Identifying the L1 of non-native writers: the CMU-Haifa system

Yulia Tsvetkov* **Naama Twitto†** **Nathan Schneider*** **Noam Ordan†**
Manaal Faruqui* **Victor Chahuneau*** **Shuly Wintner†** **Chris Dyer***

*Language Technologies Institute
Carnegie Mellon University
Pittsburgh, PA
cdyer@cs.cmu.edu

†Department of Computer Science
University of Haifa
Haifa, Israel
shuly@cs.haifa.ac.il

Abstract

We show that it is possible to learn to identify, with high accuracy, the native language of English test takers from the content of the essays they write. Our method uses standard text classification techniques based on multiclass logistic regression, combining individually weak indicators to predict the most probable native language from a set of 11 possibilities. We describe the various features used for classification, as well as the settings of the classifier that yielded the highest accuracy.

1 Introduction

The task we address in this work is identifying the native language (L1) of non-native English (L2) authors. More specifically, given a dataset of short English essays (Blanchard et al., 2013), composed as part of the *Test of English as a Foreign Language (TOEFL)* by authors whose native language is one out of 11 possible languages—Arabic, Chinese, French, German, Hindi, Italian, Japanese, Korean, Spanish, Telugu, or Turkish—our task is to identify that language.

This task has a clear empirical motivation. Non-native speakers make different errors when they write English, depending on their native language (Lado, 1957; Swan and Smith, 2001); understanding the different types of errors is a prerequisite for correcting them (Leacock et al., 2010), and systems such as the one we describe here can shed interesting light on such errors. Tutoring applications can use our system to identify the native language of students and offer better-targeted advice. Forensic linguistic applications are sometimes required to determine the L1 of authors (Estival et al., 2007b; Estival et al., 2007a). Additionally, we believe that the task is interesting in and of itself, providing a better understanding of non-native language. We are thus equally interested in defining *meaningful* features whose contribution to the task can be linguistically interpreted. Briefly, our features draw heavily on prior work in general text classification and authorship identification, those used in identifying so-called *translationese* (Volansky et al., forthcoming), and a class of features that involves determining what minimal changes would be necessary to transform the essays into "standard" English (as determined by an n-gram language model).

We address the task as a multiway text-classification task; we describe our data in §3 and classification model in §4. As in other author attribution tasks (Juola, 2006), the choice of features for the classifier is crucial; we discuss the features we define in §5. We report our results in §6 and conclude with suggestions for future research.

2 Related work

The task of L1 identification was introduced by Koppel et al. (2005a; 2005b), who work on the International Corpus of Learner English (Granger et al., 2009), which includes texts written by students from 5 countries, Russia, the Czech Republic, Bulgaria, France, and Spain. The texts range from 500 to 850 words in length. Their classification method is a linear SVM, and features include 400 standard function words, 200 letter n-grams, 185 error types and 250 rare part-of-speech (POS) bigrams. Ten-

fold cross-validation results on this dataset are 80% accuracy.

The same experimental setup is assumed by Tsur and Rappoport (2007), who are mostly interested in testing the hypothesis that an author's choice of words in a second language is influenced by the *phonology* of his or her L1. They confirm this hypothesis by carefully analyzing the features used by Koppel et al., controlling for potential biases.

Wong and Dras (2009; 2011) are also motivated by a linguistic hypothesis, namely that *syntactic* errors in a text are influenced by the author's L1. Wong and Dras (2009) analyze three error types statistically, and then add them as features in the same experimental setup as above (using LIBSVM with a radial kernel for classification). The error types are subject-verb disagreement, noun-number disagreement and misuse of determiners. Addition of these features does not improve on the results of Koppel et al.. Wong and Dras (2011) further extend this work by adding as features horizontal slices of parse trees, thereby capturing more syntactic structure. This improves the results significantly, yielding 78% accuracy compared with less than 65% using only lexical features.

Kochmar (2011) uses a different corpus, the Cambridge Learner Corpus, in which texts are 200-400 word long, and are authored by native speakers of five Germanic languages (German, Swiss German, Dutch, Swedish and Danish) and five Romance languages (French, Italian, Catalan, Spanish and Portuguese). Again, SVMs are used as the classification device. Features include POS n-grams, character n-grams, phrase-structure rules (extracted from parse trees), and two measures of error rate. The classifier is evaluated on its ability to distinguish between pairs of closely-related L1s, and the results are usually excellent.

