On the Development of a Large Scale Corpus for Native Language Identification

Thomas. G. Hudson, Sardar Jaf

Durham University, Durham, UK

g.t.hudson@durham.ac.uk, sardar.jaf@durham.ac.uk

Abstract

Native Language Identification (NLI) is the task of identifying an author's native language from their writings in a second language. In this paper, we introduce a new corpus (italki), which is larger than the current corpora. It can be used for training machine learning based systems for classifying and identifying the native language of authors of English text. To examine the usefulness of italki, we evaluate it by using it to train and test some of the well performing NLI systems presented in the 2017 NLI shared task. In this paper, we present some aspects of italki. We show the impact of the variation of italki's training dataset size of some languages on systems performance. From our empirical finding, we highlight the potential of italki as a large scale corpus for training machine learning classifiers for classifying the native language of authors from their written English text. We obtained promising results that show the potential of italki to improve the performance of current NLI systems. More importantly, we found that training the current NLI systems on italki generalize better than training them on the current corpora.

Keywords: Native Language, training data, italki, NLI, Native Language Identification, Language Identification, Dataset, Corpus.

1 Introduction

In the modern world where English is commonly used as the lingua franca of business and commerce, non-native speakers vastly outnumber native speakers of English. The study of the way non-native speakers of a language learn a new language is known as Second Language Acquisition (SLA), which focuses on the influence of the speaker's first language (FL) on their second language (SL) (Jarvis & Crossley 2012). Two types of analyses are considered in SLA: detection and comparison. The detection-based approach involves the use of large amounts of data to identify subtle patterns in the way SL usage differs from FL usage. The comparison-based approach is often used by linguists. It involves studying the differences between FLs to form hypotheses of the way these differences impact the speaker's use of their acquired language (SL).

The rise of ubiquitous computing and the availability of vast amount of data on the Internet has led to the popularity of the detection-based approach, which leads to the computational linguistics problem of native language identification.

Native language identification (NLI) is the process of identifying a writer's FL from a text (or speech). Computationally, it is a classification task. Supervised learning is the most popular approach to identifying the native language of an author from a text.

Solutions to this task have many applications. Analysis of successful systems provide insight into the theoretical linguistics, which underpins second language acquisition, types of text features that are successful are likely to emerge from some aspects of a group of FLs, which can then be examined further. Real world applications such as language teaching can be improved by identifying common mistakes to indicate areas of difficulty tailored to specific FL backgrounds (Rozovskaya & Roth 2011). NLI also has applications in forensic linguistics as part of wider studies on author profiling (Estival et al. 2007, Gibbons 2003) (e.g. Intelligence/security agencies can use NLI systems to build a profile of their target/suspect).

In the paper we provide details of the process for development a large scale dataset for NLI, empirical analyses of using the proposed dataset for the task of NLI and our future direction for further enhancing the proposed dataset . In section 2 we outline the related work. In section 3 we highlight various aspects of the proposed corpus and draw detailed comparison between the proposed corpus and the current corpora. In section 4 we evaluate the usefulness of the corpus for NLI task. Finally, section 5 concludes our work and set out our future work.

2 Related Work

Text classification systems (including solutions for native language identification) usually follow a structure outlined in Figure 1, which consists of four separate components.

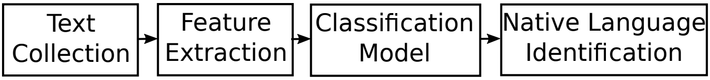

Figure 1: Framework of Tofighi et al. (2012).

Text collection involves collecting a body of text (corpus) for the task. Many approaches use pre-existing corpora made with the NLI task in mind. Next, since supervised approach is usually used for NLI, features (which are often in raw text format) are extracted from the corpus and converted to numerical attributes. These features are then supplied to a classification model, which utilizes machine learning algorithms to learn patterns and to distinguish between labelled data. Finally, the model is used to perform native language identification on unseen texts (i.e., identifying the label (FL) of the text).

The main two research events in this area were the shared task 2013 on native language identification (Tetreault et al. 2013), and more recently in 2017 (Malmasi et al. 2017). These landmark events consist of multiple teams competing to advance the state-of-the-art by designing systems to solve the task on a single shared dataset to allow a direct comparison between all approaches. The winning approaches in these tasks have become highly influential in the design of subsequent solutions to NLI.

All known solutions to the NLI problem use supervised learning. Thus, they rely on a collection of data (corpus) labelled with the native language of the author in order to train machine learning classifiers. In 2002, Granger et al. (2002) introduced the International Corpus of Learner English (ICLE) corpus to be used for the task of NLI. ICLE consists of argumentative essays written by university students of English in response to a range of prompts. Each essay is associated with a learner profile and it includes various aspects of the writer such as author's age, gender and native language.

