Real-Time Correction of Named Entities in Speech Transcriptions: A Lightweight Approach

Kiranmayi Gandikota, Anunay Katare, Chandan Kumar Pandey, Mayank Kumar Tyagi, Subhadarshi Panda, Rahul Kumar, Mohamed Asfak R
Samsung R&D Institute India-Bangalore

***Abstract*—End-to-end on-device Automatic Speech Recognition (ASR) systems have demonstrated remarkable accuracy and efficiency in recent years. However, challenges persist in correctly transcribing infrequent named entities (e.g., geographical locations, business entities, person names, etc.) and handling diverse user accents, which remain underrepresented in training datasets. While information retrieval augmentation or Retrieval Augmentation Generation (RAG) for correction of named entities has shown promise in knowledge-grounded NLP tasks when paired with large language models (LLMs), its integration into real-time on-device systems is non-trivial due to computational constraints.We introduce a novel lightweight method combining phonetic-aware retrieval, vector-based semantic search and generative correction. The system leverages a lightweight phonetic index for rapid candidate entity retrieval and a dense vector-embedding module to refine predictions as well as model the ASR error output distribution in generative space. Additionally, we introduce a novel approach to model ASR errors in natural language. Experiments on test sets emphasizing place names, monuments, airports, and landscapes yielded an increase in correct Named Entity (NE) recognition accuracy by 9.6% compared to baseline. These gains underscore the efficacy of hybrid retrieval-generation paradigms in resource-constrained environments.**

I. Introduction

The proliferation of on-device artificial intelligence has driven significant advancements in real-time speech recognition, yet challenges persist in accurately transcribing rare and context-dependent named entities (NEs), such as geographical locations, cultural landmarks, and specialized locale specific terms. These entities often exhibit low representativeness in training datasets and are susceptible to accent or dialect variability, leading to elevated word error rates (WER) in end-to-end ASR systems. Several end user applications are very susceptible to this, hampering user experience.

While retrieval-augmented generation (RAG) or information augmentation and large language models (LLMs) have demonstrated efficacy in knowledge-grounded NLP tasks [1], their integration into resource-constrained on-device environments remains computationally prohibitive or difficult to implement for realization at scale. To bridge this gap, we propose a novel hybrid framework that combines phonetic-aware retrieval and semantic vector search. To generalize and learn ASR output distribution we propose a generative on-device solution.

Our approach introduces a lightweight phonetic index (constructed via Double Metaphone [2] and Soundex [3]) for rapid candidate retrieval, complemented by a dense embedding space optimized with Triplet Loss to disambiguate semantically proximate entities. Critical to deployment, the system leverages Scalable Nearest Neighbors (ScaNN) [4] for efficient nearest-neighbor search and is tailored for on-device execution via model quantization and architecture pruning. To further generalize and model the error distribution induced by Automatic Speech Recognition (ASR), we propose a T5-based generative solution. Additionally, we introduce a novel methodology for generating training and testing data to support this framework.

Empirical evaluations on datasets emphasizing names of people, places of interest, monuments, and transportation hubs reveal a 9.6% improvement in Word Recognition Rate (WRR) for named entities, underscoring the framework's potential to enhance user experience in multilingual and low resource scenarios. This work advances the state of on-device ASR by harmonizing real-time performance with nuanced entity understanding, paving the way for more adaptive and inclusive voice interfaces for various AI based end user applications.

II. Related Work

Traditional ASR error correction techniques have leveraged noisy channel models [5] and statistical machine translation frameworks [6] to generate more plausible text sequences. These methods, while effective in general correction scenarios, often fall short in recovering rare or domain-specific entities due to limited lexical coverage. Later developments incorporated confusion networks and N-best hypotheses [7] to improve correction robustness, but such methods are often tightly coupled with the underlying ASR system and require access to decoder internals.

Post-ASR NE correction has emerged as a promising alternative, treating the task as a form of sequence correction or normalization. Earlier efforts in this direction included rule based string matching [8], fuzzy matching using phoneme level edit distances [9], and candidate reranking based on similarity measures [10]. More recently, neural approaches have shown superior performance by learning contextual representations and mapping misrecognized tokens to their correct forms. These include classification-based correction models as well as embedding-based approaches using triplet loss or contrastive learning to align corrupted and correct entity pairs [11].

