Improving Native Language Identification with TF-IDF Weighting

Binyam Gebrekidan Gebre[1], Marcos Zampieri[2], Peter Wittenburg[1], Tom Heskes[3]
[1]Max Planck Institute for Psycholinguistics
[2]University of Cologne
[3]Radboud University
bingeb@mpi.nl,mzampier@uni-koeln.de,
peter.wittenburg@mpi.nl,t.heskes@science.ru.nl

Abstract

This paper presents a Native Language Identification (NLI) system based on TF-IDF weighting schemes and using linear classifiers - support vector machines, logistic regressions and perceptrons. The system was one of the participants of the 2013 NLI Shared Task in the closed-training track, achieving 0.814 overall accuracy for a set of 11 native languages. This accuracy was only 2.2 percentage points lower than the winner's performance. Furthermore, with subsequent evaluations using 10-fold cross-validation (as given by the organizers) on the combined training and development data, the best average accuracy obtained is 0.8455 and the features that contributed to this accuracy are the TF-IDF of the combined unigrams and bigrams of words.

1 Introduction

Native Language Identification (NLI) is the task of automatically identifying the native language of a writer based on the writer's foreign language production. The task is modeled as a classification task in which automatic methods have to assign class labels (native languages) to objects (texts). NLI is by no means trivial and it is based on the assumption that the mother tongue influences Second Language Acquisition (SLA) and production (Lado, 1957).

When an English native speaker hears someone speaking English, it is not difficult for him/her to identify if this person is a native speaker or not. Moreover, it is, to some extent, possible to assert the mother tongue of non-native speakers by his/hers pronunciation patterns, regardless of their language proficiency. In NLI, the same principle that seems intuitive for spoken language, is applied to text. If it is true that the mother tongue of an individual influences speech production, it should be possible to identify these traits in written language as well.

NLI methods are particularly relevant for languages with a significant number of foreign speakers, most notably, English. It is estimated that the number of non-native speakers of English outnumbers the number of native speakers by two to one (Lewis et al., 2013). The written production of non-native speakers is abundant on the Internet, academia, and other contexts where English is used as *lingua franca*.

This study presents the system that participated in the 2013 NLI Shared Task (Tetreault et al., 2013) under the name *Cologne-Nijmegen*. The novel aspect of the system is the use of TF-IDF weighting schemes. For this study, we experimented with a number of algorithms and features. Linear SVM and logistic regression achieved the best accuracies on the combined features of unigrams and bigrams of words. The rest of the paper will explain in detail the features, methods and results achieved.

2 Motivation

There are two main reasons to study NLI. On one hand, there is a strong linguistic motivation, particularly in the field of SLA and on the other hand, there is the practical relevance of the task and its integration to a number of computational applications.

The linguistic motivation of NLI is the possibility of using classification methods to study the inter-

Proceedings of the Eighth Workshop on Innovative Use of NLP for Building Educational Applications, pages 216–223,
Atlanta, Georgia, June 13 2013.

play between native and foreign language acquisition and performance (Wong and Dras, 2009). One of the SLA theories that investigate these phenomena is contrastive analysis, which is used to explain why some structures of L2 are more difficult to acquire than others (Lado, 1957).

Contrastive analysis postulates that the difficulty in mastering L2 depends on the differences between L1 and L2. In the process of acquiring L2, *language transfer* (also known as L1 interference) occurs and speakers apply knowledge from their native language to a second language, taking advantage of their similarities. Computational methods applied to L2 written production can function as a corpus-driven method to level out these differences and serve as a source of information for SLA researchers. It can also be used to provide more targeted feedback to language learners about their errors.

NLI is also a relevant task in computational linguistics and researchers have turned their attention to it in the last few years. The task is often regarded as a part of a broader task of authorship profiling, which consists of the application of automatic methods to assert information about the writer of a given text, such as age, gender as well native language. Authorship profiling is particularly useful for forensic linguistics.

Automatic methods of NLI may be integrated in NLP applications such as spam detection or machine translation. NLP tasks such as POS tagging and parsing might also benefit from NLI, as these resources are trained on standard language written by native speakers. These tools can be more accurate to tag non-native speaker's text if trained with L2 corpora.