The main limitation of ICLE, as argued by Tofighi et al. (2012), is its heavy topic bias and its character encoding, which could lead to inflated accuracy on ICLE. These issues prevent machine learning classifiers trained on it from generalizing to real-world examples, and possibly causing the conflation of native language identification and topic identification– another task in natural language processing.

These issues led Tetreault et al. (2012) to the introduction of the 'The Test of English as a Foreign Language (TOEFL)' corpus for the first NLI shared task in 2013. TOEFL has since become the de-facto standard in NLI studies with an updated version being released for the most recent NLI shared task in 2017 (Malmasi et al. 2017). The TOEFL corpus was designed to mitigate problems of topic bias, which plagued earlier corpora, by designing the collection process to sample prompts equally across all FLs. Refined for use in the 2017 shared task, it is the standard benchmark for comparing NLI systems.

A key problem with these existing corpora is their limited size - TOEFL only provides 1000 documents per language. While the current corpus size has been sufficient for training many shallow learning algorithms (e.g., Support Vector Machines which helped many systems to obtain impressive performance) it is not sufficient for most deep neural network algorithms (which often require data sizes many orders of magnitude larger than the TOEFL dataset). Dramatically increasing the size of the dataset could significantly improve the accuracy of NLI systems and allow deep neural network algorithms to contribute to this task, as it has contributed to many natural language processing (NLP) tasks.

There have been attempts to increase dataset size. Brooke & Hirst (2012) attempted to increase the dataset size in their studies on NLI by using 2 stages of machine translation (SL→FL→SL) on a large English corpus. One of the advantage of this approach is it helps in generating a very large corpus for NLI, where features of the FL are transferred. Another advantage is the removal of any possibility of topic bias as the source documents are all randomly sampled from the same corpus. However, empirically, this produced poor but above baseline results as it eliminated more nuanced features, such as misspelling, which are commonly used to boost the accuracy of state-of-the-art systems.

An alternative approach to Brooke & Hirst (2012) is web-scraping, which is a common practice in many areas of NLP. Web-scraping can provide diverse and vast quantities of data (big data) especially important for training deep neural networks (deep learning). As some datasets (which are many times larger than TOEFL) for deep learning show promise in other areas of NLP . The introduction of a new large corpus for NLI, which could be based on the italki website[1], should improve the performance of current shallow learning based systems, and enable deep learning to algorithm to contribute to the NLI problem.

The available corpora have been used for training various learning classifiers. The most widely used classifier in NLI has been linear Support Vector Machines (SVMs) which were used by Koppel et al. (2005) in their seminal work and by the winners of both NLI shared tasks 2013 (Tetreault et al. 2013) and 2017 (Malmasi et al. 2017)

k-Nearest Neighbors and MaxEnt classifiers that have been used in the NLI task yielded a performance competitive to SVMs (Tetreault et al. 2013). Introduced by Tetreault et al. (2012), the use of multiple classifiers has become a key component in many state-of-the-art systems for NLI.

Deep learning algorithms have been tried as an alternative to shallow learning approaches in the 2017 NLI shared task (Malmasi et al. 2017). Teams experimented with simple neural models, namely multilayer perceptrons on n-gram features, but found that SVMs produce higher accuracy in shorter times (Chan et al. 2017). It is generally accepted that deep learning algorithms require large quantities of data in order to learn representations of data and perform well. We believe that the availability of an open-source large dataset will help the NLI research community to explore the application of deep learning to NLI further.

3 The italki Corpus

There are different issues with the currently available corpora, chiefly the high cost of licenses and corpus size. To address those two issues, we propose a web-based corpus. We have gathered large quantities of text from the language learning website italki. The italki website creates a community for language learners to access teaching resources, practice speaking, discuss topics and ask questions in their target language (the English language). The raw data available on the italki website is in free-form ‘Notebook’ documents, which are mainly autobiographical diary entries with connected profiles describing the native language of the author. The required text and related metadata are retrieved from the italki website via an application programming interface (API).

For this work, we have gathered data for 11 languages (Arabic (ARA), Chinese (CHI), French (FRE), German (GER), Italian (ITA), Hindi (HIN), Japanese (JPN), Korean (KOR), Telugu (TEL), and Turkish (TUR)). The collected data[2] have undergone a similar normalization process as the TOEFL corpus. In the following subsection, we describe the different normalization processes the proposed corpus undergone.

3.1 Normalization

We follow the same practice for removing noisy features from the italki corpus as used in creating the TOEFL corpus. The noisy features that we have removed are listed below:

- **URL removal**: An analysis of the corpus showed that many authors include URLs in their writing. To prevent possible confusion of the classifier, we defined multiple regular expressions

[1] Available at: www.italki.com

[2] The public repository for the corpus is available at: https://github.com/ghomasHudson/italkiCorpus/releases/tag/v1.0 and https://bitbucket.org/sardarjaf/italki-corpus/downloads/

to remove them. This problem is also somewhat mitigated in the following step.

