDER CODE-SWITCHING CONSTRAINTS

Sukhrob Avezov Sobirovich
PhD, Lecturer in the Department of Russian Language and Literature
Bukhara State University
senigama1990@mail.ru

Abstract
In this article, we investigate streaming Russian-Uzbek machine translation under dense, spontaneous code-switching. We propose a latency-controlled Transformer that combines wait-k scheduling with monotonic chunkwise attention, augmented by script and morphology-aware tokenization and a boundary-sensitive read/write policy. On chat-style test sets, the system delivers higher BLEU and chrF at the same Average Lagging and better preserves switch points and Russian stems bearing Uzbek suffixes.

Keywords: Streaming translation, Russian-Uzbek, code-switching, wait-k, monotonic attention, morphology-aware tokenization, Average Lagging, chat translation.

Introduction

Russian-Uzbek bilingual communication in Central Asia exhibits frequent intra-sentential code-switching, mixing Russian lexemes, Uzbek morphology, and dual scripts (Cyrillic/Latin). In instant messaging and customer-support chats, translation must be delivered while the source is still being typed. This setting combines three pressures: (i) unpredictable switch points, (ii) agglutinative morphology that attaches Uzbek suffixes to Russian stems (e.g., отчётlarni yuboraman), and (iii) strict latency budgets. We address these constraints with a streaming architecture that reasons about switch boundaries and inflection while maintaining controllable delay.

Methods and Related Work

Foundational work modeled code-switching as structured alternation with quantifiable constraints [1] and as matrix/embedded language interplay [2]. For streaming MT, prefix-to-prefix decoding with a fixed wait-k delay [3] established a clean latency-quality trade-off, while monotonic chunkwise attention [4] enabled incremental alignments without full look-ahead. We ground our approach in these ideas, adapting them to an agglutinative/inflectional pair with mixed scripts and pervasive lexical borrowing.

Model and training. Encoder–decoder. We start from a Transformer base. The encoder processes a growing source prefix; the decoder is trained in a prefix-to-prefix regime. We expose two latency controls: wait-k (emit after reading k tokens), and MoChA (learned monotonic attentional boundaries with small chunk glimpses).

Morphology and script-aware tokenization. We build a joint vocabulary over Russian Cyrillic and Uzbek Latin/Cyrillic. Before subwording, a light morphological splitter detaches common

Uzbek clitics and case suffixes (-ni, -ga, -da, -dan, -lar, -miz, -mi), but retains Russian stems intact to preserve cognate form — e.g., *договор+larni*. This reduces sparsity at switch sites without forcing heavy morphological analyzers.

Switch-boundary features. A shallow BiGRU tagger predicts probable switch boundaries and attaches two binary features to encoder states: SWITCH_UPCOMING and RU_STEM+UZ_SUFFIX. These are trained from weak silver labels derived from script cues, lexicon membership, and affix templates.

Read/write policy. We complement wait-k with a boundary-aware rule: if a suffixal morpheme is detected mid-chunk, we defer emission for up to 2 extra tokens to avoid truncating an inflectional unit.

Objectives. We optimize standard cross-entropy with label smoothing plus two auxiliary losses: (i) boundary preservation (penalizes deletions of predicted switch tokens), and (ii) latency regularization (discourages long consecutive READs).

Data. We compile a chat-like parallel corpus from publicly available bilingual forums and help-desk transcripts that permit research use, applying de-identification. We stratify by code-switch density: CS-light (<10% foreign tokens), CS-mid (10–30%), CS-heavy (≥30%). We construct RU→UZ and UZ→RU splits and reserve a streaming dev/test where messages arrive in 3–7-token bursts to simulate typing.

Results

Evaluation protocol. We report Average Lagging (AL) and Differentiable Average Lagging (DAL) to characterize delay, and BLEU and chrF for n-gram quality. For human-perceived adequacy under switching, we additionally collect a small Boundary Fidelity judgment: raters mark whether switch points and inflected stems are preserved without awkward reordering. Each streaming configuration is tuned on dev to meet an AL budget of ≈3, 5, or 7 tokens.

Main comparison. Under matched latency, the boundary-aware model raises quality and better preserves switch structure than standard wait-k.

