Efficient Vietnamese Name Detection Using Highly Discriminative N-Grams

Toan Luu[1], Phan Quoc Hung Mai[2], Thi Thu Phuong Dao[2], Quang Hung Nguyen[2], and Xuan Lam Pham[2(✉)]

[1] UBS Group AG, Zürich, Switzerland
[2] College of Technology, National Economics University, Hanoi, Vietnam
{11222607,11194176,11222618}@st.neu.edu.vn, lampx@neu.edu.vn

Abstract. Detecting Vietnamese names from a global name collection is essential for connecting human resources within the Vietnamese community worldwide and for developing Vietnamese Expert Finding Systems. However, applying standard classification methods to recognize Vietnamese names is challenging due to their ambiguity with names from other countries. In this study, we proposed our novel approach, Highly Discriminative N-grams (HDN), which utilizes statistics from n-grams from human names for the classification task. We show that HDN is a lightweight and simple-to-reproduce approach for this task by benchmarking with other machine learning models.

Keywords: Name-Ethnicity Classification · Machine Learning · Data Mining

1 Introduction

A recent estimate [1] suggests that over 5 million individuals of Vietnamese origin live and work outside of Vietnam. Automatically identifying Vietnamese names from a large global name dataset is essential for various applications related to the Vietnamese community worldwide. This algorithm can be utilized in developing expert-finding systems, enhancing targeted marketing strategies, and fostering connections within the community.

Various public big data platforms, such as LinkedIn, Google Scholar, and ResearchGate, can be leveraged to explore Vietnamese experts for academic and industry networking. Online phone directories can also help locate service providers, consultants, and advisors. This data supports the identification of experts across diverse fields and facilitates the connection between overseas Vietnamese professionals and research and development initiatives in Vietnam. This approach benefits both the public and private sectors by attracting Vietnamese talent worldwide, contributing to national development, and serving as a bridge for Vietnam's deeper integration into the globalized era.

A statistic in 2022 from the Vietnamese Social Science Publisher [2] examines 100 most popular surnames/family names in Vietnam, revealing that the top 15

N. T. Nguyen et al. (Eds.): ACIIDS 2025, LNAI 15684, pp. 332–347, 2025.
https://doi.org/10.1007/978-981-96-6005-6_24

most popular surnames in Vietnam cover 83% Vietnamese names. Recognizing Vietnamese names poses a challenge due to their shared Latin characters and similar word forms with names from other countries, and the transformation when the accents are dropped. For instance, the surname "Nguy n" is easily identifiable as Vietnamese. However, surnames like "Lê" when written without accents, can be confused with common French prepositions "Le" and " " may be mistaken for the English verb "Do". Furthermore, some names, such as "Vân" or "Văn", sound distinctly Vietnamese but, when written without diacritics, can be confused with the common Dutch name "Van".

The critical challenge for many systems is handling millions of names sourced from the public dataset on the Internet, encompassing various countries and languages, and automatically distinguishing between Vietnamese names and those of other nationalities. Numerous published studies detail methodologies for name classifiers [17]. Nonetheless, very few studies focus on Vietnamese names. What we know about are studies that investigate how to classify names at an ethnicity level with simple approaches [14,22], or predict gender in Vietnamese names [9,24]. This research work presents one of the initial inquiries into training a classification system to identify whether a person's name is origin from Vietnamese.

This paper is structured into three main sections. The first section provides a concise overview of name classification methods and research specific to Vietnamese names. The second section details the methods employed in Vietnamese name classification, including our novel approach Highly Discriminative N-grams (HDN) to enhance the performance and quality of existing solutions. In the third section, experimental results demonstrate that our proposed method, utilizing a rational approach. These novel findings have the potential to facilitate the development of similar name-ethnic classification applications, enabling the extraction of valuable information in today's big data landscape.

2 Related Works

2.1 Personal Name Classification

The task of detecting Vietnamese names within a comprehensive global name database can be framed as a standard name classification problem. In this context, Vietnamese names are classified as positive labels, while non-Vietnamese names serve as negative labels. Classifying the nationality of names is a challenge with numerous applications, including biomedical research, clinical practice, employment, addressing educational disparities, and delivering more precise advertisements, news, and social media content to users [5,7,29].

