Twitter Language Identification Of Similar Languages And Dialects Without Ground Truth

Jennifer Williams and Charlie K. Dagli*
Human Language Technology Group
Massachusetts Institute of Technology, Lincoln Laboratory
244 Wood Street, Lexington, MA 02420, USA
{jennifer.williams,dagli}@ll.mit.edu

Abstract

We present a new method to *bootstrap filter* Twitter language ID labels in our dataset for automatic language identification (LID). Our method combines geo-location, original Twitter LID labels, and Amazon Mechanical Turk to resolve missing and unreliable labels. We are the first to compare LID classification performance using the MIRA algorithm and $langid.py$. We show classifier performance on different versions of our dataset with high accuracy using only Twitter data, without ground truth, and very few training examples. We also show how Platt Scaling can be use to calibrate MIRA classifier output values into a probability distribution over candidate classes, making the output more intuitive. Our method allows for fine-grained distinctions between similar languages and dialects and allows us to rediscover the language composition of our Twitter dataset.

1 Introduction

Every second, the Twitter microblogging webservice relays as many as 6,000[1] short written messages (less than 140 characters), called tweets, from people around the world. The tweets are created and viewed publicly by anyone with internet access. Tweets obtained from the Twitter API are tagged with metadata such as language ID and geo-location (Graham et al, 2014).

Currently there is a mismatch between the built-in language identification support provided by the Twitter API and the needs of the natural language processing (NLP) community. While there are around 7,000[2] human languages spoken today, only 34 of the most common languages are currently recognized and tagged by Twitter[3] using automatic methods for language identification (LID). In addition to Twitter's low-coverage of languages, Twitter's default language tags are not always accurate (Zubiaga et al, 2015; Lui and Baldwin, 2014; Bergsma et al, 2012) making it very challenging to obtain the necessary ground-truth for training a language classifier.

Twitter data is linguistically diverse and has tremendous global reach and influence. Discriminating languages and dialects automatically is a critical pre-processing step for more advanced NLP applications (Dagli et al, 2016). Heavy, worldwide use of Twitter has created a very rich landscape for developing NLP applications such as support for disaster relief (Sakaki et al., 2010; Kumar et al., 2011), sentiment analysis (Volkova et al., 2013), as well as recognizing named entities (Ritter et al., 2011) and temporal reasoning for events and habits (Williams and Katz, 2012).

In this work we show how geo-location can be used to identify the language of a tweet when appropriate language tags are seemingly incorrect, or absent. Specifically, we are interested in discriminating similar languages English, Malay and Indonesian (en, ms, id) as well as dialects of Spanish from Europe and Mexico $(es\text{-}ES, es\text{-}MX)$ and dialects of Portuguese from Europe and Brazil $(pt\text{-}PT, pt\text{-}BR)$. Language names are represented using the ISO-639-2 language codes and 2-letter country abbreviation added for dialects. The methods we present in this paper provide a fast, low-cost approach to filtering Twitter LID la-

* This material is based upon work supported by the Defense Advanced Research Projects Agency under Air Force Contract No. (FA8721-05-C-0002 and/or FA8702-15-D-0001). Any opinions, findings and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the Defense Advanced Research Projects Agency.

[1] http://www.internetlivestats.com/twitter-statistics/

[2] https://www.ethnologue.com/

[3] https://dev.twitter.com/web/overview/languages

Proceedings of the Fourth Workshop on NLP for Similar Languages, Varieties and Dialects, pages 73–83,
Valencia, Spain, April 3, 2017. ©2017 Association for Computational Linguistics

bels. It is very important to have data with reliable language labels because it allows us to make fine-grained distinctions between dialects and similar languages, in order to expand the linguistic scope of NLP applications.

This paper is organized as follows: Section 2 describes related work, Section 3 describes the data collection and preparation, Section 4 describes classification algorithms, Section 5 shows our re-annotation experiments and results, Section 6 presents results using Platt Scaling, and finally Section 7 is discussion and future work.

2 Related Work

Language identification has a rich history in natural language processing (Cavnar and Trenkle, 1994; Dunning, 1994). Recently, many different language combinations have appeared in benchmark shared tasks, most notably in the DSL (Discriminating Similar Languages) Shared Task 2014, 2015, and 2016 (Lui et al, 2014; Zampieri et al, 2014; Zampieri et al, 2015, Malmasi et al, 2016). In these shared-tasks the train/test data is not composed entirely of social media while simultaneously providing support for the languages and dialects that we are interested in. Additionally, English is sometimes used by Twitter users within the country geo-boundaries of Indonesia and Malaysia. Therefore we cannot rely on user profile settings as in previous work (Saloot et al., 2016), including Kevin Scannell's ongoing Indigenous Tweets Project[4] which relies on self-reported minority language usage but does not guarantee homogeneity of labeled language collections.

Ranaivo-Malançon (2006) was the first to work on Malay-Indonesian LID using n-gram profiling and other linguistic features. While their work capitalizes on nuanced linguistic differences between Malay and Indonesian, it does not address whether or not this technique can be expanded to include English, or dialect pairs, and the results for classifier accuracy are not reported. We are also interested in discriminating dialects of Spanish and Portuguese, as these are widely spoken languages with important dialect distinctions (Zampieri et al, 2016; Çöltekin and Rama, 2016).

The 2014 DSL Shared-Task was the first large-scale task for distinguishing between similar languages and dialects in a language group, including: Malay/Indonesian, Brazilian Portuguese/Portuguese, and Spanish/Mexican Spanish. The data for this shared-task, compiled by Tan et al (2014), was collected from the web, cleaned, and consists of 18,000 training sentences per language group. Performance results per language group are reported for the top 8 systems, with the best performing system, NRC-CNRC (Goutte et al, 2014), achieving overall accuracy between 91%-99% on the language groups that we are interested in. Our work is distinct from the DSL Shared-Tasks for language and dialect identification because we are interested in learning a classifier using only Twitter data, without ground truth, using very few training examples.