To improve correction accuracy for rare or out-of-vocabulary entities, several systems augment language models

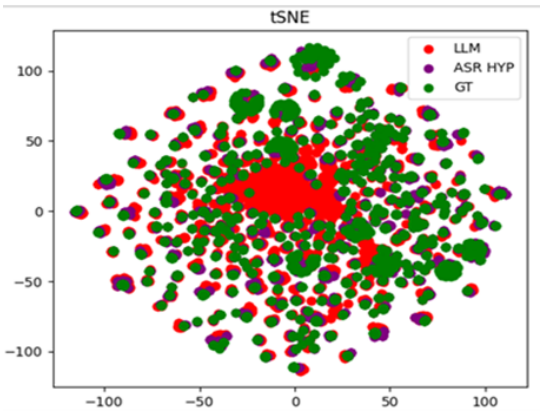

Fig. 1. t-SNE plot showing original and generated NEs.

with external knowledge sources such as entity dictionaries or vector search indices. Retrieval-augmented generation (RAG) [1] and dual encoder models [11] allow dynamic retrieval of entity representations, enabling the correction system to ground its predictions in real-world entity databases. This combination of neural correction with symbolic or retrieval-based grounding has proven effective in both academic and industrial settings [12].

III. Method

A. Data Preparation

Due to unavailability of dataset that is directly applicable for named entity correction, we developed dataset for training and evaluation internally. To enhance on-device ASR performance for Indian geographical entities (e.g., place names, monuments, airports, etc.), we systematically prioritized these categories based on empirical ASR error patterns. We focused mainly on the following categories: Person, Location, Food and Miscellaneous. Our initial experiments with limited categories (Person and Location only) resulted in overfitting of model, as several Indian person and location names may overlap with other named entities like Food. Therefore, to enable better generalisation, we added more categories like Food and Miscellaneous. The final data used had the distribution of categories as 40% Location, 30% Person, 20% Food and 10% Miscellaneous. To construct a robust dataset for rare named entities, we aggregated foundational data from structured sources (e.g., Wikipedia) and unstructured user reviews. Synthetic data was further enriched using large language models (LLMs) to generate contextually relevant sentences. Addressing the challenge of limited real-world audio data, we fine-tuned an LLM on a constrained ASR dataset to emulate realistic recognition errors. This approach enabled the simulation of phonetic and contextual ASR misinterpretations, thereby enhancing the diversity and fidelity of training data for underrepresented entities. High-quality reference sentences were generated using a large language model (LLM), validated by linguistic experts for accuracy. A data augmentation pipeline was developed to simulate realistic ASR errors: phonetic rules were applied to mimic accent-specific variations (e.g., substituting ‘s’ with ’sh’), contextual perturbations were introduced by altering either the named entities within fixed sentences or rephrasing sentences around unchanged entities, and LLMs were employed to create new sentences centered on erroneous transcriptions. This targeted approach enriches training data, improving recognition of underrepresented terms in real-world scenarios.

Data underwent rigorous filtering through a two-stage process: first, deduplication via hashing and string similarity metrics to eliminate redundant entries; second, embedding-based clustering utilizing Sentence-BERT [13] to generate semantic sentence embeddings, followed by dimensionality reduction (t-SNE [14] and PCA [15]) for cluster visualization and subsequent removal of erroneous data generated which is far from the actual error distribution. The final data used was about 50k samples large, with a split of ratio 80:10:10 for train, validation and test splits respectively for all experiments. The tSNE plot (Figure 1) shows the LLM generated erroneous NE, actual ASR error (ASR HYP) and correct NE (GT) follow similar distribution. This pipeline ensured dataset integrity while preserving linguistic diversity.

B. Phonetic retrieval

The phonetic retrieval module enhances on-device ASR by using a database-driven approach to correct named entity recognition errors, constructing a phonetic index with algorithms like Double Metaphone and Soundex. Double Metaphone improves fuzzy string matching by encoding words into phonetic codes, enabling matches for similarly sounding strings (e.g., “Robert” and “Roberts”). It generates two keys per word to account for multiple pronunciations and applies rules to replace consonants with phonetic equivalents while handling exceptions. Double Metaphone was chosen for its language-agnostic nature, enabling cross-lingual compatibility and generating phonetic encodings that reflect spoken pronunciations, enhancing search accuracy. It excels in spell correction by identifying similarly sounding words and is fast, making it ideal for performance-critical applications. Even with so many advantages of Double Metaphone, there were collisions in similar sounding words in our database. For example, consider the encoding for following two location names in India: (a) Kozhikode: KJKT vs Calicut: KLKT (b) Mangalore International: MNKLRNTRNXNL vs Bangalore International: PNKLRNTRNXNL. We are using Levenshtein distance threshold between phonetic encoding to retrieve the named entities, collisions are bound to happen in the above cases. So, we combined the two well-known encoding algorithms Double Metaphone and Soundex.