Several studies utilize personal names for classification tasks, primarily designed to categorize human race, nationality, and gender. [25] utilized crawled Wikipedia data for the name ethnicity classification task and used multinomial logistic regression for the task. This paper is known to be the first work to classify nationality based solely on personal names. They used both alphabet and phonetic sequences within names to enhance performance, applying this approach to examine the evolution of ethnicities within the computer science

research community. [15] with the same effort to predict nationality based on name, proposed a recurrent network-based model, which gains a higher accuracy rate. [29] introduced NamePrism, a novel name nationality and ethnicity classifier that provides a more detailed taxonomy of ethnic groups. They trained a word embedding model and then employed a simple naive Bayes classifier, achieving a high F1 score compared to previously published systems, serving as a benchmark.

However, very few studies have been conducted in this specific domain, and even fewer have focused on Vietnamese name detection, as we encountered minimal prior research in this area. [14] conducted a study aiming to classify Asian ethnic groups in America using surnames. This study utilized the frequency of surnames among different ethnic groups in an area, but the results of this study were not particularly significant. [22] provided a descriptive evaluation of [14] findings, specifically focusing on Vietnamese names. The authors noted that the results of [14] may vary depending on the surveyed areas since it depends on the proportion of the Vietnamese population in a specific area. Additionally, several other studies [9,24] have attempted to classify gender based on Vietnamese names.

2.2 Name Embedding and Classification Methods

Word embeddings are vector representations of words used in natural language processing. In previous work, [29] created name embedding by employing word2vec as a method to vectorize words. They proposed that this technique acts as an encoder representation, capturing gender, ethnicity, and nationality, which can be readily applied to building classifiers. Consequently, they train their embeddings by two learning methods: CBOW and Skip-Gram. [8] came up with the same approach where they confirmed the presence of ethnic and gender homophily in email communication patterns. The authors observed a higher concentration of names with matching gender and ethnicity in email users' contact lists than would be anticipated by chance.

N-grams has also been extensively applied to this task and continue to yield efficient outcomes. [25] utilize a variety of features, including basic n-grams, as well as external ones like Double Metaphone or Soundex. They found that this diverse feature set contributes to robust name classification performance. Similarly, [15] discovered that n-gram embeddings effectively capture character-level features for name classification tasks. Many other works using n-grams also show a high performance in name classification tasks [6,21]. This underscores the versatility and effectiveness of n-grams in capturing intricate patterns within textual data, particularly in tasks involving name classification.

A typical approach for classification tasks involves initially experimenting with Machine Learning methods. In this context, researchers primarily employ basic classifiers such as Logistic Regression, Naive Bayes, or Support Vector Machine. [29] used the NB approach besides their learned embeddings. Whereas, [12] carries out the task with SVM classifiers and Convolution Neural Networks.

[27] try out either LR, SVM, or NB for classifying ethnicity with a given personal name and census location in Canada.

[3] presented a name classifier using Wikipedia data. Their methodology consists of two major components: Hidden Markov Models (HMM) and Decision Trees to classify names into 13 cultural/ethnic groups. [15] proposes to use Recurrent Neural Networks (RNN) with Long Short-Term Memory (LSTM) to predict solely based on the characters for name nationality, and ethnicity classification tasks. This approach brings about a significant classification performance with high accuracy. [10] represent each character in a name as an embedding vector and model character transitions.

3 Methods of Vietnamese Name Detection

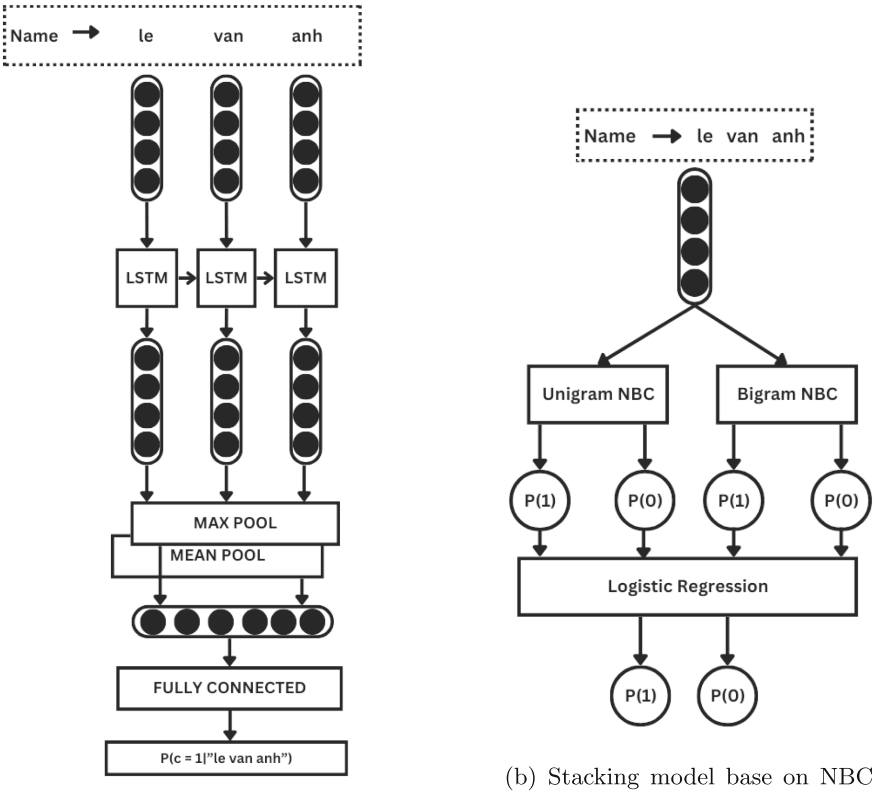
In this section, we first provide an overview of the typical classification methods used in our experiments. In the second subsection, we discuss a detailed description of our proposed approach, Highly Discriminative N-grams.