3 Related Work

In the last years, a couple of attempts at identifying native language have been described in the literature. Tomokiyo and Jones (2001) uses a Naive Bayes algorithm to classify transcribed data from three native languages: Chinese, Japanese and English. The algorithm reached 96% accuracy when distinguishing native from non-native texts and 100% when distinguishing English native speakers from Chinese native speakers.

Koppel et al. (2005) used machine learning to identify the native languages of non-native English speakers with five different mother tongues (Bulgarian, Czech, French, Russian, and Spanish), using data retrieved from the International Corpus of Learner English (ICLE) (Granger et al., 2009). The features used in this study were function words, character n-grams, and part-of-speech (POS) bigrams.

Tsur and Rappoport (2007) investigated the influence of the phonology of a writer's mother tongue through native language syllables modelled by character bigrams. Estival et al. (2007) addressed NLI as part of authorship profiling. Authors aim to attribute 10 different characteristics of writers by analysing a set of English e-mails. The study reports around 84% accuracy in distinguishing e-mails written by English Arabic and Spanish L1 speakers.

SVM, the algorithm that achieved the best results in our experiments, was also previously used in NLI (Kochmar, 2011). In this study, the author identified error types that are typical for speakers of different native languages. She compiled a set of features based on these error types to improve the classification's performance.

Recently, the TOEFL11 corpus was compiled to serve as an alternative to the ICLE corpus (Tetreault et al., 2012). Authors argue that TOEFL11 is more suitable to NLI than ICLE. This study also experimented with different features to increase results in NLI and reports best accuracy results of 90.1% on ICLE and 80.9% on TOEFL11.

4 Methods

We approach the task of native language identification as a kind of text classification. In text classification, decisions and choices have to be made at three levels. First, how do we use the training and development data? Second, what features do we extract and how do we select the most informative ones? Third, which machine learning algorithms perform best and which parameters can we tune under the constraints of memory and time? In the following subsections, we answer these questions.

4.1 Dataset: TOEFL11

The dataset used for the shared task is called TOEFL11 (Blanchard et al., 2013). It consists of 12,100 English essays (about 300 to 400 words long) from the Test of English as a Foreign Language (TOEFL). The essays are written by 11 native language speakers (L1). Table 1 shows the 11 native languages. Each essay is labelled with an English language proficiency level (high, medium, or low) based on the judgments of human assessment specialists. We used 9,900 essays for training data and 1,100 for development (parameter tuning). The shared task organizers kept 1,100 essays for testing.

Table 1: TOEFL11

L1 languages	Arabic, Chinese, French, German, Hindi, Italian, Japanese, Korean, Spanish, Telugu, Turkish
# of essays per L1	900 for training 100 for validating 100 for testing

4.2 Features

We explored different kinds and combinations of features that we assumed to be different for different L1 speakers and that are also commonly used in the NLI literature (Koppel et al., 2005; Tetreault et al., 2012). Table 2 shows the sources of the features we considered. Unigrams and bigrams of words are explored separately and in combination. One through four grams of part of speech tags have also been explored. For POS tagging of the essays, we applied the default POS tagger from NLTK (Bird, 2006).

Spelling errors have also been treated as features. We used the collection of words in Peter Norvig's website[1] as a reference dictionary. The collection consists of about a million words. It is a concatenation of several public domain books from Project Gutenberg and lists of most frequent words from Wiktionary and the British National Corpus.

Character n-grams have also been explored for both the words in the essays and for words with spelling errors. The maximum n-gram size considered is six.

All features, consisting of either characters or words or part-of-speech tags or their combinations, are mapped into normalized numbers (norm L2). For the mapping, we use TF-IDF, a weighting technique popular in information retrieval but which is also finding its use in text classification. Features that occurred in less than 5 of the essays or those that occurred in more than 50% of the essays are removed (all characters are in lower case). These cut-off values are experimentally selected.

Table 2: A summary of features used in our experiments

Word n-grams	Unigrams and bigrams of words present in the essays.
POS n-grams	One up to four grams of POS tags present in the essays; tagging is done using default NLTK tagger (Bird, 2006).
Character n-grams	One up to six grams of characters in each essay.
Spelling errors	All words that are not found in the dictionary of Peter Norvig's spelling corrector.

