Oracle and Human Baselines for Native Language Identification

Shervin Malmasi[1], **Joel Tetreault**[2] **and Mark Dras**[1]

[1]Centre for Language Technology, Macquarie University, Sydney, NSW, Australia
[2] Yahoo Labs, New York, NY, USA

shervin.malmasi@mq.edu.au, tetreaul@yahoo-inc.com
mark.dras@mq.edu.au

Abstract

We examine different ensemble methods, including an oracle, to estimate the upper-limit of classification accuracy for Native Language Identification (NLI). The oracle outperforms state-of-the-art systems by over 10% and results indicate that for many misclassified texts the correct class label receives a significant portion of the ensemble votes, often being the runner-up. We also present a pilot study of human performance for NLI, the first such experiment. While some participants achieve modest results on our simplified setup with 5 L1s, they did not outperform our NLI system, and this performance gap is likely to widen on the standard NLI setup.

1 Introduction

Native Language Identification (NLI) is the task of inferring the native language (L1) of an author based on texts written in a second language (L2). Machine Learning methods are usually used to identify language use patterns common to speakers of the same L1 (Tetreault et al., 2012). The motivations for NLI are manifold. The use of such techniques can help SLA and ESL researchers identify important L1-specific learning and teaching issues, enabling them to develop pedagogical material that takes into consideration a learner's L1. It has also been used to study language transfer hypotheses and extract common L1-related learner errors (Malmasi and Dras, 2014).

NLI has drawn the attention of many researchers in recent years. With the influx of new researchers, the most substantive work in this field has come in the last few years, leading to the organization of the inaugural NLI Shared Task in 2013which was attended by 29 teams from the NLP and SLA areas (Tetreault et al., 2013).

An interesting question about NLI research concerns an upper-bound on the accuracy achievable for a dataset. More specifically, given a dataset, a selection of features and classifiers, what is the maximal performance that could be achieved by an NLI system that always picks the best candidate? This question, not previously addressed in the context of NLI to date, is the primary focus of the present work. Such a measure is an interesting and useful upper-limit baseline for researchers to consider when evaluating their work, since obtaining 100% classification accuracy may not be a reasonable or even feasible goal. In this study we investigate this issue with the aim of deriving such an upper-limit for NLI accuracy.

A second goal of this work is to measure human performance for NLI, something not attempted to date. To this end we design and run a crowdsourced experiment where human evaluators predict the L1 of texts from the NLI shared task.

2 Oracle Classifiers

One possible approach to estimating an upper-bound for classification accuracy, and one that we employ here, is the use of an "Oracle" classifier. This method has previously been used to analyze the limits of majority vote classifier combination (Kuncheva et al., 2001). An oracle is a type of multiple classifier fusion method that can be used to combine the results of an ensemble of classifiers which are all used to classify a dataset.

The oracle will assign the correct class label for an instance if at least one of the constituent classifiers in the system produces the correct label for that data point. Some example oracle results for an ensemble of three classifiers are shown in Table 1. The probability of correct classification of a data point by the oracle is:

$$P_{\text{Oracle}} = 1 - P(\text{All Classifiers Incorrect})$$

Proceedings of the Tenth Workshop on Innovative Use of NLP for Building Educational Applications, 2015, pages 172–178,
Denver, Colorado, June 4, 2015. ©2015 Association for Computational Linguistics

		Classifier Output			
Instance	**True Label**	C_1	C_2	C_3	**Oracle**
`18354.txt`	`ARA`	`TUR`	`ARA`	`ARA`	Correct
`15398.txt`	`CHI`	`JPN`	`JPN`	`KOR`	Incorrect
`22754.txt`	`HIN`	`GER`	`TEL`	`HIN`	Correct
`10459.txt`	`SPA`	`SPA`	`SPA`	`SPA`	Correct
`11567.txt`	`ITA`	`FRE`	`GER`	`SPA`	Incorrect

Table 1: Example oracle results for an ensemble of three classifiers.