3.1 Typical Classification Methods

In our efforts to create the best classification method, we first use the commonly used classification methods. Initially, we explored fundamental Machine Learning and Deep Learning (ML/DL) algorithms, including Logistic Regression (LR), Neural Networks (NN), Long-Short Term Memory (LSTM), and stacking model to evaluate their respective performances.

LR, incorporating regularization and operating on embedding vectors, would have good performance in probabilistic classification. NN, with their multi-layered architecture and non-linear activation functions, captured intricate patterns within high-dimensional data. However, their inability to effectively model sequential dependencies in text data prompted us to explore more advanced architectures. To represent words as numerical vectors, we experimented with unigram and bigram vectorizers, but found that using them separately was insufficient. Instead, we opted to combine them, alongside pre-trained word embeddings from GloVe, which provided 300-dimensional representations trained on 42 billion tokens. For LR and NN, each name's embedding vector was computed as the mean of its token embeddings, weighted by their TF-IDF scores.

Recognizing the sequential nature of names, we employed LSTM networks, whose unique memory cells and gating mechanisms proved effective in learning the complex linguistic patterns of Vietnamese names, which often follow intricate structural rules. To further optimize performance, we initialized the LSTM embedding layer with pre-trained GloVe vectors and fine-tuned it during training. Unlike conventional LSTM models that rely solely on the last token's output for classification, assuming it encapsulates all prior information, we incorporated a pooling layer that combines mean and max pooling outputs from all LSTM cells before passing them through a final classification layer [30], enhancing the model's ability to retain essential contextual information for accurate Vietnamese name detection, as illustrated in Fig. 1a.



(a) LSTM model architecture.

(b) Stacking model base on NBC and LR (NBLR).

Fig. 1. LSTM and Stacking model architectures.

The Stacking Classifier is a widely used ensemble learning technique that enhances overall model performance by combining multiple base models, leveraging their strengths while compensating for individual weaknesses. It operates by first training base models, then using their predictions as inputs for a final estimator. Given the nature of the Naive Bayes Classifier (NBC), which relies on token frequency and assumes feature independence, its performance varies significantly depending on whether unigram or bigram features are used-yielding high recall (0.92) but low precision (0.43) for unigrams, and high precision (0.99) but low recall (0.43) for bigrams. The bigram model is more confident in detecting true positives, while the unigram model may misinterpret common words as names, such as "luc van" being mistaken for the Vietnamese names "L c" and "Văn". To address these limitations, we implemented a Stacking Classifier that combines NBC models trained with both unigram and bigram features, feeding their probability outputs into LR for final classification (we denote this model NBLR), as illustrated in Fig. 1b. Additionally, we introduced a weighting mechanism to better align with real-world data by adjusting the class probabilities in

NBC and setting label weights in LR based on the estimated Vietnamese name distribution. Specifically, we modified class probability to reflect the proportion of Vietnamese names in our evaluation datasets, and incorporated weighted logistic loss [13] to further refine predictions.

3.2 Highly Discriminative N-Grams

From what we observe in the big corpus of Vietnamese and international names, there are sets of tokens that appear only in Vietnamese names. For example "nguyen", "huyen" are rarely found in the person's name in other nations. They are considered highly discriminative unigrams. On the other hand, some unigrams, which are non-discriminative, and are common in both Vietnamese and foreign names, could be combined to create new highly discriminative bigrams. For example "tran" and "van" are non-discriminative unigrams as they appear in other nationalities, however, when they are combined, this pair becomes a discriminative feature for the Vietnamese name "tran van". This bigram "tran van" is a commonly occurring combination, typically representing the family name "Tr n" along with the middle or first name "Văn" in Vietnam.

