A study of N-gram and Embedding Representations for Native Language Identification

Sowmya Vajjala **Sagnik Banerjee**
Iowa State University, USA
sowmya,sagnik@iastate.edu

Abstract

We report on our experiments with N-gram and embedding based feature representations for Native Language Identification (NLI) as a part of the NLI Shared Task 2017 (team name: NLI-ISU). Our best performing system on the test set for written essays had a macro F1 of 0.8264 and was based on word uni, bi and trigram features. We explored n-grams covering word, character, POS and word-POS mixed representations for this task. For embedding based feature representations, we employed both word and document embeddings. We had a relatively poor performance with all embedding representations compared to n-grams, which could be because of the fact that embeddings capture semantic similarities whereas L1 differences are more stylistic in nature.

1 Introduction

Native Language Identification (NLI) refers to the task of identifying the native language (L1) of a writer based on their writings in another language (L2). Identifying the L1 of a writer is useful in applications such as authorship attribution, forensic linguistics, language instruction and Second Language Acquisition (SLA) (Koppel et al., 2005; Estival et al., 2007; Jarvis and Crossley, 2012). While early work on this problem began at the beginning of this century (Tomokiyo and Jones, 2001; Jarvis et al., 2004), there has been an increased interest in this task since 2012, with the availability of some publicly accessible corpora (Brooke and Hirst, 2012; Tetreault et al., 2012; Bykh and Meurers, 2012).

The First NLI Shared Task (Tetreault et al., 2013) and the release of large corpora such as TOEFL11 corpus of non-native English (Blanchard et al., 2013) and EFCAMDAT corpus (Geertzen et al., 2013) resulted in a surge of research in this area in the past few years. While most of the NLI research has been on English, there is a significant amount of work on other language texts such as Chinese, Finnish and Arabic (Malmasi and Dras, 2015; Malmasi, 2016). Starting form surface linguistic forms such as words and characters to deeper syntactic structures, a range of features have been explored for this task in the past five years.

The last few years saw the field of NLI advance in both the directions of feature engineering and modeling. However, irrespective of what modeling choices were made, results seem to show that word level features still are the most predictive ones as a single group (e.g., Jarvis et al., 2013; Gebre et al., 2013) for this data. So, in this paper, we take a step back from complex feature and model engineering, and explore how far can we get by doing classification using simpler feature representations based on words, characters and POS tags. While our current experiments (team name: NLI-ISU), done as a part of the NLI Shared Task 2017 (Malmasi et al., 2017), do not result in any improvements over existing approaches, we believe they provide insights into the nature of the task and why n-grams may still be needed for this task despite the presence of more compact embedding representations for texts.

The rest of this paper is organized as follows: The next section describes some of the related work and puts our experiments in context. Section 3 briefly describes the corpus used. We describe our methodology including feature description in Section 4. Our experiments and results are discussed in Section 5. Section 6 concludes the paper with pointers to future work.

Proceedings of the 12th Workshop on Innovative Use of NLP for Building Educational Applications, pages 240–248
Copenhagen, Denmark, September 8, 2017.

2 Related Work

Native Language Identification is generally treated as a supervised text classification problem in computational linguistics literature. (Koppel et al., 2005) can be described as one of the early works that considers NLI as a supervised machine learning problem. Using a corpus of texts from International Corpus of Learner English (ICLE) along with word and letter n-grams and errors made by the learners as features, they achieved a classification accuracy of over 80%.

Along with n-grams, syntactic features based on parse structures were also shown to be useful for the task in the past (Wong and Dras, 2011) resulting in accuracies in the range of 80-85% with ICLE data. Extending the n-gram based feature sets to larger n-gram sizes and using a combination of word and POS tag n-grams, Bykh and Meurers (2012) achieved an accuracy of 89.7% on the same dataset. With combinations of n-grams, lexical and syntactic features, Brooke and Hirst (2012) explored NLI with multiple corpora, and achieved accuracies of over 90% on ICLE data. Summarizing the research on NLI until then, Tetreault et al. (2012) explored a range of features on ICLE and introduced the TOEFL11 corpus for NLI (Blanchard et al., 2013).