Oracles are usually used in comparative experiments and to gauge the performance and diversity of the classifiers chosen for an ensemble (Kuncheva, 2002; Kuncheva et al., 2003). They can help us quantify the *potential* upper limit of an ensemble's performance on the given data and how this performance varies with different ensemble configurations and combinations.

One scenario is the use of an oracle to evaluate the utility of a set of feature types. Here each classifier in the ensemble is trained on a single feature type. This is the focus of our first experiment (§5).

Another scenario involves the combination of different learning algorithms trained on similar features, to form an ensemble in order to evaluate the potential benefits and limits of combining different classification approaches. This is the focus of our second experiment (§6), using all of the entries from the 2013 shared task as systems.

3 Data

Released as part of the 2013 NLI Shared task, the TOEFL11 corpus (Blanchard et al., 2013) [1] is the first dataset designed specifically for the task of NLI and developed with the aim of addressing the deficiencies of other previously used corpora. By providing a common set of L1s and evaluation standards, the shared task set out to facilitate the direct comparison of approaches and methodologies. TOEFL11 includes 12,100 learner essays sampled evenly from 11 different L1 backgrounds: Arabic, Chinese, French, German, Hindi, Italian, Japanese, Korean, Spanish, Telugu and Turkish.

4 Ensemble Combination Methods

We experiment with several ensemble combination methods to draw meaningful comparisons.

Oracle The correct label is selected if predicted by any ensemble member, as described in §2.

Plurality Voting This is a standard combination strategy that selects the label with the highest number of votes,[2] regardless of the percentage of votes it received (Polikar, 2006).

Accuracy@N To account for the possibility that a classifier may predict the correct label by chance (with a probability determined by the random baseline), we propose an Accuracy@N combiner. This method is inspired by the "Precision at k" metric from Information Retrieval (Manning et al., 2008) which measures precision at fixed low levels of results (*e.g.* the top 10 results). Here, it is an extension of the Plurality vote combiner where instead of selecting the label with the highest votes, the labels are ranked by their vote counts and an instance is correctly classified if the true label is in the top N ranked candidates.[3] In other words, it is a more restricted version of the Oracle combiner that is limited to the top N ranked candidates in order to minimize the influence of a single classifier having chosen the correct label by chance. In this study we experiment with $N = 2$ and 3. We also note that setting $N = 1$ is equivalent to the Plurality voting method.

Mean Probability All classifiers provide probability estimates for each possible class. Each class' estimates are summed and the one with the highest mean wins (Polikar, 2006, §4.2).

Simple Combination combines all features into a single feature space.

5 Feature Set Evaluation

Our first experiment attempts to derive the potential accuracy upper-limit of our feature set. We train a single linear Support Vector Machine (SVM) classifier for each feature type to create our classifier ensemble. Linear SVMs have been shown to be effective for such text classification problems and was the classifier of choice in the 2013 NLI Shared Task. We do not experiment with combining different machine learning algorithms here, instead we focus on gauging the potential of the feature set. We employ a standard set of previously used feature types: character/word n-grams, Part-of-Speech (POS) n-grams, function words, Context-free grammar production rules, Tree Substitution Grammar fragments and Stanford Dependencies. Descriptions of these features can be

[1] http://catalog.ldc.upenn.edu/LDC2014T06

[2] This differs with a *majority* vote combiner where a label must obtain over 50% of the votes.

[3] In case of ties we choose randomly from the labels with the same number of votes.

	Accuracy (%)	
	10-fold CV	**Test Set**
Random Baseline	9.1	9.1
Shared Task Best	84.3 (84.5)	83.6 (85.3)
Oracle	95.6	95.4
Accuracy@3	92.5	92.2
Accuracy@2	88.6	88.0
Plurality Vote	78.2	77.6
Simple Combination	78.2	77.5
Mean Probability	79.4	78.7

Table 2: Oracle results using our feature set.

found in §4.1 of Tetreault et al. (2012).[4]

We report classification accuracy under 10-fold cross-validation using the TOEFL11 training data and also on the test set from the 2013 shared task, shown in Table 2. For both Tables 2 and 3 we report a random baseline and the best performances on the Shared Task: the first number is the top performer from the shared task (Jarvis et al., 2013), and the number in parentheses is the best published performance after the shared task (Ionescu et al., 2014) . The cross-validation and test results are very similar, with the oracle accuracy at 95%, suggesting that for each document there is in most cases at least one feature type that correctly predicts it. This drops to 88% with the Accuracy@2 combiner, still much higher than the plurality vote and the best results from the shared task. This suggests that there is a noticeable tail of feature types dragging the plurality vote down.