3 Data Collection

We collected tweets from Twitter using the 10% firehose that we obtained from GNIP[5] between January 2014 and October 2014. The 10% firehose is a real-time random sampling of all tweets as they are relayed through the Twitter webservice. As part of their service, GNIP provided a filtering with geo-tagging enabled, so that all of the tweets in our collection were geographically tagged with longitude and latitude, allowing us to pin-point the exact location of the tweet. Initially, we collected over 25.6 million tweets during that time period. In our collection, 24 languages were automatically identified by the Twitter API using the ISO-639-2 and ISO-639-3 language codes[6].

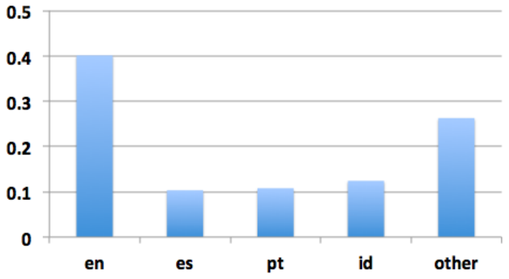

Figure 1: Twitter LID label composition (relative frequency) for our collected Twitter dataset

The most commonly occurring languages in our dataset were English, Spanish, Indonesian, and Portuguese. We note that our dataset did not contain any tweets initially identified as being in the Malay language. Figure 1 shows the distribution of languages relative to the overall collection. The

[4] http://indigenoustweets.com/

[5] https://gnip.com/

[6] https://dev.twitter.com/rest/reference/get/help/languages

language distribution in our data does not accurately represent the languages used on Twitter for two reasons: 1) Twitter's own language ID codes are not always accurate in identifying the language of a tweet, and 2) this distribution in Figure 1 represents 10% geo-enabled firehose from GNIP collected during a specific time period. Furthermore without adequate language ID technology and reliable language labels, the true distribution of languages on Twitter is not known with certainty.

4 Classification Algorithms

In this section we describe two classification algorithms that we used in our experiments. We compared performance of the MIRA algorithm with the popular pre-trained software called $langid.py$.

4.1 MIRA

Advances in statistical learning theory have made it possible to expand beyond binary classification with perceptrons (Rosenblatt, 1958) to multiclass online learners such as the Margin Infused Relaxed Algorithm (MIRA) from Crammer and Singer (2003). The MIRA algorithm is formulated as a multiclass classifier which maintains one prototype weight vector for each class. MIRA performs similar to Support Vector Machines (SVM) without batch training (Crammer et al, 2006).

For multiclass classification, MIRA is formulated as shown in equation (1):

$$c^* = \arg\max_{c \in \mathbf{C}} f_c(\mathbf{d}) \tag{1}$$

where

$$f_c(\mathbf{d}) = \mathbf{w} \cdot \mathbf{d} \tag{2}$$

and $\mathbf{w}$ is the weight vector which defines the model for class c. The output of the classifier, for each class, is the dot product between a document vector $\mathbf{d}$ and the weight vector for each class c, shown in equation (2). Therefore the predicted class is chosen by selecting the $argmax$. The values for each class, from equation (2) are neither normalized or scaled, and so they do not represent a probability distribution over candidate classes. We discuss this in greater depth in Section 6 with regard to calibrating the classifier output.

To train MIRA, we swept values for the margin slack (0.0005 to 0.00675) and number of training epochs (5 to 30). The value for training epochs denoted a hard-stop for training iterations and served as the stopping criterion. The feature vectors contained log-normalized frequency counts for word and character n-grams, with values for n swept separately for words (1 to 5) and characters (1 to 5), to allow various word and character-level n-gram combinations. After sweeping all possible feature combinations, we report experiment results based on the highest achieved overall accuracy. Words were defined by splitting on whitespace and we did not do any pre-processing or text normalization of the original tweets, similar to Lui and Baldwin (2014). For MIRA we used the open-source software suite called LLClass[7], which proved useful for other types of text categorization tasks (Shen et al, 2013).

4.2 langid.py

For comparison, we used the off-the-shelf tool *langid.py* from Lui and Baldwin (2012). This tool employs a multinomial näive Bayes classifier, and n-gram feature set. The n-gram features are selected using information gain to maximize information with respect to language while minimizing information with respect to data source. A pre-trained model also comes off-the-shelf and covers 97 languages, including the specific languages that we use for this work. At the time of this writing the pre-trained model does not include support for dialect distinction. While we did not sweep parameters for the *langid.py* software, as we wanted to evaluate off-the-shelf performance, we did use their built-in feature "label constraint" which restricts the multinomial distribution to a specified set of target labels, rather than all 97 supported languages. For example, with experiments involving English/Malay/Indonesian, we restricted the language label set to these three languages.

5 Re-Annotation Experiments

In this section we present our method to *bootstrap filter* our Twitter dataset to re-annotated data and arrive at ground truth labels. Our data processing technique is fast, easy, cheap, and independent of the classification algorithm. We also present classification results for each dataset using MIRA and $langid.py$ classifiers. All classification results are reported as the overall average accuracy with an 80/20 train/test split. Each experiment is based on N total tweets per target language and classes were stratified irrespective of tweet length.

[7] https://github.com/mitll/LLClass

5.1 Exp 1: Twitter Labels

First for Experiment 1, we used Twitter API labels as ground truth for language classification. Unfortunately, our dataset did not contain Twitter LID labels for Malay, or the Portuguese and Spanish dialects.

