Cultural Signatures in Names: An LLM Approach

LLMs enrich long-tail nationality datasets to improve local classifier performance at scale

Cong Ming, Ruixin(Jack) Shi

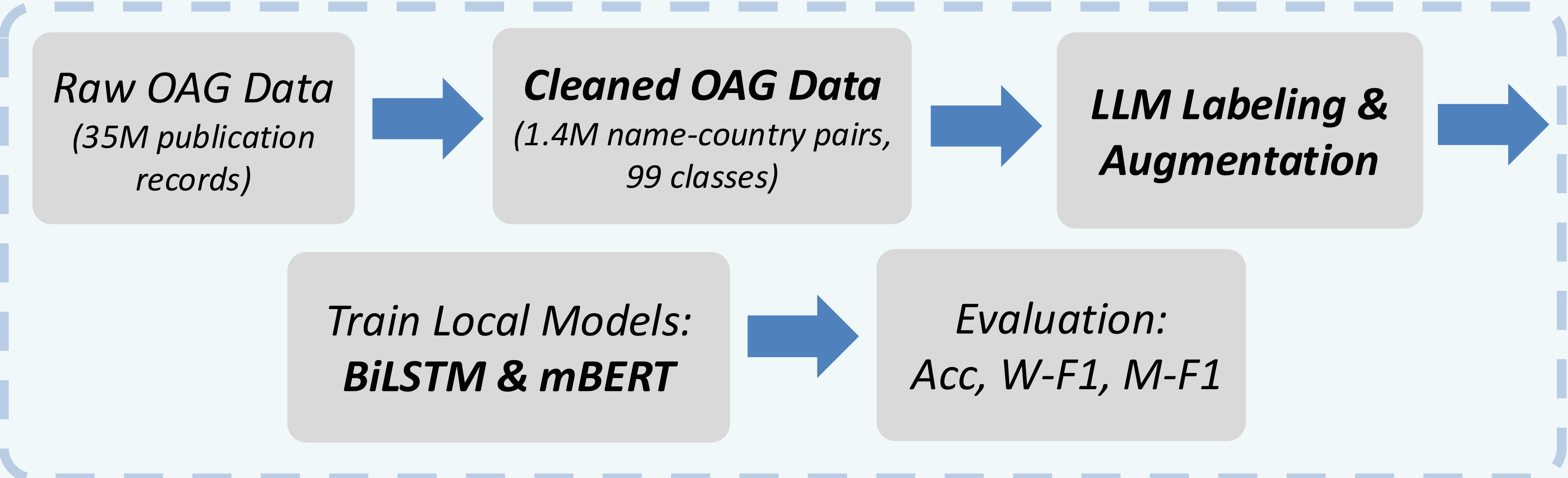
Advisor: Prof. Yifan Hu

Motivation and Pipeline

1. Motivation & Problem

- Real-world name datasets are biased and incomplete.
- Long-tail nationalities have extremely few entries.
- Legacy classifiers collapse on rare names.
- LLMs perform well but are expensive for 35M+ scale tasks.
- **Goal: build a scalable pipeline for predicting nationalities from names.**