A completely different approach is offered by Brooke and Hirst (2011). Since training corpora for this task are rare, they use mainly L1 (blog) corpora. Given English word bigrams $\langle e_1, e_2 \rangle$, they try to assess, for each L1, how likely it is that an L1 bigram was translated literally by the author, resulting in $\langle e_1, e_2 \rangle$. Working with four L1s (French, Spanish, Chinese, and Japanese), and evaluating on the International Corpus of Learner English, accuracy is below 50%.

3 Data

Our dataset in this work consists of TOEFL essays written by speakers of eleven different L1s (Blanchard et al., 2013), distributed as part of the Native Language Identification Shared Task (Tetreault et al., 2013). The training data consists of 1000 essays from each native language. The essays are short, consisting of 10 to 20 sentences each. We used the provided splits of 900 documents for training and 100 for development. Each document is annotated with the author's English proficiency level (low, medium, high) and an identification (1 to 8) of the essay prompt. All essays are tokenized and split into sentences. In table 1 we provide some statistics on the training corpora, listed by the authors' proficiency level. All essays were tagged with the Stanford part-of-speech tagger (Toutanova et al., 2003). We did not parse the dataset.

	Low	**Medium**	**High**
# Documents	1,069	5,366	3,456
# Tokens	245,130	1,819,407	1,388,260
# Types	13,110	37,393	28,329

Table 1: Training set statistics.

4 Model

For our classification model we used the `creg` regression modeling framework to train a 11-class logistic regression classifier.[1] We parameterize the classifier as a multiclass logistic regression:

$$p_{\lambda}(y \mid \mathbf{x}) = \frac{\exp \sum_j \lambda_j h_j(\mathbf{x}, y)}{Z_{\lambda}(\mathbf{x})},$$

where $\mathbf{x}$ are documents, $h_j(\cdot)$ are real-valued feature functions of the document being classified, λ_j are the corresponding weights, and y is one of the eleven L1 class labels. To train the parameters of our model, we minimized the following objective,

$$\mathcal{L} = \alpha \overbrace{\sum_j \lambda_j^2}^{\ell_2 \text{ reg.}} - \sum_{\{(\mathbf{x}_i, y_i)\}_{i=1}^{|\mathcal{D}|}} \Big(\overbrace{\log p_{\lambda}(y_i \mid \mathbf{x}_i)}^{\text{log likelihood}} + \tau \underbrace{\mathbb{E}_{p_{\lambda}(y' \mid \mathbf{x}_i)} \log p_{\lambda}(y' \mid \mathbf{x}_i)}_{-\text{conditional entropy}} \Big),$$

[1] `https://github.com/redpony/creg`

which combines the negative log likelihood of the training dataset $\mathcal{D}$, an ℓ_2 (quadratic) penalty on the magnitude of $\boldsymbol{\lambda}$ (weighted by α), and the *negative* entropy of the predictive model (weighted by τ). While an ℓ_2 weight penalty is standard in regression problems like this, we found that the the additional entropy term gave more reliable results. Intuitively, the entropic regularizer encourages the model to remain maximally uncertain about its predictions. In the metaphor of "maximum entropy", the entropic prior finds a solution that has *more* entropy than the "maximum" model that is compatible with the constraints.

The objective cannot be minimized in closed form, but it does have a unique minimum and is straightforwardly differentiable, so we used L-BFGS to find the optimal weight settings (Liu et al., 1989).

5 Feature Overview

We define a large arsenal of features, our motivation being both to improve the accuracy of classification and to be able to interpret the characteristics of the language produced by speakers of different L1s.

While some of the features were used in prior work (§2), we focus on two broad novel categories of features: those inspired by the features used to identify translationese by Volansky et al. (forthcoming) and those extracted by automatic statistical "correction" of the essays. Refer to figure 1 to see the set of features and their values that were extracted from an example sentence.

POS *n*-grams Part-of-speech *n*-grams were used in various text-classification tasks.

Prompt Since the prompt contributes information on the domain, it is likely that some words (and, hence, character sequences) will occur more frequently with some prompts than with others. We therefore use the prompt ID in conjunction with other features.

Document length The number of tokens in the text is highly correlated with the author's level of fluency, which in turn is correlated with the author's L1.

