# Introduction

Machine transliteration is a part of machine translation to deal proper nouns, which are translated with preserved pronunciation onto the target language [[1](#_ENREF_1)]. There are many sources of complications in machine transliteration, and these problems are usually specific to the nature of the source and target languages [[1](#_ENREF_1)]. The key challenge in transliteration is to ensure that the sound is preserved in the target language. For instance, a Chinese name when written in English may never capture the exact pronunciation, but the English spelling would attempt to match the Chinese pronunciation as much as possible within the constraints of the English letters.

In this experiment, we attempt to recover the correct Latin spellings of names transliterated into Persian, a process called as back transliteration. In this experiment, we assume that the correct Latin spelling for any Persian name is available in a dictionary, so the problem attempts to identify the correct spelling from the dictionary.

# Method

We predict the correct Latin spelling from a transcripted Persian pronunciation, by treating the problem as a spelling correction problem, using a dictionary of correct Latin spellings.

We first use global edit distance with weighted operator cost assigning different costs for insertion, deletion and replacement.

We then improve on this model by adding weighted character substitution and insertion matrix to account for similar sounding letters. The character replacement weights are shown in substitution matrix Table 1. The insertion cost matrices for different runs are shown in Table 2 and Table 3.

We also then use Soundex with edit distance and n-gram methods to evaluate if they provide better precision and recall compared to global edit distance.

Table - Replacement cost matrix

|  |
| --- |
| (Source char, Destination Char, Cost) |
| ('a', 'e', .1) |
| ('e', 'i', .1) |
| ('a', 'o', .2) |
| ('o', 'a', .2) |
| ('p', 'f', .1) |
| ('k', 'c', .1) |
| ('k', 'x', .1) |
| ('x', 'k', .1) |
| ('y', 'i', .1) |
| ('y', 'e', .1) |
| ('v', 'u', .1) |
| ('v', 'o', .1) |
| ('v', 'w', .1) |
| ('z', 's', .1) |
| ('z', 'j', .2) |
| ('s', 'c', .4) |
| ('\'', 'a', .1) |

Table - Insertion cost matrix A

|  |
| --- |
| (Character, Cost) |
| ('a', .1) |
| ('e', .1) |
| ('i', .1) |
| ('o', .1) |
| ('u', .1) |
| ('h', .2) |

Table - Insertion cost matrix B

|  |
| --- |
| (Character, Cost) |
| ('a', .01) |
| ('e', .01) |
| ('i', .01) |
| ('o', .01) |
| ('u', .01) |
| ('h', .02) |

## Data

The data set used in this experiment is curated by Karimi et al [[2](#_ENREF_2)] [[3](#_ENREF_3)].

The training dataset contains the Persian name spelling and the corresponding Latin spelling.

The dataset also includes a Latin names dictionary containing the list of valid Latin names.

## Scoring Method

We use precision and recall to evaluate the effectiveness of using edit distance to predict the correct Latin spelling. The correct spellings predicted by the system are the ones with the smallest distance (lowest cost) required to edit the Persian name to the names available in the Latin names dictionary.

# Results

The results using global edit distance, Soundex and n-gram methods are summarized in Table 4.

The best scores were found using insertion and substitution cost matrices for global edit distance with a top precision of 49.27% and a recall of 58.89%. The least suitable distance measure was n-gram, with a best score of just 23.64% recall and 15.58% precision

Table – Precision and recall for various methods

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Method** | **I Cost** | **R Cost** | **D Cost** | **Recall** | **Precision** |
| Global Edit | 1 | 1 | 1 | 42.99% | 04.45% |
| Global Edit | 1 | 2 | 3 | 48.3% | 17.7% |
| Global Edit | 1 | Default 1,  \*Unless in Table 1 | 2 | 44.6% | 16.4% |
| Global Edit | 1 | Default 2,  \*Unless in Table 1 | 3 | 57.7% | 37.7% |
| Global Edit | Default 1,  \*Unless in Table 2 | Default 1,  \*Unless in Table 1 | 2 | 57.91% | 40.21% |
| Global Edit | Default 1,  \*Unless in Table 2 | Default 2  \*Unless in Table 1 | 3 | 61.05% | 46.56% |
| Global Edit | Default 1,  \*Unless in Table 3 | Default 2  \*Unless in Table 1 | 3 | 58.89% | 49.27% |
| Soundex | 1 | 1 | 1 | 56.38% | 02.2% |
| 2-Gram | - | - | - | 23.64% | 15.58% |
| 1-Gram | - | - | - | 14.9% | 06.34% |
| \* Indicates use of cost matrix, with default cost specified here. E.g. 1 with Table 2 indicates that a default cost of 1 is used unless specified in the matrix Table 2 referenced | | | | | |

# Discussion

## Nature of the dataset

Approximately 75% of the Latin name spellings are longer than the Persians ones. A major contributor to the difference in the length is the lack of vowels [a, e, i, o u] between consonants in the Persian spellings. E.g. Persian spelling JRMY corresponds to the Latin name Jeremy. This observation suggests that inserting vowels is more likely to produce the right spelling compared to replacing or deleting a character. Only 3% of the Latin spellings are shorter than the corresponding English ones, and the rest of the names (22%) have the same length.