6 2013 Shared Task Evaluation

In the second experiment we apply our methods to the submissions in the 2013 NLI Shared Task, aiming to quantify the potential upper limit for combining a range of different systems.

The data comes from the closed-training sub-task.[5] Each team was allowed to submit up to 5 different runs for each task, allowing them to experiment with different feature and parameter variations of their system. Each team's systems produce predictions using their own set of features and learning algorithms, with several of these systems using ensembles themselves.

In total, 115 runs were submitted by 29 teams, with the winning entry achieving the highest accuracy of 83.6% on the test set. We experiment under two conditions: using only each team's best run and using all 115 runs. Results are compared against the random baseline and winning entry.

	Accuracy (%)	
	Best Run	**All Runs**
Random Baseline	9.1	9.1
Shared Task Best	84.3 (84.5)	83.6 (85.3)
Oracle	97.9	99.5
Accuracy@3	95.5	95.6
Accuracy@2	92.2	92.5
Plurality Vote	84.5	84.4

Table 3: Oracle results on the shared task systems.

Table 3 shows the results for this experiment. The oracle results are higher than the previous experiment, which is not unexpected given the much larger number of predictions per document. Results for the other combiners are also higher here.

The Accuracy@2 results are 92% in both conditions, much higher than the winning entry's 83%. Results from the Accuracy@2 combiner, both here and in the previous experiment, show that a great majority of the texts are close to being correctly classified: this value is significantly higher than the plurality combiner[6] and not much lower than the oracle. This shows that the correct label receives a significant portion of the votes and when not the winning label, it is often the runner-up.[7]

One implication of this concerns practical applications of NLI, *e.g.* in a manual analysis, where it may be worthwhile for researchers to also consider the runner-up label in their evaluation.

This knowledge could also be used to increase NLI accuracy by aiming to develop more sophisticated classifiers that can take into account the top N labels in their decision making, similar to discriminative reranking methods applied in statistical parsing (Charniak and Johnson, 2005).

Using the Accuracy@2 combiner, we isolate the cases where the actual label was the runner up and extract the most frequent pairs of top 2 labels, presented in Table 4. We see that a quarter of the errors are confusion between Hindi and Telugu. The Korean and Turkish confusion could be due to both being Altaic languages.

We also examine the confusion matrices for the plurality, Accuracy@2 and oracle combiners,[8] shown

[4] For features comparisons see Malmasi and Cahill (2015)

[5] The shared task consisted of three sub-tasks. For each task, the test set was TOEFL11-TEST; only the type of training data varied by task where the other two sub-tasks allowed the use of external training data.

[6] Which is itself equivalent to an Accuracy@1 combiner.

[7] In approx. 8% of the cases here, to be more precise.

[8] Where the Accuracy@2 and oracle combiners could not predict the correct label the plurality vote was used.

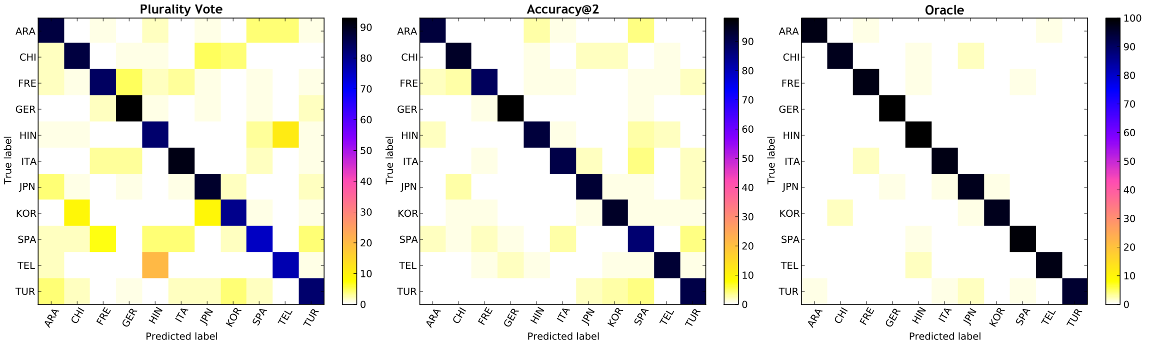

Figure 1: Confusion matrices for the plurality (L), Accuracy@2 (M) and oracle (R) combiners..