Using double representation helps in rich encoding and reduces collision. For the above two examples the combined encodings become: (a) Kozhikode: KJKTK223 vs Calicut: KLKTC423 (b) Mangalore International: MNKLRNTRNXNLM524 vs Bangalore International: PNKLRNTRNXNLB524

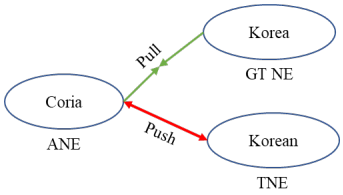

Fig. 2. Triples.

Clearly, the distance and overlap between the colliding entities has increased and hence resolving the collision.

During inference, the module calculates both phonetic (phoneme level) and character (word-level) distances between the input entity and indexed entries. A cascaded filtering process then applies edit and phonetic distance-based thresholds and re-ranking mechanisms to dynamically refine candidate entities, prioritizing those with the highest similarity scores. This hybrid distance metric and adaptive rejection strategy significantly improve correction accuracy for underrepresented or mispronounced entities in real-time scenarios. The module enables efficient resolution of ASR-related issues, ensuring prompt mitigation of errors.

C. Vector retrieval

To address phonetically and semantically similar erroneous named entities (NEs), we fine-tuned a dense embedding model using a Triplet Loss framework. The anchor samples were derived from ASR-transcribed NEs(ANE), while the positive samples corresponded to ground-truth (GT) NEs. Negative samples were selected as the top 10 most similar NEs (TNE) from the original embedding space of the GT entities, ensuring robust discrimination between correct and near-duplicate distractors(Figure. 2). Margin is a hyperparameter that enforces a minimum difference between the two distances. The resulting embeddings were integrated with the ScaNN [4] library to enable efficient nearest-neighbor search, and the model was optimized for on-device deployment by porting to a lightweight, compatible architecture. This pipeline enhances error correction by leveraging both semantic proximity and computational efficiency in resource-constrained environments.

$$\begin{aligned}\text{Triplet Loss} = max(&\text{distance(ANE,GT NE)} \\ &-\text{distance(ANE,TNE)} + \text{margin}, 0)\end{aligned}$$

D. Joint NER and Vector Embedding

Named Entity Recognition (NER) is pivotal for correcting ASR outputs, addressing challenges in identifying NEs across diverse regions and resolving errors caused by acoustic ambiguities or dialectal variations. To address these challenges, we propose a multi-task NER framework that concurrently optimizes two objectives: (1) standard NE classification to predict entity types, and (2) triplet loss training to align latent representations of ASR-generated errors with ground-truth text distributions. This joint training strategy enhances the model's ability to recognize erroneous NEs by enabling the base encoder to learn relationships between ASR error patterns and correct entity distributions.

Our architecture uses BERT-Mini [16] (Figure 3) as a lightweight encoder, leveraging token-level encodings for NE classification and pooled outputs optimized via triplet loss to align correct and ASR-distorted entities, focusing on domain-specific nuances for conversational applications while ensuring efficiency. The loss is taken as weighted sum of Triplet Loss and cross-entropy loss for NER (Equation. 1). N is the total number of tokens, y_i is true NER label, p_i is predicted probability distribution of the token, w_1 and w_2 are the weights to combine the losses.

$$\text{Joint Loss} = w_1 \cdot \text{Triplet Loss} + w_2 \cdot \Big(\sum_{i=1}^{N} -y_i log(p_i) \Big) \quad (1)$$

Consequently, when used in actual ASR systems, the model demonstrates better performance, as it can effectively handle the inherent variability and imperfect outputs typical of speech-to-text transformations. This training strategy ultimately enhances the model's adaptability to diverse and unpredictable input scenarios encountered in real applications.

E. Generative Correction

Retrieval and vector-based approaches excel in NE correction using structured knowledge bases but face scalability limits due to predefined systems, while generative models enhance generality by resolving ambiguities through contextual understanding. A primary challenge involves achieving robust named entity (NE) correction while minimizing generative overhead to reduce computational latency. The proposed approach encloses target NEs within brackets (e.g., [Entity]) as model input (Figure. 4). The system then either generates a corrected NE if the original is erroneous, or returns the original NE if correct, or preserves the input if the model lacks relevant knowledge. This selective generation strategy, producing output only for targeted NEs significantly reduces generated token counts, thereby optimizing inference speed. The model used is t5-efficient-tiny [17], which is smaller in size (15.58 M parameters) and suitable for on-device deployment.