- **Long word removal**: We remove words that are more than fifteen characters of length. These words usually consist of missed URLs or sequences of control/HTML characters that do not generalize to other, off-line, scenarios.
- **Non-standard spacing removal**: Some authors used multiple spaces between words. These are removed. The above processes resulted in substantially cleaner data. Figure 2 shows an example of a normalized text. In the following subsection, we compare various aspects of the italki corpus against the TOEFL corpus.

> We had power cut until 2 hours ago besides the water cut . Just now weve had both thankfully . Being a crowded family , the absence of any of these creates some problems for us . The stars are shining in the sky now . Our house overlooks a barren valley in which horses , goats or sheep graze sometimes . This pastoral view feels as if I were in a village rather than a city . Theres hardly any rush in the vicinity no matter what time it is . Although our house doesnt possess a spectacular view , I like it for its tranquil Environment . Perhaps this is because of the fact that its one of the houses on the road to a nearby village . Weve been informed long before that power network in our apartment is connected to the nearby villages power network , which accounts for why we become extremely indignant when this power cut repeats . I am looking forward to Friday when waters said to will be provided to our vicinity . Water coming from Water trucks is not quite handy nor incessant since a whole building makes use of it .

Figure 2: Example of normalisation.

3.2 Comparing italki to TOEFL

The result of the post-normalization stage was a large corpus suitable for training machine learning algorithms for classifying text and identifying the native language of authors from their written English text. The data format in italki matches that of the TOEFL dataset. This allows the research community to easily use italki in future work without changing data import routines significantly. The structure of the corpus is shown in Figure 3. The corpus is organized in sub folders. The entire corpus is within the 'data' folder, which contains the essays, features and labels. The data in the corpus was randomized and it was divided to different sections. The essays and labels folders contain subfolders (development (dev), train and test folders). The data in the dev, train and test are used for validation, training and testing machine learning classifiers, respectively. The subfolders in the essays folder contain the raw data in 'txt' file format. The subfolders in the labels folder contain comma separated value (csv) files which have information about the labels (native languages) for

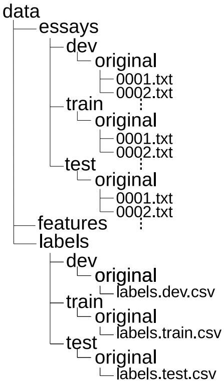

Figure 3: Corpus folder structure.

each document in each subfolder in the essays folder. The features folder is empty but can be used to temporarily store features such as part-of-speech tags by NLI systems.

Detailed information about the size of the data in the train, development and test folders for each language is presented in Table 1 The number of documents per language in italki is significantly larger than the number of documents per language in TOEFL (which is 1100 documents per language) except for Telugu. The italki corpus contains approximately 122,000 documents in total across a wide range of realistic topic areas, and it contains data for the same languages as in the TOEFL corpus. The content of the documents are raw text written in English by authors whose English is their second language. For each document, the native language of the author is used as the label for that document.

Lang	**Train**			**Dev**			**Test**		
	#Docs	**#Sent**	**Words**	**Docs**	**Sent**	**Words**	**Docs**	**Sent**	**Words**
Arabic	12035	55117	869008	1281	5759	87858	1465	6346	102175
Chinese	12113	120429	1998294	1348	13557	223239	1530	14815	244141
French	8129	66338	1001566	954	7782	118755	994	8311	127138
German	1563	15496	227172	178	1869	28359	188	1722	25499
Hindi	3942	29790	450335	438	3004	44826	469	3642	53427
Italian	8455	68017	1052870	952	7961	123068	991	7945	122275
Japanese	14327	137946	1648510	1542	15179	178820	1776	17226	205242
Korean	11205	121776	1488519	1239	13142	160926	1389	14604	177479
Spanish	12183	111117	1867441	1325	11947	199401	1006	13658	224963
Telugu	509	2498	32472	57	337	4944	48	229	2910
Turkish	6175	36027	409410	747	4049	48292	875	5049	57732

Table 1: Data sizes for train, development (Dev) and test. Lang=languages, #Docs=number of documents, #Sents=number of sentences and #Words=number of words.

Table 2 shows the data size across the 11 languages in italki and TOEFL corpora. For many of the languages we see 10-fold increase in the document size in italki compared to TOEFL. However, for Telugu, the number of documents in italki is less than that in TOEFL. The number of documents in italki for German, Hindi and Turkish, though it is larger than TOEFL, it is comparatively smaller than for other languages in italki. This creates data imbalance in italki. One of our future goal is to increase the data size for Telugu, German, Hindi and Turkish to eliminate the current data imbalance issue. The total number of words in italki for the majority of the languages is larger than the number of words in TOEFL. italki contains more words for Arabic, Chinese, French, Italian, Japanese, Korean, and Spanish than TOEFL. Hindi and Turkish also has more words in italki than TOEFL. However, there are less words for German in italki than there are in TEOFL. Telugu has exceptionally smaller number of words in italki than in TOEFL.