Confused Pair	Percent	Cumulative Percent
HIN–TEL	15.9	15.9
TEL–HIN	10.2	26.1
CHI–KOR	6.8	33.0
JPN–KOR	6.8	39.8
KOR–TUR	4.5	44.3

Table 4: Most commonly predicted top 2 label pairs where the runner-up is the true label.

in Figure 1. They show that Hindi–Telugu is the most commonly confused pair and confirm the directionality of the confusion: more Telugu texts are misclassified as Hindi than vice versa.

7 Human NLI Performance

While most NLI work has focused on improving system performance, to our knowledge there has not been any corresponding study which looks at human performance for this task. To give our preceding results more context, as well as the results of the field, we ran an exploratory study to determine how accurate humans are for this task.

7.1 Experiment Design

Our initial hypothesis was to use the Amazon Mechanical Turk to collect crowdsourced judgments. However, unlike simpler NLP tasks, e.g. sentiment analysis and word sense disambiguation, which can be effectively annotated by untrained Turkers (Snow et al., 2008), NLI requires raters with knowledge and exposure to writers with different L1s. Optimally, one would use a set of ESL teachers and researchers who have experience in working with ESL writers from all of the 11 L1s, though such people are rarity. As a reasonable compromise, we chose 10 professors and researchers who have varied linguistic backgrounds, speak multiple languages, and have had exposure with the particular L1s, either as a speaker or through working with ESL students. We also constrained the task from 11 L1s to 5 (Arabic, Chinese, German, Hindi, and Spanish) as we believed that 11 L1s would be too much of an overload on the judges. The 5 L1s were selected since they all belong to separate language families.

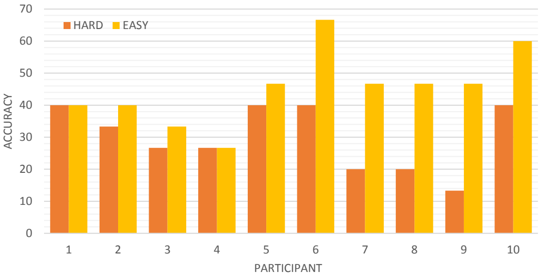

Figure 2: Prediction accuracy for each of our 10 participants under both easy and hard conditions.

The experiment consisted of rating 30 essays from TOEFL11-TEST, 15 of which most Shared Task systems could predict correctly (easy), and the remaining 15 were essays in which the Shared Task systems had difficulty (hard). The L1s were distributed evenly over the essays and easy/hard conditions (3 "easy" and 3 "hard" essays per L1).

7.2 Results

Figure 2 shows the accuracy for each rater in this pilot study. The top rater accurately identified 16 out of 30 L1s (53.3%), with the lowest raters at 30.0% overall and an average of 37.3%. All raters did better on the "easy" cases than on the "hard." A paired-samples t-test was conducted to compare human accuracy in the easy and hard conditions. A significant difference was found for easy (M=45.33, SD=11.67) and hard (M=30, SD=10.06), t(9)=−3.851, $p = .004$.

Next, we compared human accuracy with our NLI system, which we re-trained using only the five selected L1s. Results are shown in Table 5. All ensembles outperform human raters and a plurality vote composed of the human raters. Interestingly, the human plurality vote was only 3% higher than the top human score, suggesting that the raters tended to get the same essays correct.