The model is fine-tuned on a balanced dataset comprising ASR (Automatic Speech Recognition) output transcripts and synthetically generated NE errors.

F. Optimizations for On-device Adaptation

Joint modelling of NER and vector embedding of named entities reduces the memory footprint, we further quantize the model using tf-lite dynamic quantization for faster inference and less memory requirement. The model size reduces by 74%.

The base model t5-efficient-tiny [17] is the smallest T5 model with less than 16M parameters. To optimize inference efficiency and extend compatibility with TensorFlow Lite (TFLite), we decoupled the model's encoder and decoder components. This architectural separation enables independent parallel execution, significantly improving throughput.

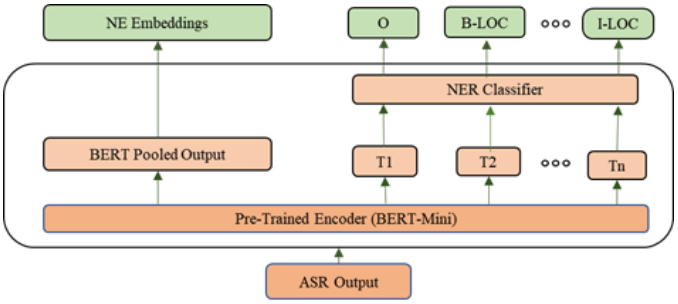

Fig. 3. Joint architecture.

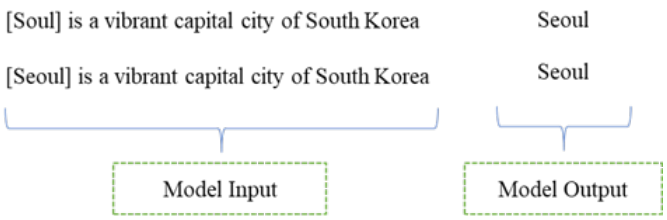

Fig. 4. Example.

Additionally, we applied post-training quantization with 16-bit floating point for activations and 8-bit integers for weights. The model size reduces by 65%.

These optimizations collectively reduce computational latency, memory footprint, and power consumption while maintaining task accuracy.

IV. Results and Discussion

A. NEC and NER Results

We evaluated our method using internal test set, the result shows 9.6% accuracy improvement for NE Word Recognition Rate (WRR) over plain ASR without NEC which was 69.02%.

NER model is an important component to the pipeline. It generalizes well on Indian locale entities and noisy ASR named entities. We have compared our NER on Indian names with INDIC NER [18] on internal dataset. The NER performance outperforms (Table I) INDIC NER with one-tenth of its size.

B. Generative NEC Model Results

To evaluate model robustness, we measure two key metrics: (a) Correction Rate: The proportion of incorrect named entities (NEs) successfully corrected by the model. This metric indicates the model's proficiency in learning ASR error patterns and generating contextually appropriate corrections. (b) Retention Rate: The proportion of correct NEs preserved without modification. This assesses the model's precision in avoiding over-correction, thereby preventing the introduction of errors in real-world scenarios. Note that the test set has all the correct NEs. The above metrics collectively ensure the model balances correction efficacy with reliability. In order to measure the capability of larger LLM in correcting named entities in ASR transcripts, we chose Llama 3.1 8B model since it has been effective in wide range of NLP tasks demonstrating state of the art performance [19]. We ran experiments using few-shot prompting and the results are shown in Table 2 (Table. II). Results show that even larger models lack regional named entity understanding and tend to hallucinate and over correct it.

NER Method	Acc (%)	Size (No. of Parameters)
NEC NER	**96.48**	**14.27M**
INDIC NER	68.98	166.67M

TABLE I
Comparative performance of the proposed NER model with INDIC NER.

Model	Correction Rate (%)	Retention Rate (%)
t5-efficient-tiny	**79.96**	**98.43**
Llama 3.1 8B	35.47	51.5

TABLE II
Comparison of t5-efficient-tiny with a larger LLM for Named Entity Correction

V. Conclusion

In this study, we present a robust lightweight system for named entity correction, characterized by its agility in adapting to diverse dialects and real-time operational capabilities. Our experiments show 9.6% improvement compared to baseline. In the future, we intend to leverage the modular architecture of the solution to expand its coverage to additional NE domains, locales and cultures.