Despite the fact that italki appears smaller than TOEFL in terms of the average number of words per document/sentence and average sentence length per document, the total number of words and total number of sentences, for most of the languages, is much larger than those in TOEFL (as shown in and Table 2). However, the total number of sentences and the total number of words per language in italki is substantially larger than those in TOEFL for most of the languages, with the exception of Telugu and German, as shown in Table 2.

Language	italki			TOEFL		
	#docs	#sentences	#words	#docs	#sentences	#words
Arabic	14781	67222	1059041	1100	13813	314666
Chinese	14991	148801	2465674	1100	19883	358605
French	10077	82431	1247459	1100	18662	354484
German	1929	19087	281030	1100	19769	381161
Hindi	4849	36436	548588	1100	19620	391970
Italian	10398	83923	1298213	1100	14588	326679
Japanese	17645	170351	2032572	1100	19171	306794
Korean	13833	149522	1826924	1100	20530	332850
Spanish	15003	136722	2291805	1100	15695	363675
Telugu	614	3064	40326	1100	18626	368882
Turkish	7797	45125	515434	1100	19693	352911
Total	111,917	942,684	13,607,066	12,100	200,050	3,852,677

Table 2: Number of documents per language in italki and TOEFL corpora.

There are few major differences between italki and TOEFL. In italki, each document contains 60-150 words per languages. This is significantly different from TOEFL[3], where each document per language contains between 300 and 400 words. Figure 4 shows a comparison between italki and TOEFL with regards to this feature.

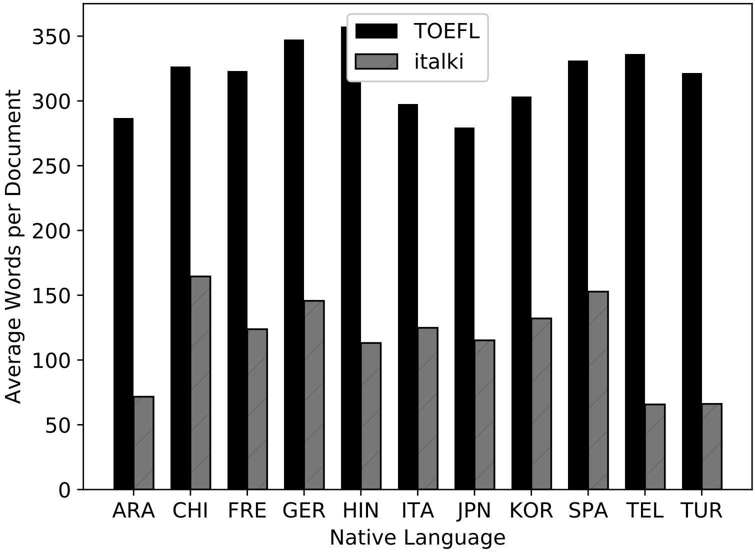

Figure 4: Average words per document for different languages in italki and TOEFL.

To further explore the differences between italki and TOEFL, we compare a range of metrics on the documents. From Figures 5 and 6 we can see that, on average, italki documents contains shorter

[3] This particular difference could make italki more appropriate for training NLI systems targeting short text, such as identifying the native language of authors of short messages.

sentences and fewer sentences than TOEFL, respectively. This reflects the nature of the italki website as a collection of free-form text, often diacritic entries as opposed to the structured essay responses in TOEFL. However, this feature could be useful for training system to perform NLI task on social media data, which often has a similar profile.

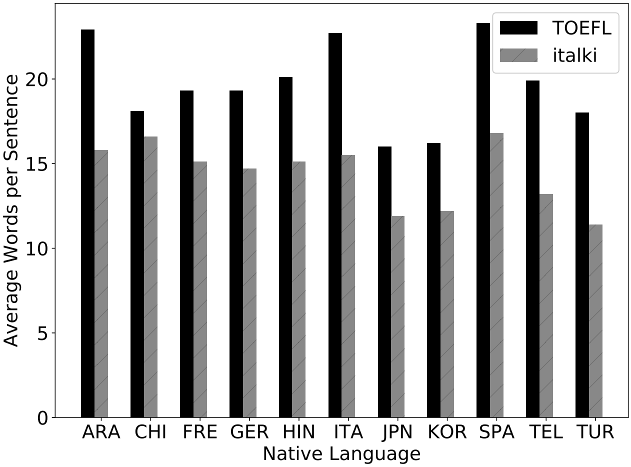

Figure 5: Comparing italki and TOEFL: Average sentence length per document.

