Accurate Language Identification of Twitter Messages

Marco Lui and **Timothy Baldwin**
NICTA VRL
Department of Computing and Information Systems
The University of Melbourne, VIC 3010, Australia
mhlui@unimelb.edu.au, tb@ldwin.net

Abstract

We present an evaluation of "off-the-shelf" language identification systems as applied to microblog messages from Twitter. A key challenge is the lack of an adequate corpus of messages annotated for language that reflects the linguistic diversity present in Twitter. We overcome this through a "mostly-automated" approach to gathering language-labeled Twitter messages for evaluating language identification. We present the method to construct this dataset, as well as empirical results over existing datasets and off-the-shelf language identifiers. We also test techniques that have been proposed in the literature to boost language identification performance over Twitter messages. We find that simple voting over three specific systems consistently outperforms any specific system, and achieves state-of-the-art accuracy on the task.

1 Introduction

Twitter[1] has captured the attention of various research communities as a potent data source, because of the immediacy of the information presented, the volume and variability of the data contained, the potential to analyze networking effects within the data, and the ability to (where GPS data is available) geolocate messages (Krishnamurthy et al., 2008). Although individual messages range from inane through mundane right up to insane, the aggregate of these messages can lead to profound insights in real-time. Examples include real-time detection of earthquakes (Sakaki et al., 2010), analysis of the location and prevalence of flu epidemics (Lampos et al., 2010; Culotta, 2010), news event detection (Petrović et al., 2010), and prediction of sporting match outcomes (Sinha et al., 2013).

Text analysis of social media has quickly become one of the "frontier" areas of Natural Language Processing (NLP), with major conferences opening entire tracks for it in recent years. The challenges in NLP for social media are many, stemming primarily from the "noisy" nature of the content. Research indicates that English Twitter in particular is more dissimilar to the kinds of reference corpora used in NLP to date, compared to other forms of social media such as blogs and comments (Baldwin et al., 2013). This has led to the development of techniques to "normalize" Twitter messages (Han et al., 2013), as well as Twitter-specific approaches to conventional NLP tasks such as part-of-speech tagging (Gimpel et al., 2011) and information extraction (Bontcheva et al., 2013). Even so, a precondition of NLP techniques is that the language of the input data is known, and this has led to interest in "language identification" (LangID) of Twitter messages. Research has shown that "off-the-shelf" LangID systems appear to perform fairly well on Twitter (Lui and Baldwin, 2012), but Twitter-specific systems seem to perform better (Carter et al., 2013; Tromp and Pechenizkiy, 2011; Bergsma et al., 2012; Goldszmidt et al., 2013).

Twitter recognizes the utility of language metadata in enabling new applications, and as of March 2013 includes language predictions with results from its API (Roomann-Kurrik, 2013). These predictions are not perfect (see Section 3.2), and at time of writing do not cover some languages (e.g. Romanian). Furthermore, some research groups

[1] http://www.twitter.com

Lui, Marco and Timothy Baldwin (to appear) Accurate Language Identification of Twitter Messages, In Proceedings of the EACL 2014 Workshop on Language Analysis in Social Media (LASM 2014), Gothenburg, Sweden.

have collected a substantial cache of Twitter data from before the availability of built-in predictions. Motivated by the need to work with monolingual subsets of historical data, we investigate the most practical means of carrying out LangID of Twitter messages, balancing accuracy with ease of implementation. In this work, we present an evaluation of "off-the-shelf" language identifiers, combined with techniques that have been proposed for boosting accuracy on Twitter messages.

A major challenge that we have had to overcome is the lack of annotated data for evaluation. Bergsma et al. (2012) point out that in LangID research on microblog messages to date, only a small number of European languages has been considered. Baldwin and Lui (2010) showed that, when considering full documents, good performance on just European languages does not necessarily imply equally good performance when a larger set of languages is considered. This does not detract from work to date on European languages (Tromp and Pechenizkiy, 2011; Carter et al., 2013), but rather highlights the need for further research on LangID for microblog messages.

Manual annotation of Twitter messages is a challenging and laborious process. Furthermore, Twitter is highly multilingual, making it very difficult to obtain annotators for all of the languages represented. Previous work has attempted to crowdsource part of this process (Bergsma et al., 2012), but such an approach requires substantial monetary investment, as well as care in ensuring the quality of the final annotations. In this paper, we propose an alternative, "mostly-automated" approach to gathering language-labeled Twitter messages for evaluating LangID. A corpus constructed by direct application of automatic LangID to Twitter messages would obviously be unsuitable for evaluating the accuracy of LangID tools. Even with manual post-filtering, the remaining dataset would be biased towards messages that are easy for automated systems to classify correctly. The novelty of our approach is to leverage user identity, allowing us to construct a corpus of language-labeled Twitter messages without using automated tools to determine the languages of the messages. This quality makes the corpus suitable for use in the evaluation of automated LangID of Twitter messages.