Pronouns The use of pronouns varies greatly among different authors. We use the same list of 25 English pronouns that Volansky et al. (forthcoming) use for identifying translationese.

Punctuation Similarly, different languages use punctuation differently, and we expect this to taint the use of punctuation in non-native texts. Of course, character *n*-grams subsume this feature.

Passives English uses passive voice more frequently than other languages. Again, the use of passives in L2 can be correlated with the author's L1.

Positional token frequency The choice of the first and last few words in a sentence is highly constrained, and may be significantly influenced by the author's L1.

Cohesive markers These are 40 function words (and short phrases) that have a strong discourse function in texts (*however*, *because*, *in fact*, etc.). Translators tend to spell out implicit utterances and render them explicitly in the target text (Blum-Kulka, 1986). We use the list of Volansky et al. (forthcoming).

Cohesive verbs This is a list of manually compiled verbs that are used, like cohesive markers, to spell out implicit utterances (*indicate*, *imply*, *contain*, etc.).

Function words Frequent tokens, which are mostly function words, have been used successfully for various text classification tasks. Koppel and Ordan (2011) define a list of 400 such words, of which we only use 100 (using the entire list was not significantly different). Note that pronouns are included in this list.

Contextual function words To further capitalize on the ability of function words to discriminate, we define pairs consisting of a function word from the list mentioned above, along with the POS tag of its adjacent word. This feature captures patterns such as verbs and the preposition or particle immediately to their right, or nouns and the determiner that precedes them. We also define 3-grams consisting of one or two function words and the POS tag of the third word in the 3-gram.

Lemmas The content of the text is not considered a good indication of the author's L1, but many text categorization tasks use lemmas (more precisely, the stems produced by the tagger) as features approximating the content.

Misspelling features Learning to perceive, produce, and encode non-native phonemic contrasts

Firstly the employers live more savely because they are going to have more money to spend for luxury .

	Presence		Considered alternatives/edits	
Characters	"CHAR_l_y_ ":	log 2 + 1	"DeleteP_p_.":	1.0
	"CharPrompt_P5_g_o_i":	log 1 + 1	"InsertP_p_,":	1.0
	"MFChar_e_ ":	log 1 + 1	"MID:SUBST:v:f":	log 1 + 1
	"Punc_period":	log 1 + 1	"SUBST:v:f":	log 1 + 1
Words	"DocLen_":	log 19 + 1	"MSP:safely":	log 1 + 1
	"MeanWordRank":	422.6	"Match_p_to":	0.5
	"CohMarker_because":	log 1 + 1	"Delete_p_to":	0.5
	"MostFreq_have":	log 1 + 1	"Delete_p_are":	1.0
	"PosToken_last_luxury":	log 1 + 1	"Delete_p_because":	1.0
	"Pronouns_they":	log 1 + 1	"Delete_p_for":	1.0
POS	"POS_VBP_VBG_TO":	log 1 + 1		
	"POS_p_VBP_VBG_TO":	0.059		
Words + POS	"VBP_VBG_to":	log 1 + 1		
	"FW__more RB":	log 1 + 1		

Figure 1: Some of the features extracted for an L1 German sentence.

is extremely difficult for L2 learners (Hayes-Harb and Masuda, 2008). Since English's orthography is largely phonemic—even if it is irregular in many places, we expect leaners whose native phoneme contrasts are different from those of English to make characteristic spelling errors. For example, since Japanese and Korean lack a phonemic /l/-/r/ contrast, we expect native speakers of those languages to be more likely to make spelling errors that confuse *l* and *r* relative to native speakers of languages such as Spanish in which that pair is contrastive. To make this information available to our model, we use a noisy channel spelling corrector (Kernighan, 1990) to identify and correct misspelled words in the training and test data. From these corrections, we extract minimal edit features that show what insertions, deletions, substitutions and joinings (where two separate words are written merged into a single orthographic token) were made by the author of the essay.

Restored tags We focus on three important token classes defined above: punctuation marks, function words and cohesive verbs. We first remove words in these classes from the texts, and then recover the most likely hidden tokens in a sequence of words, according to an *n*-gram language model trained on all essays in the training corpus corrected with a spell checker and containing both words and hidden tokens. This feature should capture specific words or punctuation marks that are consistently omitted (deletions), or misused (insertions, substitutions). To restore hidden tokens we use the `hidden-ngram` utility provided in SRI's language modeling toolkit (Stolcke, 2002).