Highly Discriminative N-Grams. (HDN) is inspired by the research [20] which uses Highly Discriminative Keys (HDK) for efficiently querying documents from the P2P search engine system. The research has demonstrated that, due to the adherence of word frequency to Zipf's law, the proliferation of word combinations for creating HDKs cannot increase too rapidly as the document collection expands. This implies that we can feasibly manage to store the quantity of HDKs in practice.

In Vietnamese names, the distribution of token frequency approximates Zipf's law, wherein a small number of tokens appear frequently in the majority of names. When a list of measured tokens is arranged in descending order of frequencies, the value of the nth entry is inversely proportional to n. Consequently, we can employ the same approach to generate combinations of name tokens for creating new discriminative n-grams without concern for exponential growth in the number of combinations. The HDK approach relies on document frequency with a fixed threshold to constrain the size of the inverted index during transmission across low-latency networks. Conversely, our approach relies on a confidence score to determine whether to proceed with combining tokens to create more discriminative n-grams.

The simplified version of the algorithm is represented in the Algorithm 1. The input of the algorithm is the name of any person. The output is labeled "*true*" which is Vietnamese or "*false*" which is not a Vietnamese name. In real-world applications, we return as well confidence scores and features used to identify the labels for debugging or to apply post-processing for the score that is close to the positive threshold.

Algorithm 1. Name classifier using Highly Discriminative N-grams

```
1: tokens = tokenized(name)
2: for t in tokens do
3:    if isDiscriminative(t) then
4:       return true
5:    end if
6: end for
7: for (t1,t2) in bigrams(tokens) do
8:    if isDiscriminative(t1,t2) then
9:       return true
10:   end if
11: end for
12: if highCombinedConfidences(tokens) then
13:    return true
14: end if
15: return false
16:
```

At line 3, the function **isDiscriminative** checks if a token is discriminative if it satisfies the following criteria:

- $positive\_frequency(t) > rare\_threshold$: This threshold is used to filter out some spelling mistakes tokens in the training set or abnormal names from positive name set. In the beginning, we set $rare\_threshold$ as 10.
- $confidence(t) > confidence\_threshold$: which is set extremely high, e.g. 0.995 in our current setting. When the training collection is bigger we increase thresholds.

The $confidence$ score is calculated based on the estimated probability of a token being a Vietnamese name, derived from the training dataset. Due to the fact that it is impossible to collect large negative cases of names from global databases for training set, we enhance the negative cases of tokens to ensure that tokens frequently appearing in negative label names receive a very low confidence score. This negative amplification ensures that if a token is highly prevalent in Vietnamese names but is not rare in negative names, it is not considered highly discriminative.

$$confidence(t) = \left(\frac{f(t|1)}{f(t)}\right)^{1+\frac{f(t|0)}{2}} \tag{1}$$

In the formula (1), $f(t|1)$, $f(t|0)$, and $f(t)$ are respectively frequencies of positive, negative, and total names contain token t in the training dataset. The frequency of negative names is put on power to make the confidence score reduce drastically if the number of negative labels increases. The confidence score always

has a value between 0 and 1 (when $f(t|0) = 0$). For example, the token *"van"* in our training set has a high positive ratio (0.98%) due to its frequent occurrence in Vietnamese names. However, since its negative frequency exceeds 1023, the resulting confidence score is only 0.0011.

The function **bigrams** at line 7 returns bigrams from a list of tokens. For instance name *"le van anh"* generates 3 bigrams: *"anh le"*, *"anh van"*, *"le van"*. The tokens in a bigram are put in alphabetic order to avoid duplication of bigrams. We disregard the order of tokens in the original name for detection, as the placement of first and last names in Vietnam is reversed compared to Western countries. The bigrams are generated solely from non-discriminative unigrams because the algorithm returns immediately upon detecting a discriminative unigram in previous lines.