Our main contributions are: (1) we provide a manually-labeled dataset of Twitter messages, adding Chinese (zh) and Japanese (ja) to the set of Twitter messages with human annotation for language; (2) we provide a second dataset constructed using a mostly-automated approach, covering 65 languages; (3) we detail the method for constructing the dataset; (4) we provide a comprehensive empirical evaluation of the accuracy of off-the-shelf LangID systems on Twitter messages, using published datasets in addition to the new datasets we have introduced; and (5) we discuss and evaluate a simple voting-based ensemble for LangID, and find that it outperforms any individual system to achieve state-of-the-art results.

2 Background

LangID is the problem of mapping a document onto the language(s) it is written in. The best-known technique classifies documents according to rank order statistics over character n-gram sequences between a document and a global language profile (Cavnar and Trenkle, 1994). Other statistical approaches applied to LangID include Markov models over n-gram frequency profiles (Dunning, 1994), dot products of word frequency vectors (Darnashek, 1995), and string kernels in support vector machines (Kruengkrai et al., 2005). In contrast to purely statistical methods, linguistically-motivated models for LangID have also been proposed, such as the use of stop word lists (Johnson, 1993), where a document is classified according to its degree of overlap with lists for different languages. Other approaches include word and part-of-speech (POS) correlation (Grefenstette, 1995), cross-language tokenization (Giguet, 1995) and grammatical-class models (Dueire Lins and Gonçalves, 2004).

LangID of short strings has attracted recent interest from the research community. Hammarstrom (2007) describes a method that augments a dictionary with an affix table, and tests it over synthetic data derived from a parallel bible corpus. Ceylan and Kim (2009) compare a number of methods for identifying the language of search engine queries of 2 to 3 words. They develop a method which uses a decision tree to integrate outputs from several different LangID approaches. Vatanen et al. (2010) focus on messages of 5–21 characters, using n-gram language models over data drawn from UDHR in a naive Bayes classifier. Carter et al. (2013) focus specifically on LangID in Twitter messages by augmenting stan-

dard methods with LangID priors based on a user's previous messages and the content of links embedded in messages; this is also the method used in TwitIE (Bontcheva et al., 2013). Tromp and Pechenizkiy (2011) present a method for LangID of short text messages by means of a graph structure, extending the standard 'bag' model of text to include information about the relative order of tokens. Bergsma et al. (2012) examine LangID for creating language-specific twitter collections, finding that a compressive method trained over out-of-domain data from Wikipedia and standard text corpora performed better than the off-the-shelf language identifiers they tested. Goldszmidt et al. (2013) propose a method based on rank-order statistics, using a bootstrapping process to acquire in-domain training data from unlabeled Twitter messages. Recent work has also put some emphasis on word-level rather than document-level LangID (Yamaguchi and Tanaka-Ishii, 2012; King and Abney, 2013), including research on identifying the language of each word in multilingual online communications (Nguyen and Dogruoz, 2013; Ling et al., 2013). In this paper, we focus on monolingual messages, as despite being simpler, LangID of monolingual Twitter messages is far from solved.

In Section 1, we discussed some work to date on LangID on Twitter data. Some authors have released accompanying datasets: the dataset used by Tromp and Pechenizkiy (2011) was made available in its entirety, consisting of 9066 messages in 6 Western European languages. Other authors have released message identifiers with associated language labels, including Carter et al. (2013), with 5000 identifiers in 5 Western European languages, and Bergsma et al. (2012), providing 13190 identifiers across 9 languages from 3 language families (Arabic, Cyrillic and Devanagari). To date, only the dataset of Tromp and Pechenizkiy (2011) has been used by other researchers (Goldszmidt et al., 2013). With the kind co-operation of the authors, we have obtained the full datasets of Carter et al. (2013) and Bergsma et al. (2012), allowing us to present the most extensive empirical evaluation of LangID of Twitter messages to date. However, the total set of languages covered is still very small. In Section 2.1, we present our own manually-annotated dataset, adding Chinese (zh) and Japanese (ja) to the languages that have manually-annotated data.

	English	Chinese	Japanese
Initial	0.906	0.773	0.989
Post-review	0.930	0.916	0.998

Table 1: Inter-annotator agreement measured using Fleiss' kappa (Fleiss, 1971) over annotations for TWITTER.