Brown clusters (Brown et al., 1992) describe an algorithm that induces a hierarchical clustering of a language's vocabulary based on each vocabulary item's tendency to appear in similar left and right contexts in a training corpus. While originally developed to reduce the number of parameters required in *n*-gram language models, Brown clusters have been found to be extremely effective as lexical representations in a variety of regression problems that condition on text (Koo et al., 2008; Turian et al., 2010; Owoputi et al., 2013). Using an open-source implementation of the algorithm,[2] we clustered 8 billion words of English into 600 classes.[3] We included log counts of all 4-grams of Brown clusters that occurred at least 100 times in the NLI training data.

5.1 Main Features

We use the following four feature types as the baseline features in our model. For features that are sensitive to frequency, we use the log of the (frequency-plus-one) as the feature's value. Table 2 reports the accuracy of using each feature type in isolation (with

[2] https://github.com/percyliang/brown-cluster

[3] http://www.ark.cs.cmu.edu/cdyer/en-600/cluster_viewer.html

Feature	Accuracy (%)
POS	55.18
FreqChar	74.12
CharPrompt	65.09
Brown	72.26
DocLen	11.81
Punct	27.41
Pron	22.81
Position	53.03
PsvRatio	12.26
CxtFxn (bigram)	62.79
CxtFxn (trigram)	62.32
Misspell	37.29
Restore	47.67
CohMark	25.71
CohVerb	22.85
FxnWord	42.47

Table 2: Independent performance of feature types detailed in §5.1, §5.2 and §5.3. Accuracy is averaged over 10 folds of cross-validation on the training set.

10-fold cross-validation on the training set).

POS Part-of-speech *n*-grams. Features were extracted to count every POS 1-, 2-, 3- and 4-gram in each document.

FreqChar Frequent character *n*-grams. We experimented with character *n*-grams: To reduce the number of parameters, we removed features only those character *n*-grams that are observed more than 5 times in the training corpus, and *n* ranges from 1 to 4. High-weight features include: `TUR:<Turk>`; `ITA:<Ital>`; `JPN:<Japa>`.

CharPrompt Conjunction of the character *n*-gram features defined above with the prompt ID.

Brown Substitutions, deletions and insertions counts of Brown cluster unigrams and bigrams in each document.

The accuracy of the classifier on the development set using these four feature types is reported in table 3.[4]

5.2 Additional Features

To the basic set of features we now add more specific, linguistically-motivated features, each adding a small number of parameters to the model. As above, we indicate the accuracy of each feature type in isolation.

[4]For experiments in this paper combining multiple types of features, we used Jonathan Clark's workflow management tool, `ducttape` (`https://github.com/jhclark/ducttape`).

Feature Group	# Params	Accuracy (%)	ℓ_2
POS	540,947	55.18	1.0
+ FreqChar	1,036,871	79.55	1.0
+ CharPrompt	2,111,175	79.82	1.0
+ Brown	5,664,461	81.09	1.0

Table 3: Dev set accuracy with MAIN feature groups, added cumulatively. The number of parameters is always a multiple of 11 (the number of classes). Only ℓ_2 regularization was used for these experiments; the penalty was tuned on the dev set as well.

DocLen Document length in tokens.

Punct Counts of each punctuation mark.

Pron Counts of each pronoun.

Position Positional token frequency. We use the counts for the first two and last three words before the period in each sentence as features. High-weight features for the *second* word include: `ARA:2<,>`; `CHI:2<is>`; `HIN:2<can>`.

PsvRatio The proportion of passive verbs out of all verbs.

CxtFxn Contextual function words. High-weight features include: `CHI:<some JJ>`; `HIN:<as VBN>`.

Misspell Spelling correction edits. Features included substitutions, deletions, insertions, doubling of letters and missing doublings of letters, and splittings (*alot*→*a lot*), as well as the word position where the error occurred. High-weight features include: `ARA:DEL<e>`, `ARA:INS<e>`, `ARA:SUBST<e>/<i>`; `GER:SUBST<z>/<y>`; `JPN:SUBST<l>/<r>`, `JPN:SUBST<r>/<l>`; `SPA:DOUBLE<s>`, `SPA:MID_INS<s>`, `SPA:INS<s>`.