4.2.1 Term Frequency (TF)

Term Frequency refers to the number of times a particular term appears in an essay. In our experiments, terms are n-grams of characters, words, part-of-speech tags or any combination of them. The intuition is that a term that occurs more frequently identifies/specifies the essay better than another term that occurs less frequently. This seems a useful heuristic but what is the relationship between the frequency of a term and its importance to the essay? From among many relationships, we selected a logarithmic relationship (sublinear TF scaling) (Manning et al., 2008):

$$wf_{t,e} = \begin{cases} 1 + \log(tf_{t,e}) & \text{if } tf_{t,e} > 0 \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

[1] http://norvig.com/spell-correct.html

where $wf_{t,e}$ refers to weight and $tf_{t,e}$ refers to the frequency of term t in essay e.

The $wf_{t,e}$ weight tells us the importance of a term in an essay based on its frequency. But not all terms that occur more frequently in an essay are equally important. The effective importance of a term also depends on how infrequent the term is in other essays and this intuition is handled by Inverse Document Frequency(IDF).

4.2.2 Inverse Document Frequency(IDF)

Inverse Document Frequency (IDF) quantifies the intuition that a term which occurs in many essays is not a good discriminator, and should be given less weight than one which occurs in fewer essays. In mathematical terms, IDF is the log of the inverse probability of a term being found in any essay (Salton and McGill, 1984):

$$idf(t_i) = \log \frac{N}{n_i}, \tag{2}$$

where N is the number of essays in the corpus, and term t_i occurs in n_i of them. IDF gives a new weight when combined with TF to form TF-IDF.

4.2.3 TF–IDF

TF–IDF combines the weights of TF and IDF by multiplying them. TF gives more weight to a frequent term in an essay and IDF downscales the weight if the term occurs in many essays. Equation 3 shows the final weight that each term of an essay gets before normalization.

$$w_{i,e} = (1 + \log(tf_{t,e})) \times \log(N/n_i) \tag{3}$$

Essay lengths are usually different and this has an impact on term weights. To abstract from different essay lengths, each essay feature vector is normalized to unit length. After normalization, the resulting essay feature vectors are fed into classifiers.

4.3 Classifiers

We experimented with three linear classifiers - linear support vector machines, logistic regression and perceptrons - all from scikit-learn (Pedregosa et al., 2011). These algorithms are suitable for high dimensional and sparse data (text data is high dimensional and sparse). In the following paragraphs, we briefly describe the algorithms and the parameter values we selected.

SVMs have been explored systematically for text categorization (Joachims, 1998). An SVM classifier finds a hyperplane that separates examples into two classes with maximal margin (Cortes and Vapnik, 1995) (Multi-classes are handled by multi one-versus-rest classifiers). Examples that are not linearly separable in the feature space are mapped to a higher dimension using kernels. In our experiments, we used a linear kernel and a penalty parameter of value 1.0.

In its various forms, logistic regression is also used for text classification (Zhang et al., 2003; Genkin et al., 2007; Yu et al., 2011) and native language identification (Tetreault et al., 2012). Logistic regression classifies data by using a decision boundary, determined by a linear function of the features. For the implementation of the algorithm, we used the LIBLINEAR open source library (Fan et al., 2008) from scikit-learn (Pedregosa et al., 2011) and we fixed the regularization parameter to 100.0.

For baseline, we used a perceptron classifier (Rosenblatt, 1957). Perceptron (or single layer network) is the simplest form of neural network. It is designed for linear separation of data and works well for text classification. The number of iterations of the training algorithm is fixed to 70 and the rest of parameters are left with their default values.

5 Results and Discussion

For each classifier, we ran ten-fold cross-validation experiments. We divided the training and development data into ten folds using the same fold splitting ids as requested by the shared task organizers and also as used in (Tetreault et al., 2012). Nine of the folds were used for training and the tenth for testing the trained model. This was repeated ten times with each fold being held out for testing. The performance of the classifiers on different features are presented in terms of average accuracy.

Table 3 gives the average accuracies based on the TF-IDF of word and character n-grams. Linear SVM gives the highest accuracy of 84.55% using features extracted from unigrams and bigrams of words. Logistic regression also gives comparable accuracy of 84.45% on the same features.