Languages	N/class	MIRA	langid.py
en, id	500	98.0	90.1
pt, es, en, id	500	93.5	85.95

Table 1: Exp 1 results using Twitter API language labels as ground truth

The performance shown for the English/Indonesian pair in Table 1 is competitive with the DSL Shared Task performance for this language pair (Zampieri et al, 2016). We also used Twitter labels to evaluate multiclass classification for pt, es, en, id and note that the MIRA classifier outperforms langid.py for this set.

5.2 Exp 2: Geo-Boundary Filtering

In Experiment 2, we filtered our Twitter dataset by establishing geo-bounding boxes to geographically define countries where the language of interest is suspected to be most prominent. For example, we used the country Malaysia as a representative geo-source for Malaysian tweets. We used a free website to set up the latitudinal and longitudinal geo-bounding boxes around the countries [8] and there are additional alternative websites to obtain similar geo-boundaries[9][10]. Each bounding box corner was defined by a latitude/longitude coordinate pair corresponding to SW, NW, SE, NE. Multiple bounding boxes were used for approximating the shape of each country and we made every effort to include major metropolitan cities within the bounds. In some cases, our bounding boxes were slightly overspecified and slightly underspecified depending on the geometric shape of the country as shown for Portugual in Figure 2.

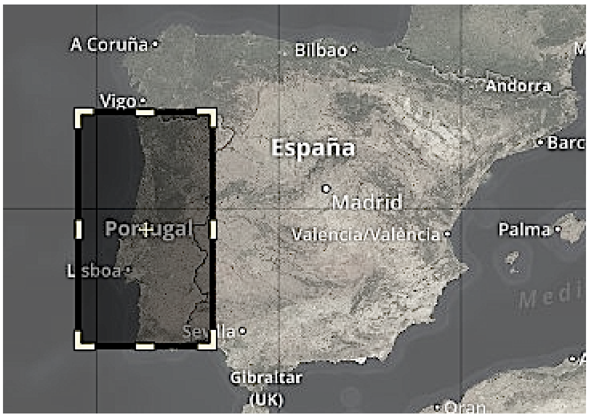

Figure 2: Example of geo-bounding box to identify tweets that originated from Portugal

We recognize that Twitter users in each of the geo-bounded countries are able to tweet in any language. Our data filtering method was based on the assumption that the majority of tweets from a country would be composed in that country's most common language. We calculated how frequently different Twitter API language labels occurred within the bounds of the target country define a target label *purity*, with respect to the expected majority language. This is the conditional probability of the target Twitter LID label occurring in the target country, shown in equation (3)

$$\mathbf{p}(\mathbf{label}|\mathbf{country}) = \frac{count_{label}}{count_{country}} \qquad (3)$$

Geo-Bound Country	Language	Label	Purity
Malaysia	Malaysian	ms	0%
Indonesia	Indonesian	id	63%
United States	English	en	85%
Portugal	Portuguese	pt	68%
Brazil	Portuguese	pt	71%
Spain	Spanish	es	72%
Mexico	Spanish	es	69%

Table 2: Twitter LID label purity within geographic country boundaries

The majority of tweets originating from Malaysia were tagged as id and en. We observed similar scarcity of Malay tweets in Twitter's publicly released language identification datasets [11]. In fact, Malay tweets make up less than 0.001% of Twitter's uniformly sampled dataset despite API support for Malay language identification. Our estimates of label purity, in addition to Twitter's dataset coverage of Malay, emphasize the persisting need for automatic language disambiguation. We compared classifier performance using geo-boundary as a stand-in for ground truth labels, and our results are shown in Table 3.

5.3 Exp 3: Geo Filtering + Twitter Labels

To generate ground truth in Experiment 3, we took the intersection of labels from geo-bounds and

[8] http://boundingbox.klokantech.com/

[9] http://www.naturalearthdata.com/

[10] https://help.openstreetmap.org/

[11] https://blog.twitter.com/2015/evaluating-language-identification-performance

Languages	N/class	MIRA	langid.py
en, id, ms	1000	80.8	54.2
en, id	1000	93.5	79.5
id, ms	1000	86.3	51.7
en, ms	1000	86.0	76.6
pt-PT, pt-BR	1000	75.0	–
es-ES, es-MX	1000	66.8	–
en, id, ms, pt-PT, pt-BR, es-ES, es-MX	1000	68.5	–

Table 3: Exp 2 results using geo-boundaries to represent ground truth LID labels (i.e. country labels = language labels)

original Twitter LID labels. For example, we extracted all tweets from Brazil that the Twitter API had labeled as *pt* for Portuguese, and re-labeled them as Brazilian Portuguese, *pt-BR*. We repeated the classification experiment using a separate subset of tweets and these new labels. As shown in Table 4, the classification results for MIRA in Experiment 3 are competitive with results from related benchmarking tasks, such as DSL 2016 (Malmasi et al, 2016).

Languages	N/class	MIRA	langid.py
en, id, ms	1000	85.5	60.7
en, id	1000	99.5	92.8
id, ms	1000	90.5	49.0
en, ms	1000	88.7	78.9
pt-PT, pt-BR	1000	80.5	–
es-ES, es-MX	1000	67.2	–
en, id, ms, pt-PT, pt-BR, es-ES, es-MX	1000	77.2	–

Table 4: Exp 3 results using combined geo-boundary definitions and Twitter LID labels

5.4 Exp 4: Mechanical Turk-Verified Labels

Finally, in Experiment 4 we further refined the ground truth labels obtained from earlier experiments. We verified the target language of tweets using Amazon Mechanical Turk Human Intelligence Tasks (HITs), using the same train/test data from Experiment 3 (before classification). Each HIT contained one tweet. We assigned 3 workers per HIT at the rate of $0.02 USD per HIT and the total cost for MTurk annotation in this work was $360.00 USD. In an effort to ensure that workers were qualified for the task, we allowed only workers who had an MTurk approval rating >95%, however we did not administer a language performance test in this work. To complete a HIT, workers selected one answer to a multiple-choice question, described below, and we did not inform workers that the text was from Twitter.