Restore Counts of substitutions, deletions and insertions of predefined tokens that we restored in the texts. High-weight features include: `CHI:DELWORD<do>`; `GER:DELWORD<on>`; `ITA:DELWORD<be>`

Table 4 reports the empirical improvement that each of these brings independently when added to the main features (§5.1).

5.3 Discarded Features

We also tried several other feature types that did not improve the accuracy of the classifier on the development set.

CohMark Counts of each cohesive marker.

Feature Group	# Params	Accuracy (%)	ℓ_2
MAIN + Position	6,153,015	81.00	1.0
MAIN + PsvRatio	5,664,472	81.00	1.0
MAIN	5,664,461	81.09	1.0
MAIN + DocLen	5,664,472	81.09	1.0
MAIN + Pron	5,664,736	81.09	1.0
MAIN + Punct	5,664,604	81.09	1.0
MAIN + Misspell	5,799,860	81.27	5.0
MAIN + Restore	5,682,589	81.36	5.0
MAIN + CxtFxn	7,669,684	81.73	1.0

Table 4: Dev set accuracy with MAIN features plus additional feature groups, added independently. ℓ_2 regularization was tuned as in table 3 (two values, 1.0 and 5.0, were tried for each configuration; more careful tuning might produce slightly better accuracy). Results are sorted by accuracy; only three groups exhibited independent improvements over the MAIN feature set.

CohVerb Counts of each cohesive verb.

FxnWord Counts of function words. These features are subsumed by the highly discriminative CxtFxn features.

6 Results

The full model that we used to classify the test set combines all features listed in table 4. Using all these features, the accuracy on the development set is 84.55%, and on the test set it is 81.5%. The values for α and τ were tuned to optimize development set performance, and found to be $\alpha = 5, \tau = 2$.

Table 5 lists the confusion matrix on the test set, as well as precision, recall and F_1-score for each L1. The largest error type involved predicting Telugu when the true label was Hindi, which happened 18 times. This error is unsurprising since many Hindi and Telugu speakers are arguably native speakers of Indian English.

Production of L2 texts, not unlike translating from L1 to L2, involves a tension between the imposing models of L1 (and the source text), on the one hand, and a set of cognitive constraints resulting from the efforts to generate the target text, on the other. The former is called *interference* in Translation Studies (Toury, 1995) and *transfer* in second language acquisition (Selinker, 1972). Volansky et al. (forthcoming) designed 32 classifiers to test the validity of the forces acting on translated texts, and found that features sensitive to interference consistently yielded the best performing classifiers. And indeed, in this work too, we find fingerprints of the source language are dominant in the makeup of L2 texts. The main difference, however, between texts translated by professionals and the texts we address here, is that more often than not professional translators translate into their mother tongue, whereas L2 writers write out of their mother tongue by definition. So interference is ever more exaggerated in this case, for example, also phonologically (Tsur and Rappoport, 2007).

We explore the effects of interference by analyzing several patterns we observe in the features. Our classifier finds that the character sequence *alot* is overrepresented in Arabic L2 texts. Arabic has no indefinite article and we speculate that Arabic speakers conceive *a lot* as a single word; the Arabic equivalent for *a lot* is used adverbially like an *-ly* suffix in English. For the same reason, another prominent feature is a missing definite article before nouns and adjectives. Additionally, Arabic, being an Abjad language, rarely indicates vowels, and indeed we find many missing *e*'s and *i*'s in the texts of Arabic speakers. Phonologically, because Arabic conflates /ɪ/ and /ə/ into /i/ (at least in Modern Standard Arabic), we see that many *e*'s are indeed substituted for *i*'s in these texts.

We find that essays that contain hyphens are more likely to be from German authors. We again find evidence of interference from the native language here. First, relative clauses are widely used in German, and we see this pattern in L2 English of L1 German speakers. For example, *any given rational being – let us say Immanual Kant – we find that*. Another source of extra hyphens stems from compounding convention. So, for example, we find *well-known*, *community-help*, *spare-time*, *football-club*, etc. Many of these reflect an effort to both connect and separate connected forms in the original (e.g., *Fussballklub*, which in English would be more naturally rendered as *football club*). Another unexpected feature of essays by native Germans is a frequent substitution of the letter *y* for *z* and vice versa. We suspect this owes to their switched positions on German keyboards.