Table 3: Cross-validation results; accuracy in %

N-gram	Linear SVM	Logistic Regression	Perceptron
Words			
1	74.73	74.18	65.45
2	80.91	80.27	75.45
1 and 2	**84.55**	84.45	78.82
(1 and 2)*	83.36	83.27	78.73
* minus country and language names			
Characters			
1	18.45	19.27	9.09
2	43.27	40.82	10.36
3	71.36	68.00	36.91
4	80.36	79.91	59.64
5	83.09	82.64	73.91
6	**84.09**	84.00	76.45

The size of the feature vector of unigrams and bigrams of words is 73,626[2]. For each essay, only a few of the features have non-zero values. Which features are active and most discriminating in the classifiers? Table 4 shows the ten most informative features for the 10th run in the cross-validation (as picked up linear SVM).

Table 4: Ten most informative features for each L1

ARA	many reasons / from / self / advertisment / , and / statment / any / thier / alot of / alot
CHI	in china / hold / china / time on / may / taiwan / just / still / , the / . take
FRE	french / conclude , / even if / in france / france / to conclude / indeed , / ... / . indeed / indeed
GER	special / furthermore / might / germany / , because / have to / . but / - / often / , that
HIN	which / and concept / various / hence / generation / & / towards / then / its / as compared
ITA	in italy / , for / infact / that a / italy / i think / in fact / italian / think that / :
JPN	, and / i disagree / is because / . it / . if / i think / japan , / japanese / in japan / japan
KOR	. however / however , / even though / however / these days / various / korea , / korean / in korea / korea
SPA	an specific / because is / moment / , etc / going to / , is / necesary / , and / diferent / , but
TEL	may not / the statement / every one / days / the above / where as / with out / when compared / i conclude / and also
TUR	ages / istanbul / addition to / conditions / enough / in turkey / the life / ; / . because / turkey

The ten most informative features include country and language names. For example, for Japanese and Korean L1s, four of the ten top features include Korea or Korean in the unigrams or bigrams. How would the classification accuracy decrease if we removed mentions of country or language names?

We made a list of the 11 L1 language names and the countries where they are mainly spoken (for example, German, Germany, French, France, etc.). We considered this list as stop words (i.e. removed them from corpus) and ran the whole classification experiments. The new best accuracy is 83.36% (a loss of just 1.2%). Table 3 shows the new accuracies for all classifiers. The new top ten features mostly consist of function words and some spelling errors. Table 5 shows all of the new top ten features.

The spelling errors seem to have been influenced by the L1 languages, especially for French and Spanish languages. The English words *example* and *developed* have similar sounding/looking equivalents in French (*exemple* and *développé*) . Similarly, the English words *necessary* and *different* have similar sounding/looking words in Spanish (*necesario* and *diferente*). These spelling errors made it to the top ten features. But how discriminating are they on their own?

Table 5: Ten most informative features (minus country and language names) for each L1

ARA	many reasons / from / self / advertisment / , and / statment / any / thier / alot of / alot
CHI	and more / hold / more and / time on / taiwan / may / just / still / . take / , the
FRE	conclude / exemple / developped / conclude , / even if / to conclude / indeed , / ... / . indeed / indeed
GER	has to / special / furthermore / might / , because / have to / . but / - / often / , that
HIN	and concept / which / various / hence / generation / & / towards / then / its / as compared
ITA	possibility / probably / particular / , for / infact / that a / i think / in fact / think that / &
JPN	i agree / the opinion / tokyo / two reasons / is because / , and / i disagree / . it / . if / i think
KOR	creative / , many / 's / . also / . however / even though / however , / various / however / these days
SPA	activities / an specific / moment / , etc / going to / , is / necesary / , and / diferent / , but
TEL	may not / the statement / every one / days / the above / where as / when compared / with out / i conclude / and also
TUR	enjoyable / being / ages / addition to / istanbul / enough / conditions / the life / ; / . because

We ran experiments with features extracted from

[2]features that occur less than 5 times or that occur in more than 50% of the essays are removed from the vocabulary

Table 6: Confusion matrix: Best accuracy is for German (95%) and the worst is for Hindi (72%)