Instructions: *Please indicate which language the text is in. Some text snippets are full sentences while others are partial sentences or phrases. If the text contains more than one language, indicate that in your response. Note that you can ignore URLs, punctuation, and emoticons to decide the language. In order to be paid you must answer each question correctly.*

The authors would like to note that this final statement of the instructions to workers was to motivate them to complete the task meaningfully. All workers who completed tasks in the allotted time frame were paid automatically.

Workers were asked to select one of the following three statements, where language *X* the language label used for train/test in Experiment 3.

A1. The text is entirely composed in language *X*

A2. The text is composed in language *X* and at least one other language

A3. None of the text is composed in language *X*

Target	# HITs	A1	A2	A3
ms	900	**614**	205	81
id	912	**736**	158	18
pt-PT	904	**816**	66	22
pt-BR	874	**778**	66	30
es-ES	889	**845**	36	8
es-MX	838	**762**	72	4

Table 5: MTurk annotations per language

The annotation results of our MTurk experiment are shown in Table 5. Columns $A1$, $A2$, and $A3$ show the frequency that at least 2 of 3 human annotators agreed on the language condition. We began with 1000 tweets per language for annotation. If fewer than 2 annotators agreed on a condition, the HIT for that tweet was not counted in this analysis. This method of filtering both reduced the amount of data and simultaneously increased our confidence in the labels as ground truth. Our analysis with MTurk shows that the majority of train/test tweets in Experiment 3 were composed entirely in the target language *X*, with some instances of code-mixing of two or more languages. We used the tweets verified by Mechanical Turk to learn another set of classifiers for Experiment 4, shown in Table 6. The number of tweets per language class is reduced in this dataset, because we used only tweets verified as being 100% in the target language (column A1 from Table 5). While the classifier accuracy between Experiment 3 and Experiment 4 is similar, we believe that the performance is lower in Experiment 4 because of fewer training examples.

Languages	N/class	MIRA	langid.py
en, id, ms	600	92.5	63.8
id, ms	600	87.9	53.4
pt-PT, pt-BR	750	79.6	–
es-ES, es-MX	750	70.3	–
en, id, ms, pt-PT, pt-BR, es-ES, es-MX	1000	79.3	–

Table 6: Exp 4 results using MTurk verified labels

6 MIRA Classifier Calibration

Classifier output scores for MIRA and similar algorithms, like SVM, do not correspond to probabilities. For example, the raw score cannot guide the researcher or end user to knowing if a tweet is 80% likely to be English or 50% likely to be English. The ability to transform raw classifier scores into probabilities is very important if the technology is to be used as a consumable for text analytics or as part of an advanced NLP pipeline. In this section, we show how we calibrated scores using output from the MIRA classifier for 3 different experiments from Section 5. As with many classifiers, the raw score output can be difficult to interpret intuitively since the scalar values for each class do not represent a probability distribution over the classes. We used a technique called Platt Scaling, which learns logistic regression from the raw score output of the MIRA classifier. The Platt Scaling technique provides us with a probability distribution on classes and is easy to train and test. For our reliability plots and calibration, we used classifier output scores of test sets from experiments described in Section 5. For the purpose of brevity, we describe classifier scaling using results for one language pair: Indonesian and Malay.

6.1 Score Reliability Plots

Reliability plots show how well a classifier's output is calibrated when the true probability distribution for classes is not known (Niculescu-Mizil and Caruana, 2005; Zadronzy and Elkan, 2002; DeGroot and Feinberg, 1983). For this visualization, the classifier output scores, also called *predicted values*, are normalized between 0 and 1 and then values are binned into 10 bins. The values plotted are the binned scores s versus the conditional probability of correct class prediction given the score, $P(c|s(x) = s)$. A classifier that is well-calibrated will have values that fall close to the diagonal line $x = y$.

We normalized the raw classifier output values so that the scores fell between 0 and 1, using exponent-normalization as in equation (4), for a given tweet:

$$\mathbf{exp_c} = \frac{e^{s_c}}{\sum_{c \in \mathbf{C}} e^{s}} \tag{4}$$

where $\mathbf{exp_c}$ is the normalized score for class c, and s_c is the raw classifier output score for class c. We further divide by the sum, so that the normalized class scores for a given tweet sum to 1.

We created reliability plots for the id, ms prediction task from Experiments 2, 3, and 4. Figures 3 - 11 show the histogram distribution of normalized classifier scores with the corresponding reliability plot. Recall that each experiment was based on different kinds of ground truth. All of the reliability plots before Platt-scaling exhibit a sigmoidal distribution. The prevalence of our observed sigmoidal distribution is similar to findings from Niculescu-Mizil and Caruana (2005), who noted this shape for learning algorithms based on maximum margin methods, such as SVM. MIRA and SVM both use maximum margin principles and are known to perform similarly, with the additional benefit that MIRA does not require batch training because it is online (Crammer et al., 2006)

6.2 Platt Scaling

Platt scaling uses logistic regression to learn a mapping between classifier output scores and probability estimates (Platt, 1999). The output of Platt scaling is a probability distribution over candidate classes, rather than raw scores from the classifier which are often non-intuitive and difficult to interpret (Zadronzy and Elkan, 2002). Platt scaling is traditionally used in binary problems, and adapted to multiclass problems by developing the original classifier as an ensemble of one-vs-all classifiers, then fitting logistic regression for each binary model (Niculescu-Mizil and Caruana, 2005; Zadronzy and Elkan, 2002). We trained and tested logistic regression on a binary class problem with MIRA output using the Logistic Regression library in Python Scikit-Learn, which is designed to handle binary, one-vs-rest, and multinomial logistic regression (Pedregosa et al, 2011).