Lexical item frequency also provides clues to the L1 of the essay writers. The word *that* occurs more frequently in the texts of German L1 speakers. We

true↓	ARA	CHI	FRE	GER	HIN	ITA	JPN	KOR	SPA	TEL	TUR	Precision (%)	Recall (%)	F_1 (%)
ARA	80	0	2	1	3	4	1	0	4	2	3	80.8	80.0	80.4
CHI	3	80	0	1	1	0	6	7	1	0	1	88.9	80.0	84.2
FRE	2	2	81	5	1	2	1	0	3	0	3	86.2	81.0	83.5
GER	1	1	1	93	0	0	0	1	1	0	2	87.7	93.0	90.3
HIN	2	0	0	1	77	1	0	1	5	9	4	74.8	77.0	75.9
ITA	2	0	3	1	1	87	1	0	3	0	2	82.1	87.0	84.5
JPN	2	1	1	2	0	1	87	5	0	0	1	78.4	87.0	82.5
KOR	1	5	2	0	1	0	9	81	1	0	0	80.2	81.0	80.6
SPA	2	0	2	0	1	8	2	1	78	1	5	77.2	78.0	77.6
TEL	0	1	0	0	18	1	2	1	1	73	3	85.9	73.0	78.9
TUR	4	0	2	2	0	2	2	4	4	0	80	76.9	80.0	78.4

Table 5: Official test set confusion matrix with the full model. Accuracy is 81.5%.

hypothesize that in English it is optional in relative clauses whereas in German it is not, so German speakers are less comfortable using the non-obligatory form. Also, *often* is over represented. We hypothesize that since it is cognate of German *oft*, it is not cognitively expensive to retrieve it. We find *many times*—a literal translation of *muchas veces*—in Spanish essays.

Other informative features that reflect L1 features include frequent misspellings involving confusions of *l* and *r* in Japanese essays. More mysteriously, the characters *r* and *s* are misused in Chinese and Spanish, respectively. The word *then* is dominant in the texts of Hindi speakers. Finally, it is clear that authors refer to their native cultures (and, consequently, native languages and countries); the strings *Turkish*, *Korea*, and *Ita* were dominant in the texts of Turkish, Korean and Italian native speakers, respectively.

7 Discussion

We experimented with different classifiers and a large set of features to solve an 11-way classification problem. We hope that studying this problem will improve to facilitate human assessment, grading, and teaching of English as a second language. While the core features used are sparse and sensitive to lexical and even orthographic features of the writing, many of them are linguistically informed and provide insight into how L1 and L2 interact.

Our point of departure was the analogy between translated texts as a genre in its own and L2 writers as pseudo translators, relying heavily on their mother tongue and transferring their native models to a second language. In formulating our features, we assumed that like translators, L2 writers will write in a simplified manner and overuse explicit markers. Although this should be studied vis-à-vis comparable outputs of mother tongue writers in English, we observe that the best features of our classifiers are of the "interference" type, i.e. phonological, morphological and syntactic in nature, mostly in the form of misspelling features, restoration tags, punctuation and lexical and syntactic modeling.

We would like to stress that certain features indicating a particular L1 have no bearing on the quality of the English produced. This has been discussed extensively in Translation Studies (Toury, 1995), where interference is observed by the overuse or underuse of certain features reflecting the typological differences between a specific pair of languages, but which is still within grammatical limits. For example, the fact that Italian native speakers favor the syntactic sequence of determiner + adjective + noun (e.g., *a big risk* or *this new business*) has little prescriptive value for teachers.

A further example of how L2 quality and the ability to predict L1 are uncorrelated, we noted that certain L2 writers often repeat words appearing in their essay prompts, and including information about whether the writer was reusing prompt words improved classification accuracy. We suggest this reflects different educational backgrounds. This feature says nothing about the quality of the text, just as the tendency of Korean and Italian writers to mention their home country more often does not.

Acknowledgments

This research was supported by a grant from the Israeli Ministry of Science and Technology.

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.

Shoshana Blum-Kulka. 1986. Shifts of cohesion and coherence in translation. In Juliane House and Shoshana Blum-Kulka, editors, *Interlingual and intercultural communication Discourse and cognition in translation and second language acquisition studies*, volume 35, pages 17–35. Gunter Narr Verlag.

Julian Brooke and Graeme Hirst. 2011. Native language detection with 'cheap' learner corpora. In *Conference of Learner Corpus Research (LCR2011)*, Louvain-la-Neuve, Belgium. Presses universitaires de Louvain.