This corpus was used in the first Native Language Identification shared task (Tetreault et al., 2013). 29 teams participating in the task, and wide range of lexical and syntactic feature representations were explored. The best performing system (Jarvis et al., 2013) resulted in an accuracy of 83.6% and used word, char, POS n-gram features.

After this shared task, interest in NLI continued with different groups exploring both finer feature representations and diverse ensemble methods for combining multiple classification models. These explorations resulted in an accuracy gain of up to 2% on the 2013 shared task test set (Ionescu et al., 2014; Bykh and Meurers, 2014, 2016). More recently, (Malmasi and Dras, 2017) reported an accuracy of 87.1% on the 2013 test set, using an ensemble of meta classifiers and a range of word level and syntactic features. Apart from TOEFL11, other corpora such as EFCAM-DAT (Geertzen et al., 2013) were also used for NLI in the recent past (e.g., Nisioi, 2015).

While most of the work in NLI happened in English, a substantial body of NLI research happened in the past two years covering at least six other languages (cf. Malmasi and Dras, 2015; Malmasi, 2016). In addition to using the written responses, a recent development has been the use of speech transcripts and audio features for dialect identification (Malmasi et al., 2016) and native language identification (Schuller et al., 2016). In this background, the NLI Shared Task 2017 was proposed, with an additional spoken language component.

While a range of feature representations and modeling representations have been explored from this task, it has been shown that word/character level n-grams have been unreasonably effective as a single feature group (e.g., Jarvis et al., 2013; Gebre et al., 2013; Bykh et al., 2013). As Jarvis et al. (2013) concluded, "complex features" such as suffixes, length, lexical variety etc did not result in any major improvement over n-gram features. Further, other complex and memory intensive representations such as constituency and dependency parses did not result in large performance improvements without the support of stronger models and ensemble learners.

In this background, in this paper, we take a step back from exploring new feature extraction methods and new modeling techniques, and re-investigate the role of surface feature representations in NLI. Word and document embeddings became popular and useful alternatives to n-gram features in several classification tasks in the recent past as they result in dense representation compared to sparse n-gram features. Hence, in addition to word, character and POS n-grams, we also explored the use of embedding based feature representations for this task.

3 Data

We used a corpus of standardized assessment of English proficiency for academic purposes provided by the shared task organizers. It is a corpus of non-native speaker English essays and speech transcripts. The written corpus has a training data of 11000 essays written by learners with 11 native language backgrounds (Arabic, Chinese, French, German, Hindi, Italian, Japanese, Korean, Spanish, Telugu, Turkish). The essays are written in response to 8 prompts, and essays are evenly distributed across L1s (1000 essays per L1). The development set had 1100 essays (100 per L1) and the prompt information was provided. Exact text for the prompts was not provided in the corpus. No information was given about the proficiency

scores for the essays. We discarded two texts from the training data which had only two-three token responses (e.g., "I agree") before starting with feature extraction.

The shared task task also had a speech track, where the goal is to predict the speaker's L1 based on a transcription of a 45 second recording. There were 11000 spoken transcriptions (1000 per L1) in the training data and 1100 (100 per L1) in the development set, similar to the essay section of the corpus. The transcriptions were produced in response to 9 prompts. While the original recordings were not provided, i-vectors, which are low-dimensional representations of the speech signals were provided using the Kaldi toolkit (http://kaldi-asr.org). We did not use the i-vectors and only did preliminary n-gram based experiments on that data as well.

Test Data: Test data for both written and spoken texts had 1100 texts each (100 per L1 in each case). i-vectors were provided for the spoken files in the test data as well.

4 Features

As mentioned earlier, we explored two kinds of feature representations in this task: n-grams and embeddings.[1]

4.1 N-gram Representations

We explored N-gram representations at the level of words, characters, POS tags and mixed word-POS representations. Binary feature representation with a minimum n-gram frequency of 10 was used as a common setting for across all features. The maximum number of features was capped at 100K for most of the experiments, to limit feature explosion and over-fitting to rare n-grams. We did not find any significant differences between using binary, count and TF-IDF representations.