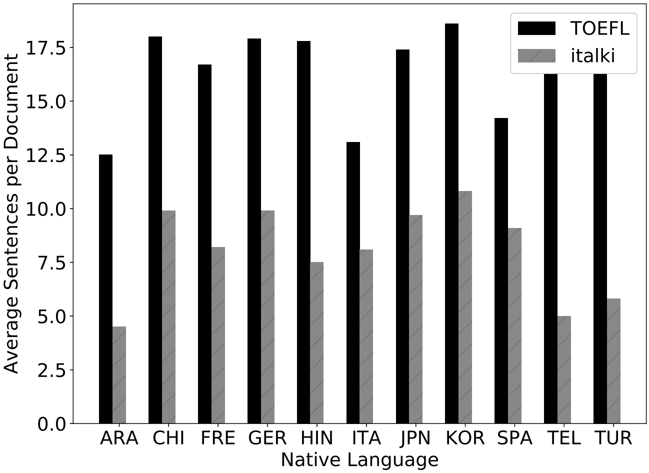

Figure 6: Comparing italki and TOEFL: Average number of sentences per document.

4 Corpus Evaluation

Supervised learning approach is usually used for the NLI problem. For this approach, a corpus of data, labelled with the native language of the author, is required. We evaluate the suitability of italki for the NLI problem by using it to train and evaluate several existing NLI systems. We choose four systems from the teams in the 2017 shared task (Malmasi et al. 2017) where implementations are readily available:

- **Groningen** (Kulmizev et al. 2017) - Character 1-9-grams classified with a linear SVM.
- **Tubasfs** (Rama & Coltekin 2017) - Character 7-grams and word bigrams classified with a linear SVM.
- **NLI-ISU** (Vajjala & Banerjee 2017) - 1-3-grams classified with MaxEnt, a probabilistic classifier which picks a model based on its entropy (Nigam et al. 1999).
- **Uvic-NLP** (Chan et al. 2017) - Character 4-5-grams and word 1-3-grams classified with a linear SVM.

The performance of several systems in the NLI 2017 shared task, when trained and tested on the TOEFL corpus, is shown in Table 3

System	**F1**
Groningen	0.8756
Tubasfs	0.8716
Uvic-NLP	0.8633
NLI-ISU	0.8264

Table 3: System performance from NLI 2017 shared task (Malmasi et al. 2017).

These systems were chosen based on their performance ranking (1, 2 and 3) in the NLI 2017 shared task and the retraining and on italki data. In the following subsection, we show our evaluation of those systems when they are trained and tested on the italki corpus.

Following the change in the 2017 shared task, We evaluate and rank the systems identified in Section 4 by their macro F1 score (averaging the per class F1 score across all classes (languages)) rather than accuracy as in earlier studies. We use the definition from Yang & Liu (1999) in order to combine recall (r) and precision (p) over q classes. This ensures systems which perform consistently across all classes are rewarded.

4.1 The Impact of Data Size on System Performance

One of the main advantages of using italki over the current corpora is the data size. In this section we will investigate the impact of a larger web-scraped corpus on the performance of the current, or potential, state-of-the-art systems. For this objective, we explore the affect of different training data sizes on four different NLI systems that were presented in the NLI 2017 shared task(Malmasi et al. 2017).

Experiment #1: NLI systems' performance on italki. For this experiment, we incrementally increase the size of the training set from 10,000 documents across all the eleven languages to 110,000 documents but we keep the test size to 1,000 documents. It is important to note that the training set is not balanced. Some languages have less training data than others. For example, Telugu has just over 500 documents while many other languages have more than 10,000 documents. As presented in Figure 7, we find that increasing the training data size to around 60000 documents it improves the systems' performance linearly. It is worth noting that the systems do not improve when the training size goes beyond 60000 documents. Although this is likely due to the nature of shallow learning algorithms (where even significant increase in the training size does not contribute to their learning) and the selected systems utilize shallow learning algorithms for training. However, the main reason is most likely due to the data imbalance of italki.

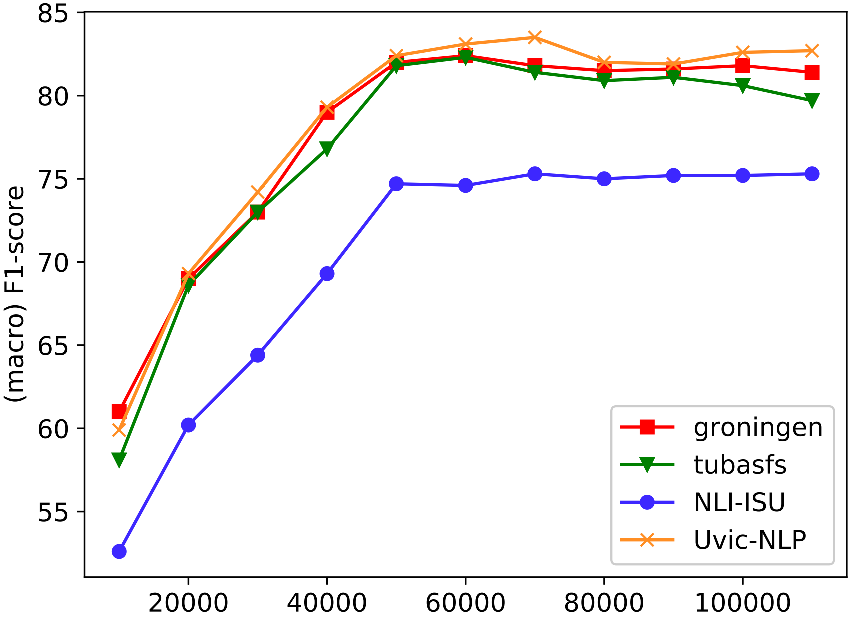

Figure 7: Systems performance when trained on italki corpus.

