Determining an Author's Native Language by Mining a Text for Errors

Moshe Koppel
Computer Science Department
Bar-Ilan University
Ramat-Gan, 52900, ISRAEL
+972-3-5317874
koppel@cs.biu.ac.il

Jonathan Schler
Computer Science Department
Bar-Ilan University
Ramat-Gan, 52900, ISRAEL
+972-3-5317874
schlerj@cs.biu.ac.il

Kfir Zigdon
Computer Science Department
Bar-Ilan University
Ramat-Gan, 52900, ISRAEL
+972-3-5317874
zigdonk@cs.biu.ac.il

ABSTRACT

In this paper, we show that stylistic text features can be exploited to determine an anonymous author's native language with high accuracy. Specifically, we first use automatic tools to ascertain frequencies of various stylistic idiosyncrasies in a text. These frequencies then serve as features for support vector machines that learn to classify texts according to author native language.

Categories and Subject Descriptors

I.2.6 [**Artificial Intelligence**]: Learning – *Analogies, Concept learning, Connectionism and neural nets, Induction, Knowledge acquisition, Language acquisition, Parameter learning*

General Terms

Algorithms, Measurement, Experimentation

Keywords

Text mining, author profiling

1. INTRODUCTION

Stylistic analysis of a text might offer hints towards a psychological or demographic profiling of the text's author. For example, it has already been shown that automated text analysis methods can be used to identify an anonymous author's gender with accuracy above 80% [1].

In this paper, we will show that stylistic idiosyncrasies can be used to identify the native language of the author of a given English language text. Writers' spelling, grammar and usage in a second language are often influenced by patterns in their native language [2] [3]. Thus, it is plausible that certain writing patterns – function word selection, syntax and errors – might be

particularly prevalent for native speakers of a given language.

Some work [4] has been done on categorizing transcripts of English speech utterances as by native or non-native English speakers. In our experiments, we know that the writer is not a native English speaker but we wish to determine which language is native to the author. We consider written text, which offers the benefit of grammar and spelling cues, but loses the benefit of mispronunciation cues. To the best of our knowledge, this is the first published work on the automated determination of author native language from written text.

2. Stylistic Features

Identifying an author's native language is a type of authorship attribution problem. Instead of identifying a particular author from among a closed list of suspects, we wish to identify an author class, namely, those authors who share a particular native language.

Researchers in authorship attribution typically seek the kinds of features use of which is roughly invariant for a given author (or author class) across topics but which might vary from one author (or author class) to another. Generally, researchers use feature sets that are relatively common. Thus, for example, the seminal authorship attribution work of Mosteller and Wallace [5] on the Federalist Papers used a set of several hundred function words, that is, words that are context-independent and hence unlikely to be biased towards specific topics. Other features used in even earlier work [6] are complexity-based: average sentence length, average word length, type/token ratio and so forth. Recent technical advances in automated parsing and part-of-speech (POS) tagging have facilitated the use of syntactic and quasi-syntactic features such as POS n-grams [7] [8] [9] [10]. Other recent work [11] considers language modeling using letter n-grams.

However, human experts working on real-life authorship attribution problems do not work this way. They typically seek idiosyncratic usage by a given author that serves as a unique fingerprint of that author. For example, Foster [12] describes his techniques for identifying a variety of notorious anonymous authors including the author of the novel, Primary Colors, and the Unabomber. These techniques include repeated use of particular

types of neologisms or unusual word usage. Significantly, Foster identifies these linguistic idiosyncrasies manually. In the case of unedited texts, spelling and grammatical errors, which are typically eliminated in the editing process, can be exploited as well.

In this paper, we will use a variety of stylistic feature types that might be helpful for determining an author's native language. Very crudely, we can break these feature types into three broad categories.

1. Function words – As noted above, function words are useful for authorship attribution [5]. It stands to reason that such words might also be useful for native language identification since certain function words are liable to be used more or less frequently by native speakers of a given language, depending on the presence or absence of analogues for those words in the given language. A good example is the word *the*, which is typically used less frequently by native speakers of languages, such as Russian, that do not use a definite article.

2. Letter n-grams – As noted, letter n-grams have also been shown to be useful for authorship attribution [11]. It is likely that this is simply an artifact of variable usage of particular words, which in turn might be the result of different thematic preferences. In the case of native language attribution, however, letter n-grams might reflect the orthographic conventions of an author's native language (at least for those cases in which the native language uses the Latin alphabet).