The function **isDiscriminative** at line 8 is calculated in the same way for bigrams, but based on a pair of tokens with smaller $rare\_threshold$ (because frequency of 2 combined tokens always smaller than frequency of each token)

At line 12, the function **highCombinedConfidences** is introduced to address cases where a name contains multiple candidate discriminative features, but individually they are not highly discriminatory. Then when combined in the formula (2), they form a highly discriminative combination. This function is employed to handle scenarios where the discriminative bigram is not generated from the training dataset due to the rarity of the name.

$$combinedConfidence(t_1..t_i) = 1 - \prod(1 - confidence(t_i)) \tag{2}$$

For example, token t_1 has a confidence score of 0.97, and token t_2 has a confidence score of 0.98 which is lower than $confidence\_threshold$. The name containing only tokens t_1 and t_2 will have combined confidences $1 - (1 - 0.97) * (1 - 0.98) = 0.999$ which is higher than $confidence\_threshold$ to be detected as positive.

Both unigram and bigram sets, along with their corresponding statistics and confidence scores, are pre-computed from the training name dataset and stored in a hashmap for rapid retrieval. The maximum number of tokens in a person's name is approximately 5, which results in a constant upper bound on the number of unigrams and bigrams generated by a name. As a result, the complexity of this algorithm is constant, making it highly efficient for name detection in large datasets.

4 Experiment

4.1 Training Dataset

In our experiment, we assembled both training and evaluation datasets. The training dataset consists of positive and negative examples of Vietnamese names

gathered from various sources: (1) names extracted from the Vietnamese phonebook through web crawling, (2) a compilation of student names from several universities in Vietnam, (3) names extracted from a European country's phonebook via web crawling, (4) names obtained from Wikipedia, and (5) names sourced from LinkedIn profiles. To ensure label accuracy, we utilized multiple validation methods, including cross-checking by volunteers, verifying Vietnamese names through Wikipedia or LinkedIn profiles, and examining personal homepages or biographies available on the Internet and social media.

All names undergo normalization, which involves removing accents. For example, "Tr n Văn" is transformed into "tran van". Additionally, single-character tokens are excluded to reduce the risk of errors with abbreviations. This normalization process is implemented because Vietnamese individuals and professionals often present their names in the international format on global websites and publications. Training name data statistics are shown in Table 1.

Table 1. Statistic from training name dataset

Total names	4,126,815
Positive names (Vietnamese names)	635,678
Negative names (Non-Vietnamese names)	3,491,137
Total tokens	9,423,808
Average tokens per name	2.38
Unique unigrams	276,890
Positive unigrams	4,099
Discriminative unigrams	124
Positive bigrams	37,914
Discriminative bigrams	8,476

The top 10 unigrams and bigrams from Vietnamese names are presented in Table 2, revealing that a few tokens dominate the majority of names. Approximately 4,000 unigrams constitute over 635,000 Vietnamese names, with 124 of them classified as highly discriminative (primarily appearing in Vietnamese names but rarely in names from other nations). These 124 discriminative names encompass over 84% of Vietnamese names in our training dataset, with "nguyen" being the most prevalent, accounting for nearly 9%.

Table 2. Top 10 unigrams and bigrams from 635,678 Vietnamese names

Unigram	Percentage	Bigram	Percentage
nguyen	8.90	tran van	0.99
thi	7.90	le van	0.88
van	3.74	van vu	0.45
tran	2.97	bui van	0.34
le	2.83	do van	0.34
hoang	2.27	duc tran	0.30
thanh	2.11	hong le	0.30
ngoc	2.06	thu tran	0.30
pham	2.01	dinh van	0.29
anh	1.82	duong van	0.29

It is worth noting that the most popular bigram "nguyen thi" is not included here because we generate bigrams only when the unigram is not a discriminative token. Since "nguyen" is a discriminative unigram for Vietnamese names, all bigrams containing it are not generated. This approach helps store the number of HDN in memory for the efficient classifier in practice. In fact, the total generated bigrams are only 37,914 and 8,476 among them are discriminative which is simple to manage.

4.2 Evaluating Datasets

The experiment makes use of 3 datasets (see Table 3) with different characteristics for evaluating the model performance:

Evaluating Dataset 1 (**ED1-career2019**): This dataset contains names of global scientists mentioned in the research [11]. For the unsure Vietnamese names, we verified the information of scientists based on their homepage or bio on the internet to confirm they are originally Vietnamese. As the number of positive labels are small and we can verify carefully profile of scientists, we consider this is the golden set of our application.

Evaluating Dataset 2 (**ED2-behindthename**): Random names were generated from the BehindTheName website. BehindTheName is a website for learning about all aspects of names. All given names from all cultures and periods are eligible to be included in the main name database, which currently having more than 25,000 names. The website provided the tool to generate random names of any nation. Therefore we can easily have the correct label for Vietnamese names and names from other countries.

