Evaluating the Impact of Model Complexity on the Classification of Eastern and Western European Authors in Subreddit-based Binary Classification

Pamela Cece p.cece@student.tue.nl
Julia Dobladez Brisquet j.dobladez@student.tue
Isha Narang i.narang@student.tue.nl
Sneha Ajay s.ajay@student.tue.nl

Abstract

In the context of author classification in Eastern and Western, this article investigates the comparative efficacy of augmenting complexity in binary classification models. The core findings underscore the importance of advanced model architecture in author attribution. The broader significance lies in refining frameworks enhance information retrieval systems, contributing to a nuanced understanding of online discourse.

1 Introduction

[1] This paper is focused on a binary text classification on author attribution and profiling. The central research question of this paper is: What is the comparative efficacy of increasing complexity in binary classification models as compared to baseline models for classifying Eastern and Western authors based on subreddits?

Many studies have been performed on binary classification models, multi-classifications, or even comparison of models, but not a comparison where the complexity of the model is being increased while performing the task. The models compared were Simple Logistic Regression (SLR), a Naive Bayesian (NB) model, SLR with word embedding, a Neural Network with word embedding, and FastText (from lowest to highest complexity). Using different metrics such as F1 score, Recall, and Matthew's Correlation Coefficient (MCC). The central findings were that FastText performed the best, hence to answer the core question the efficacy is better when increasing complexity.

2 Related Works

Many researchers have offered different approaches towards performing tasks in NLP by designing new transfer learning algorithms. (Howard and Ruder, 2018) proposed Universal Language Model Fine-tuning (ULMFiT), an effective transfer learning method that can be applied to any task in NLP. As a starting point to start performing classification tasks in NLP, ULMFiT was used but was later discarded as compatibility issues occurred. Meanwhile, (Ahmed et al., 2004) targeted language identification, hence presenting a procedure based on N-grams and cumulative frequency addition for automatically determining the language of a given text. This paper applied said procedure on the binary classification task, as well as used different types of features such as subjectivity, Named Entity Recognition (NER), and number of unique words.

In his book *Authorship Attribution*, Juola discussed various techniques for identifying the authors of written texts as well as the assumptions that most researchers make. However, Juola argues that there are reasons to believe that said patterns may be too complex to detect through average word length or vocabulary size. (Luyckx and Daelemans, 2008) stated that most studies performed on authorship attribution have focused on few authors, plus debated that these studies overestimated the performance of their system and the importance of linguistic features. The obtained results showed that increasing the number of authors caused a significant decrease in the performance of models, which concluded that similar types of features work well for different amounts of authors but generalizations about individual features were not helpful.

Additionally, investigated the effectiveness of feature selection and robustness of a memory-based learning approach doing authorship attribution and limited training data compared to eager learning methods such as SVMs. Similarly, this paper compares the efficacy of increasing complexity in binary classification on authorship attribution by having as baseline models a Simple Logistic Regression (SLR) and a Naive Bayesian (NB) model, while the more complex models are SLR with word embedding, a Neural Network with word embedding, and FastText.

3 Data

The dataset comprises 76,405 entries collected from Reddit, with 53,288 posts from Western users and 15,471 from Eastern users. Notably, Germany emerges as the most represented country within the subgroup of Western users, while Poland leads in representation within the Eastern subgroup. In the graph. The graph also visualizes our classification criteria for Eastern European and Western countries, with distinctions made based on geographical and cultural considerations. 2, 3

To address the substantial difference in entry count between Eastern and Western user posts, undersampling was employed for balanced representation. This approach ensures a fair evaluation of classification models across the diverse dataset. The final dataset used came to 30942 entries, with half being posts from western users and the other half from eastern.