	ARA	CHI	FRE	GER	HIN	ITA	JPN	KOR	SPA	TEL	TUR
ARA	**83**	1	4	1	1	3	1	2	3	1	0
CHI	0	**88**	2	0	2	0	2	5	1	0	0
FRE	3	0	**88**	2	1	2	0	1	2	0	1
GER	2	0	1	**95**	0	0	0	0	1	0	1
HIN	2	1	1	1	**72**	0	0	0	2	***18***	3
ITA	0	0	6	3	0	**84**	0	0	6	0	1
JPN	1	2	0	1	1	0	**84**	***10***	0	0	1
KOR	0	3	0	2	3	0	***8***	**81**	1	1	1
SPA	6	2	5	2	0	4	0	0	**79**	0	2
TEL	0	0	0	0	***16***	0	1	0	0	**83**	0
TUR	1	1	0	1	3	0	0	0	1	0	**93**

only spelling errors. For comparison, we also ran experiments with POS tags with and without their words. None of these experiments beat the best accuracy obtained using unigram and bigram of words - not even the unigram and bigram of POS tagged words. See table 7 for the obtained results.

Table 7: Cross-validation results; accuracy in %

N-gram	Linear SVM	Logistic Regression	Perceptron
POS			
1	17.00	17.09	9.09
2	43.45	40.00	11.18
3	55.27	53.55	35.36
4	56.09	56.18	48.64
POS + Word			
1	75.09	74.18	64.09
2	80.45	80.64	76.18
1 and 2	83.00	83.36	79.09
Spelling errors - characters			
1	20.36	21.00	9.09
2	34.09	32.64	9.73
3	47.00	44.64	26.82
4	50.82	48.09	41.64
1–4	51.82	48.27	34.18
words	42.73	39.45	28.73

All our reported results so far have been global classification results. Table 6 shows the confusion matrix for each L1. The best accuracy is 95% for German and the worst is for Hindi (72%). Hindi is classified as Telugu (18%) of the times and Telugu is classified as Hindi 16% of the times and only one Telugu essay is classified as any other than Hindi. More generally, the confusion matrix seems to suggest that geographically closer countries are more confused with each other: Hindi and Telugu, Japanese and Korean, Chinese and Korean.

The best accuracy (84.55%) obtained in our experiments is higher than the state-of-the-art accuracy reported in (Tetreault et al., 2012) (80.9%). But the features we used are not different from those commonly used in the literature (Koppel et al., 2005; Tetreault et al., 2012) (n-grams of characters or words). The novel aspect of our system is the use of TF-IDF weighting on all of the features including on unigrams and bigrams of words.

TF-IDF weighting has already been used in native language identification (Kochmar, 2011; Ahn, 2011). But its importance has not been fully explored. Experiments in Kochmar (2011) were limited to character grams and in a binary classification scenario. Experiments in Ahn (2011) applied TF-IDF weighting to identify content words and showed how their removal decreased performance (Ahn, 2011). By contrast, in this paper, we applied TF-IDF weighting consistently to all features - same type features (e.g. unigrams) or combined features (e.g. unigram and bigrams).

How would the best accuracy change if TF-IDF weighting is not applied? Table 8 shows the changes to the best average accuracies with and without TF/IDF weighting for the three classifiers.

Table 8: The importance of TF-IDF weighting

TF	IDF	SVM	LR	Perceptron
Yes	Yes	84.55	84.45	78.82
Yes	No	80.82	80.73	63.18
No	Yes	82.36	82.27	78.82
No	No	79.18	78.55	56.36

6 Conclusions

This paper has presented the system that participated in the 2013 NLI Shared Task in the closed-training track. Cross-validation testing on the TOEFL11 corpus showed that the system could achieve an accuracy of about 84.55% in categorizing unseen essays into one of the eleven L1 languages.

The novel aspect of the system is the use of TF-IDF weighting schemes on features – which could be any or combination of n-gram words/characters/POS tags. The feature combination that gave the best accuracy is the TF-IDF of unigrams and bigrams of words. The next best feature class is the TF-IDF of 6-gram characters , which achieved 84.09%, very close to 84.55%. Both linear support vector machines and logistic regression classifiers have performed almost equally.

To improve performance in NLI, future work should examine new features that can classify geographically or typologically related languages such as Hindi and Telugu. Future work should also analyze the information obtained in NLI experiments to quantify and investigate differences in the usage of foreign language lexicon or grammar according to the individual's mother tongue.