Word n-grams : Word n-grams are used in almost all the previous NLI approaches, and we start with them as well. We explored 1–8 lower cased n-grams with/without punctuation, with/without stemming and with/without spell check. We used the Enchant spell checker through the PyEnchant library (http://pyenchant.readthedocs.io). We considered two n-gram representations using spell-checker:

- replace the spelling error with the most likely word suggested by the checker
- replace the error with a pseudo-word

Spelling errors were used as features in earlier NLI approaches (Koppel et al., 2005; Gebre et al., 2013). But we are not aware of any previous work that pre-processed for spelling errors before n-gram extraction.

Char n-grams : We explored 2–10 character grams (lower cased), with/without crossing over word boundaries for n-gram extraction. Punctuation was not included while extracting character n-grams.

POS n-grams : We explored 1–5 POS grams. We extracted features using both NLTK tagger and Stanford POS tagger.

Word-POS mixed n-grams : (Bykh and Meurers, 2012) in the past used Open Class POS n-grams where n-grams for open class words (nouns, verbs, adjectives, and cardinal numbers) were replaced by their POS tags and the other words are left as is while calculating n-grams. Similarly, skip word n-grams have also been explored in NLI research before (Malmasi and Cahill, 2015). We extended such feature representations further by other Word-POS mixed representations such as: replacing only nouns, or only verbs with their tags, or replacing all except prepositions etc. We consider such mixed representations as a form of skip gram representations, where the gap has a name (POS tag, for example). We used NLTK tagger for feature extraction.

4.2 Embedding Representations

Embedding based representations are seen as an alternative to the sparse n-gram based representations in the recent past as they resulted in dense feature representations for text. Hence, we explored word and document level embeddings for this task, using several models. We used gensim[2] to train and classify using embedding features.

Word Embeddings : We trained the embeddings using the entire training corpus, and tuning the number of dimensions using cross validation. We tried with both CBOW (continuous bag-of-words) and skip-gram. In our experience, CBOW generated the vector representations better

[1] code for the feature extraction and classification is hosted at: https://github.com/nishkalavallabhi/NLIST2017/ for replication purposes.

[2] https://radimrehurek.com/gensim/

than skip-grams. For all the settings a minimum word count of 5 was set. The negative sampling rate, for all the settings, was left at default. We did change the negative sampling rate in the hope of obtaining better results but the results we obtained was not significantly better than the default case. For each of the settings, we explored 100 to 1000 features in an increment of 100 and a window setting of 5 to 15 in increments of 2.

Three different methods were used to get the vector representations for documents using these word embeddings:

- summed vector of all the word embeddings
- averaged vector for all the word embeddings
- a combination of average and standard deviation

For building these word embeddings, we used Word2Vec (Mikolov et al., 2013) and FastText (Joulin et al., 2016).

Document Embeddings : In addition to word embeddings, vector representations were also generated for an entire document with distributed memory (dm) and distributed bag-of-words (dbow) architectures (Le and Mikolov, 2014) using Doc2Vec tool. Number of dimensions ranged from 100 to 500 in 100 increments and the window size ranged from 5 to 50. We did not use negative sampling as it was shown in previous research that negative sampling will result in a document embedding that is biased to content words (Lau and Baldwin, 2016) whereas function words are important in the task of NLI.

For document embeddings, we used two representations:

- Doc2Vec-Full: Using the entire training data to construct an unsupervised Doc2Vec model
- Doc2Vec-PerL1: Using training data per L1 to build 11 Doc2Vec models, and use the concatenation of vectors from all 11 models per text during classification training and testing.