2. Data Flow & Pipeline


Raw OAG Data *(35M publication records)*

Cleaned OAG Data *(1.4M name-country pairs, 99 classes)*

LLM Labeling & Augmentation

Train Local Models: ***BiLSTM & mBERT***

Evaluation: Acc, W-F1, M-F1

3. Baselines & Models

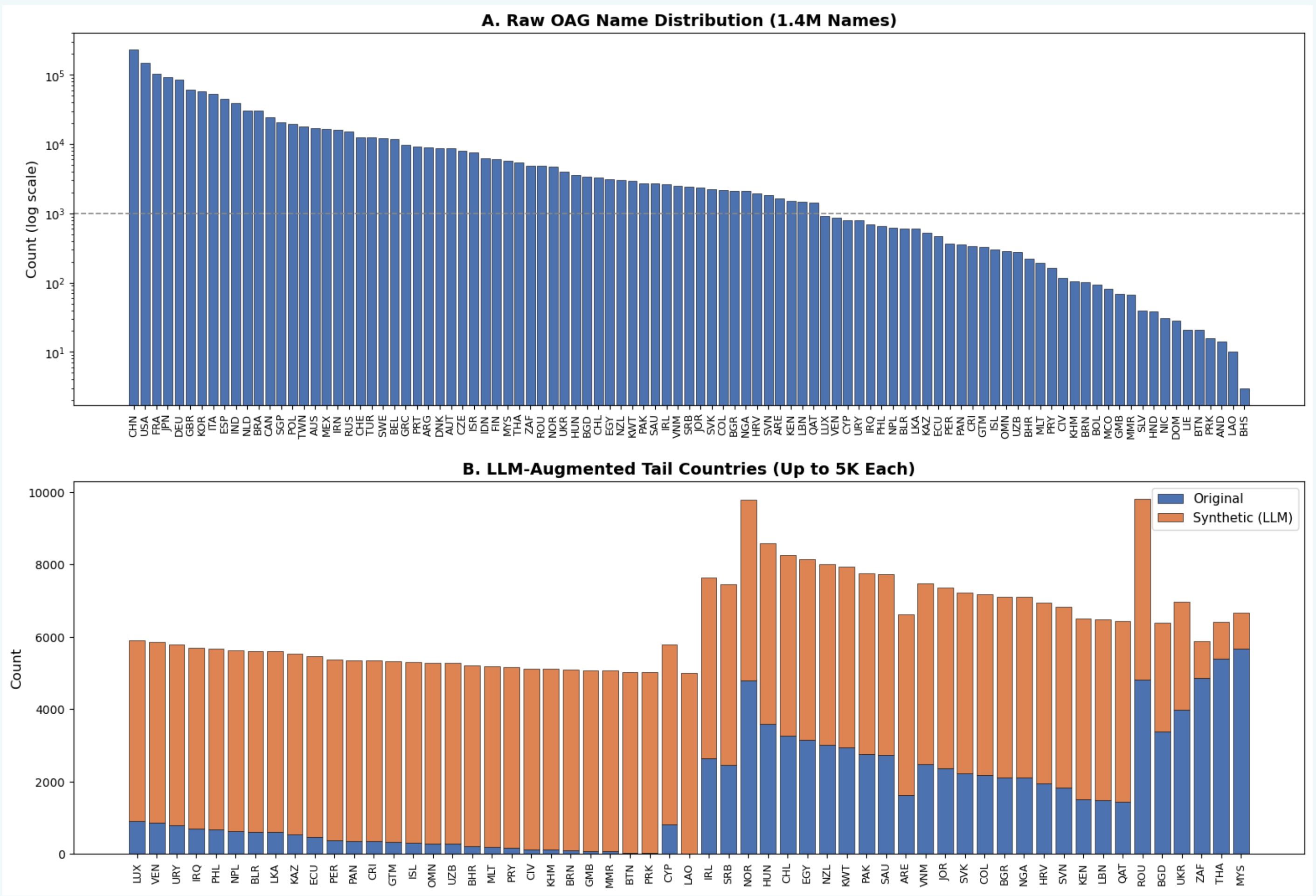
Baselines

- EthnicSeer
  - 12-class, logistic regression
- NamePrism
  - 39-class, Naïve Bayes
- Name2Nat
  - 170-class GRU

Our Models

- BiLSTM
  - L=40, d=64
  - fine-tuned 99-way
- mBERT
  - bert-base-multilingual-cased
  - fine-tuned 99-way