[1] GitHub Repository: https://github.com/jdobladez/LangAI-Assignment (Accessed: January 12, 2024)

Examining bigram distributions among classes reveals interesting patterns. The top 30 bigrams for Eastern users include common phrases like "I'm," "can't," and "that's," reflecting colloquial usage. In comparison, Western users exhibit similar trends with frequent bigrams such as "I'm," "can't," and "that's." However, there are some differences between the two groups, for example the bigram (”die”, “die”) appears in the Western top 30, reflecting the German language. These bigrams are absent in the Eastern ones, which indicates that the model might need to capture language nuances.4, 5

The top bigrams provide a foundation for understanding language intricacies within each subgroup, aiding in the design and evaluation of classification models. Further analysis and interpretation of these patterns will be crucial in drawing meaningful conclusions about the comparative efficacy of increasing complexity in binary classification models for distinguishing Eastern and Western authors based on subreddit posts.

4 Method and Experimental Setup

4.1 Preprocessing

In preparing the data for experimentation, we employed several preprocessing steps to enhance the quality of the text features:
Lowercased: Text was converted to lowercase to ensure uniformity in word representation. Removed Stop Words: Common stop words were eliminated using the NLTK library, reducing noise and improving model focus.
Lemmatization: Employed NLTK's lemmatization to reduce words to their base form, enhancing semantic understanding.
Tokenization: Utilized spaCy's tokenization to break down text into meaningful units, aiding in subsequent analysis.
Added Features: Extracted additional features including average word length, average sentence length, type-token ratio (TTR), normalized counts of punctuation, number of unique words, number of characters, number of contractions, and text subjectivity (using TextBlob). These features aimed to provide a richer context for the models.

4.2 Models Chosen

1. **Simple Logistic without Embeddings**: Chosen as a baseline model due to its simplicity and interpretability, facilitating a clear
2. **Naive Bayes (N-gram)**: Selected as another baseline, leveraging n-gram features to capture bi- and tri-gram patterns. Laplace smoothing was employed for generalization.
3. **Simple Logistic Regression with Embeddings**: Introduced embeddings to the logistic regression model to capture semantic relationships between words. Embeddings were obtained using GloVe pretrained vectors as initial weights for word2vec models.
4. **Neural Network with Embeddings**:Employed neural networks with the same embeddings as the past model.
5. **FastText**: Applied for its ability to handle text classification efficiently. Preprocessed data as required for FastText and experimented with bigram representation, adjusting epochs and learning rates.

4.3 Training Setup

1. **Simple Logistic without Embeddings**: Trained with an 80/20 split, L2 regularization penalty, and 10,000 epochs.
2. **Naive Bayes (N-gram)**: Utilized an 80/20 split, n-gram features, and Laplace smoothing. Employed NLTK library for bi- and tri-gram extraction. Applied 5-fold cross-validation.
3. **Simple Logistic Regression with Embeddings**: Trained without additional complexity.
4. **Neural Network with Embeddings**:Employed early stopping and hyperparameter tuning for hidden layer sizes, learning rate initialization, and alpha values.
5. **FastText**: Trained with bigram representation, 10 epochs, and a learning rate of 0.1. Applied 5-fold cross-validation.

4.4 Performance Metrics

To assess the performance of each model comprehensively, a suite of metrics was employed:
Precision: assesses positive prediction accuracy, prioritizing reliability in labeling positive instances and minimizing false positives
Recall: measures the model's sensitivity, identifying all relevant instances, crucial when capturing all positive cases
F1 Score: strikes a balance between Precision and Recall, providing a harmonic mean for a nuanced compromise between false positives and false negatives.
Matthews Correlation Coefficient (MCC): offers a balanced measure by considering all elements of the confusion matrix, making it particularly useful for binary classification tasks in imbalanced datasets. This metric provides a robust assessment of classification effectiveness.
Accuracy: while offering an overall correctness measure, is complemented by the other metrics. It ensures a holistic view.

5 Results

5.1 Evaluation Metrics

Legend:

- Simple Logistic regression - SLR base
- Naive Bayes - NB
- Simple Logistic regression (with word embedding) - SLR2
- Neural Network (with word embedding) - NN
- FastText - FT
- Matthew's Correlation Coefficient - MCC
- Class 0 = Eastern Europe
- Class 1 = Western Europe

In Table 1, the MCC shows a huge improvement from the SLR base to NB. It continues to improve steadily with each model. The NN and FT have identical F1 scores of 0.91 over both classes but differ significantly in MCC.