Additionally, we also explored the use of Efficient, Compositional, Order-sensitive n-gram Embeddings (ECO) proposed recently by Poliak et al. (2017) for constructing the document embeddings. In ECO embeddings, vector representation of neighbouring words (both occurring before and after) are averaged to obtain the numeric representation of the current word. We used the pre-trained word vectors from Wikipedia dump with dimensionality ranging from 100 to 700 as provided by Poliak et al. (2017).[3] to generate document embeddings

5 Results

We used Logistic Regression and SVMs with default parameters to train our classification models. While there are no significant differences between both the algorithms, logistic regression was much faster. So, unless otherwise stated, we report the results with logistic regression in the rest of this paper. We submitted runs for both the ESSAY track and the SPEECH track. For the SPEECH track we worked with the transcripts directly and not with the i-vectors. macro-F1 and classification accuracy were used as the evaluation measures for this task.

- Run 1: Word 1-3 grams + incl. punctuation + no stemming
- Run 2: Word 1-3 grams + POS Bi, Tri grams
- Run 3: Character n-grams (2–10), crossing word boundaries.

Table 1 shows the results on test set for our submitted systems using Logistic Regression.

System	F1 (macro)	Accuracy
Random Baseline	0.0909	0.0909
Official Baseline (Essay)	0.7104	0.7109
Official Baseline (Speech-transcriptions only)	0.5435	0.5464
Official Baseline (Speech-with ivectors)	0.7980	0.7982
Run 1-Essay	**0.8264**	**0.8264**
Run 2-Essay	0.8201	0.8200
Run 3-Essay	0.7829	0.7836
Run 1-Speech	**0.4282**	**0.4259**
Run 2-Speech	0.4036	0.4000

Table 1: Official Submissions for the ESSAY and SPEECH tracks

Word n-grams (range: 1–3) turned out to be most predictive feature representation among the ones we tried. N-grams beyond 3 did not result

[3] https://zenodo.org/record/439387

in a significant improvement in accuracy. While adding bi-, tri-grams resulted in about 9% improvement in accuracy over unigrams, adding 4-8 grams did not result in any significant performance difference on development set.

Stemming consistently resulted in a decrease in performance compared to non-stemmed features, and including punctuation always resulted in a 2-3% increase in accuracy on the development set for all settings we explored.[4] Adding POS based features to word n-grams did not result in any significant difference in the accuracy. For character n-grams, there was a 3.2% decrease in accuracy on the development set when we did not consider n-grams across word boundaries.

Spell checking: We did not find spell checking particularly useful for this task. Both our spell check feature representations did not result in any improvement in the results on the development set. It could be because we set our minimum frequency threshold to 10 and the error patterns are not frequent and consistent enough in the dataset. On the other hand, this may also imply that the people from the same native language may not always have a consistent spelling error pattern significant enough to be distinguishable from another native language group.

Using only POS n-grams did not result in an accuracy beyond 60% using both the taggers, for n = 1 to 8. Combining them with word n-grams did not result in any improvement either, as it was seen in Run 2 results in Table 1.

Mixed Word-POS representations: In terms of mixed word-POS representations, we explored the following representations using the NLTK tagger:

- Rep 1: Replace all nouns, pronouns and punctuation markers with a single string for each category.
- Rep 2: Same as the above representation, but having retaining punctuation tags for all punctuation markers
- Rep 3: Same as Rep 2, but replacing all verb tags with a single string.
- Rep 4: All words except prepositions were replaced with a common tag, and all punctuations were replaced with a common tag.
- Rep 5: OCPOS representation as described in Bykh and Meurers (2012).

For all these cases, we trained classification models with 1–8 n-grams, minimum frequency of 10, and up to 300K features. While some of these mixed word-POS representations were not explored for this task before, none of the models give an accuracy beyond 75% on the development set. It has to be noted that we used only Logistic Regression and SVM for classification. But, it is unlikely that another classification algorithm would result in a dramatic increase with these feature representations. We did not explore ensemble models where different feature representations are combined as multiple models instead of a large single model.

