Nativeness Modeling Decisions

Name: Shayne Miel
Date: 02/27/2017

This document will present the decisions made around the modeling of nativeness in ELL writing. I will describe the problem and constraints, the data that I'm using, potential confounds and what I'm doing to mitigate them, the modeling task, and the architecture of the baseline and the intended approach.

Problem Statement

English Language Learners around the world would like to have rapid and reliable feedback on the quality of their English writing, both to assess their progress and to point out ways in which they might improve. One of the aspects of writing that ELL writers struggle with, broadly speaking, is making their writing look like the kind of writing produced by native English writers. In order to provide feedback on that aspect, we would like to be able to predict whether a section of text looks like it was generated by a native English writer. Ideally, we would like to be able to provide the probability that any given section of text was generated by a native English writer. This is sort of like an ELL Turing Test - if an English Language Learner can successfully fool the model into thinking that their writing is produced by a native writer, then they are doing very well on this aspect of writing.

Constraints

We do not have any labeled data that says explicitly how "native" a piece of writing is. Instead, we'll use the fact that we know some writing is from native writers and some is from ELL writers as binary labels. Unfortunately, we also don't have any large datasets that include both native and ELL writing.[1]

In order to have enough data to build a model, we will need to use an amalgamation of different data sets. I'll describe what we have below. Some of the data sets I was able to find do have a few prompts with both native and ELL writing. Not enough to use as a training set, but we can use it as our test set (more on this below). Some of the data also contains labels at the essay level and the student level that are somewhat related to writing proficiency (TOEFL levels, CEFR levels, Language trait scores, etc.). These are not standardized or prevalent enough to want to use them as labels when training, but it will be useful to look at whether they correlate with the predicted probability that writing is generated by a native writer.

[1] There may be something available in the Turnitin database. I have yet to explore that option.

Table 1: Datasets

Data Set	*n* essays	*n* prompts	L1s
ELL			
ICNALE	5,600	2	CHN, **ENG**, FIL, HKG, IND, JPN, KOR, PAK, SIN, THA, TWN
NUCLE	1,397	3	CHN
FCE	2,481	44	CAT, CHN, FRA, GER, GRC, ITA, JPN, KOR, NL, POL, PRT, RUS, SPA, SWE, THA, TUR
CEEAUS	1,008	2	CHN, **ENG**, JPN
MOECS	199	1	**ENG**, JPN
Gachon	15,831	20	KOR
TECCL	9,864	???	CHN
Chungdahm	550	1	KOR
NOG	4,694	4	CHN
TOEFL-11	12,100	8	ARA, CHN, FRA, GER, IND, ITA, JPN, KOR, SPA, TEL, TUR
Native English			
RA-2016	41,227	53	**ENG**
SBAC-Field	29,559	21	**ENG**
ASAP	17,677	8	**ENG**

In order to make an ELL product feasible, it is important that our models be prompt- and L1-independent, if possible.

Data

The following data sets were collected from online sources and data that was submitted to Turnitin privately. Some are only available as research corpora so we may need to replace them with other data if we want to operationalize this model.[2] 1 lists the 13 data sets that make up our corpus and 2 shows the meanings of the L1 abbreviations used.

In this collection there are a large number of prompts and L1s represented, with an approximately even split between native and non-native L1s. Three data sets (ICNALE, CEEAUS, and MOECS) have both native and non-native writing on the same prompt.

[2] Let's ask a lawyer though.

Table 2: L1s

Abbreviation	Language				
ARA	Arabic	GRC	Greek	PRT	Portuguese
BUL	Bulgarian	HKG	Hong Kong Cantonese	RUS	Russian
CAT	Catalan	IND	Indian languages	SIN	Singapore languages
CHN	Chinese	ITA	Italian	SPA	Spanish
CZE	Czech	JPN	Japanese	SWE	Swedish
ENG	English	KOR	Korean	TEL	Telugu
FIL	Filipino	NL	Dutch	THA	Thai
FIN	Finnish	NOR	Norwegian	TSW	Tswana
FRA	French	PAK	Urdu	TUR	Turkish
GER	German	POL	Polish	TWN	Taiwanese

Possible Confounds

There are all sorts of things that could trick our models into learning a correlate of non-native writing besides writing proficiency. I'll list them in 3 and talk about the ways that I've tried to mitigate the risk.