To build and evaluate logistic regression, we used the test data from our previous experiments, as in Section 6.1, and divided that data into train and test sets with an 80/20 split. For example, the test data from Experiment 2 for id, ms consisted

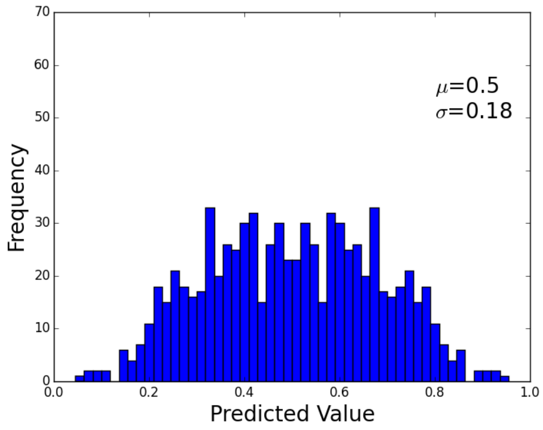

Figure 3: Geo-only, normalized scores

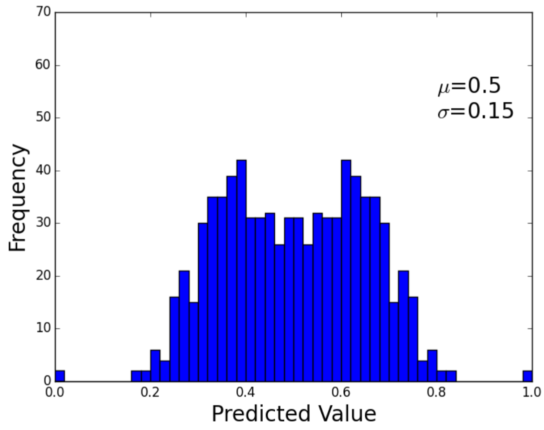

Figure 6: Geo+Twitter, normalized scores

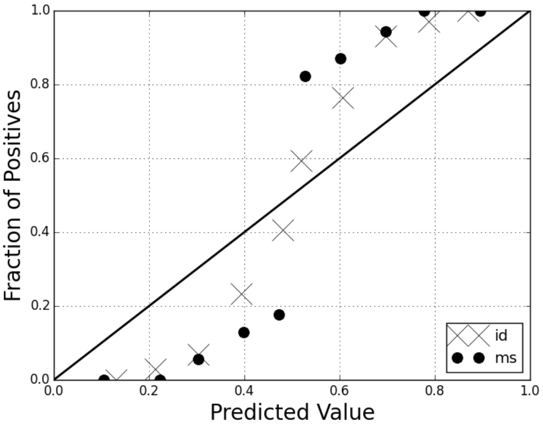

Figure 4: Geo-only, reliability plot

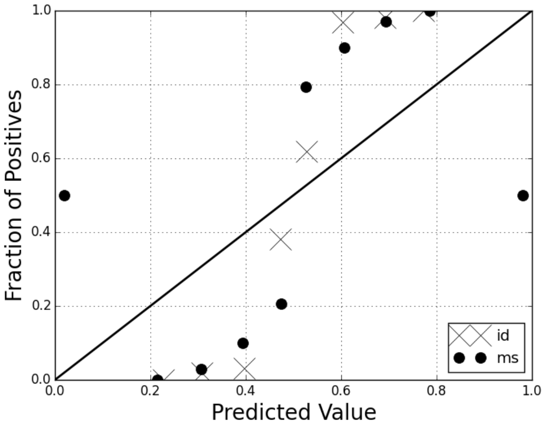

Figure 7: Geo+Twitter, reliability plot

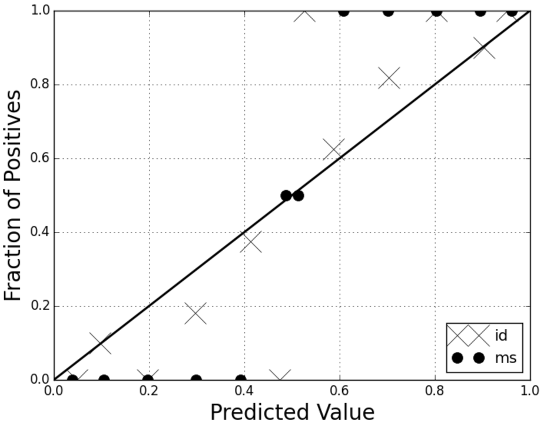

Figure 5: Geo-only, with Platt-scaling

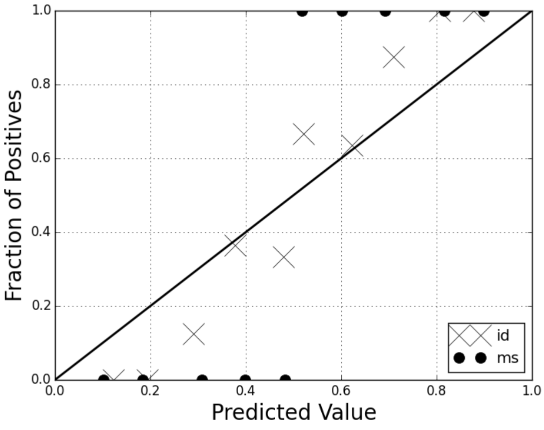

Figure 8: Geo+Twitter, with Platt-scaling

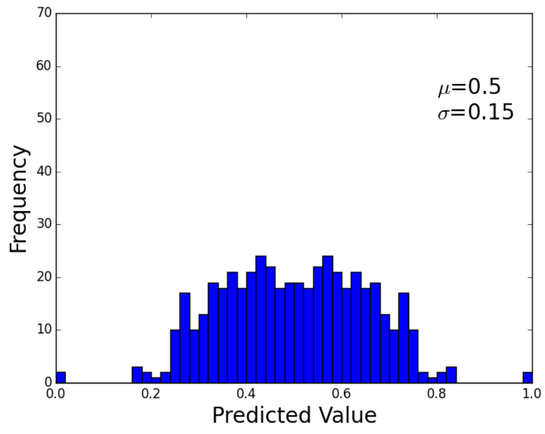

Figure 9: MTurk, normalized scores

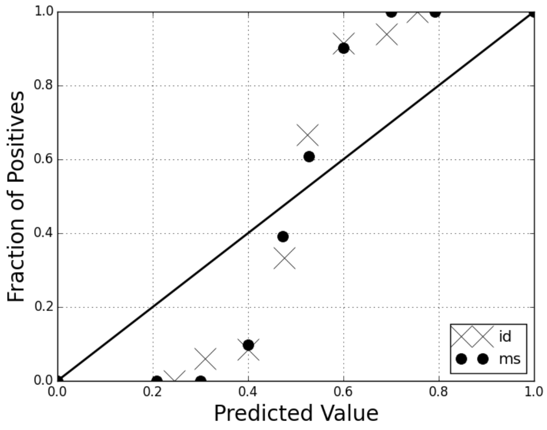

Figure 10: MTurk, reliability plot

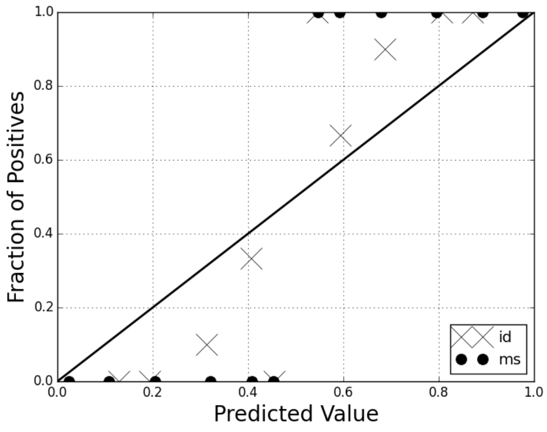

Figure 11: MTurk, with Platt-scaling

of 200 tweets per class. To do Platt scaling on this dataset, we used 160 tweets per class for training and 40 per class for testing.

With each of the datasets, Platt-scaling tends to affect calibration probabilities for Indonesian tweets more than for Malay tweets. This is observed as Indonesian data points are closer to the diagonal line. At the same time, the Platt-scaling plots also reveal that predicted values, especially for Malay, are pushed closer to 0 and 1. For example, logistic regression will always correctly predict ms for Malay, when the probability of Malay is > 0.5, but not for Indonesian. This could indicate a need for further data purification.

We examined the accuracy of logistic regression, where the predicted class is taken to be the $argmax$ class probability. In Figure 12, the overall classification accuracy on each dataset is similar for MIRA with and without Platt-scaling. We think this is an important finding because it shows that LID classifier output can be converted into probability distributions without loss of accuracy.