In addition to training classifiers, we also briefly explored using distance measures from stylistics and authorship attribution research such as Burrow's Delta (Burrows, 2002) and other related measures (Evert et al., 2015) using 100-1000 most frequent word, character and POS n-grams in the corpus. We did not find them particularly useful for this task, with highest accuracies of less than 60% on the development set. This could be due to the fact that Delta based measures are usually used on much longer texts, typically full length texts or novels.[5]

Speech Data: As mentioned earlier, for speech transcripts, we did not use the i-vectors and only used the above mentioned n-gram features. They were not as useful predictors for speech as they were for essays. One possible reason could be the fact that we have much smaller texts compared to written texts. However, i-vectors, which capture the acoustic features, clearly play an important role in NLI for speech data, as it was seen from the improvement over baseline they achieved on development set, as it was indicated in the documentation for corpus release.

5.1 With Embeddings on Development Set

In addition to the submitted runs, we explored word and document embedding based feature rep-

[4] Since the classification accuracy was very sensitive to decisions such as stemming and punctuation, and to how the features are extracted, we are sharing our final list of word trigram features for both essay and speech tracks extracted using LightSide (Mayfield and Rosé, 2013) on github for replication purposes.

[5] We used Stylo (Eder et al., 2016) and JGAAP (https://github.com/evllabs/JGAAP) libraries for calculating Delta scores

resentations that were described in Section 4 for this task. Our experiments with these representations did not result in better results than word and character n-grams. Table 2 shows a summary of the most predictive results with embedding features in our experiments.

System	F1 (macro)	Accuracy
Random Baseline	0.0909	0.0909
Word2vec (dim:200,window:11)	0.6311	0.6312
Word2vec (Nouns and Numbers sub.) (dim:200,window:11)	0.6311	0.6312
ECO	0.5744	0.5742
Doc2Vec-full (dim:100,window:10)	0.5440	0.5463
Doc2Vec-full (dim:500,window:25)	0.6276	0.6291
Doc2Vec-byL1 (dim:100,window:10)	0.6169	0.6190
Doc2Vec-byL1 (dim:500,window:25)	**0.7119**	**0.7127**

Table 2: Results for the ESSAY track with Embedding Features on Development data

For word embeddings, we achieved a macro F1 of 0.63 with Word2Vec (number of features 200 and window size 11), using SVM. We experimented with various levels of negative sampling but we could not attain any improvement. What is more interesting to note is that the system performance remains the same even when nouns and numbers are substituted. We repeated our experiments by averaging the word vectors with their corresponding TF-IDF values but we did not any improvement of performance. Training the embeddings on spell corrected data did not produce better results.

For larger number of features we noticed that the system performed better on the training set than it did on the development set clearly hinting at over-fitting. We performed 5 fold cross-validation, with multiple parameter settings and using linear, rbf and polynomial kernels, in a bid to find optimum parameter settings which would lead to the best classifier. Linear kernel emerged out as the winner for the optimum parameter settings for Word2Vec.

We got a macro F1 of 0.57 with ECO embeddings (number of features 700 and window size 4) using SVM. The reason for a poorer performance of ECO embeddings compared to Word2Vec could be the training corpus. ECO embeddings were trained on Wikipedia dump and not the training corpus as was the case for Word2Vec embeddings. Training embeddings on the shared task's training corpus could have possibly captured the specific features of the corpus instead of more general language features from Wikipedia corpus.

FastText performed much worse than Word2Vec and ECO, and was even below baseline with some of the parameter settings. We found that the performance of the system did not change appreciably when the number of features was increased, indicating that a large number of features may not be essential or desirable to capture all the stylistic differences in the corpus.

With Doc2Vec, concatenating the vectors from L1 specific doc2vec models performed much better than training a single Doc2Vec model on the entire dataset, giving a macro F1 of 0.7119 (500 features per L1, window size 25, dbow representation) using Logistic Regression. Doc2Vec-byL1 was consistently better than Doc2Vec-full in all the parameter settings we explored, always resulting in over 7% increase in accuracy.