	Accuracy (%)		
	Easy	**Hard**	**All**
Random Baseline	20.0	20.0	20.0
NLI Plurality Vote	100.0	33.3	66.7
NLI Mean Probability	**100.0**	**46.7**	**73.3**
Top Human	66.7	40.0	53.3
Human Plurality Vote	73.3	40.0	56.7

Table 5: Comparing human participant performance against an NLI system on 30 selected texts.

We also note that some L1s received more correct predictions than others,[9] but the difference is not statistically significant.[10] Some participants noted that while they had familiarity with L1 Spanish/Chinese non-native writing, they did not have much exposure to the other L1s, possibly due to international student cohorts.

Our belief, based on these pilot results, is that as the number of classes increases, the system will outpace the human raters by a widening margin. It should also be noted that we purposefully selected disparate L1s to make easier for the human raters. As there are several other L1s in the TOEFL11 that are in the Romance family class, and others where it is less likely for raters to have seen student essays (such as Telugu), including those will also likely affect human performance.

8 Related Work

Prior work has shown that ensemble classification can improve NLI performance. Tetreault et al. (2012) established that ensembles composed of classifiers trained on different feature types were useful for NLI and we also take this approach. Several shared task systems also found improvements using different ensemble classifications. Goutte et al. (2013) used plurality voting in their shared task submission which placed seventh. Cimino et al. (2013) found that a meta-classifier approaches outperformed plurality voting, while both outperformed their basic system. Malmasi et al. (2013) experimented with 7 different methods of ensemble classification and found that the mean probability method performed best, though they note that all ensemble methods were within about 1% of each other. This method, performed after the final submission phase, performed at 83.6%, the same as the top performing system (Jarvis et al., 2013).

More recently, Bykh and Meurers (2014) extended their shared task submission (Bykh et al., 2013) by investigating the use of model selection and tuning for ensemble classification. Their method outperformed plurality voting, and when combined with improvements to syntactic and n-gram features, produced a performance of 84.82%. Finally, Ionescu et al. (2014) used string kernels to achieve the highest reported result on the TOEFL11-TEST: 85.3% and 10-fold CV: 84.5%.

In contrast to the prior work, our work in combining the outputs of each system could not make use of the development set since that would require the actual code from all 29 systems. If that were available, then a meta-classifer could be used to further improve performance.

9 Discussion

We presented a novel analysis for predicting the "potential" upper limit of NLI accuracy on a dataset. This upper limit can vary depending on which components – feature types and algorithms – are used in the system. Alongside other baselines, oracle performance can assist in interpreting the relative performance of an NLI system.[11]

A useful application of this method is to isolate the subset of wholly misclassified texts for further investigation and error analysis. This segregated data can then be independently studied to better understand the aspects that make it hard to classify them correctly. This can also be used to guide feature engineering practices in order to develop features that can distinguish these challenging texts. In practice, this type of oracle measure can be used to guide the process of choosing the pool of classifiers that form an ensemble.

We also note that these oracle figures would be produced by an optimal system that always makes the correct decision using this pool of classifiers. While these oracle results could be interpreted as potentially attainable, this may not be feasible and practical limits could be substantially lower.

A potentially fruitful direction for future work is the investigation of meta-classification methods that can overcome the limitations of the plurality voting methods to achieve higher results. It should be noted that the human study described in this paper is a pilot. We plan on conducting a larger rating where we sample randomly across essays and include more experts for each L1.

[9] CHI: 50%, SPA: 46.7%, HIN: 33.3%, GER: 31.7%, ARA: 26.7%

[10] Our sample size is too small, but this is still suggestive.

[11] *e.g.* an NLI system with 70% accuracy against an Oracle baseline of 80% is relatively better compared to one with 74% accuracy against an Oracle baseline of 93%.

Acknowledgments

We would like to thank the three anonymous reviewers as well as our raters: Martin Chodorow, Carla Parra Escartin, Marte Kvamme, Aasish Pappu, Dragomir Radev, Patti Spinner, Robert Stine, Kapil Thadani, Alissa Vik and Gloria Zen.

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.

Serhiy Bykh and Detmar Meurers. 2014. Exploring Syntactic Features for Native Language Identification: A Variationist Perspective on Feature Encoding and Ensemble Optimization. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, pages 1962–1973, Dublin, Ireland, August. Dublin City University and Association for Computational Linguistics.