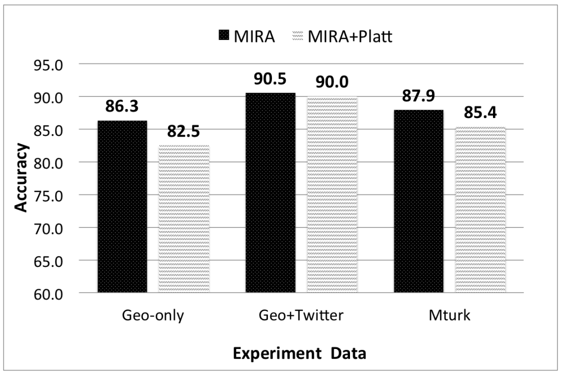

Figure 12: MIRA and Platt-scaling Test Accuracy

What do scores look like for a given tweet? In Table 7 we show raw classifier output scores, normalized scores, and probabilities from Platt-scaling for the following Malay tweet:

Malay: *Nak tengok wayang. Rindu tempat kerja. Hehehe*

English[12]**:** *Want to see a movie. Miss work. hehehe*

	ma	id
Raw scores	**0.514**	-0.514
Exponent Normalized	**0.737**	0.263
Platt + Exponent Normalized	**0.535**	0.465

Table 7: Score distribution for Malay tweet

[12] Translation obtained from https://translate.google.com/

The raw output scores from MIRA, while clearly separating binary classes, are not easily interpreted as a measure of certainty or probability. While the exponent normalized scores do sum to 1, and appear to situate probability mass towards the predicted class, it is not a true probability. The probabilities that are output during Platt-scaling are true probabilities and this method preserves the original MIRA classifier accuracy, thus it is a valid and meaningful technique, especially when language ID is a consumable pre-processing technology for NLP pipelines.

7 Discussion and Future Work

In this work, we showed that geo-bounding combined with "best-guess" language labels can be used to annotate language labels on easily confused language pairs and dialects, when ground truth is unreliable. In each experiment, we showed how our data purification method resulted in increasing accuracy and classifier performance for both classifiers, MIRA and *langid.py*. Further, our method to purify language labels is easy to implement for tweets that are geo-tagged with latitude and longitude. Once a model has been learned from geo-tagged tweets, the model can also be used for tweets that are not geo-tagged.