Number of features and window size seemed to have a good influence on the classification performance and window sizes below 10 resulted in low performance for L1 classification. It was also shown in a previous empirical evaluation that dbow favors larger window sizes (Lau and Baldwin, 2016), although the longest they had was 15. Overall, from what we observed so far, training L1 specific Doc2Vec models may result in better performance for this task. Finding a better way to combine L1 specific features instead of just concatenating everything may boost the performance further.

5.2 Prompt based classification

Our results so far seem to show that embedding based representations are not particularly useful for this task. We hypothesized that this could be due to the fact that most of what embeddings capture is semantic similarity, while NLI involves capturing stylistic choices such as use of function words, punctuation markers etc, along with content word choices. To test this hypothesis, we did prompt based classification instead of L1 classifi-

cation.

Doc2Vec-Full models for prompt based classification achieved accuracy of over 95% on the development for smaller feature/dimension sizes (10–20) and window sizes (5–10) using logistic regression. A dimensionality of 5 already gave an accuracy of 73% on the development set for prompt classification (8 prompts in essays corpus). This clearly indicates that the embeddings were able to capture topical differences between prompts easily even in a low dimensional space.

From a comparison of Doc2vec experiments for L1 and prompt classification, we can conclude that embeddings are more suitable when the categories have more semantic and less stylistic differences. However, an interesting observation from L1 classification using Doc2Vec was the influence of window size on classification performance. Performance steadily improved with both larger dimensions and larger window sizes. Whether this captures something unique about stylistic variation is something that should be more systematically explored in future.

6 Discussion

We described some of our experiments that study the usefulness of n-gram and embedding based feature representations for Native Language Identification as a part of the NLI Shared Task 2017. Our main conclusions so far are:

- Word uni–trigram features performed the best as a single group for classifying written texts, and there is no significant improvement in terms of adding infrequent trigrams or adding n-grams beyond 3.
- Character n-grams (n=2–10) were the second best performing feature group for written texts.
- Results with word and character n-grams could not be replicated with speech transcripts.
- Word and document embedding features did not give better results than n-grams, possibly because they capture semantic similarities instead of stylistic aspects.

6.1 Outlook

While modeling innovations may result in performance improvement, they make predictions more and more opaque. For NLI to be useful in applications such as language instruction or in language generation (e.g. generating texts with individual writing style in applications such as machine translation) we may need interpretable models. More qualitative analysis and eventually more concrete stylistic features for specific L1 backgrounds need to be developed. With this goal, and inspired by previous work on learning stylistic variation for language generation (Lin, 2012) and learning to segment phrasal features (instead of words) for sentiment analysis (Tang et al., 2014), we plan to focus on working towards better feature representations that may result in generalizable insights into the nature of L1 influence on L2 writing.

References

Daniel Blanchard, Joel Tetreault, Derrick Higgins, Aoife Cahill, and Martin Chodorow. 2013. Toefl11: A corpus of non-native english. Technical report, Educational Testing Service.

Julian Brooke and Graeme Hirst. 2012. Robust, Lexicalized Native Language Identification. In *Proceedings of COLING 2012*. The COLING 2012 Organizing Committee, Mumbai, India, pages 391–408. http://www.aclweb.org/anthology/C12-1025.

John Burrows. 2002. delta: a measure of stylistic difference and a guide to likely authorship. *Literary and linguistic computing* 17(3):267–287.

Serhiy Bykh and Detmar Meurers. 2012. Native Language Identification using Recurring n-grams – Investigating Abstraction and Domain Dependence. In *Proceedings of COLING 2012*. The COLING 2012 Organizing Committee, Mumbai, India, pages 425–440. http://www.aclweb.org/anthology/C12-1027.

Serhiy Bykh and Detmar Meurers. 2014. Exploring Syntactic Features for Native Language Identification: A Variationist Perspective on Feature Encoding and Ensemble Optimization. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*. Dublin, Ireland, pages 1962–1973.

Serhiy Bykh and Detmar Meurers. 2016. Advancing linguistic features and insights by label-informed feature grouping: An exploration in the context of native language identification. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*. The COLING 2016 Organizing Committee, Osaka, Japan, pages 739–749.