Some of the Persian names have more than 1 correct Latin spelling, e.g Persian name KYN has 8 Latin spellings - cain / kain/ kane/ kean / keane / keen / kien / kin. Around 10% of the Persian names have more than 1 correct spelling.

All the Latin spellings in the training set are available in the names lookup dataset, making dictionary-based methods for predicting the Latin spelling an appropriate baseline solution.

## Global edit distance to predict spelling

Using a single parameter set scoring method, as detailed in Table 4, for calculating the edit distance has a maximum precision 17%, with a maximum recall of 48%. The low precision is expected, as the scoring method does not account for similarity in the phonemes of the letters, such as a & o when used in spellings.

Using the substitution matrix, see Table 1, to account for similar sounding letters almost doubles the precision from 17% to 37% and the recall increases by almost 10% to 57.7%.

The biggest increase in precision (top score is ~ 49%) is when an insertion matrix, Table 2 & Table 3 is used to lower the cost of inserting vowels. This is inline with the observations of the dataset, given a majority of the Persian names lack vowels between consonants, also substantiated by Karimi et al in his thesis [[4](#_ENREF_4)]

The main shortcoming of this approach is that it does not predict Latin names with 2 consecutive consonants repeated, e.g. Manning (nn). Around 23% of the names in the training dataset exhibit this characteristic.

The next problem with this approach is that it does not account for specific character sequences. For instance, Persian spellings starting with “as” corresponding Latin names starting with “S”, e.g. astyplz needs to be corrected to staples. This is implemented as a finite state transducer to take into account character sequences[[4](#_ENREF_4)]

## N-Gram distance

As a majority of Persian names lack vowels between consonants, the 2 n-gram distances fail to detect similarity between English and Persian name, because there are no n-grams in common. This problem of lack of ability to detect similar words with no-ngrams in common and the lack of sensitivity to order using n-gram distance is also demonstrated by Kondrak [[5](#_ENREF_5)] . For instance the predicted spelling using 2-gram for Persian spelling admvnd, (Edmonds) is predicted as “and” because that is the closest 2-gram distance (due to gram “nd”). 1-gram fares worse than 2 –gram because it seems look for closest anagrams. E.g adryn, with correct Latin spelling Adriane is predicted as randy using 1-gram distance.

## Soundex distance

Although Soundex seems a logical fitting solution to find a spelling that retains the original pronunciation, it has low precision ~ 2.2% despite having ~56.38 recall. For e.g the Persian spelled name amy (Latin ami) with soundex code A500 is matched with 74 names ranging from ann to aoyoun. This problem of soundex matching dissimilar names is also identified by Zobel et al[[6](#_ENREF_6)]. Zobel [[6](#_ENREF_6)] proposes a variation of the soundex algorithm, mainly Phonix+ & editex, Phonix+ doesn’t truncate codes to a 4 letter code, like soundex does, reducing false matches [[6](#_ENREF_6)] .

# Conclusion

As there are no short vowels in the Persian language[[4](#_ENREF_4)]assigning low insert cost to vowels improves the precision & recall. This can be further improved to take into account specific character sequences, using methods like the finite state transducer [[4](#_ENREF_4)].

N-gram does not perform well because of the lack of common n-gram between Persian and Latin spellings. Soundex method matched too many dissimilar names significant lowering precision.

# Citations

1. Karimi, S., F. Scholer, and A. Turpin, *Machine transliteration survey.* ACM Computing Surveys (CSUR), 2011. **43**(3): p. 17.

2. Karimi, S., A. Turpin, and F. Scholer, *English to persian transliteration*, in *Proceedings of the 13th international conference on String Processing and Information Retrieval*. 2006, Springer-Verlag: Glasgow, UK. p. 255-266.

3. Karimi, S., A. Turpin, and F. Scholer. *Corpus effects on the evaluation of automated transliteration systems*. in *ANNUAL MEETING-ASSOCIATION FOR COMPUTATIONAL LINGUISTICS*. 2007.

4. Karimi, S., *Machine transliteration of proper names between English and Persian*. 2008, RMIT University, Melbourne.

5. Kondrak, G. *N-gram similarity and distance*. in *International Symposium on String Processing and Information Retrieval*. 2005. Springer.

6. Zobel, J. and P. Dart. *Phonetic string matching: Lessons from information retrieval*. in *Proceedings of the 19th annual international ACM SIGIR conference on Research and development in information retrieval*. 1996. ACM.