We uncovered hidden Malay tweets in our dataset with high accuracy. We also showed that MIRA is useful for LID, with performance accuracy near state-of-the-art on very few training examples without pre-processing or text cleaning. While previous work has shown that Malay/Indonesian can be learned using 18,000 training sentences with accuracy as high as 99.6% (Goutte et al., 2014), our result of 90.5% trained on 1600 tweets is competitive with previous work. We believe performance will further increase as more training examples are added with high confidence ground truth labels. Using geo-bounding, we were also able to separate dialects of Spanish and Portuguese to achieve finer-grained distinctions at the dialect level, which the Twitter API does not currently provide.

The highest weighted MIRA n-gram features correspond to high-frequency characters in each target language, suggesting that MIRA is learning features of languages and not Twitter artifacts (URLs, hashtags, @mentions, emoticons, etc).

In future work, we want to explore other easily confused language pairs, such as Ukrainian and Russian. Also, since MIRA is well-formulated for multiclass classification, we are interested in seeing how well it performs on a large multi-language dataset that includes several easily confused language pairs. Sometimes a single tweet will be written in more than one language, for example with code-switching or code-mixing (Barman et al, 2014). We are especially interested in adapting the MIRA classifier for code-switching and language segmentation problems. In the case of code-switching, it may be possible to utilize raw scores from classifier output or the results of Platt-scaling to construct a model that predict language mixture in a single utterance.

References

Ustab Barman, Amitava Das, Joachim Wagner, and Jennifer Foster. "Code Mixing: A Challenge for Language Identification in the Language of Social Media". In Proceedings of *First Workshop on Computational Approaches to Code Switching, Empirical Mdethods in Natural Language Processing (EMNLP)*, Doha, Qatar, (2014): 13-23.

Shane Bergsma, Paul McNamee, Mossaab Bagdouri, Clayton Fink, and Theresa Wilson. "Language Identification for Creating Language-Specific Twitter Collections." In Proceedings of *North American Chapter of the Association for Computational Linguistics (NAACL-HLT)* 2012.

William B. Cavnar and John M. Trenkle. "N-gram Based Text Categorization". *Ann Arbor, MI, 48113(2)*, 1994, 161-175.

Çagri Çöltekin, and Taraka Rama. "Discriminating similar languages: experiments with linear SVMs and neural networks". In Proceedings of *3rd Workshop on Language Technology for Closely Related Languages, Varieties and Dialects (VarDial)*, Osaka, Japan. 2016.

Koby Crammer and Yoram Singer. "Ultraconservative Online Algorithms for Multiclass Problems". In *Journal of Machine Learning Research*, 2003, Jan(3): 951-991.

Koby Crammer, Ofer Dekel, Joseph Keshet, Shai Shalev-Schwartz, and Yoram Singer. "Online Passive-Aggressive Algorithms". In *Journal of Machine Learning Research*, 2006, Mar(7): 551-585.

Charlie K. Dagli, William M. Campbell, Lin Li, Jennifer Williams, Kelly Geyer, Gordon Vidaver, Joel Acevedo-Aviles, Esther Wolf, Jonathan Taylor, Joseph P. Campbell. "LLTools: Machine Learning for Human Language Processing. In Proceedings of *Neural Information Processing Systems (NIPS) Workshop on Machine Learning Systems*, Barcelona, Spain, December 2016.

Morris H. DeGroot and Stephen E. Feinberg. "The comparison and evaluation of forecasters.". The Statistician (1983): 12-22.

Ted Dunning. "Statistical Identification of Language". Computing Research Laboratory, New Mexico State University, 1994, 10-03.

Cyril Goutte, Serge Léger, Marine Carpuat. "The NRC System for Discriminating Similar Languages". In Proceedings of *First Workshop on Applying NLP Tools to Similar Languages, Varieties, and Dialects*, 139-145, Dublin, Ireland, August 2014.

Cyril Goutte, Serge Lger, Shervin Malmasi and Marcos Zampieri. "Discriminating Similar Languages: Evaluations and Explorations". In Proceedings of *10th International Conference on Language Resources and Evaluation (LREC)*, 2016.

Mark Graham, Scott A. Hale, and Devin Gaffney "Where in the world are you? Geolocation and language identification in Twitter". *The Professional Geographer*, 66, no. 4 (2014): 568-578.

Shamanth Kumar, Geoffrey Barbier, Mohammad Ali Abbasi, and Huan Liu. "TweetTracker: An Analysis Tool for Humanitarian and Disaster Relief." In Proceedings of *International Conference on Weblogs and Social Media (ICWSM)*, AAAI, 2011.

Marco Lui and Tim Baldwin. "langid. py: An off-the-shelf language identification tool.". In Proceedings of *ACL 2012 system demonstrations*, pp. 25-30. Association for Computational Linguistics, 2012.

Marco Lui and Tim Baldwin. "Accurate Language Identification of Twitter Messages". In Proceedings of *Joint Workshop on Language Technology for Closely Related Languages, Varieties and Dialects (LT4VarDial)*, 2015, 35-43.

Marco Lui, Ned Letcher, Oliver Adams, Long Duong, Paul Cook, Timothy Baldwin. "Exploring methods and resources for discriminating similar languages". In Proceedings of *First Workshop on Applying NLP Tools to Similar Languages, Varieties and Dialects*, pp. 129-138. 2014.

