Group 8:An approach to Digaai

Yueh Ying Lee, James Craver, Jeffrey Wu, Bowen Zhang

Department of Computer Science, Boston University

Goal

The goal of this project is to identify inputs, which are first and last names, as Brazilian or not, as shown in Table 1. As many Romance languages share similar features, it can sometimes be difficult to identify the particular nationality to which a name belongs.

Id	Firstname	Lastname	IsBrazilian
0	Ithalohfonseca	Chaverinho	1
1	Gustavo	Gon\xc3\xa7alves	1
2	Rafael	Geraldo	1
3	Cristyan	Victor	1

Table 1. A list of entries from our training set

Our method

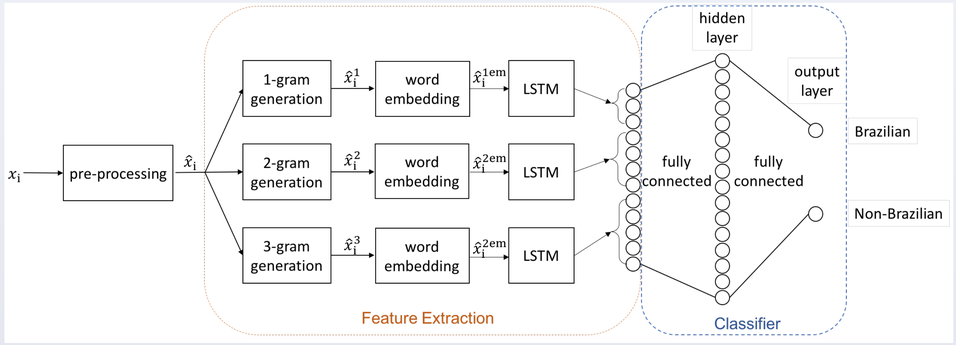

Figure 1: System overview, inspired by [2]

Pre-processing scheme:
(1) decoding (UTF-8), (2) converting to lower case, (3) concatenating first name and last name using $ and + as delimiters
Example: x_i=(Gustavo, Gon\xc3\xa7alves) → $\hat{x}_i$ = $gustavo$+gonçalves+

Feature extraction scheme :
(1) generate 1-gram, 2-gram, and 3-gram sequences for each training entry
(2) embed using the pre-trained skip-gram model proposed in [2].
*using efficient implementation Word2Vec [3].
(3) Further feature extraction using many-to-one (last output) LSTMs, and then concatenate the 3 LSTM outputs as our final feature.

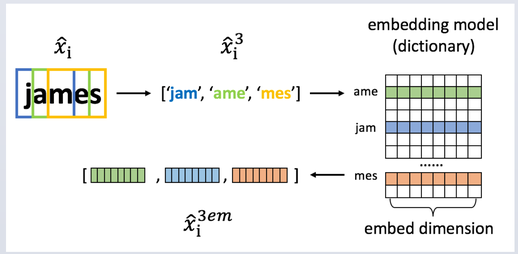

Figure 2: A 3-gram embedding example

Classifier :
Classify using 2 fully-connected layers.
Loss function : softmax loss.
To avoid overfitting, dropout is applied to the hidden layer.
Output : 2 logits, a pair of scores (non-Brazilian score, Brazilian score);

Predict label : $\hat{y} = \begin{cases} 1 & Brazilian \geq non\ Brazilian \\ 0 & otherwise \end{cases}$

Dataset breakdown and Evaluation Metric

Training (pos/ neg/ total)	23965 / 24048 / 48013
Testing (pos/ neg/ total)	NA / NA / 11941
Validation (pos/neg/total)	NA / NA / NA

Table 2, Dataset summary

Evaluation metric: classification error $\frac{FN+FP}{TP+TN+FN+FP}$
Since there is no validation set in the dataset,
we use 10-fold cross validation to choose our hyper-parameters.

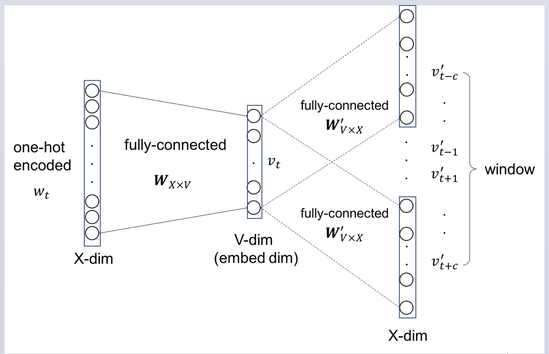

Figure 3: Skip-gram model. Note the shared output weight matrix $W'_{V\times X}$

Results

a. Baseline Method

Features : $(\hat{x}_i^{1em}, \hat{x}_i^{2em}, \hat{x}_i^{3em})$
Classifier : random forest
Hyper-parameters and performance summarized in Table 3.
Test on 1 validation set is shown in Figure 4.

Note fixed the embedding parameters as shown in Table 4.

Hyper-parameter	Optimal
Tree depth (1~14)	10
Number of trees (20~140)	50
10-fold cross-valid error	0.8324

Table 3: Baseline hyper parameters and performance

1-gram embed dim	10
2-gram embed dim	100
3-gram embed dim	200
window size	5
# of training iteration	10

Table 4: Skip-model parameters

b. Main approach

Table 5 shows a summary of various hyper-parameters and performance.
The optimal value is chosen based on average error of 10-fold cross validation.
Figure 5, 6 shows a diagram of error rate vs number of epochs on different setups.