In the following experiments, we evaluate the performance of Tubasfs by incrementally increasing the size of the training data for each language from 500 to 7000 documents. The main reason for using Tubasfs is due to time constraint. The training time of Tubasfs was significantly shorter than other systems.[4]

Experiment #2: Tubasfs system performance on unbalanced italki. Training one of the top performing systems of the NLI 2017 shared task on italki yield lower accuracy than training them on TOEFL. As we mentioned earlier, currently, italki dataset is not well balanced across all the eleven languages. For a few languages (Telugu, German and Hindi) the training data is much smaller than the available training data for other languages. To understand the impact of the data imbalance of italki on the performance of the NLI systems, we evaluated the Tubasfs system[5] performance for each language by increasing the size of the training data incrementally while keeping the test size (1000 documents) constant as before. Table 4 shows the system's performance on different languages. We have gradually increased the number of documents in the training set from 500 to 7000 documents. The system performance degrades when the training size is increased beyond

[4] We have conducted 27 experiments in total to assess the impact of the data imbalance of italki, hence the use of a fast system for training, such as Tubasfs, was important.

[5] We have not experimented with the other available systems because of the time and space constraint. Our choice of using Tubasfs is mainly because of the speed of training it compared to other systems.

the available training data for some languages (Telugu, German and Hindi). It also appears that the data imbalance affects some of those languages with substantial training data (such as French, Italian and Spanish). The average system performance declines when the training set goes beyond 6000 documents. However, as presented in Figure 7 it appears the system achieves its optimum performance (83%) if the training size is dramatically increased (40,000 document). The impact of the data imbalance is evident in those languages with small training size in the corpus, such as Telugu, as it can be noted from Table 4.

Training Data	ARA	CHI	FRE	GER	HIN	ITA	JPN	KOR	SPA	TEL	TUR	AVG
500	0.349	0.573	0.374	0.550	0.776	0.487	0.581	0.535	0.451	0.557	0.360	0.508
1000	0.424	0.581	0.457	0.613	0.888	0.537	0.602	0.577	0.505	0.556	0.395	0.558
1500	0.534	0.599	0.544	0.612	0.900	0.591	0.632	0.617	0.578	0.504	0.429	0.594
2000	0.530	0.644	0.570	0.631	0.818	0.653	0.673	0.616	0.630	0.422	0.834	0.638
2500	0.574	0.693	0.578	0.607	0.786	0.627	0.676	0.624	0.616	0.274	0.875	0.630
3000	0.543	0.689	0.632	0.599	0.780	0.663	0.861	0.698	0.624	0.248	0.916	0.659
4000	0.567	0.708	0.860	0.570	0.747	0.684	0.896	0.705	0.680	0.150	0.946	0.683
5000	0.629	0.772	0.907	0.581	0.744	0.711	0.952	0.908	0.694	0.216	0.960	0.734
6000	0.650	0.765	0.920	0.545	0.762	0.724	0.947	0.930	0.708	0.183	0.946	0.735
7000	0.629	0.751	0.907	0.563	0.770	0.694	0.938	0.938	0.702	0.216	0.914	0.730

Table 4: Systems performance per language when trained on italki.

Experiment #3: Tubasfs system performance on relatively balanced italki. Once we identified the impact of the sparse training set of some languages, such as Telugu, in italki on the system, we conducted further experiment by excluding Telugu from the experiment because its training much smaller than other languages. We also conducted controlled experiment by omitting German and Hindi from this experiment once the training size reached a rate where their performance degraded in the experiment #2 (Hindi at 2000 documents and German at 2500 documents). As shown in Table 5, it can be noted that the system performance improved for all the languages. The average accuracy of the system increased by approximately 13%.

Training Data	ARA	CHI	FRE	GER	HIN	ITA	JPN	KOR	SPA	TUR	AVG
1000	0.477	0.583	0.484	0.630	0.917	0.545	0.602	0.593	0.542	0.435	0.581
1500	0.563	0.599	0.555	0.611	0.957	0.602	0.635	0.623	0.583	0.472	0.620
2000	0.579	0.635	0.573	0.641	_	0.649	0.670	0.631	0.638	0.871	0.654
2500	0.627	0.706	0.625	_	_	0.667	0.689	0.660	0.687	0.920	0.698
3000	0.636	0.706	0.667	_	_	0.709	0.861	0.731	0.716	0.956	0.748
4000	0.677	0.750	0.912	_	_	0.717	0.904	0.731	0.747	0.965	0.801
5000	0.724	0.817	0.942	_	_	0.778	0.957	0.930	0.773	0.965	0.861
6000	0.729	0.806	0.956	_	_	0.787	0.966	0.947	0.773	0.956	0.865
7000	0.729	0.806	0.956	_	_	0.787	0.966	0.947	0.773	0.956	0.865

Table 5: Systems performance per language when trained on italki.

