TurboBias: Universal ASR Context-Biasing Powered by GPU-Accelerated Phrase-Boosting Tree

Andrei Andrusenko, Vladimir Bataev, Lilit Grigoryan, Vitaly Lavrukhin, Boris Ginsburg | ***NVIDIA***

ASRU 2025

Poster No. 142

Introduction

Motivation: Recognizing domain-specific key phrases is crucial for contextual Automatic Speech Recognition (ASR). However, most existing biasing approaches either **require additional model training** (deep-fusion), **slow down decoding considerably** (shallow-fusion), **restrict the range of supported ASR model architectures**, or are **limited in open access**.

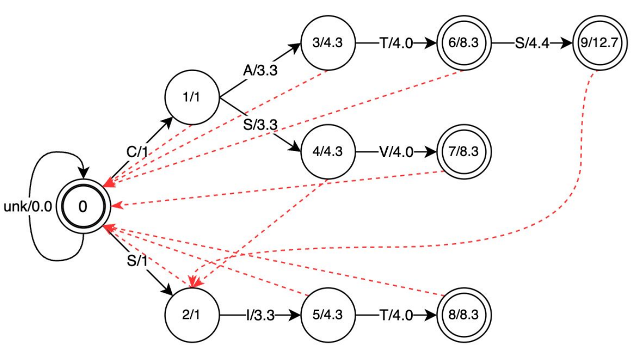
Solution: This work introduces a universal **GPU-accelerated** phrase-boosting framework (GPU-PB) supporting **CTC**, **Transducer (RNN-T)**, and **AED** ASR models. The method employs a phrase-boosting tree **with a modified scoring wight distribution**. This enables high-accuracy recognition in both **greedy** and **beam-search modes** with only 2–5% runtime overhead. The approach maintains strong efficiency even with large vocabularies (up to 20K phrases) and consistently outperforms open-source biasing baselines in both speed and accuracy.

The proposed method is open-sourced in the NVIDIA NeMo toolkit.

Method

1. Build a standard prefix tree based on the Aho-Corasick algorithm for tokenized phrases
2. Modify a uniform weights distribution by increasing the next transition score based on the depth of the tree with logarithmic dependency
3. Convert the obtained prefix tree into an efficient GPU-based structure (Triton kernel) supporting the use of queries across the entire vocabulary through index-select operation

The method supports GPU-based greedy and beam-search decoding for CTC, RNN-T, and AED models



T	H	E	C	A	T	I	S	S	I	T	T	I	N	G
0	0	0	1	3.3	4	0	1	-1+1	3.3	4	-8.3	0	0	0

Figure 1. An example of a boosting tree for words "CAT, CATS, CSV, SIT" with character-level tokenization

Experimental setup

ASR models: Hybrid Transducer-CTC, and AED (Canary) with the same FastConformer encoder (114M), trained on ~24k hours of English data with 1024 BPE tokens

Test data: CSTalks (8.3h of computer science domain), Earnings 21 (10h of earnings calls), MultiMed (13.5h of medical data).

Metrics: overall WER, F-score for key phrases, inverse Real Time Factor (RTFx) for speed

Table 1. Performance evaluation of the proposed GPU-PB method in the context-biasing task for CTC, RNN-T, and AED models in greedy and beam-search decoding modes

Model	Decod	GPU PB	CSTalks F-score (P/R)↑	CSTalks WER↓	Earnings21 (10h) F-score (P/R)↑	Earnings21 (10h) WER↓	MultiMed F-score (P/R)↑	MultiMed WER↓	RTFx↑ Avg.
CTC	greedy	–	35.0 (97/21)	13.7	45.7 (94/30)	15.6	54.0 (95/38)	15.0	2181
		✓	64.8 (94/50)	11.9	53.5 (92/38)	15.6	60.2 (93/45)	14.9	2067
	beam	–	35.0 (97/21)	13.8	45.7 (94/30)	15.6	54.0 (95/38)	15.0	1874
		✓	**83.2** (90/77)	**10.2**	**67.8** (89/55)	**15.5**	**71.8** (89/60)	**14.3**	1786
RNN-T	greedy	–	42.5 (96/27)	12.8	56.0 (95/40)	15.1	60.4 (95/44)	13.9	1822
		✓	70.4 (92/57)	10.7	63.3 (93/48)	15.0	66.3 (91/52)	13.7	1751
	beam	–	44.2 (97/29)	12.8	55.8 (93/40)	14.3	62.5 (95/47)	13.6	1466
		✓	**82.9** (90/76)	**9.6**	**74.0** (88/64)	**14.2**	**75.8** (89/66)	**12.9**	1420
AED	greedy	–	52.6 (97/36)	12.7	54.7 (92/39)	15.4	64.0 (94/49)	14.1	356
		✓	75.6 (93/64)	10.4	63.8 (91/49)	15.3	69.3 (89/57)	13.9	350
	beam	–	53.7 (97/37)	12.5	54.6 (92/39)	15.2	65.7 (94/51)	13.7	145
		✓	**82.4** (94/73)	**10.2**	**66.2** (88/53)	**15.1**	**75.5** (88/66)	**13.1**	141

Table 2. Performance comparison of GPU-PB

Decod.	C-Biasing	F-score (P/R)↑	WER↓	RTFx↑
CTC				
greedy	–	35.0 (97/21)	13.7	2232
	CTC-WS	**79.8** (90/72)	**10.9**	906
	NGPU-LM	58.5 (96/42)	11.9	2123
	GPU-PB_{uw}	53.7 (96/37)	13.2	2181
	GPU-PB	64.8 (94/50)	11.9	1991
beam	–	35.0 (97/21)	13.8	1883
	Pyctcdecode	74.0 (93/62)	11.5	29
	NGPU-LM	55.2 (97/39)	12.5	1807
	GPU-PB_{uw}	75.0 (94/62)	11.5	1784
	GPU-PB	**83.2** (90/77)	**10.2**	**1777**
RNN-T				
greedy	–	42.5 (96/27)	12.8	1832
	CTC-WS	**80.0** (90/72)	**10.1**	639
	NGPU-LM	68.5 (96/54)	10.8	1812
	GPU-PB_{uw}	65.0 (94/50)	12.1	1759
	GPU-PB	70.4 (92/57)	10.7	1753
beam	–	44.2 (97/29)	12.8	1467
	NGPU-LM	58.6 (97/42)	12.0	1426
	GPU-PB_{uw}	80.9 (93/72)	10.1	1417
	GPU-PB	**82.9** (90/76)	**9.6**	**1430**

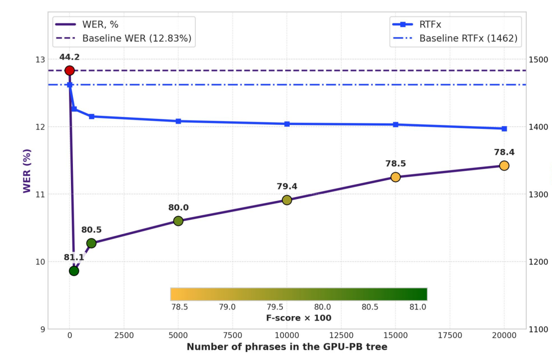


Figure 2. GPU-PB robustness to the number of key phrases

Conclusion

We proposed a universal ASR context-biasing framework with the following:

- Support all major ASR models: CTC, RNN-T, and AED
- Application in greedy and beam-search with only 2-5% RTFx overhead
- Average F-score improvement by 12-15% in greedy and 17-20% beam-search
- Robustness to the context list size growth up to 20K phrases
- Open-sourced implementation as a part of NeMo toolkit