Shervin Malmasi and Mark Dras "Language Identification Using Classifier Ensembles". In Proceedings of *Fifth Workshop on Language Analysis for Social Media (LASM), European Association for Computational Linguistics (EACL)*, Gothenburg, Sweden, 2014: 17-25.

Shervin Malmasi, Marcos Zampieri, Nikola Ljubešic, Preslav Nakov, Ahmed Ali, and Jörg Tiedemann. "Discriminating between similar languages and arabic dialect identification: A report on the third dsl shared task". In Proceedings of *3rd Workshop on Language Technology for Closely Related Languages, Varieties and Dialects (VarDial)*, Osaka, Japan. 2016.

Alexandru Niculescu-Mizil, and Rich Caruana. "Predicting good probabilities with supervised learning". In Proceedings of *22nd International Conference on Machine learning (ICML)*, Opp. 625-632. ACM, 2005.

Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel et al. "Scikit-learn: Machine learning in Python". In *Journal of Machine Learning Research*, 12, no. Oct (2011): 2825-2830.

John Platt. "Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods". In *Advances in large margin classifiers*, 10.3 (1999): 61-74.

Bali Ranaivo-Malançon. "Automatic Identification of Close Languages - Case Study: Malay and Indonesian." In *ECTI Transaction on Computer and Information Technology*, 2006, 2(2): 126-133.

Alan Ritter, Sam Clark, and Oren Etzioni. "Named Entity Recognition in Tweets: An Experimental Study." In Proceedings of *Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2011, 1524-1534.

Frank Rosenblatt. "The Perceptron: A Probabilistic Model For Information Storage and Organization in the Brain". In *Psychological Review*, 1958, 65(6): 386.

Takeshi Sakaki, Makoto Okazaki, and Yutaka Matsuo. "Earthquake Shakes Twitter Users: Real-Time Event Detection by Social Sensors." In Proceedings of the *19th International Conference on World Wide Web*, Association for Computing Machinery, 2010, 851-860.

Mohammad Arshi Saloot, Norisma Idris, AiTi Aw, and Dirk Thorleuchter. "Twitter Corpus Creation: The Case Of A Malay Chat-Style Text Corpus (MCC)". *Digital Scholarship in the Humanities* 2016, 31(2), 227-243.

Wade Shen, Jennifer Williams, Tamas Marius, and Elizabeth Salesky. "A language-independent approach to automatic text difficulty assessment for second-language learners". In Proceedings of *Second Workshop on Predicting and Improving Text Readability for Target Reader Populations (PITR), Association for Computational Linguistics (ACL)* 2013, 30-38.

Thamar Solorio, Elizabeth Blair, Suraj Maharjan, Steven Bethard, Mona Diab, Mahmoud Gohneim, Abdelati Hawwari, Fhad AlGhamdi, Julia Hirschberg, Alison Chang, and Pascale Fung. "Overview for the First Shared Task on Language Identification in Code-Switched Data". In Proceedings of *First Workshop on Computational Approaches to Code Switching, Empirical Methods in Natural Language Processing (EMNLP)* 2014, 62-72.

Liling Tan, Marcos Zampieri, Nikola Ljubešic, and Jörg Tiedemann. “Merging comparable data sources for the discrimination of similar languages: The DSL corpus collection”. In Proceedings of *7th Workshop on Building and Using Comparable Corpora (BUCC)*, pp. 11-15. 2014.

Svitlana Volkova, Theresa Wilson, and David Yarowsky. “Exploring Demographic Language Variations to Improve Multilingual Sentiment Analysis in Social Media”. In Proceedings of *Empirical Methods in Natural Language Processing (EMNLP)*, 2013, 1815-1827.

Jennifer Williams and Graham Katz. “Extracting and modeling durations for habits and events from Twitter.” In Proceedings of *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Short Papers-Volume 2*, Association for Computational Linguistics, 2012.

Yin-Lai Yeong and Tien-Ping Tan. “Language Identification of Code Switching Sentences and Multilingual Sentences of Under-Resourced Languages by Using Multi-Structural Word Information”. In Proceedings of *INTERSPEECH* 2014, 3052-3055.

Bianca Zadronzy and Charles Elkan. “Transforming classifier scores into accurate multiclass probability estimates.”. In Proceedings of *Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* pp. 694-699. ACM, 2002.

Marcos Zampieri and Binyam Gebrekidan Gebre. “Automatic identification of language varieties: The case of Portuguese”. In Proceedings of *KONVENS2012 - The 11th Conference on Natural Language Processing*. Österreichischen Gesellschaft für Artificial Intelligende (ÖGAI), 2012, 233-237.

Marcos Zampieri, Liling Tan, Nikola Ljubešic, and Jörg Tiedemann. “A report on the DSL shared task 2014.” In Proceedings of *First Workshop on Applying NLP Tools to Similar Languages, Varieties and Dialects* 2014, 58-67.

Marcos Zampieri, Liling Tan, Nikola Ljubešic, and Jörg Tiedemann. “Overview of DSL Shared Task 2015”. In Proceedings of *Joint Wokrshop on Closely Related Languages, Varieties, and Dialects* 2015.

Marcos Zampieri, Shervin Malmasi, Octavia-Maria Sulea and Liviu P. Dinu. “A Computational Approach to the Study of Portuguese Newspapers Published in Macau”. In Proceedings of *Workshop on Natural Language Processing Meets Journalism (NLPMJ)* 2016, 47-51.

Arkaitz Zubiaga, Iñaki San Vincente, Pablo Gamallo, José Ramon Pichel, Iñaki Alegria, Nora Aranberri, Aitzol Ezeiza, and Víctor Fresno. “TweetLID: A Benchmark For Tweet Language Identification.” In Proceedings of *Language Resources and Evaluation Conference (LREC)* 2015: 1-38.