Experiment #4: Tubasfs system performance on balanced italki. The final experiment, which was to identify the impact of large and balanced dataset on the system's performance, focused on training the system only on those languages (Arabic, Chinese, Japanese, Korea and Spanish) where their training sizes is more than 11,000 documents. We found that the system performance when trained on 7000 documents improved noticeably for Arabic, Chinese and Spanish. Moreover, the average system accuracy also improved.

Training Data	ARA	CHI	JPN	KOR	SPA	AVG
1000	0.608	0.657	0.670	0.703	0.716	0.671
2000	0.657	0.708	0.680	0.635	0.764	0.689
3000	0.701	0.756	0.868	0.729	0.784	0.768
4000	0.722	0.778	0.908	0.737	0.806	0.790
5000	0.774	0.845	0.966	0.943	0.831	0.872
6000	0.753	0.833	0.966	0.952	0.834	0.868
7000	0.750	0.839	0.966	0.952	0.840	0.869

Table 6: Systems performance per language when trained on italki.

The previous experiments have indicated that the availability of a large, and balanced, corpus could significantly improve NLI system performance. From those experiments we identify the need to increase the training size for some of the languages in italki[6]

Experiment #5: Testing for generalization. The previous experiment (experiment #1) showed that the performance of NLI systems when they are trained on italki may potentially be much lower than when they are trained on TOEFL dataset. One of the reasons could be because of the differences between italki and TOEFL from a number of aspect (see section 3.2 for more details). However, the main reason, as highlighted through experiments #2, #3 and #4, is due to the unbalanced size of the training set for some languages in italki.

Since one of the main goal of generating a classification model on a training set for any problem is to generalize the classification model to unseen data. In this experiment, we evaluate the appropriateness of italki for the task of NLI in terms of model generalization. Machine learning algorithms are trained and tuned on a set of training data in the hope that they can perform well on real world data, i.e., generalize to unseen data. One of the most suitable evaluation of the corpus to achieve this goal is via transfer learning.

We trained the selected NLI systems in section 4 on one corpus and tested them on another corpus to examine how well they generalize to data in other corpora (unseen data). We note that in this experiment, we use the unbalanced version of italki, i.e., we use the training set available for all the languages (see Table 1 for more detail).

Table 7 shows the result of the systems evaluation when trained on one corpus (such as italki) and tested on another corpus (such as TOEFL). The systems generalize better when training them on italki than on TOEFL. The empirical results indicate that italki is a better corpus for model generalization.

Table 8 shows the accuracy gain by individual system when trained on italki and tested on TOEFL. Uvic-NLP generalizes the most to unseen data while NLI-ISU generalizes the least. Groningen is second in place followed by Tubasfs in third place.

5 Conclusions

Native Language Identification (NLI) has benefited from the availability of data and advances in machine learning algorithms. Supervised approaches, where machine learning algorithms are trained on labelled data, to classifying the native language of author of text is the dominant approach

[6] Due to time constraint we have reported empirical result for a specific size of training set.

			Test	
			TOEFL	italki
Train	TOEFL	groningen	_	0.4042
		tubasfs	_	0.4035
		NLI-ISU	_	0.3374
		Uvic-NLP	_	0.4136
	italki	groningen	0.5879	_
		tubasfs	0.5807	_
		NLI-ISU	0.5035	_
		Uvic-NLP	0.6177	_

Table 7: Inter-corpus Performance based on F1 measure.

Systems	Generalization difference
Uvic-NLP	0.2041
Groningen	0.1837
Tubasfs	0.1777
NLI-ISU	0.1666

Table 8: Systems generalization measure.

to NLI problem. Although there are a number of corpora available for the NLI task, there exist some limitations in using them to train machine learning classifiers. In this study, we presented a web-scraped corpus (italki) which is larger than the current corpora. We have evaluated several publicly available systems, which performed well in the NLI 2017 shared task, on italki. We have empirically demonstrated that the current approaches, mainly shallow learning where the selected NLI systems utilize, benefit from a training data many times larger than the training data of the TOEFL corpus (which is the most heavily used corpus in previous work). We have evaluated the proposed corpus to identify its contribution to system's generalization to unseen data. We found that systems trained on italki generalize better than those trained on existing corpora.

