Striking the Right Chord: A Comprehensive Approach to Amazon Music Search Spell Correction

Siddharth Sharma
eshasidd@amazon.com
Amazon Inc
Sunnyvale, CA, USA

Shiyun Yang
shiyuny@amazon.com
Amazon Inc
Sunnyvale, CA, USA

Ajinkya Walimbe
wajinkya@amazon.com
Amazon Inc
Sunnyvale, CA, USA

Tarun Sharma
tarunsh@amazon.com
Amazon Inc
Sunnyvale, CA, USA

Joaquin Delgado
dejoaqui@amazon.com
Amazon Inc
Sunnyvale, CA, USA

ABSTRACT

Music and media search spell correction is distinct as it involves named entities like artist, album and podcast names, keywords from track titles and catchy phrases from lyrics. Users often mix artist names and keywords from track title or lyrics making spell correction highly contextual. Data drift in search queries caused during calendar event days or a newly released music album, brings a unique challenge of quickly adapting to new data points. Scalability of the solution is an essential requirement as the Music catalog is extremely large. In this work, we build a multi-stage framework for spell correction solution for music, media and named entity heavy search engines. We offer contextual spelling suggestions using a generative text transformer model and a mechanism to rapidly adapt to data drift as well as different market needs by using parameter efficient based fine tuning techniques. Furthermore, using a reinforcement learning approach our spell correction system can learn from a user's implicit and explicit feedback in real-time. Some key components of this system are being used in search at Amazon Music and showing significant improvements in customer engagement rate and other relevant metrics.

CCS CONCEPTS

• **Applied Computing** → **Document management and text processing**.

KEYWORDS

spellcheck, spell correction, neural networks, transformers, sequence to sequence, BART, performance efficient fine tuning, lora

ACM Reference Format:
Siddharth Sharma, Shiyun Yang, Ajinkya Walimbe, Tarun Sharma, and Joaquin Delgado. 2024. Striking the Right Chord: A Comprehensive Approach to Amazon Music Search Spell Correction. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '24), July 14–18, 2024, Washington, DC, USA.* ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/3626772.3661344

SIGIR '24, July 14–18, 2024, Washington, DC, USA

ACM ISBN 979-8-4007-0431-4/24/07
https://doi.org/10.1145/3626772.3661344

1 INTRODUCTION

Music search spell correction presents a distinctive set of challenges compared to traditional spell correction tasks. Unlike generic text, music queries involve intricate entity searches, encompassing artist names, title track, album names and even catchy phrases from lyrics. The amalgamation of words in artist, album and track titles further complicates the correction process. The examples below show misspellings in music search queries encompassing a wide array of variations

(1) spacing issue: *"doda day" -> "do da day"*
(2) error in multiple words : *"lead zepelin" -> "led zeppelin"*
(3) phonetic issues : *"the beetles" -> "the beatles"*
(4) missing special characters: *"pink" -> "p!nk"*
(5) other issues: *"defffleppard" -> "def leppard"*

The contextual nature of music search, heavily reliant on named entities, amplifies the complexity of spell correction. Valid English words may appear misspelled when used in a different context; for instance, *"Weekend"* in 'The Weekend' versus the artist *"The Weeknd"*. Additionally, customer intent in music searches can be ambiguous, with single-word queries often resembling multiple entities in the catalog, like *"Christoph"* potentially referring to the artist *"Christophe"* or *"Christopher"*. This further necessitates leveraging user's behavioural engagement data for accurate correction

Multilingual transliterations with non-Latin languages pose multiple edit distance challenges e.g. *Gaddiyan Uchhiya Rakhiya*[1] was a trending viral song in India last year, but a large proportion of search queries were having more than 2 edit distance issues, thereby negatively impacting retrieval. The dynamic elements introduced by unfamiliar terms and shifts in distribution due to new releases by popular artists, trending music tracks and calendar event days further complicates spell correction process, necessitating an agile system that can adapt to shift in distribution of search queries.

Scalability issues and contextual oversights persist in many existing spell correction methods. The legacy spell corrector at Amazon Music using Noisy Channel method[6] not only lacks capability to solve complex spelling issues but also fails to capture one-off errors observed in low volume queries. [9] uses Symmetric Delete [2] for candidate generation requiring pre-computation and indexing of 1 to 2 edit distance variants per spelling, which is impractical

for extensively large media catalogs. Moreover, it lacks contextual sensitivity. These limitations of existing spell correction methods necessitates a more sophisticated candidate generation approach for Music spell correction issues.