Task Description

Our eventual goal is to be able to give sub-essay feedback on which areas look the most/least proficient. In the absence of proficiency labels, I will use nativeness as a proxy. So, the eventual goal becomes to be able to give sub-essay feedback on which areas look the most/least native.

Since we don't really care about the accuracy of our native essay classifier, I'll use AUC as the metric of comparison between models. That metric tells us "for an essay randomly sampled from the native essays and an essay randomly sampled from the ELL essays, what's the probability that the native essays is ranked higher?"

This is also a convenient metric because we want to use the ICNALE, CEEAUS, and MOECS data sets as our test set, and the class distribution between native and ELL writers in those data sets is heavily imbalanced. See "The Splits.pdf" for a description of the train, dev and test sets.

The models will be trained to predict the binary native/non-native classification, using native as the positive class. Inputs to the classifier will be sliding windows of characters, such that every document $d \in D$ has k_d windows of n contiguous characters. If I have enough time, I will try three scenarios for predicting the essay-level label from the sliding window of characters:

Table 3: Confounds

Potential Confound	Mitigation
Formatting differences may cause the model to learn that an entire sub-dataset is native or ELL.	1. Use entirely separate data sets for train and test splits. 2. Restrict the formatting so that there is no repeated whitespace and all characters fall between \x32 (Space) and \x126 (~).
Prompt differences may cause the model to learn that an entire prompt is native or ELL.	1. Use entirely separate prompts for train, dev and test splits. 2. Make sure the prompts we do have that contain both ELL and native writing are present in the test set. 3. Control for the contribution of the prompt in the model (add an explanatory variable if using regression, add a secondary objective ala [1] if using a neural network, etc).
Native writers tend to write longer essays and longer sentences than ELL students. While "write more" is something we would want to encourage, it doesn't provide the kind of meaningful proficiency feedback we'd want to give.	Rather than using the full essay or even a particular sentence as an input, I will use sliding windows of $n = 100$ characters and take either the average or the max of the predicted probabilities.

1. Every window is given the label of the essay from which it came. The loss is a standard log loss across all windows:

$$L(D,Y) = -\frac{1}{K}\sum_{d\in D}\sum_{i=0}^{|d|-n} y_d log(p_{di}) + (1-y_d)log(1-p_{di})$$

   where $K = \sum_{d\in D} k_d$, and $p_{di} = f(d[i : i+n])$

2. Window predictions are averaged across an essay and the loss is a modified version of the loss described above:

$$L(D,Y) = -\frac{1}{|D|}\sum_{d\in D} y_d log\Big(\frac{1}{|d|-n}\sum_{i=0}^{|d|-n} p_{di}\Big) + (1-y_d)log\Big(1 - \frac{1}{|d|-n}\sum_{i=0}^{|d|-n} p_{di}\Big)$$

3. The essay prediction is the minimum window prediciton in that essay and the loss is another modified version of the loss described above:

$$L(D,Y) = -\frac{1}{|D|}\sum_{d\in D} y_d log\Big(min(p_{di} : i \in [0,\ldots,|d|-n]\Big) + (1-y_d)log\Big(1-min(p_{di} : i \in [0,\ldots,|d|-n])\Big)$$

Baselines

I'll use two baselines: perplexity from a pretrained langauge model, trained on the billion word corpus

References

[1] Zhong, Yu, and Gil Ettinger. ”Enlightening Deep Neural Networks with Knowledge of Confounding Factors.” *arXiv preprint* arXiv:1607.02397 (2016).