3. Errors and Idiosyncrasies – As noted, errors and idiosyncrasies are the features commonly used by attribution experts analyzing data manually. Their utility for native language attribution is obvious: writers might be expected to transport orthographic or syntactic conventions from their native languages over to English in ways that result in non-conventional English. One of the main contributions of this paper will be to fully automate the process of idiosyncrasy detection. The next two sections of the paper will be devoted to describin

3. Error Types

Flagging of various types of writing errors has been used in a number of applications including teaching English as a foreign language (EFL) [13] [14] student essay grading [15] and, of course, word processing. The approaches used are either partially manual or do not flag the types of errors relevant to our task. Consequently, we develop our own automated methods for error-tagging.

Our first challenge is to identify the error types we are interested in tagging. After that we will show how to automate the tagging

process. The error types we consider fall into the following four categories:

1. Orthography – We consider here a range of spelling errors as follows:

- Repeated letter (e.g. remmit instead of remit)
- Double letter appears only once (e.g. comit instead of commit)
- Letter α instead of β (e.g. firsd instead of first)
- Letter inversion (e.g. fisrt instead of first)
- Inserted letter (e.g. friegnd instead of friend)
- Missing letter (e.g. frend instead of friend)
- Conflated words (e.g stucktogether)
- Abbreviations

The first six of these represents multiple error types since the specific letter(s) are specified as part of the error. For example, a missing i is a different error than a missing n. Thus, in principle, "Letter α instead of β" represents 26*25/2 = 325 separate error types. We will see below though that most of these occur so infrequently that we can consider only a small subset of them.

It should be emphasized that we use the term "error" or "idiosyncrasy" to refer to non-standard usage or orthography in U.S. English, even though often such usage or orthography simply reflects different cultural traditions or deliberate author choice.

2. Syntax – We consider non-standard usage as follows:

- Sentence Fragment
- Run-on Sentence
- Repeated Word
- Missing Word
- Mismatched Singular/Plural
- Mismatched Tense
- *that/which* confusion

Our system supports these error types for use in a variety of applications not considered in this paper. These errors are not appropriate for the native language problem we consider here, so we do not use them in the experiments reported below.

3. Neologisms – In order to leverage an observation of Foster [12] that certain writers tend to create neologistic adjectives (like *fantabulous*) while others create neologistic verbs, nouns, etc., we note for each POS (other than proper nouns), entirely novel exemplars (i.e. those for which there is no near match) of that POS.

4. Parts-of-speech bigrams – We consider 250 rare POS bigrams in the Brown corpus [16]. Such pairs might be presumed to be in error, or at least non-standard. Chodorow and Leacock [15] flag errors for essay grading by checking those POS pairs which appear less frequently in the corpus than would be expected based on the frequency of the pair's constituent individual POS. For our purposes, any rare POS bigram that shows up in a text is worth noting.

4. Automated Error Tagging

Of course, we can conjure many sophisticated error types, based upon deeper linguistic analysis of the text, besides those that were presented in the previous section. However, we restrict ourselves here to those considered above because they can be identified with relative ease, as we now show.

In order to tag errors in the above list, we exploit existing tools. Thus, for most of the error types in categories 1 and 2 above, we use the following procedure:

We run a text through the MS-Word application and its embedded spelling and grammar checker. Each error found in the text by the spell checker is recorded along with the best suggestion (to correct the error) suggested by the checker. Each pair <error, suggestion> is then processed by another program, which assigns it an "error type" from among those in the list we constructed.

Obviously, automated spelling and grammar checkers are far from perfect: certainly, suggested corrections may not reflect an author's intention. Nevertheless, since we are not interested in any individual error but rather to gather statistics on error-type frequencies, such automated checkers are adequate for our purpose. Still, for certain classes of errors we found MSWord's spell and grammar checker to be especially inadequate, so we prepared scripts ourselves for capturing them. In particular, we found that MSWord's spell checker was very weak at handling non-standard words with grammatical suffixes (-ism, -ist, -ble, -ive, -logy, -tion, etc.)

For categories 3 and 4, we run a text through the Brill [17] tagger. For category 4, we juxtapose results from MSWord's spelling checker (and our own routines for words with identifiable grammatical suffixes) with results of the Brill tagger.

When we ran our entire corpus of flawed texts through this process, we found that many error types on our list are so infrequent as to not be worth considering. Consequently, we reduced our list of error types to only those 185 types that occurred at least three times in a large corpus of chat group posts used for gathering error statistics (in addition to the 250 rare part-of-speech bigrams).

5. Experimental Setup

We use the International Corpus of Learner English [18], which was assembled for the precise purpose of studying the English writing of non-native English speakers from a variety of countries. All the writers included in the corpus are university students (mostly in their third or fourth year) studying English as a second language. All are roughly the same age (in their twenties) and are assigned to the same proficiency level in English. We consider sub-corpora contributed from Russia, Czech Republic, Bulgaria, France and Spain. The Czech sub-corpus, consisting of essays by 258 authors, is the smallest of these, so we take exactly 258 authors from each sub-corpus (randomly discarding the surplus). Each of the texts in the collection is of length between 579-846 words.