Serhiy Bykh, Sowmya Vajjala, Julia Krivanek, and Detmar Meurers. 2013. Combining shallow and

linguistically motivated features in native language identification. *the 8th NAACL Workshop on Innovative Use of NLP for Building Educational Applications (BEA8)* pages 197–206.

Maciej Eder, Jan Rybicki, and Mike Kestemont. 2016. Stylometry with r: a package for computational text analysis. *R Journal* 8(1):107–121. http://journal.r-project.org/archive/2016-1/eder-rybicki-kestemont.pdf.

Dominique Estival, Tanja Gaustad, Son Bao Pham, Will Radford, and Ben Hutchinson. 2007. Author profiling for English emails. In *Proceedings of the 10th Conference of the Pacific Association for Computational Linguistics*. Melbourne, Australia, pages 263–272.

Stefan Evert, Thomas Proisl, Thorsten Vitt, Christof Schöch, Fotis Jannidis, and Steffen Pielström. 2015. Towards a better understanding of burrows's delta in literary authorship attribution. In *Proceedings of the Fourth Workshop on Computational Linguistics for Literature*. Association for Computational Linguistics, Denver, Colorado, USA, pages 79–88. http://www.aclweb.org/anthology/W15-0709.

Binyam Gebrekidan Gebre, Marcos Zampieri, Peter Wittenburg, and Tom Heskes. 2013. Improving native language identification with tf-idf weighting. In *the 8th NAACL Workshop on Innovative Use of NLP for Building Educational Applications (BEA8)*. pages 216–223.

Jeroen Geertzen, Theodora Alexopoulou, and Anna Korhonen. 2013. Automatic linguistic annotation of large scale l2 databases: The ef-cambridge open language database (efcamdat). In *Proceedings of the 31st Second Language Research Forum. Somerville, MA: Cascadilla Proceedings Project*.

Radu Tudor Ionescu, Marius Popescu, and Aoife Cahill. 2014. Can characters reveal your native language? A language-independent approach to native language identification. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Association for Computational Linguistics.

Scott Jarvis, Yves Bestgen, and Steve Pepper. 2013. Maximizing Classification Accuracy in Native Language Identification. In *Proceedings of the Eighth Workshop on Innovative Use of NLP for Building Educational Applications*. Association for Computational Linguistics, Atlanta, Georgia, pages 111–118.

Scott Jarvis, Gabriela Castaneda-Jiménez, and Rasmus Nielsen. 2004. Investigating l1 lexical transfer through learners wordprints. In *Second Language Research Forum (SLRF). State College, PA*.

Scott Jarvis and Scott Crossley, editors. 2012. *Approaching Language Transfer Through Text Classification: Explorations in the Detection-based Approach*, volume 64. Multilingual Matters Limited, Bristol, UK.

Armand Joulin, Edouard Grave, Piotr Bojanowski, and Tomas Mikolov. 2016. Bag of tricks for efficient text classification. *arXiv preprint arXiv:1607.01759* .

Moshe Koppel, Jonathan Schler, and Kfir Zigdon. 2005. Determining an author's native language by mining a text for errors. In *Proceedings of the eleventh ACM SIGKDD international conference on Knowledge discovery in data mining*. ACM, Chicago, IL, pages 624–628.

Jey Han Lau and Timothy Baldwin. 2016. An empirical evaluation of doc2vec with practical insights into document embedding generation. In *Proceedings of the 1st Workshop on Representation Learning for NLP*. Association for Computational Linguistics, Berlin, Germany, pages 78–86. http://anthology.aclweb.org/W16-1609.

Quoc Le and Tomas Mikolov. 2014. Distributed representations of sentences and documents. In *Proceedings of the 31st International Conference on Machine Learning (ICML-14)*. pages 1188–1196.

Jing Lin. 2012. *Using a rewriting system to model individual writing styles*. Ph.D. thesis, University of Aberdeen.

Shervin Malmasi. 2016. *Native Language Identification: Explorations and Applications*. Ph.D. thesis, Macquarie University. http://hdl.handle.net/1959.14/1110919.