Model	Policy	AL ↓	RU→UZ BLEU ↑	UZ→RU BLEU ↑	chrF ↑	Boundary Fidelity ↑
Offline Transformer	full-context	∞	29.8	27.3	58.4	0.90
Re-translation (chunk, 6-token)	fixed chunk	6.1	27.2	25.3	55.6	0.78
Wait-k baseline	k=3	3.2	26.1	24.3	54.1	0.79
Ours: wait-k + MoChA + boundary	k=3	3.3	27.0	25.1	55.0	0.86
Wait-k baseline	k=5	5.1	27.1	25.4	55.2	0.82
Ours: wait-k + MoChA + boundary	k=5	5.2	28.0	26.2	56.2	0.88
Wait-k baseline	k=7	7.0	27.6	25.9	55.7	0.84
Ours: wait-k + MoChA + boundary	k=7	7.1	28.6	26.8	57.0	0.89

Observations. Gains are largest at lower AL, where the boundary-aware delay avoids splitting Uzbek suffix chains attached to Russian stems. The gap narrows as AL grows and the task approaches offline conditions.

Impact of code-switch density. When stratified, improvements concentrate in CS-mid and CS-heavy subsets. At k=5, RU→UZ BLEU rises +1.4 in CS-mid and +1.8 in CS-heavy over the vanilla wait-k, versus +0.6 in CS-light. Boundary Fidelity increases by ~0.06–0.09 absolute in the denser regimes, reflecting better handling of embedded Russian nouns with Uzbek case markers (справка-ni, маршрут-ga).

Ablations. Removing the morphology-aware pretokenizer reduces chrF by ~0.9 and increases AL by +0.2 (the policy hesitates more often). Dropping switch-boundary features costs ~0.8 BLEU and lowers Boundary Fidelity by 0.05, primarily due to premature emission at the clitic -mi and at encliticized case markers. Replacing MoChA with soft attention under wait-k slightly helps BLEU in CS-light but hurts latency stability in CS-heavy, raising the standard deviation of segment-level AL.

Qualitative analysis. We examine frequent error patterns.

1. Premature commitment. Baselines emit a Russian lemma before seeing the Uzbek suffix, forcing later repairs. Our policy waits briefly to ingest the suffix, producing natural Uzbek case morphology in one pass.
2. Script normalization. Mixed script in source (договоplarni) is standardized internally but surfaces as expected in target.
3. Anticipation vs. safety. At low k, anticipating verbs from context occasionally misfires in polite forms; boundary cues reduce such errors by deferring until the Uzbek politeness clitic appears.

Discussion

Linguistic and engineering implications. The results support long-standing observations that code-switching is not random but patterned [1] [2]. In a streaming MT system, honoring those patterns means aligning read/write behavior with morphosyntactic units. Wait-k provides a simple contract with the user about latency; MoChA adds elasticity around prosodic/morphological boundaries; and light, language-specific tokenization yields most of the benefit without heavyweight analyzers. For production chat, the takeaway is pragmatic: carry a small look-ahead budget that you can «spend» only when boundary predictors fire.

Conclusion

We presented a streaming Russian–Uzbek MT system tailored to code-switching. By combining wait-k, monotonic chunkwise attention, and lightweight morphology/script-aware preprocessing with boundary-sensitive read/write decisions, we improved translation quality at matched latency and preserved switch structure. The approach is simple to deploy, language-pair aware, and ready to extend to other agglutinative/inflectional pairs in live chat.

References

1. Poplack S. “Sometimes I'll start a sentence in Spanish Y TERMINO EN ESPAÑOL”: Toward a typology of code-switching //Linguistics. – 2013. – Т. 51. – №. s1. – С. 11-14.
2. Myers-Scotton C. Duelling languages: Grammatical structure in codeswitching. – Oxford University Press, 1997.
3. Ma M. et al. STACL: Simultaneous translation with implicit anticipation and controllable latency using prefix-to-prefix framework //arXiv preprint arXiv:1810.08398. – 2018.
4. Chiu C. C., Raffel C. Monotonic chunkwise attention //arXiv preprint arXiv:1712.05382. – 2017.
5. Авезов С. О КОРПУСНОЙ ЛИНГВИСТИКЕ, ТРУДНОСТЯХ ПЕРЕВОДА И ПРИНЦИПАХ ОРГАНИЗАЦИИ ПАРАЛЛЕЛЬНЫХ КОРПУСОВ ТЕКСТОВ //«УЗБЕКСКИЕ НАЦИОНАЛЬНЫЕ ОБРАЗОВАТЕЛЬНЫЕ ЗДАНИЯ ТЕОРЕТИЧЕСКОЕ И ПРАКТИЧЕСКОЕ СОЗДАНИЕ ВОПРОСЫ" Международная научно-практическая конференция. – 2022. – Т. 1. – №. 1.

Web of Discoveries: Journal of Analysis and Inventions
webofjournals.com/index.php/3