Models / Class		SLR base	NB	SLR 2	NN	FT
Precision	*Class 0*	*0.58*	1.00	0.80	0.90	0.90
	Class 1	*0.60*	0.63	0.81	0.92	0.92
Recall	*Class 0*	0.62	0.39	0.81	0.93	0.93
	Class 1	0.57	0.63	0.80	0.89	0.89
F1 score	*Class 0*	0.60	0.56	0.80	0.91	0.91
	Class 1	0.59	0.77	0.81	0.91	0.91
MCC		0.19	0.53	0.61	0.71	0.82
Accuracy		0.59	0.70	0.81	0.88	0.90

Table 1: Evaluation metrics calculated on the test set (20% of the data)

5.2 Over-fitting Analysis

Laplace Smoothing Validation Accuracy for Naive Bayes Model to mitigate issues on unseen n-grams: 0.714
Cross validation scores were used as a metric for accessing overfitting in all the models except the neural network:

Models	SLR base	NB	SLR2	FT
Fold 1:	0.609	0.876	0.813	0.925
Fold 2:	0.564	0.839	0.777	0.919
Fold 3:	0.554	0.811	0.705	0.920
Fold 4:	0.612	0.826	0.612	0.922
Fold 5:	0.577	0.777	0.731	0.921

Table 2: 5-fold Cross Validation Scores across models

For the neural network, the training and testing graphs were used to test for overfitting. 1

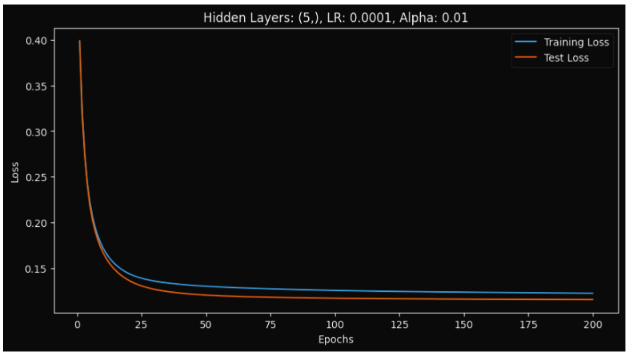

Figure 1: Training and test loss graph for NN

5.3 Feature importance

The SLR base model's absolute coefficients for features are shown in the table below. For SLR2, there were a total 111 features (including the 100 dimensional features of the document embeddings). The rankings are calculated in descending order of coefficient values. (refer to table 3 in appendix)

6 Discussion

In our binary classification task, minimizing false positives does not have precedence over minimizing false negatives since the assignment of class 1 to Western is completely arbitrary. Therefore, the F1 score, the harmonic mean between recall and precision, is more relevant to our task. The MCC also provides a balanced measure (taking into account both false positives and negatives).

In terms of overfitting, all of the models excluding the NN, the CV scores are relatively similar across the 5 folds with no extreme values, indicating lower possibility of overfitting. For the NN, the training and test losses (refer to Fig 1) decrease rapidly within about 10 epochs and then plateau at a loss of around 0.07. This is indicative of a low possibility of overfitting.

6.1 Model Discussion

6.1.1 Simple Logistic Regression (baseline model)

The SLR base model exhibits moderate performance with precision scores of 0.58 and 0.60 for Eastern and Western instances, indicating a tendency for false positives for Eastern authors, indicating challenges in correctly identifying non-Eastern authors. However, recall scores of 0.62 and 0.57 reveal a possibility of false negatives for Western authors and a precision-recall trade-off. F1 scores reflect this trade-off, with Class 0 at 0.60 and Class 1 at 0.59. The MCC value of 0.19 suggests moderate agreement between predicted and actual values, emphasizing the need for improvements in robust classification.