In this paper, we propose a multi-stage framework designed to tackle the challenges associated with different misspelling variations. Our approach is currently being used for offering spell correction suggestions on Amazon Music Search in US. To address the multiple challenges discussed above, the proposed framework comprising of four key components:

(1) A BART[7] Encoder-Decoder Transformer model trained in synthetic misspellings for contextual candidate generation
(2) Performance Efficient Fine Tuning(PEFT) [3] for candidate generator model on user behavioral and other high quality data points using LoRA[4]
(3) Post processing business guardrails to avoid generative model hallucinations and bad suggestions
(4) A UCB[1] Bandit based strategy to downgrade bad spelling auto corrects and adapt to data drift occurred due to new release of a music album or holiday (or other calendar event).

2 TRAINING DATASETS

When delving into music spell correction, a notable challenge arises due to the absence of publicly available datasets tailored to this domain. Moreover, the dynamic nature of music catalogs, constantly evolving with new artists, tracks, albums, and podcasts, further complicates the reliance on static datasets, a mechanism capable of accommodating this ever-expanding catalog is imperative. Given the vast scale of music catalogs, obtaining manually annotated data proves to be an impractical and non-scalable approach. In response to the aforementioned challenges we relied on the following training dataset creation methodologies.

2.1 Synthetic Training Data

To address the challenge of scalability and dynamic nature of our of catalog, we devised a mechanism to synthetically generate misspelled dataset to train a BART Transformer model. This involved utilizing various custom techniques to inject noise in Artist names, Albums names, Track and Podcast Tiles. We utilized *nlpaug*[2]. We generated more than *500 million* synthetic misspelled queries. Table 1 shows types of misspellings generated.

2.2 Catalog MetaData

A must have requirement for generative spelling corrector model is not to misspell an already correct input, as this would lead to very bad customer experience. We included almost a *million* data points from our catalog meta data that consist of top performing music entities in our training data. Replicating high-quality data points in training generative text models proved to an effective data augmentation techniques and prevented the model from *"over-correcting"* a correct user input.

[1]https://www.youtube.com/watch?v=pWYblEaUFng
[2]https://github.com/makcedward/nlpaug

Table 1: Synthetic Misspellings

Type of misspelling	Correct Spelling	Misspelling
Character Deletion	The Beatles	The Betles
Character Substitution	Taylor Swift	Tailor Swift
Swap Characters	Confessions	Confessoins
Remove Space	Fine Line	Fineline
Remove Space And Add Noise	Joe Rogan	Joerogen
Word Split	Folklore	Folk Lore
Word Split And Add Noise	Deftones	Deff Tones
Issue In Multiple Words	Led Zeppelin	Lead Zepelin
Issues in first character	Metallica	Netallica
Last character missing	Blinding Lights	Blinding Light

2.3 Customer Feedback Data

Over time, we have collected customer feedback data on spell suggestions offered by a legacy noisy channel spell corrector. For every spelling suggestion offered as an Auto Correct (AC) or Did You Mean (DYM), we collect implicit and explicit customer feedback by measuring metrics like click through rate(CTR). We approximately selected *100 thousand* spelling suggestions where CTR was above a certain pre-selected threshold. We harnessed this high-quality dataset to fine tune the BART model using the LoRA adapter-based fine tuning technique.

2.4 Other Sources

We incorporated following supplementary sources to enhance accuracy of the BART model

(1) Human annotation of misspellings in *top 10 thousand* most searched queries.
(2) We incorporated user query refinement pairs in training data where misspelled queries were refined for correct spelling by users themselves.

3 SYSTEM DESIGN

Our music search spell correction system comprise of two key phases, each addressing a specific challenge associated with spell correction in music search. The candidate generation phase involves a Bart Transformer model trained from scratch on large scale data to provide contextual spellcheck. It is further fine tuned using PEFT[3] technique on high quality data. Next phase involves filtering outright incorrect as well as low performing candidates using business guardrails and online Bandit based solution that effectively stops low performing spelling suggestions in real time.

3.1 Custom BART Transformer

We trained a BART Encoder-Decoder transformer model on a large-scale synthetic misspellings dataset 2.1. The model offers coverage for different types of spelling correction issues discussed in Table 1, particularly for popular entities within our catalog. Additionally, it excels in contextual spelling correction, enhancing its effectiveness in real-world scenarios.