4. Dataset Overview

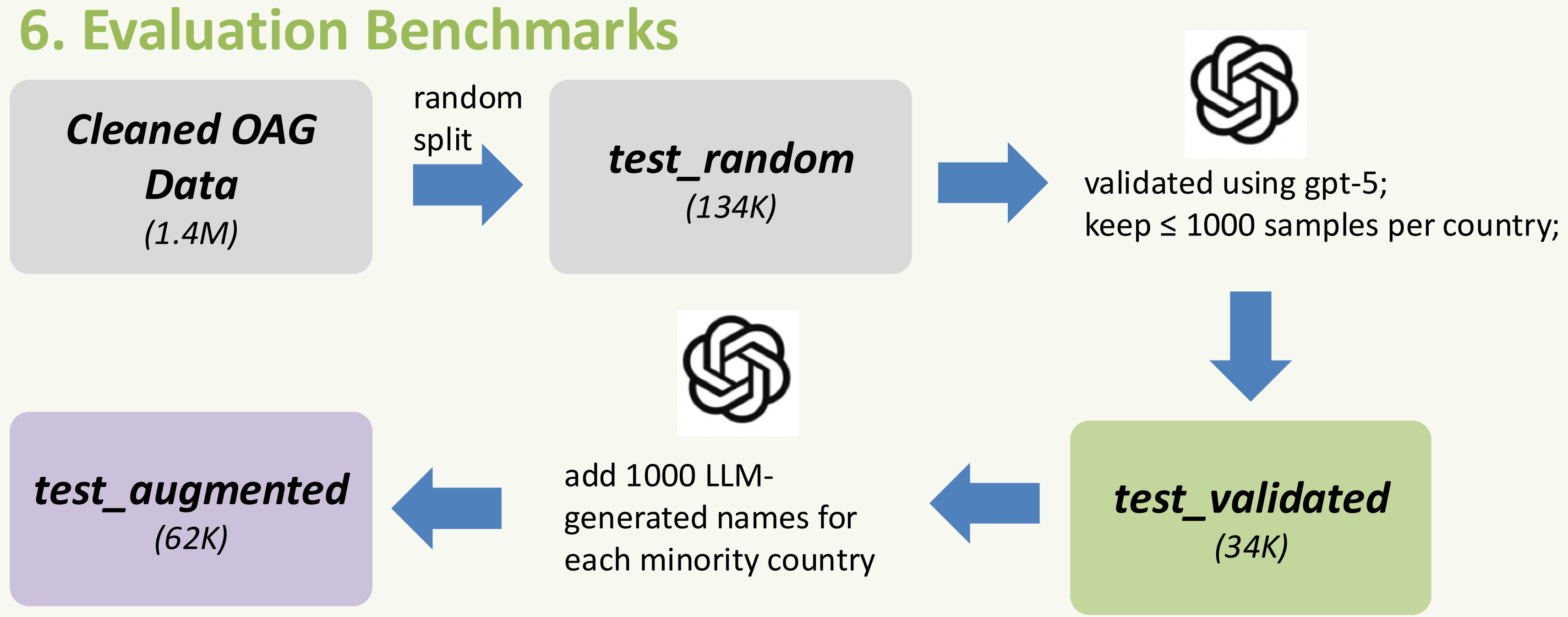


Key Techniques and Experiments

5. LLM as Data Enricher

- **LLM Labeling**: GPT-5 assigns nationality labels to raw names
  - Create the golden test set (test_validated)
- **LLM Augmentation**: Generates synthetic names for low-resource nationalities
  - Augment training: *train_original (1.07M) -> train_augmented (1.16M)*
  - Not evaluation *(test_augmented only)*

6. Evaluation Benchmarks


Cleaned OAG Data *(1.4M)*

random split

test_random *(134K)*

validated using gpt-5;
keep ≤ 1000 samples per country;

test_validated *(34K)*

add 1000 LLM-generated names for each minority country

test_augmented *(62K)*

7. Model Comparison

- ***model A:*** *BiLSTM (train_original)* ***model B:*** *BiLSTM (train_augmented)*
- ***model C:*** *mBERT(train_original)* ***model D:*** *mBERT(train_augmented)*

	Acc	W-F1	M-F1
model A	0.660	0.630	0.279
model B	0.660	0.628	0.272
model C	**0.699**	0.679	0.349
model D	0.698	**0.680**	**0.364**

test_random

	Acc	W-F1	M-F1
model A	0.556	0.525	0.316
model B	0.553	0.522	0.313
model C	0.651	0.630	0.416
model D	**0.652**	**0.635**	**0.442**

test_validated

	Acc	W-F1	M-F1
model A	0.352	0.297	0.243
model B	0.641	0.626	0.459
model C	0.466	0.417	0.350
model D	**0.741**	**0.732**	**0.569**

test_augmented

	Acc	W-F1	M-F1
EthnicSeer	0.677	0.670	0.548
model A	**0.782**	**0.794**	**0.713**
	Acc	**W-F1**	**M-F1**
NamePrism	0.589	0.603	0.383
model A	**0.752**	**0.746**	**0.524**
	Acc	**W-F1**	**M-F1**
Name2Nat	0.467	0.442	0.139
model A	**0.660**	**0.630**	**0.279**

test_random, mapped to corresponding taxonomies

8. Best Overall Model

model D on test_validated:
Acc = 0.652, W-F1 = 0.635, M-F1 = 0.442

Findings and Why It Matters

9. Key Findings

- Overall performance:

  Baselines < model B < model A < model C < model D
- LSTM gets **worse** with augmented training data, but mBERT gets **better**.
- Augmentation improves **fairness** (macro-F1) more than accuracy.

10. Which Countries Are Easy or Hard?

A. Countries with Very High F1 (Easy Classes)

Country	F1
Japan	0.96
South Korea	0.95
Turkey	0.94
Iran	0.90
Finland	0.92
Thailand	0.91
Poland	0.91
Greece	0.91

Country	F1
Czech Republic	0.82
Indonesia	0.82

- **Highly characteristic morphology** (e.g., Japanese Kanji/Katakana names, Korean syllabic patterns)
- **Low ambiguity** with other countries
- **Strong pattern regularity** in spelling and phonetics
- **Large validated datasets** → stable patterns

B. Countries With Moderate F1 (Partially Ambiguous Groups)

Country	F1
Germany	0.49
Spain	0.60
Argentina	0.49
USA	0.31

- **Mixture of linguistic influences**
- **High internal diversity** (e.g., USA, Singapore)
- **Overlap with neighboring cultures** (e.g., Spain–Latin America)

C. Countries With Very Low F1 (Hard Classes)

Country	F1
Venuszuela	0.00
Honduras	0.00
Panama	0.00
Luxembourg	0.00

- **Very small support** (< 50 samples → extremely unstable)
- **Culturally similar to neighboring**
- **Highly multilingual/immigrant populations** (e.g., Luxembourg, New Zealand)

11. Why Not Use LLMs Directly?

- **Too slow:** 35M names = days to months even with batching, while local models run thousands/sec at near-zero cost.
- **Too expensive:** API inference cost is prohibitive.
- **Privacy + governance issues**

12. Conclusion

- LLMs act as data multipliers instead of classifiers.
- Synthetic long-tail augmentation improves fairness and recall.
- Practical for real-world demographic inference at population scale.

Code: https://github.com/MingCong19/NameBERT
Open Academic Graph: https://www.microsoft.com/en-us/research/project/open-academic-graph/

Northeastern University