Each document in the corpus is represented as a numerical vector of length 1035, where each vector entry represents the frequency (relative to document length) of a given feature in the document. The features are:

- 400 standard function words
- 200 letter n-grams
- 185 error types
- 250 rare POS bigrams

We use multi-class linear support vector machines (SVM) [19] to learn models for distinguishing vectors belonging to each of the five classes. The efficacy of linear SVMs for text categorization is already well attested [20].

In order to test the effectiveness of models learned by SVMs to properly categorize unseen documents, we ran ten-fold cross-validation experiments: the corpus was divided randomly into ten sets of (approximately) equal size, nine of which were used for training and the tenth of which was used for testing the trained model. This was repeated ten times with each set being held out for testing exactly once.

6. Results

In Figure 1, we show accuracy results of ten-fold cross-validation experiments for various combinations of feature classes. As can be seen, when all feature types are used in tandem we obtain accuracy of 80.2%. The confusion matrix for the experiment with all features is shown in Table 1. It should be noted that a document is only regarded as being correctly classed if it is assigned to its correct class and to no other class. Thus, since we have five possible classes of roughly equal size, 20% accuracy is a reasonable baseline for this experiment.
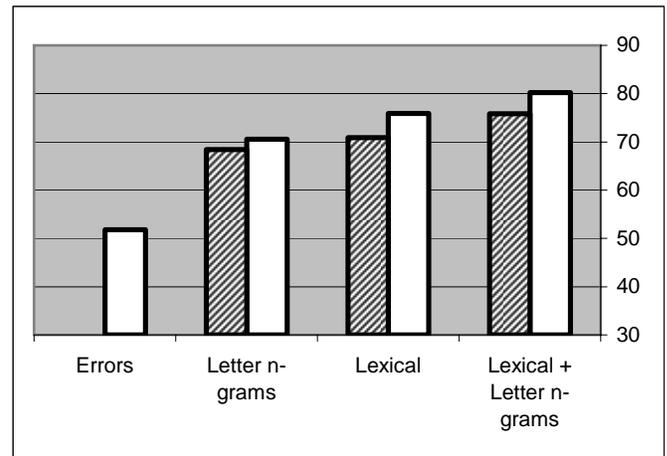

Figure 1 Accuracy (y-axis) on ten-fold cross-validation using various feature sets (x-axis) without (diagonal lines) and with (white) errors, and classifying with multi-class linear SVM. Note that "errors" refers to all error types, including rare POS bigrams.

Table 1 Confusion matrix for the author's native language identification using SVM

		Classified As				
		Czech	French	Bulgarian	Russian	Spanish
Actual	Czech	**209**	1	18	20	10
	French	9	**219**	13	12	5
	Bulgarian	14	8	**211**	18	7
	Russian	24	8	24	**194**	8
	Spanish	16	10	10	7	**215**

Table 2 A selection of features and the number of documents in each sub-corpus in which they appear.

Feature	Bulgarian	Czech	French	Russian	Spanish
cannot	131	100	79	65	57
however	127	65	92	45	81
indeed	25	3	86	15	9
over	66	72	61	116	52
*most*_ADVERB	20	3	3	8	1
NUMBER_*more*	5	6	2	17	5
there_to	2	2	14	2	0
NUMBER MODAL	16	8	54	23	5
DOUBLED CONSONANT	31	106	91	46	108
MR (no period)	1	14	39	1	16
VOWELo	6	20	37	22	46
UNDOUBLED CONSONANT	43	113	59	45	167
CONSnm	1	8	2	2	47
CONScq	0	0	0	0	16

The success of the system depends on the interaction of hundreds of features. Nevertheless, it is instructive to consider some of the features that proved particularly helpful in the learned models. Table 2 shows a variety of features along with the number of documents in each category in which the feature appears. We find that a number of distinctive patterns were exploitable for identifying native speakers of particular languages. For example:

- Some features that appear in more documents in the Bulgarian corpus than in the other corpora are the POS pair *most*–ADVERB and the somewhat formal function words *cannot* and *however*.
- A relatively large number of authors in the Spanish corpus had difficulty with doubling consonants, either doubling unnecessarily (*dissappear, fullfill, opening) or* omitting one of a double (*efect, inteligent*). Some errors that are almost exclusive to authors in the Spanish corpus derive directly from orthographic or pronunciation conventions of Spanish: confusion of *m* and *n* (*confortable*) or *q* and *c* (*cuantity, cuality*).
- A relatively high number of authors of documents in the Czech corpus also doubled letters in a non-standard way.
- Documents in the French corpus are characterized by relatively frequent use of the word *indeed* as well as *Mr* (without the period) and, as in the Spanish corpus, incorrect use of the vowel o (*outhor, psychodelic*). The POS pair*s number—modal_verb* (*one must, one could*) and *there–to* (*there* and *to* are each assigned their own POS in the Brill tag set) also appears more frequently in the French corpus.
- Authors in the Russian corpus are more prone to use the word *over* as well as the POS pair NUMBER—*more*.