From our experiment we have identified some limitations in italki. Chiefly, the data imbalance, where for a few languages (Telugu, Hindi and German) the training data is vastly smaller than for other languages. We aim to explore two approaches to address this issue: (i) to collect more data for those languages with small data size, and (ii) to explore the possibility of text generation model (which are based on training deep learning algorithms on the current data set) for automatically generating text for some of the languages with small data size.

Because of the unavailability of large training data, deep learning algorithms, unlike in other natural languages processing tasks, have not made headway in NLI task. Italki provides large quantities of training data. It allows us to experiment with deep learning classifiers and evaluate their performance on NLI. This narrowing of the gap between deep and shallow learning should serve as a motivation for further application of deep learning to NLI, which we aim to investigate in the future.

6 Acknowledgement

This work was supported by the CRITiCaL - combating cRiminals In The CLoud project (EPSRC ref: EP//M020576//1).

References

Brooke, J. & Hirst, G. (2012), Robust, Lexicalized Native Language Identification, *in* 'COLING2012: Conference on Computational Linguistics', The COLING 2012 Organizing Committee, Mumbai, India, pp. 391–408.

Chan, S., Jahromi, M. H., Benetti, B., Lakhani, A. & Fyshe, A. (2017), Ensemble Methods for Native Language Identification, *in* 'BEA2017: Workshop on Innovative Use of NLP for Building Educational Applications', Association for Computational Linguistics, Copenhagen, pp. 217–223.

Estival, D., Gaustad, T., Pham, S. B., Radford, W. & Hutchinson, B. (2007), Author profiling for English emails, *in* 'PACLING2007: Conference of the Pacific Association for Computational Linguistics', Melbourne, Australia, pp. 263–272.

Gibbons, J. (2003), *Forensic Linguistics: An Introduction to Language in the Justice System*, John Wiley & Sons.

Granger, S., Dagneaux, E., Meunier, F. & Paquot, M. (2002), *International corpus of learner English*, Presses universitaires de Louvain.

Jarvis, S. & Crossley, S. A. (2012), *Approaching Language Transfer Through Text Classification: Explorations in the Detection based Approach*, Vol. 64, Multilingual Matters, Bristol, UK.

Koppel, M., Schler, J. & Zigdon, K. (2005), 'Automatically determining an anonymous author's native language', *Intelligence and Security Informatics* pp. 41–76.

Kulmizev, A., Blankers, B., Bjerva, J., Nissim, M., Van Noord, G., Plank, B. & Wieling, M. (2017), The Power of Character N-grams in Native Language Identification, *in* 'BEA2017: Workshop on Innovative Use of NLP for Building Educational Applications', Association for Computational Linguistics, Copenhagen, pp. 382–389.

Malmasi, S., Evanini, K., Cahill, A., Tetreault, J., Pugh, R., Hamill, C., Napolitano, D. & Qian, Y. (2017), A Report on the 2017 Native Language Identification Shared Task, *in* 'BEA2017: Workshop on Innovative Use of NLP for Building Educational Applications', Association for Computational Linguistics, Copenhagen, pp. 62–75.

Nigam, K., Lafferty, J. & Mccallum, A. (1999), Using Maximum Entropy for Text Classification, *in* 'IJCAI1999: Workshop on Machine Learning for Information Filtering', Stockholm, Sweden, pp. 61–67.

Rama, T. & Coltekin, C. (2017), Fewer features perform well at Native Language Identification task, *in* 'BEA2017: Workshop on Innovative Use of NLP for Building Educational Applications', Association for Computational Linguistics, Copenhagen, pp. 255–260.

Rozovskaya, A. & Roth, D. (2011), Algorithm Selection and Model Adaptation for ESL Correction Tasks, *in* 'ACL2011: Meeting of the Association for Computational Linguistics)', Portland, Oregon, USA.

Tetreault, J., Blanchard, D. & Cahill, A. (2013), A report on the first native language identification shared task, *in* 'BEA2013: Workshop on innovative use of NLP for building educational applications', Association for Computational Linguistics, Atlanta, Georgia, pp. 48–57.

Tetreault, J., Blanchard, D., Cahill, A. & Chodorow, M. (2012), Native Tongues, Lost and Found: Resources and Empirical Evaluations in Native Language Identification, *in* ‘COLING2012: Conference on Computational Linguistics’, Vol. 2, Mumbai, India, pp. 2585–2602.

Tofighi, P., Köse, C. & and Leila Rouka (2012), ‘Author’s native language identification from web-based texts’, *International Journal of Computer and Communication Engineering* **1**(1), 47–50.

Vajjala, S. & Banerjee, S. (2017), A study of N-gram and Embedding Representations for Native Language Identification, *in* ‘BEA2017: Workshop on Innovative Use of NLP for Building Educational Applications’, Association for Computational Linguistics, Copenhagen, pp. 240–248.

Yang, Y. & Liu, X. (1999), A re-examination of text categorization methods, *in* ‘Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval’, ACM, pp. 42–49.