The testing error of almost all parameter combinations converged to approximately 10% after 1500 epochs, so we choose the number of training epochs as 2000.

We furthermore observed that while training error decreased with more epochs, testing error did not increase. We therefore conclude that we have not succumbed to overfitting to the training data.

After the hyper-parameters were determined, we tried different skip-gram model parameters and found little effect, as shown in Figure 7. We therefore used the same parameters as in Table 3.

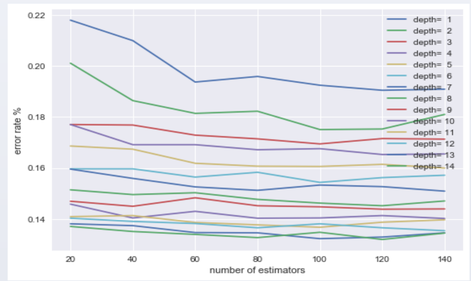

Figure 4: Baseline parameter configuration experiment. As in our main approach, the metric is classification error. This diagram illustrates testing on 1 validation set.

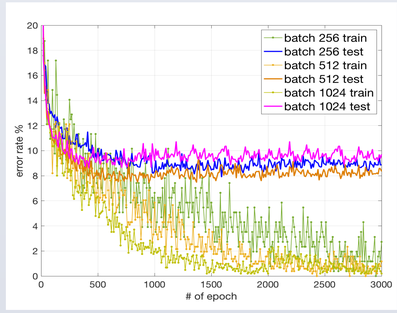

Figure 5: Error rate for chosen batch sizes, testing on one validation set. Conducted with learning rate is 1e-3, dropout ratio 0.5, hidden layer dimension 200.

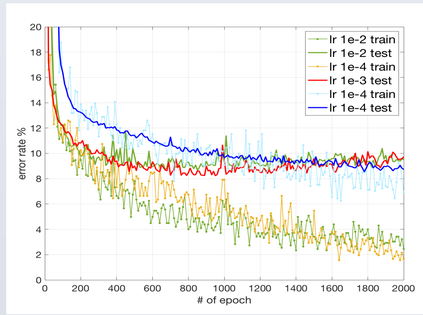

Figure 6: Error rate for chosen learning rates, testing on one validation set. Conducted with batch size 1024, dropout ratio 0.5, hidden layer dimension 200.

Hyper-parameters	optimal
Batch size (128, 256, 512, 1024, 2048)	512
Dropout ratio (0.25, 0.5, 0.75)	0.5
Learning rate (1e-2~1e-5)	3.5e-3
Hidden layer dimension (50~300)	200
Average cross validation error	0.9128

Table 5: Hyper-parameter summary and performance

rank	Team	accuracy
1	**gonna_GG (our approach)**	0.9083
2	Yuxi Jiang	0.9026
3	Federico Siano	0.8944
4	**RF_GG (our baseline)**	0.8380

Table 6: Kaggle submission rank

Typical mistake

To inspect why our validation error does not decrease, we analyzed some incorrect predicted validation entries. The false positive ratio and false negative ratio had similar amounts, both around 5%.

Major FP entries breakdown : (1) rarely used names (2) one-sided labels
Major FN entries breakdown : (1) rarely used names (2) names possibly corresponding to immigrants

One-sided labels example : Last name Victor*, 107 entries, 105/107 are Brazilian
Possible immigrants names : Asian names like Linh Hoang Len Da

Future work

To reduce the error of rare names, more training entries would be required, but for second kind error in both the FP and FN cases, more entries would not necessarily help, as these correspond to outliers in their respective group. Further features not available to us would be required to reduce this kind of error, as names have proven to be insufficient.

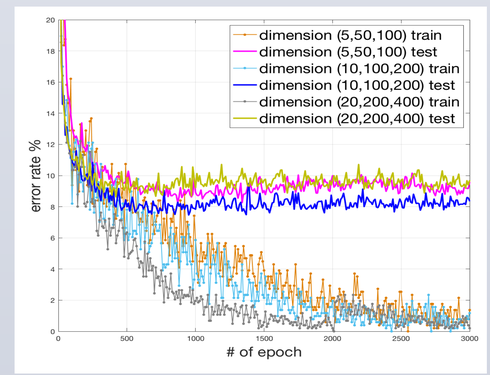

Figure 7: Experiment on different embedding dimensions in skip-gram model. Hyper-parameters were fixed according to Table 5.The triplet in legend means (1-gram embed dimension, 2-gram embed dimension, 3-gram embed dimension).

References

[1] Mikolov, Tomas, Ilya Sutskever, Kai Chen, Greg S. Corrado, and Jeff Dean Distributed representations of words and phrases and their compositionality, NIPS 2013
[2] Jinhyuk Lee, Hyunjae Kim, Miyoung Ko, Donghee Choi, Jaehoon Choi, Jaewoo Kang, Name Nationality Classification with Recurrent Neural Networks, IJCAI 2017
[3] Word2Vec library, https://radimrehurek.com/gensim/models/word2vec.html

BOSTON UNIVERSITY