Peter F. Brown, Peter V. de Souza, Robert L. Mercer, Vincent J. Della Pietra, and Jenifer C. Lai. 1992. Class-based *n*-gram models of natural language. *Computational Linguistics*, 18(4).

Dominique Estival, Tanja Gaustad, Son Bao Pham, Will Radford, and Ben Hutchinson. 2007a. Author profiling for English emails. In *Proc. of PACLING*, pages 263–272, Melbourne, Australia.

Dominique Estival, Tanja Gaustad, Son Bao Pham, Will Radford, and Ben Hutchinson. 2007b. TAT: An author profiling tool with application to Arabic emails. In *Proc. of the Australasian Language Technology Workshop*, pages 21–30, Melbourne, Australia, December.

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

Rachel Hayes-Harb and Kyoko Masuda. 2008. Development of the ability to lexically encode novel second language phonemic contrasts. *Second Language Research*, 24(1):5–33.

Patrick Juola. 2006. Authorship attribution. *Foundations and Trends in Information Retrieval*, 1(3):233–334.

Mark D. Kernighan. 1990. A spelling correction program based on a noisy channel model. In *Proc. of COLING*.

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

Terry Koo, Xavier Carreras, and Michael Collins. 2008. Simple semi-supervised dependency parsing. In *Proc. of ACL*.

Moshe Koppel and Noam Ordan. 2011. Translationese and its dialects. In *Proc. of ACL-HLT*, pages 1318–1326, Portland, Oregon, USA, June. Association for Computational Linguistics.

Moshe Koppel, Jonathan Schler, and Kfir Zigdon. 2005a. Automatically determining an anonymous author's native language. *Intelligence and Security Informatics*, pages 41–76.

Moshe Koppel, Jonathan Schler, and Kfir Zigdon. 2005b. Determining an author's native language by mining a text for errors. In *Proc. of KDD*, pages 624–628, Chicago, IL. ACM.

Robert Lado. 1957. *Linguistics across cultures: applied linguistics for language teachers*. University of Michigan Press, Ann Arbor, Michigan, June.

Claudia Leacock, Martin Chodorow, Michael Gamon, and Joel Tetreault. 2010. *Automated Grammatical Error Detection for Language Learners*. Morgan and Claypool.

Dong C. Liu, Jorge Nocedal, Dong C. Liu, and Jorge Nocedal. 1989. On the limited memory BFGS method for large scale optimization. *Mathematical Programming B*, 45(3):503–528.

Olutobi Owoputi, Brendan O'Connor, Chris Dyer, Kevin Gimpel, Nathan Schneider, and Noah A. Smith. 2013. Improved part-of-speech tagging for online conversational text with word clusters. In *Proc. of NAACL*.

Larry Selinker. 1972. Interlanguage. *International Review of Applied Linguistics in Language Teaching*, 10(1–4):209–232.

Andreas Stolcke. 2002. SRILM—an extensible language modeling toolkit. In *Procedings of International Conference on Spoken Language Processing*, pages 901–904.

Michael Swan and Bernard Smith. 2001. *Learner English: A Teacher's Guide to Interference and Other Problems*. Cambridge Handbooks for Language Teachers. Cambridge University Press.

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

Gideon Toury. 1995. *Descriptive Translation Studies and beyond*. John Benjamins, Amsterdam / Philadelphia.

Kristina Toutanova, Dan Klein, Christopher D. Manning, and Yoram Singer. 2003. Feature-rich part-of-speech tagging with a cyclic dependency network. In *Proc. of HLT-NAACL*, pages 173–180, Edmonton, Canada, June. Association for Computational Linguistics.

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 *Proc. of the Workshop on Cognitive Aspects of Computational Language Acquisition*, pages 9–16, Prague, Czech Republic, June. Association for Computational Linguistics.

Joseph Turian, Lev Ratinov, and Yoshua Bengio. 2010. Word representations: A simple and general method for semi-supervised learning. In *Proc. of ACL*.

Vered Volansky, Noam Ordan, and Shuly Wintner. forthcoming. On the features of translationese. *Literary and Linguistic Computing*.

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

Sze-Meng Jojo Wong and Mark Dras. 2011. Exploiting parse structures for native language identification. In *Proc. of EMNLP*, pages 1600–1610, Edinburgh, Scotland, UK., July. Association for Computational Linguistics.