Thus, the model faces challenges related to sensitivity to subreddit context and linguistic nuances (like punctuations), evident from the recall values and absolute feature coefficients (as shown in Table 3).

6.1.2 Naive Bayes (baseline model)

The Naive Bayes model employs both 2-gram and 3-gram features, providing comparatively better performance metrics than the simple logistic regression model. Table 1 shows that while exhibiting lower precision in classifying Western European authors, recall values of 0.39 for Class 0 reflect challenges capturing all Eastern European linguistic patterns. The F1 scores further highlights the precision-recall trade-off, signaling potential for enhanced performance.

With an improved MCC of 0.53 from the previous SLR base model, the NB model excels in certain aspects of author classification as also supported in research (Luo, 2021) and (Pranckevičius and Marcinkevičius, 2017). The validation accuracy of 0.714 with Laplace smoothing showcases the model's generalization to new, unseen n-grams. Despite this, the observed disparity in recall scores between classes highlights limitations. These findings underscore the need for advanced models to improve performance and robustness across diverse linguistic contexts, as explored in the sections below:

6.1.3 Simple Logistic Regression (with embeddings)

The accuracy, precision, recall and F1 score are consistently around 0.80. This indicates that the model does not outperform in the classification of one label over the other. The grammatical and linguistic features are not as influential in the prediction task as compared to the dimensionality features (document embeddings). For example, the highest coefficient ranking is 49 out of 111 (refer to Table 3). This is also supported by the huge improvement in MCC (+0.34) from SLR base to SLR with embeddings. These embeddings can capture semantic and contextual information, which can be essential for understanding the nuances in communication styles across the regions (Selva Birunda and Kanniga Devi, 2021). Moreover, the embeddings were trained using pre-trained GloVe vectors from the Twitter database thereby providing the model with a "basic vocabulary" (Rios and Lwowski, 2020) upon which the embeddings were re-trained, essentially allowing the model to start training with better initial weights.

However, the MCC is relatively low at a value of 0.61 and can be further improved. This is potentially because the logistic regression model relies on manually engineered features and is not able to learn non-linear relationships.

6.1.4 Neural Network (with embeddings)

The precision, recall and accuracy The F1 score for the NN is consistent across both classes at 0.91, indicating steady performance across classes. The increase in MCC compared to SLR2, is possibly due to its ability to capture non-linear relationships between the document embeddings through representation learning.

The NN utilizes word embeddings (obtained from pre-trained GloVe vectors). Therefore, the lower MCC compared to FastText is most likely due to its inability to handle out-of-vocabulary (OOV) words since the principle of GloVe vectors is word co-occurrence matrices calculated from the training corpus (Kandi, 2018).

6.1.5 FastText

The F1 score of FastText is again consistent across both classes at 0.91 (identical to that of NN). Therefore, similar to NN, FastText shows steady performance across classes.

The higher MCC of 0.82 suggests that FastText is more adept at balancing all four elements of the confusion matrix. This increased MCC is possibly attributed to the fact that alongside word embeddings, sub-word embeddings are included in FastText. These help capture OOV words through n-grams (bi-gram implemented here) as these are unique representations of the training corpus (Minaee et al., 2021). For example, suppose the word "Parisian" is an OOV word. In that case, FastText can calculate its word embedding using the embeddings of the subwords (n-grams) like "Pari", "is", "and sian", thus conceptualizing the context of the word (Minaee et al., 2021). Therefore, FastText can handle morphologically rich language. Such language is evident in the social media sphere (Kincl et al., 2019).

6.2 Limitations & Future Works

Therefore, the strengths of the strongest model, FastText include its ability to use word embeddings created through bigrams to understand the intrinsic semantic differences in language between Eastern and Western European Reddit authors. However, the limitations of the model arise from its "black-box" nature. It is virtually impossible to understand or assign *characteristics* to each dimension of the word vector. Without a clear understanding of what each dimension represents, it is difficult to understand which features or essentially dimensions contributed the most to the classification task. FastText's performance on our binary classification task reveals that there are differences in the way Eastern and Western European authors engage on Reddit however what these differences represent and where they arise from are overlooked in our research paper.