Shervin Malmasi and Aoife Cahill. 2015. Measuring feature diversity in native language identification. In *Proceedings of the Tenth Workshop on Innovative Use of NLP for Building Educational Applications*. Association for Computational Linguistics, Denver, Colorado, pages 49–55. http://www.aclweb.org/anthology/W15-0606.

Shervin Malmasi and Mark Dras. 2015. Multilingual native language identification. *Natural Language Engineering* pages 1–53.

Shervin Malmasi and Mark Dras. 2017. Native language identification using stacked generalization. *CoRR* abs/1703.06541. http://arxiv.org/abs/1703.06541.

Shervin Malmasi, Keelan Evanini, Aoife Cahill, Joel Tetreault, Robert Pugh, Christopher Hamill, Diane Napolitano, and Yao Qian. 2017. A Report on the 2017 Native Language Identification Shared Task. In *Proceedings of the 12th Workshop on Building Educational Applications Using NLP*. Association for Computational Linguistics, Copenhagen, Denmark.

Shervin Malmasi, Marcos Zampieri, Nikola Ljubešić, Preslav Nakov, Ahmed Ali, and Jörg Tiedemann. 2016. Discriminating between Similar Languages and Arabic Dialect Identification: A Report on the Third DSL Shared Task. In *Proceedings of the VarDial Workshop*. Osaka, Japan.

Elijah Mayfield and Carolyn Penstein Rosé. 2013. 8 lightside. *Handbook of Automated Essay Evaluation: Current Applications and New Directions* page 124.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*. pages 3111–3119.

Sergiu Nisioi. 2015. *Feature Analysis for Native Language Identification*, Springer International Publishing, Cham, pages 644–657. https://doi.org/10.1007/978-3-319-18111-0$_4$9.

Adam Poliak, Pushpendre Rastogi, M. Patrick Martin, and Benjamin Van Durme. 2017. Efficient, compositional, order-sensitive n-gram embeddings. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*. Association for Computational Linguistics, Valencia, Spain, pages 503–508. http://www.aclweb.org/anthology/E17-2081.

Bjrn Schuller, Stefan Steidl, Anton Batliner, Julia Hirschberg, Judee K. Burgoon, Alice Baird, Aaron Elkins, Yue Zhang, Eduardo Coutinho, and Keelan Evanini. 2016. The INTERSPEECH 2016 Computational Paralinguistics Challenge: Deception, Sincerity & Native Language. In *Interspeech 2016*. pages 2001–2005. https://doi.org/10.21437/Interspeech.2016-129.

Duyu Tang, Furu Wei, Bing Qin, Li Dong, Ting Liu, and Ming Zhou. 2014. A joint segmentation and classification framework for sentiment analysis. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Association for Computational Linguistics, Doha, Qatar, pages 477–487. http://www.aclweb.org/anthology/D14-1054.

Joel Tetreault, Daniel Blanchard, and Aoife Cahill. 2013. A Report on the First Native Language Identification Shared Task. In *Proceedings of the Eighth Workshop on Building Educational Applications Using NLP*. Association for Computational Linguistics, Atlanta, GA, USA.

Joel Tetreault, Daniel Blanchard, Aoife Cahill, and Martin Chodorow. 2012. Native tongues, lost and found: Resources and empirical evaluations in native language identification. In *Proceedings of COLING 2012*. The COLING 2012 Organizing Committee, Mumbai, India, pages 2585–2602. http://www.aclweb.org/anthology/C12-1158.

Laura Mayfield Tomokiyo and Rosie Jones. 2001. You're not from'round here, are you?: naive bayes detection of non-native utterance text. In *Proceedings of the second meeting of the North American Chapter of the Association for Computational Linguistics on Language technologies*. Association for Computational Linguistics, pages 1–8.

Sze-Meng Jojo Wong and Mark Dras. 2011. Exploiting Parse Structures for Native Language Identification. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, Edinburgh, Scotland, UK., pages 1600–1610. http://www.aclweb.org/anthology/D11-1148.