References

[1] V. Karpukhin, B. Oguz, S. Min, P. Lewis, L. Wu, S. Edunov, D. Chen, and W.-t. Yih, "Dense passage retrieval for open-domain question answering," in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, B. Webber, T. Cohn, Y. He, and Y. Liu, Eds. Online: Association for Computational Linguistics, Nov. 2020, pp. 6769–6781. [Online]. Available: https://aclanthology.org/2020.emnlp-main.550/

[2] L. Philips, "The double metaphone search algorithm," *C/C++ Users Journal*, vol. 18, no. 6, pp. 38–43, 2000.

[3] R. Russell and M. Odell, "The soundex indexing system," National Archives and Records Administration, Tech. Rep., 1918.

[4] R. Guo, P. Sun, E. Lindgren, Q. Geng, D. Simcha, F. Chern, and S. Kumar, "Accelerating large-scale inference with anisotropic vector quantization," in *Proceedings of the 37th International Conference on Machine Learning*, H. D. III and A. Singh, Eds. PMLR, 2020, pp. 3887–3896.

[5] E. K. Ringger and J. F. Allen, "Error correction via a noisy channel model," in *Proceedings of the 34th annual meeting on Association for Computational Linguistics*. ACL, 1996, pp. 295–302.

[6] N. Ueffing, G. Haffari, and A. Sarkar, "Translating with learned edit operations on n-best hypotheses," in *HLT-NAACL*, 2003.

[7] L. Mangu, E. Brill, and A. Stolcke, "Finding consensus in speech recognition: Word error minimization and other applications of confusion networks," *Computer Speech & Language*, vol. 14, no. 4, pp. 373–400, 2000.

[8] S. V. Pakhomov *et al.*, "Automated correction of misspellings and incorrect drug names in clinical text," *Journal of the American Medical Informatics Association*, vol. 18, no. 5, pp. 516–525, 2011.

[9] L. Derczynski, A. Ritter, S. Clark, and K. Bontcheva, "Phonetic similarity and edit distance for named entity normalization in noisy user-generated text," in *Proceedings of the ACL 2014 Workshop on Noisy User-generated Text (W-NUT)*, 2014, pp. 19–26.
[10] S. Louvan and B. Magnini, "Neural reranking for named entity recognition from speech," in *Interspeech*, 2020.
[11] R. Xie, Y. Shen, Y. Liu, and X. Ren, "Entity correction via contrastive learning," in *EMNLP*, 2022.
[12] Y. Zhang, J. Lee, and D. Bahdanau, "Improving ner robustness on noisy inputs with knowledge base grounding," in *ACL*, 2023.
[13] N. Reimers and I. Gurevych, "Sentence-bert: Sentence embeddings using siamese bert-networks," 2019.
[14] L. J. P. van der Maaten and G. E. Hinton, "Visualizing high-dimensional data using t-sne," *Journal of Machine Learning Research*, vol. 9, no. Nov, pp. 2579–2605, 2008.
[15] H. Hotelling, "Analysis of a complex of statistical variables into principal components," *Journal of Educational Psychology*, vol. 24, no. 6, pp. 417–520, 498–520, 1933.
[16] I. Turc, M.-W. Chang, K. Lee, and K. Toutanova, "Well-read students learn better: On the importance of pre-training compact models," 2019. [Online]. Available: https://doi.org/10.48550/arXiv.1908.08962
[17] Y. Tay, M. Dehghani, J. Rao, W. Fedus, S. Abnar, H. W. Chung, S. Narang, D. Yogatama, A. Vaswani, and D. Metzler, "Scale efficiently: Insights from pre-training and fine-tuning transformers," 2021.
[18] A. Mhaske *et al.*, "Naamapadam: A large-scale named entity annotated data for indic languages," in *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, A. Rogers, J. L. Boyd-Graber, and N. Okazaki, Eds. Toronto, Canada: Association for Computational Linguistics, July 2023, pp. 10 441–10 456. [Online]. Available: https://doi.org/10.18653/V1/2023.ACL-LONG.582
[19] A. Grattafiori, M. Valko, M. Restrepo, M. Patel, M. Vyatskov, M. Samvelyan, and N. Usunier, "The llama 3 herd of models," 2024. [Online]. Available: https://doi.org/10.48550/arXiv.2407.21783