A possible future work could be to look into the interoperability of word embeddings or dimensions. We could look into lower dimensional representation of the word embeddings to find patterns that could be encoded as features into the more interpretable models like the SLR. To further understand the nuances in the language used between the two regions, certain cultural phrases could be encoded as embeddings from the trained word embedding models (either FastText or Word2Vec) and the cosine similarities between each document embedding and the *cultural phrases* could be included as features in the model.

7 Conclusion

In categorizing Eastern and Western European authors based on subReddit content, the study initiated with SLR base and the NB model using 2-gram and 3-gram features as baseline models. Despite moderate performance,and insensitivity to linguistic nuances, these models set the stage for our subsequent analyses. NB displayed improved precision (MCC: 0.53) over SLR base, but its recall scores highlighted the need for more advanced models.

Increasingly intricate models enhanced classification performance. Embeddings improved the SLR model, boosting MCC. Pre-trained GloVe vectors from Twitter refined regional communication nuances. The NN model, with heightened complexity, outperformed the SLR, highlighting non-linear relationships' importance. FastText excelled, attaining a 0.82 MCC, showcasing efficiency in balancing morphological information and handling OOV words. These findings affirm the potential of sophisticated models in achieving balanced classification across diverse elements.

The substantial MCC improvement from the logistic baseline to the Naive Bayes classifier and subsequent complex models highlights the importance of word representation in text classification. Research consistently emphasizes the superiority of n-grams and logistic regression models enriched with insightful features over Naive Bayes baselines (Pranckevičius and Marcinkevičius, 2017). This influenced the decision to incorporate word embeddings and complex models. The results affirm the direct correlation between model complexity and overall performance, substantiating existing literature (Weatherwax and Epstein, 2013).

Authorship Statement

- *Preprocessing*: Pamela Cece
- *Model Training*: Isha Narang, Sneha Ajay
- *Research*: Isha Narang, Sneha Ajay, Pamela Cece, Julia Dobladez Brisquet
- *Balancing Data*: Julia Dobladez Brisquet
- *Writing Report*: Isha Narang, Sneha Ajay, Pamela Cece, Julia Dobladez Brisquet

Each author played a crucial role in different aspects of the project. Pamela Cece was responsible for preprocessing the data, while Isha Narang and Sneha Ajay led the model training process. The research phase involved the collective efforts of all four authors, contributing diverse perspectives. Julia Dobladez Brisquet took charge of balancing the data, ensuring its integrity. The final report was collaboratively written by all authors, consolidating their findings and insights.

References

Bashir Ahmed, Sung-Hyuk Cha, and Charles C. Tappert. 2004. Language identification from text using n-gram based cumulative frequency addition.

Jeremy Howard and Sebastian Ruder. 2018. Universal language model fine-tuning for text classification.

Patrick Juola. 2008. Authorship attribution. *Foundations and Trends® in Information Retrieval*, 1:233–334.

Shabeel Meemulla Kandi. 2018. Language modelling for handling out-of-vocabulary words in natural language processing. *Diss. Doctoral dissertation.*

Tomáš Kincl, Michal Novák, and Jiří Přibil. 2019. Improving sentiment analysis performance on morphologically rich languages: Language and domain independent approach. *Computer Speech & Language*, 56:36–51.

Xiaoyu Luo. 2021. Efficient english text classification using selected machine learning techniques. *Alexandria Engineering Journal*, 60(3):3401–3409.

Kim Luyckx and Walter Daelemans. 2008. Authorship attribution and verification with many authors and limited data. *International Conference on Computational Linguistics*, 22:513–520.

Shervin Minaee, Nal Kalchbrenner, Erik Cambria, Narjes Nikzad, Meysam Chenaghlu, and Jianfeng Gao. 2021. Deep learning–based text classification: a comprehensive review. *ACM computing surveys (CSUR)*, 54(3):1–40.