One of the key challenge in using Transformer models in production is their high inference latency. By performing extensive model format compile time optimizations and robust tuning of the

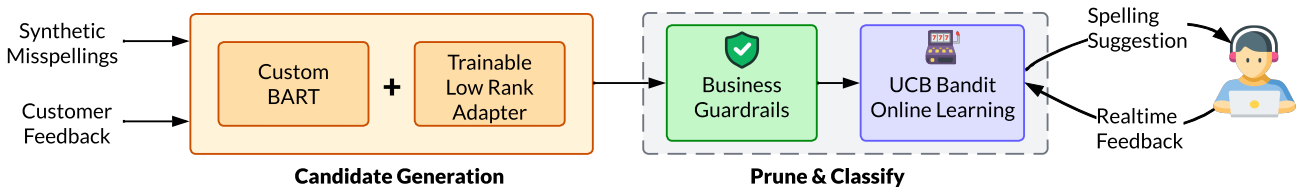

Figure 1: High Level Architecture

model server for *GPU based inference*[5], we were able to achieve p99 model server latency under 25 ms, making this approach pragmatic for use in production settings.

We discovered that instead of using pre-trained *bart-base checkpoint*[3] and then fine tuning it, training the model from scratch on our domain specific data helped boost the performance. Interestingly we didn't find much gains from unsupervised pre-training by token masking objective[7] on Music Catalog data and then fine tuning it on our spelling correction dataset. Moreover, due to highly domain specific nature of our problem, we found that model's performance remains effectively the same even when we reduced the size of Encoder and Decoder layers by half, which helped us in bringing down the inference latency. Training a new tokenizer using Byte Pair Encoding[8] on our catalog metadata, helped us drastically boost the model performance and significantly reduce *"over-correction"* issues (model generating incorrect output when provided with correctly spelled input).

3.2 Fine Tuning With Low Rank Adapter

To counter distribution drift from new catalog entities and calendar event influences like Halloween and Valentine's Day, frequent model retraining is crucial. Retraining for transformer models is time-consuming and resource-intensive. Adopting LoRA Adapter mitigates these challenges, making fine-tuning more efficient and cost-effective. We found that using conventional fine tuning techniques that involve varying learning rates for different transformer layers, adding new layers or freezing some while fine tuning others layers of a pre-trained BART checkpoint are not only slow but also lead to *catastrophic forgetting*[4] where our model abruptly and drastically forgot previously learned information upon learning new information. LoRA Adapter based fine tuning enabled our model to swiftly adapt to new additions in the catalog and mitigate data drift in search queries as well as catalog while still retaining information it learned during previous large scale training.

3.3 Business Guardrails

We implemented a set of post-processing guardrails to further refine the suggestions generated by the BART transformer model. These guardrails are strategically designed to filter out suggestions based on business rules and domain-specific knowledge e.g. preemptively omitting spelling suggestions that contain culturally inappropriate and profane language.

Some examples business guardrails include *Prefix Guardrail*, where we refrain from changing initial characters of short one word queries, as this can change meaning of the query e.g. *fight song* to *night song*. Another example is *Numeric Guardrail*, which forbids us to change numeric part of query e.g. given query *top songs 2024* don't change the year component to *top songs 2023* as this will change the intent of the query.

3.4 UCB Bandit Online Learning

In the landscape of music search spell correction, it's inevitable that the spell corrector model may occasionally offer sub-optimal suggestions, particularly when faced with out-of-vocabulary tokens or contextual nuances. For instance, during calendar events such as *"Halloween"*, the model might erroneously correct a search query like *"Halloween 2024"* to *"Halloween 2023"*, based on its training data, leading to potential user frustration or embarrassment. These issues highlight the importance of addressing bad or embarrassing suggestions promptly to maintain a positive user experience.

Traditionally, mitigating such issues involved manual intervention, such as adding erroneous suggestions to a block list or retraining the model. To proactively tackle these challenges, we propose use of a real-time UCB[1] algorithm. The algorithm dynamically evaluates the effectiveness of a spelling suggestions by using implicit as well as explicit user feedback and it will intervene in real-time to suppress a bad spelling suggestion.

This proactive approach ensures that users are not repeatedly presented with erroneous or embarrassing suggestions, thereby enhancing the overall search experience.