Acknowledgments

The research leading to these results has received funding from the European Commissions 7th Framework Program under grant agreement no 238405 (CLARA). The authors would like to thank the organizers of the NLI Shared Task 2013 for providing prompt reply to all our inquiries and for coordinating a very interesting and fruitful shared task.

References

Charles S. Ahn. 2011. *Automatically detecting authors' native language*. Ph.D. thesis, Monterey, California. Naval Postgraduate School.

Steven Bird. 2006. NLTK: the natural language toolkit. In *Proceedings of the COLING/ACL on Interactive presentation sessions*, pages 69–72. Association for Computational Linguistics.

Daniel Blanchard, Joel Tetreault, Derrick Higgins, Aoife Cahill, and Martin Chodorow. 2013. TOEFL11: A Corpus of Non-Native English. Technical report, Educational Testing Service.

Corinna Cortes and Vladimir Vapnik. 1995. Support-vector networks. *Machine learning*, 20(3):273–297.

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 (PACLING)*, pages 263–272.

Rong-En Fan, Kai-Wei Chang, Cho-Jui Hsieh, Xiang-Rui Wang, and Chih-Jen Lin. 2008. Liblinear: A library for large linear classification. *The Journal of Machine Learning Research*, 9:1871–1874.

Alexander Genkin, David D Lewis, and David Madigan. 2007. Large-scale bayesian logistic regression for text categorization. *Technometrics*, 49(3):291–304.

Sylviane Granger, Estelle Dagneaux, and Fanny Meunier. 2009. *International Corpus of Learner English*. Presses Universitaires de Louvain, Louvain-la-Neuve.

Thorsten Joachims. 1998. *Text categorization with support vector machines: Learning with many relevant features*. Springer.

Ekaterina Kochmar. 2011. Identification of a writer's native language by error analysis. Master's thesis, University of Cambridge, United Kingdom.

Moshe Koppel, Jonathan Schler, and Kfir Zigon. 2005. Automatically determining an anonymous author's native language. *Lecture Notes in Computer Science*, 3495:209—217.

Robert Lado. 1957. *Applied Linguistics for Language Teachers.* University of Michigan Press.

Paul Lewis, Gary Simons, and Charles Fennig. 2013. *Ethnologue: Languages of the World, Seventeeth Edition*. SIL International.

Christopher D. Manning, Prabhakar Raghavan, and Hinrich Schütze. 2008. *Introduction to information retrieval*, volume 1. Cambridge University Press Cambridge.

Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, Jake Vanderplas, Alexandre Passos, David Cournapeau, Matthieu Brucher, Matthieu Perrot, and Édouard Duchesnay. 2011. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.

Frank Rosenblatt. 1957. *The perceptron, a perceiving and recognizing automaton Project Para*. Cornell Aeronautical Laboratory.

Gerard Salton and Michael McGill. 1984. *Introduction to Modern Information Retrieval*. McGraw-Hill Book Company.

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*, pages 2585–2602, Mumbai, India, December. The COLING 2012 Organizing Committee.

Joel Tetreault, Daniel Blanchard, and Aoife Cahill. 2013. Summary report on the first shared task on native language identification. In *Proceedings of the Eighth Workshop on Building Educational Applications Using NLP*, Atlanta, GA, USA, June. Association for Computational Linguistics.

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 (NAACL '01)*.

Oren Tsur and Ari Rappoport. 2007. Using classifier features for studying the effect of native language on the choice of written second language words. In *Proceedings of the Workshop on Cognitive Aspects of Computational Language Acquisition*, pages 9–16.

Sze-Meng Jojo Wong and Mark Dras. 2009. Contrastive analysis and native language identification. In *Proceedings of the Australasian Language Technology Association Workshop*, pages 53–61. Citeseer.

Hsiang-Fu Yu, Fang-Lan Huang, and Chih-Jen Lin. 2011. Dual coordinate descent methods for logistic regression and maximum entropy models. *Machine Learning*, 85(1-2):41–75.

Jian Zhang, Rong Jin, Yiming Yang, and Alexander G. Hauptmann. 2003. Modified logistic regression: An approximation to svm and its applications in large-scale text categorization. In *ICML*, pages 888–895.