Tomas Pranckevičius and Virginijus Marcinkevičius. 2017. Comparison of naive bayes, random forest, decision tree, support vector machines, and logistic regression classifiers for text reviews classification. *Baltic Journal of Modern Computing*, 5(2):221.

Anthony Rios and Brandon Lwowski. 2020. An empirical study of the downstream reliability of pre-trained word embeddings. In *Proceedings of the 28th International Conference on Computational Linguistics (COLING 2020)*.

S Selva Birunda and R Kanniga Devi. 2021. A review on word embedding techniques for text classification. *Innovative Data Communication Technologies and Application: Proceedings of ICIDCA 2020*, pages 267–281.

John L Weatherwax and David Epstein. 2013. A solution manual and notes for: The elements of statistical learning by jerome friedman, trevor hastie, and robert tibshirani.

A Appendix

Features	**Absolute Coefficient (SLR_base)**	**Ranking (SLR2)**
Nr. of commas	1.984	49
Type Token Ratio	0.954	94
Nr. of periods	0.602	75
Average word length	0.312	82
Subjectivity	0.306	53
Nr. of question marks	0.185	105
Nr. of exclamation marks	0.136	95
Nr. of contradictions	0.025	107
Nr. of unique words	0.009	110
Nr. of characters	0.001	111
Average sentence length	0.0003	109

Table 3: Feature importance coefficients and ranking for models SLR base and SLR 2