4 EVALUATION RESULTS

We conducted extensive online and offline evaluations on diverse datasets. For offline evaluation, we curated a synthetic misspelling dataset covering various types of misspellings mentioned earlier in Table 1. This dataset was augmented with manually labeled correct spellings of poor performing misspelled queries. Figure 2 illustrates the distribution of edit distance and types of misspellings within our offline evaluation dataset. We evaluated accuracy of our transformer model and our legacy noisy channel spell corrector (as a baseline) on different types of misspellings. The results presented in Table 2 depict the relative improvement in accuracy achieved by comparing the difference between accuracy numbers from both methodologies.

During A/B experiment on Amazon Music Search we observed significant positive impact of over 100 bps on our top level business metrics like CTR, Music Play Rate etc, as well as positive impact on spelling suggestion coverage (percentage of queries we offer spelling suggestion on).

[3]https://huggingface.co/facebook/bart-base
[4]https://en.wikipedia.org/wiki/Catastrophic_interference

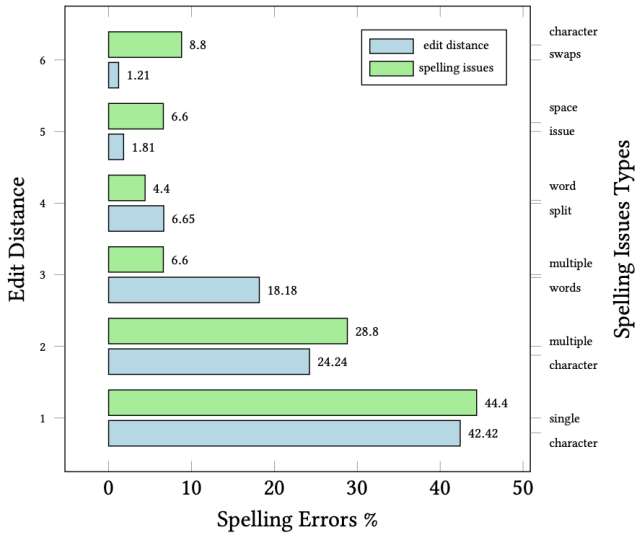

Figure 2: Edit Distance Distribution In Spelling Issues.

Type of misspelling	$\frac{\text{Transformer - Legacy Model}}{\text{Legacy Model}} \times 100$
Character Deletion	758.67 %
Character Substitution	1153.7 %
Swap Characters	555.68 %
Remove Space And Add Noise	41600 %
Word Split	1812.6 %
Word Split And Add Noise	436.36 %
Issues In Multiple Words	3817.95 %

Table 2: Relative Improvement In Accuracy Between Transformer Model And Legacy Noisy Channel Spell Corrector On Different Kind Of Misspellings

In a separate experiment, we evaluated the efficacy of the UCB Bandit in selecting the best performing spelling suggestion when provided with two different alternatives. Table 3 showcases performance data of two spelling variants of the misspelled query *boyz in the hood*. This data was collected in an online test.

	Spelling Variant 1	Spelling Variant 2
Suggested Spellings	*boys in the hood*	*boyz n the hood*
Suggestions Offered	65	96
Responses Clicked	48	77
CTR	73.84	80.2

Table 3: Performance of different spelling candidates

Using the data in Table 3 we simulated request and rewards for our UCB agent. In this scenario Bandit has two arms, each arm representing the action of selecting one of the spelling suggestion. Our agent is unaware of mean rewards of both arms and will have to explore each arm initially to generate an empirical reward estimate for both arms. We expect that while minimizing *regret*[6] Bandit should converge to selecting the arm that has higher CTR. Figure 3 shows the results of the simulation. Although the CTR of both spelling candidates were close, UCB Bandit spent almost 40% of the total trials in exploration and while rest in exploration phase. This showcases that Bandit based approach can be very effective in choosing a better spelling suggestion or avoiding a bad suggestion in real time while minimizing bad customer experience.

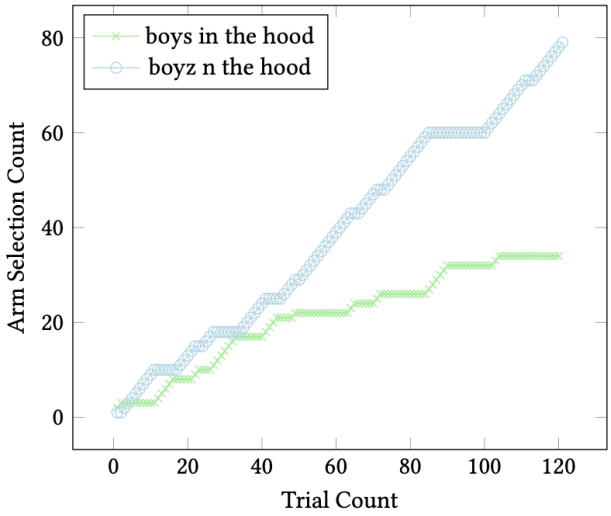

Figure 3: UCB Bandit Simulation - Selecting Different Spelling Variants For The Misspelled Query "boyz in the hood"