Evaluating Dataset 3 (**ED3-phonebook**): This dataset is extracted from a phone book published on the internet of an European country. The name of the country is not exposed for privacy reasons. To set a high barrier for the solutions, we selected a subset of names from the phone book for evaluation which contains

at least 1 token overlap with Vietnamese names from the training set. The reason is for the name which does not contain any Vietnamese name token (e.g. Antony White) it is trivial to identify this as a non-Vietnamese name.

Table 3. Evaluating datasets

Evaluating Dataset	Total names	Positive labels
ED1-career2019	150,000	141
ED2-behindthename	30,000	469
ED3-phonebook	130,000	3,349

4.3 Classification Methods Setup

To compare with our new proposal method using Highly Discriminative N-grams (HDN) we also run experiments on 4 different classification methods with some parameter tuning to make sure we have the best results from the applied models.

Through the initial experiments, we implement two different types of text embedding, and for each classification method, we opt for the embedding that produces a better performance on the 3 evaluating datasets. Firstly, we used pre-trained word embedding GloVe [19] for LSTM, combining with TF-IDF for LR, NN. Secondly, we apply counting vectors [4,16] for NGrams-based models including HDN, and Stacking NBLR.

Table 4. Classifier setup

Method	Feature Embedding	Positive Threshold
HDN	N-Grams	0.995
LR	GloVe	0.7
NN	GloVe	0.5
LSTM	GloVe	0.7
NBLR	N-Grams	0.55

Furthermore, we observed that in the presence of highly unbalanced training and testing datasets, adjusting the prediction threshold for P(1) within the range of 0 to 1 can significantly impact the performance of the models. Different thresholds are used to assign a positive label for each method to achieve the best performance on the training data set. After the correct threshold is chosen, we keep it for all 3 evaluation sets. All classifier setup is illustrated in Table 4

4.4 Experimental Results

The summary results with standard metrics Precision, Recall and F1-Score of all methods applied on 3 different evaluation name collections are summarized in the Table 5.

The HDN method consistently delivers top performance across all three evaluation datasets, achieving high Precision, Recall, and F1 scores. LSTM and NBLR perform similarly, with score differences of 0.01 to 0.05. On the ED3 dataset, NBLR achieves a precision of 0.985 but suffers from lower recall, resulting in a lower F1 score, while HDN maintains the highest F1 score of 0.934. All models perform exceptionally well on the ED2 dataset. However, since ED2 is generated from a predefined name set, its high discriminative features make the task easier. Additionally, the BehindTheName dataset contains unrealistic names, such as missing surnames, gender-mixed names, and duplicate tokens (e.g.,"Văn Vân") which reduced recall of HDN.

The more challenging ED1 dataset reveals major performance gaps, particularly in traditional models like LR and NN, which achieve low precision scores of 0.531 and 0.310, respectively. In contrast, HDN outperforms all models on ED1 with a precision of 0.927 and recall of 0.992, while LSTM follows closely, trailing by about 0.03. This highlights HDN's robustness, as it excels in the hardest dataset while maintaining strong performance across others. Overall, HDN proves to be the most reliable model for real-world name evaluation sets ED1-career2019 and ED3-phonebook. LSTM and NBLR also demonstrate strong results, making them competitive alternatives.

Often, recognizing the names of Vietnamese experts and scientists can be difficult and time-consuming, even for humans. These names, commonly encountered in published papers and other sources, often comprise multiple components,

Table 5. Score of all methods on 3 evaluation datasets.

Method	HDN	LR	NN	LSTM	NBLR
ED1 - Career2019					
Precision	**0.927**	0.531	0.310	0.895	0.880
Recall	**0.992**	0.936	0.991	0.971	0.943
F1-Score	**0.958**	0.677	0.473	0.931	0.910
ED2 - BehindTheName					
Precision	**0.984**	0.955	0.963	0.982	0.978
Recall	0.997	**1.0**	0.998	**1.0**	0.998
F1-Score	**0.991**	0.977	0.980	**0.991**	0.988
ED3 - Phonebook					
Precision	0.970	0.855	0.918	0.967	**0.985**
Recall	0.901	0.887	**0.918**	0.895	0.832
F1-Score	**0.934**	0.871	0.918	0.929	0.902

such as a Vietnamese surname, an abbreviated middle name, and a foreign first name in an international format (e.g. Thomas V. Nguyen). Hence, in the real-world application of the Vietnamese Expert Finding System, we are inclined to choose to utilize the HDN method since the model landed the highest scores on ED1, which contains names of Vietnamese scientists and experts.