Serhiy Bykh, Sowmya Vajjala, Julia Krivanek, and Detmar Meurers. 2013. Combining Shallow and Linguistically Motivated Features in Native Language Identification. In *Proceedings of the Eighth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 197–206, Atlanta, Georgia, June. Association for Computational Linguistics.

Eugene Charniak and Mark Johnson. 2005. Coarse-to-fine n-best parsing and MaxEnt discriminative reranking. In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, pages 173–180. Association for Computational Linguistics.

Andrea Cimino, Felice Dell'Orletta, Giulia Venturi, and Simonetta Montemagni. 2013. Linguistic Profiling based on General–purpose Features and Native Language Identification. In *Proceedings of the Eighth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 207–215, Atlanta, Georgia, June. Association for Computational Linguistics.

Cyril Goutte, Serge Léger, and Marine Carpuat. 2013. Feature Space Selection and Combination for Native Language Identification. In *Proceedings of the Eighth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 96–100, Atlanta, Georgia, June. Association for Computational Linguistics.

Radu Tudor Ionescu, Marius Popescu, and Aoife Cahill. 2014. Can characters reveal your native language? a language-independent approach to native language identification. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1363–1373, Doha, Qatar, October. Association for Computational Linguistics.

Scott Jarvis, Yves Bestgen, and Steve Pepper. 2013. Maximizing Classification Accuracy in Native Language Identification. In *Proceedings of the Eighth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 111–118, Atlanta, Georgia, June. Association for Computational Linguistics.

Ludmila I Kuncheva, James C Bezdek, and Robert PW Duin. 2001. Decision templates for multiple classifier fusion: an experimental comparison. *Pattern Recognition*, 34(2):299–314.

Ludmila I Kuncheva, Christopher J Whitaker, Catherine A Shipp, and Robert PW Duin. 2003. Limits on the majority vote accuracy in classifier fusion. *Pattern Analysis & Applications*, 6(1):22–31.

Ludmila I Kuncheva. 2002. A theoretical study on six classifier fusion strategies. *IEEE Transactions on pattern analysis and machine intelligence*, 24(2):281–286.

Shervin Malmasi and Aoife Cahill. 2015. Measuring Feature Diversity in Native Language Identification. In *Proceedings of the Tenth Workshop on Innovative Use of NLP for Building Educational Applications*, Denver, Colorado, June. Association for Computational Linguistics.

Shervin Malmasi and Mark Dras. 2014. Language Transfer Hypotheses with Linear SVM Weights. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, Doha, Qatar, October. Association for Computational Linguistics.

Shervin Malmasi, Sze-Meng Jojo Wong, and Mark Dras. 2013. NLI Shared Task 2013: MQ Submission. In *Proceedings of the Eighth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 124–133, Atlanta, Georgia, June. Association for Computational Linguistics.

Christopher D Manning, Prabhakar Raghavan, and Hinrich Schütze. 2008. Evaluation in information retrieval. In *Introduction to Information Retrieval*, pages 151–175. Cambridge university press Cambridge.

Robi Polikar. 2006. Ensemble based systems in decision making. *Circuits and systems magazine, IEEE*, 6(3):21–45.

Rion Snow, Brendan O'Connor, Daniel Jurafsky, and Andrew Ng. 2008. Cheap and Fast – But is it Good? Evaluating Non-Expert Annotations for Natural Language Tasks. In *Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing*, pages 254–263, Honolulu, Hawaii, October. Association for Computational Linguistics.

Joel Tetreault, Daniel Blanchard, Aoife Cahill, and Martin Chodorow. 2012. Native Tongues, Lost and Found: Resources and Empirical Evaluations in Native Language Identification. In *Proceedings of COLING 2012*, pages 2585–2602, Mumbai, India, December. The COLING 2012 Organizing Committee.

Joel Tetreault, Daniel Blanchard, and Aoife Cahill. 2013. A Report on the First Native Language Identification Shared Task. In *Proceedings of the Eighth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 48–57, Atlanta, Georgia, June. Association for Computational Linguistics.