5 CONCLUSION AND NEXT STEPS

In this paper we described a comprehensive framework to solve different aspects of spell correction involved in entity heavy Music search spell correction. This involved contextual spelling suggestion using generative models, fine tuning them on high quality data using adapter based techniques and demoting bad suggestions in real time using Bandit based approach. Using this approach we were able to attain high accuracy on different kind of misspellings as well as developed a mechanism for efficient fine-tuning.

The diverse customer base of Amazon Music across territories with unique linguistic landscapes suggests future investigation into multi-lingual transformer model architectures. Recent advancements in decoder-only architectures like Mixtral Mixture Of Experts[5] (MOEs), have exhibited SOTA performance on various benchmark datasets. This suggests promising avenues for exploring alternative architectures and driving further advancements in the field. There are also scaling challenges with the current bandit approach; a more lightweight classifier leveraging search queries performance data as well as lexical features might offer an acceptable trade-off of accuracy for efficiency.

MAIN AUTHOR BIO

Siddharth Sharma is a senior machine learning engineer at Amazon Music Inc. He earned his Master's degree from North Carolina State University, Raleigh, NC. Siddharth's work focuses on search, retrieval and ranking models.

[5]https://aws.amazon.com/blogs/machine-learning/how-amazon-music-uses-sagemaker-with-nvidia-to-optimize-ml-training-and-inference-performance-and-cost/

[6]https://en.wikipedia.org/wiki/Regret_(decision_theory)

REFERENCES

[1] Peter Auer, Nicolò Cesa-Bianchi, and Paul Fischer. 2002. Finite-time analysis of the multiarmed bandit problem. In *Proceedings of the 15th Annual Conference on Learning Theory*. Springer, 36–50.

[2] Wolf Garbe. 2012. 1000x Faster Spelling Correction algorithm. (2012). (2012). https://seekstorm.com/blog/1000x-spelling-correction

[3] Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin De Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. 2019. Parameter-Efficient Transfer Learning for NLP. In *Proceedings of the 36th International Conference on Machine Learning (Proceedings of Machine Learning Research, Vol. 97)*, Kamalika Chaudhuri and Ruslan Salakhutdinov (Eds.). PMLR, 2790–2799. https://proceedings.mlr.press/v97/houlsby19a.html

[4] Edward J Hu, yelong shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2022. LoRA: Low-Rank Adaptation of Large Language Models. In *International Conference on Learning Representations*. https://openreview.net/forum?id=nZeVKeeFYf9

[5] Albert Q. Jiang, Alexandre Sablayrolles, Antoine Roux, Arthur Mensch, Blanche Savary, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Emma Bou Hanna, Florian Bressand, Gianna Lengyel, Guillaume Bour, Guillaume Lample, Lélio Renard Lavaud, Lucile Saulnier, Marie-Anne Lachaux, Pierre Stock, Sandeep Subramanian, Sophia Yang, Szymon Antoniak, Teven Le Scao, Théophile Gervet, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. 2024. Mixtral of Experts. arXiv:2401.04088 [cs.LG]

[6] Mark D. Kemighan, Kenneth W. Church, and William A. Gale. 1990. TA Spelling Correction Program Based on a Noisy Channel Model. *COLING* volume II., 1 (1990).

[7] Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, Dan Jurafsky, Joyce Chai, Natalie Schluter, and Joel Tetreault (Eds.). Association for Computational Linguistics, Online, 7871–7880. https://doi.org/10.18653/v1/2020.acl-main.703

[8] Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Neural Machine Translation of Rare Words with Subword Units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Association for Computational Linguistics, Berlin, Germany, 1715–1725. https://www.aclweb.org/anthology/P16-1162

[9] Sanat Sharma, Josep Valls-Vargas, Tracy Holloway King, Francois Guerin, and Chirag Arora. 2023. Contextual Multilingual Spellchecker for User Queries. SIGIR '23: Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval, 3395–3399. https://doi.org/10.1145/3539618.3591861