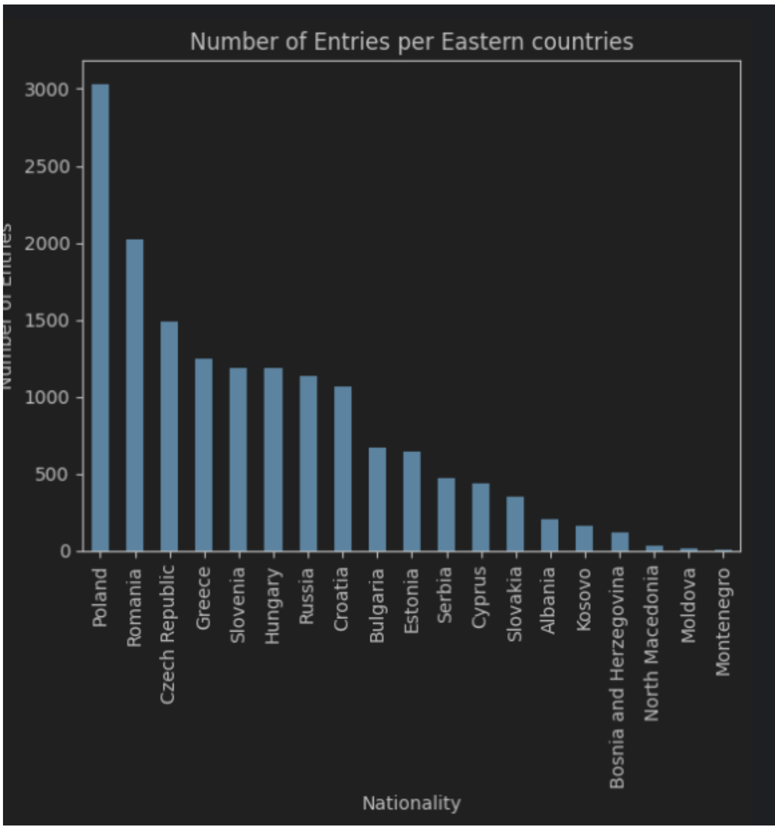

Figure 2: Number of Entries per Eastern Country

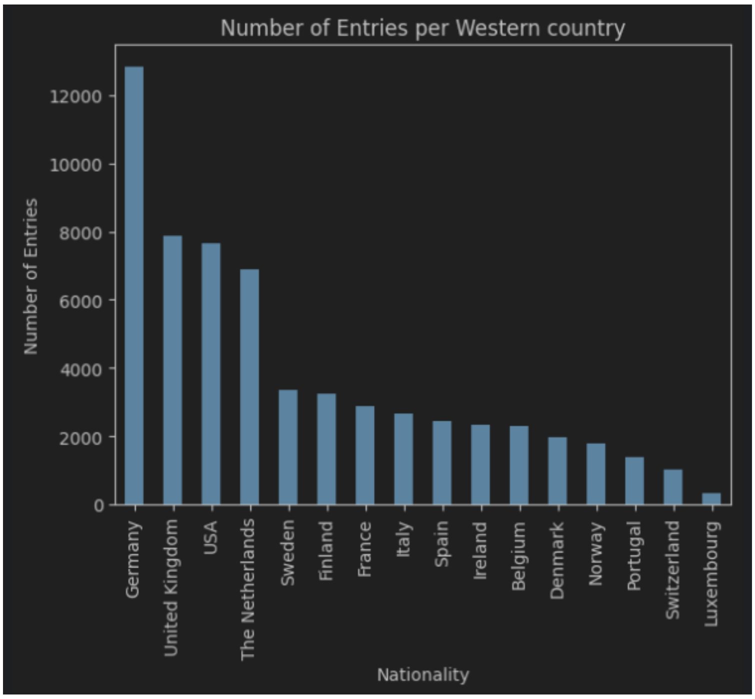

Figure 3: Number of entries per Eastern Country

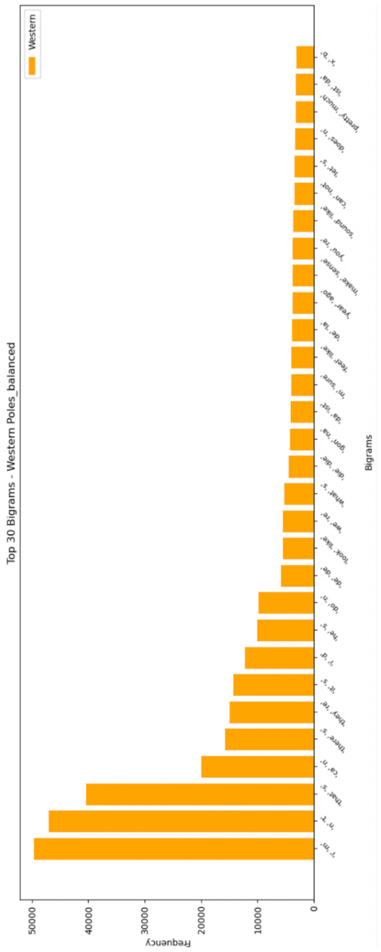

Figure 5: Top 30 most frequent bi-grams for Western users

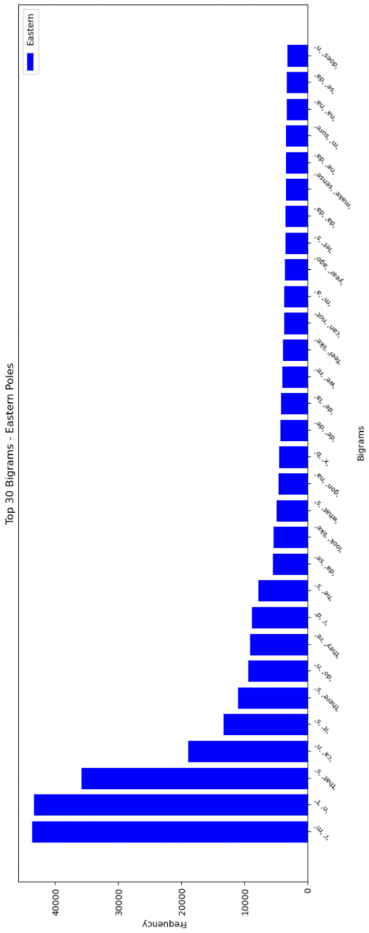

Figure 4: Top 30 most frequent bi-grams for Eastern users