The above examples are all features that distinguish one language corpus from all the rest. Of course there are many features that distinguish some subset of languages from the others. For example, the frequency of the word the is significantly less frequent in the documents by Czech (47.0 per 1000 words), Russian (50.1) and Bulgarian (52.3) authors than in those by French (63.9) and Spanish (61.4) authors. (Russian and Czech both do not use a definite article and Bulgarian uses it only as a suffix.)

Unsurprisingly, as can be seen in Table 1, most mistakes were among the three Slavic languages (Russian, Czech, Bulgarian).

7. Conclusions

We have implemented a fully automated method for determining the native language of an anonymous author. In experiments on a corpus including authors from five different countries, our method achieved accuracy of above 80% in categorizing unseen documents.

The authors of these documents were generally reasonably proficient in English (and may have even used automated spell-checkers), which made the task particularly difficult. It may be, however, that we were able to take unfair advantage of differences in overall proficiency among the different sub-corpora. For example, the Bulgarian authors were on average considerably less prone to errors than the Spanish authors. One way to ensure robustness against such artifacts of the available data would be to run similar experiments in which error frequency is normalized not against document length but rather against overall error frequency.

The applicability of these methods depends on a number of factors. Is the method precise enough to handle tens if not hundreds of different candidate native languages? How short can the documents be and still permit accurate categorization? Each of these questions requires further investigation.

8. REFERENCES

[1] Koppel, M., S. Argamon, A. Shimony. Automatically categorizing written texts by author gender. (2002) Literary and Linguistic Computing 17(4).

[2] Lado,R. Linguistics Across Cultures, Ann Arbor: (1961) University of Michigan Press.

[3] Corder, S. P. Error Analysis and Interlanguage. (1981) Oxford: Oxford University Press.

[4] Tomokiyo, L.M. and R. Jones. "You're Not From 'Round Here, Are You? Naive Bayes Detection of Non-native Utterance Text" (2001) NAACL 2001.

[5] Mosteller, F. and Wallace, D. L. Inference and Disputed Authorship: The Federalist.
Reading, Mass. : Addison Wesley, (1964).

[6] Yule, G.U. 1938. On sentence length as a statistical characteristic of style in prose with application to two cases of disputed authorship. Biometrika, 30, (1938) 363-390.

[7] Baayen, H., H. van Halteren, F. Tweedie, Outside the cave of shadows: Using syntactic annotation to enhance authorship attribution.In "Literary and Linguistic Computing, 11, (1996).

[8] Argamon-Engelson, S., M. Koppel, G. Avneri. Style-based text categorization: What newspaper am I reading?. in Proc. of AAAI Workshop on Learning for Text Categorization, (1998), pp. 1-4

[9] Stamatatos, E., N. Fakotakis & G. Kokkinakis, Computer-based authorship attribution without lexical measures. Computers and the Humanities 35, (2001) pp. 193—214.

[10] Koppel, M., J. Schler. Exploiting Stylistic Idiosyncrasies for Authorship Attribution. in Proceedings of "IJCAI'03 Workshop on Computational Approaches to Style Analysis and Synthesis", (2003) Acapulco, Mexico

[11] Peng, F., D. Schuurmans, S. Wang. Augmenting Naive Bayes Classifiers with Statistical Language Models. Inf. Retr. (2004) 7(3-4): 317-345

[12] Foster, D. Author Unknown: On the Trail of Anonymous, (2000) New York: Henry Holt.

[13] Dagneaux, E., Denness, S. & Granger, S. Computer-aided Error Analysis. System. An International Journal of Educational Technology and Applied Linguistics. Vol 26 (2), (1998) 163-174.

[14] Tono, Y., Kaneko, T., Isahara, H., Saiga, T. and Izumi, E. The Standard Speaking Test (SST) Corpus: A 1 million-word spoken corpus of Japanese learners of English and its implications for L2 lexicography. Second Asialex International Congress, Korea, (2001) pp. 257-262

[15] Chodorow, M. and C. Leacock. An unsupervised method for detecting grammatical errors, Proceedings of 1st Meeting of N. American Chapter of Assoc. for Computational Linguistics, (2000) 140-147