4.5 Large Language Models Benchmark

Recently Large Language Models (LLMs), known for their advanced capabilities in understanding and generating text, have become a standard for evaluating linguistic tasks [28]. To benchmark their performance, we conducted a comparison between our proposed method and LLMs to evaluate their ability to detect Vietnamese names. Benchmarking against LLMs is essential because these models set a high baseline for multilingual understanding, providing a clear reference point for assessing the effectiveness of our method. In this experiment, we focus on recall metrics, meaning we exclusively compare the models' performance when only given Vietnamese names. The evaluated dataset consists of 3,490 Vietnamese names from ED1 and ED3, tested on the common property LLMs: GPT 4o [18] and Gemini 1.5 Flash (version 002) [23].

Table 6. Comparision between LLM and our methods

Method	Recall	
	ED1 - Career2019	ED3 - Phonebook
HDN	**0.992**	**0.901**
LSTM	0.971	0.895
NBLR	0.943	0.832
Gemini-1.5-flash-002	0.915	0.802
Gemini-1.5-flash-002	0.921	0.839

Table 6 highlight the comparison between LLMs and HDN, focusing on recall metrics, where HDN achieves significantly higher recall scores, outperforming GPT-4o-mini and Gemini-1.5-Flash-002 by 7–10%. Upon analysis, LLMs tend to classify names as non-Vietnamese when encountering names that contain non-Vietnamese words or do not follow the typical Vietnamese name format (e.g., reversed first and last names such as Nguyen Audrey or Delley Thy-Vu). Moreover, since names are short texts and do not require significant contextual understanding [26]—using LLMs for this task is highly inefficient due to their high computational cost and resource demands. In contrast, our model is specifically designed for Vietnamese name detection, offering better performance with significantly lower costs and higher security, making it a more practical and effective solution.

5 Conclusion and Future Works

This work addresses the challenge of detecting Vietnamese names in a global dataset, achieving an F1 score greater than 93% despite overlaps with names from other countries. HDN proved to be a promising approach due to its simplicity and ability to capture common Vietnamese name patterns using Ngrams. Hence, our experiments demonstrated consistently strong performance across different settings. While this study focused on Vietnamese names, our methods can be adapted for other nationalities, as each country can leverage its own expertise to curate more accurate training data.

Our HDN algorithm still has a weakness: it is highly sensitive to threshold values when determining whether a term is "discriminative" and assigning a positive label. This leads to cases, for instance, the bigram "Hai Long" is classified as Vietnamese, even in non-Vietnamese names like "Jiang Hai Long". While this approach boosts recall, it may reduce precision. To address this, we can refine the discriminative feature verification function to achieve a balance between precision and recall, or consider trigram which will cost more storage and processing time. Additionally, incorporating supplementary expert profile information—such as language proficiency, education background, and professional networks-can further improve the accuracy of identifying Vietnamese experts.

There is a need for methods for discovering outliers and their proper recognition as certain names are not successfully identified by none of the proposed models. To address this limitation, we will incorporate contextual information and external knowledge sources, such as name databases and expert profile attributes, to enhance accuracy. Additionally, most false positives from the HDN method originate from two specific Asian countries when written in Latin characters. We will integrate HDN with a reliable classifier for these names to filter out false positives and improve precision.

References

1. Overseas vietnamese. https://en.wikipedia.org/wiki/Overseas_Vietnamese
2. 100 họ phổ biến ở Việt Nam. NXB Khoa học Xã Hội (2022)
3. Ambekar, A., Ward, C., Mohammed, J., Male, S., Skiena, S.: Name-ethnicity classification from open sources. In: Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD 2009, Association for Computing Machinery, New York (2009)
4. Azzopardi, L., Girolami, M., Crowe, M.: Probabilistic hyperspace analogue to language. In: Proceedings of the 28th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (2005)
5. Burchard, E.G., et al.: The importance of race and ethnic background in biomedical research and clinical practice. N. Engl. J. Med. **348**(12) (2003)
6. Chen, Y., You, J., Chu, M., Zhao, Y., Wang, J.: Identifying language origin of person names with N-grams of different units. In: 2006 IEEE International Conference on Acoustics Speech and Signal Processing Proceedings, vol. 1 (2006)

7. Coldman, A.J., Braun, T., Gallagher, R.P.: The classification of ethnic status using name information. J. Epidemiol. Commun. Health **42**(4) (1988)
8. Han, S., et al.: Generating look-alike names for security challenges. In: Proceedings of the 10th ACM Workshop on Artificial Intelligence and Security, AISec 2017, Association for Computing Machinery, New York (2017)
9. Ho Huong, T., Tran-Trung, K., Truong Hoang, V., et al.: A computational linguistic approach for gender prediction based on vietnamese names. Mobile Inf. Syst. (2022)
10. Hu, Y., Hu, C., Tran, T., Kasturi, T., Joseph, E., Gillingham, M.: What's in a name?–Gender classification of names with character based machine learning models. Data Min. Knowl. Disc. **35**(4) (2021)
11. Ioannidis, J., Baas, J., Klavans, R., Boyack, K.: A standardized citation metrics author database annotated for scientific field. PLoS Biol. (2019)
12. Kang, Y.: Name-nationality classification technology under Keras deep learning. In: BDE 2020, Association for Computing Machinery, New York (2020)
13. King, G., Zeng, L.: Logistic regression in rare events data. Polit. Anal. **9**(2) (2001)
14. Lauderdale, D.S., Kestenbaum, B.: Asian American ethnic identification by surname. Popul. Res. Policy Rev. **19** (2000)
15. Lee, J., Kim, H., Ko, M., Choi, D., Choi, J., Kang, J.: Name nationality classification with recurrent neural networks. In: Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI-2017 (2017)
16. Lund, K., Burgess, C.: Producing high-dimensional semantic spaces from lexical co-occurrence. Behav. Res. Methods Instrum. Comput. **28**(2) (1996)
17. Mateos, P.: A review of name-based ethnicity classification methods and their potential in population studies. Popul. Space Place **13**(4) (2007)
18. OpenAI: GPT-4o system card (2024). https://openai.com/index/gpt-4o-system-card/
19. Pennington, J., Socher, R., Manning, C.: GloVe: global vectors for word representation. In: Moschitti, A., Pang, B., Daelemans, W. (eds.) Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), Association for Computational Linguistics, Doha (2014)
20. Podnar, I., Rajman, M., Luu, T., Klemm, F., Aberer, K.: Scalable peer-to-peer web retrieval with highly discriminative keys. In: Proceedings of the 23rd International Conference on Data Engineering, ICDE 2007, IEEE Computer Society (2007)
21. Schnell, R., et al.: A new name-based sampling method for migrants using N-grams. German Record Linkage Center, Working Paper Series, No. WP-GRLC-2013-04 (2013)
22. Taylor, V.M., Nguyen, T.T., Hoai Do, H., Li, L., Yasui, Y.: Lessons learned from the application of a Vietnamese surname list for survey research. J. Immigr. Minor. Health **13** (2011)
23. Team, G.: Gemini 1.5: Unlocking multimodal understanding across millions of tokens of context (2024). https://arxiv.org/abs/2403.05530
24. To, H.Q., Nguyen, K.V., Nguyen, N.L.T., Nguyen, A.G.T.: Gender prediction based on vietnamese names with machine learning techniques. In: Proceedings of the 4th International Conference on Natural Language Processing and Information Retrieval, NLPIR 2021, Association for Computing Machinery, New York (2021)
25. Treeratpituk, P., Giles, C.: Name-ethnicity classification and ethnicity-sensitive name matching. In: Proceedings of the 26th AAAI Conference on Artificial Intelligence and the 24th Innovative Applications of Artificial Intelligence Conference, Proceedings of the National Conference on Artificial Intelligence (2012)

26. Vaswani, A., et al.: Attention is all you need. In: Proceedings of the 31st International Conference on Neural Information Processing Systems, NIPS 2017, Curran Associates Inc., Red Hook (2017)
27. Wong, K.O., Zaïane, O.R., Davis, F.G., Yasui, Y.: A machine learning approach to predict ethnicity using personal name and census location in Canada. PLoS ONE **15** (2020)
28. Yao, Y., Duan, J., Xu, K., Cai, Y., Sun, Z., Zhang, Y.: A survey on large language model (LLM) security and privacy: The good, the bad, and the ugly. High Confidence Comput. **4**(2) (2024)
29. Ye, J., et al.: Nationality classification using name embeddings. In: Proceedings of the 2017 ACM on Conference on Information and Knowledge Management, CIKM 2017, Association for Computing Machinery, New York (2017)
30. Zhou, P., Qi, Z., Zheng, S., Xu, J., Bao, H., Xu, B.: Text classification improved by integrating bidirectional LSTM with two-dimensional max pooling. arXiv preprint arXiv:1611.06639 (2016)