2.1 Manual annotation of ZHENJA

A manual approach to constructing a LangID dataset from Twitter data is difficult due to the wide variety of languages present in Twitter — Bergsma et al. (2012) report observing 65 languages in a 10M message sample, and Baldwin et al. (2013) report observing 97 languages in a 1M message sample. While this is encouraging in terms of sourcing data for lower-density languages, the distribution of languages is Zipfian, and the relative proportion of data in most languages is very small. Manually retrieving all available messages in a language would require a native speaker to view and reject a huge number of messages in other languages in order to collect the small number that are written in the target language. We initially attempted this, building ZHENJA, a dataset derived from a set of 5000 messages randomly sampled from a larger body of 622192 messages collected from the Twitter streaming API over a single 24-hour period in August 2010. The messages are a 1% representative sample of the total public messages posted on that day. Each of the 5000 selected messages was annotated by speakers of three languages: English, Japanese and Mandarin Chinese. For each message, three annotators were asked if the message contained any text in languages which they spoke, as well as if it appeared to contain text in (unspecified) languages which they did not speak. The latter label was introduced in order to make a distinction between text in languages not spoken by our annotators (e.g. Portuguese) and text with no linguistic content (e.g. URLs). After the initial annotation, annotators were asked to review messages where there was disagreement, and messages were assigned labels given by a majority of annotators post-review. Inter-annotator agreement (Table 1) is strong for the task: only 20 out of 5000 messages have less than 80% majority in annotations. In many instances, the disagreement was due to messages consisting entirely of a short sequence of hanzi/kanji, which both Chinese and Japanese speakers recognized as valid (these messages are

excluded from our set of labeled messages). Out of the 5000 messages, 1953 (39.1%) were labeled as English, 16 were labeled as Chinese (0.3%) and 1047 were labeled as Japanese (20.9%), for a total of 3016 labeled messages.

A total of 8 annotators each invested 2–4 hours in this annotation task, and the final dataset only covers 3 languages (which includes the top-2 highest-density languages in Twitter). Obviously, constructing a dataset of language-labeled Twitter messages is a labor-intensive process, and the lower density the language, the more expensive our methodology becomes (as more and more documents need to be looked over to find documents in the language of interest). Ideally, we would like to be able to use some form of automated LangID to accelerate the process without biasing the data towards easy-to-classify messages.

2.2 A broad-coverage Twitter corpus

Based on our discussion so far, our desiderata for a LangID dataset of Twitter messages are as follows: (1) achieve broader coverage of languages than existing datasets; (2) minimize manual annotation; and (3) avoid bias induced by selecting messages using LangID. (2) and (3) may seem to be conflicting objectives, but we sidestep the problem by first identifying monolingual users, then produce a dataset by sampling messages by these users from a held-out collection.

The overall workflow for constructing a dataset is summarized in Algorithm 1. For each user we consider, we divide all their messages into two disjoint sets. One set (M_u^{main}) is used to determine the language(s) spoken by the user. If only one language is detected, the user is added to a pool of candidate users (U^{accept}). A fixed number of users is sampled for each language (U^{sample}), and for each sampled user a fixed number of messages is sampled from the held-out set ($M_u^{heldout}$) and added to the final dataset. We sample a fixed number of users per language to limit the amount of data in the more-frequent languages, and we only sample a small number of messages per user in order to avoid biasing the dataset towards the linguistic idiosyncrasies of any specific individual. For both sampling steps, if the number of items available is less than the number required, all the available items are returned.

Algorithm 1 uses automated LangID to detect the language of messages in M_u^{main} (line 8). The accuracy of this identifier is not critical, as any misclassifications for a monolingual user would cause them to be rejected, as they would appear multilingual. Hence, the risk of false positives at the user-level LangID is very low. However, incorrectly rejecting users reduces the pool of data available for sampling, so a higher-accuracy solution is preferable. We compared the performance of 8 off-the-shelf (i.e. pre-trained) LangID systems to determine which would be the most suitable for this role.

Algorithm 1 Procedure for building a Twitter LangID dataset

1: $U \leftarrow$ active users
2: $L^{accept}, M^{accept}, U^{accept} \leftarrow \{\}, \{\}, \{\}$
3: **for each** $u \in U$ **do**
4: $\quad M_u \leftarrow$ all messages by user u
5: $\quad M_u^{main}, M_u^{heldout} \leftarrow$ RandomSplit(M_u)
6: $\quad L_u \leftarrow \{\}$
7: $\quad$ **for each** $m \in M_u^{main}$ **do**
8: $\quad\quad l_u \leftarrow$ LangID(m)
9: $\quad\quad$ **if** $l_u \neq$ unknown **then**
10: $\quad\quad\quad L_u \leftarrow L_u \cup \{l_u\}$
11: $\quad\quad$ **end if**
12: $\quad$ **end for**
13: $\quad$ **if** $len(L_u) = 1$ **then**
14: $\quad\quad U^{accept} \leftarrow U^{accept} \cup \{(u, L_u)\}$
15: $\quad\quad L^{accept} \leftarrow L^{accept} \cup L_u$
16: $\quad$ **end if**
17: **end for**
18: **for each** $l \in L_{accept}$ **do**
19: $\quad U^{sample} \leftarrow$ Sample(U_l^{accept}, K)
20: $\quad$ **for each** $u \in U^{sample}$ **do**
21: $\quad\quad M^{sample} \leftarrow$ Sample($M_u^{heldout}, N$)
22: $\quad\quad M^{accept} \leftarrow M^{accept} \cup \{(M^{sample}, l)\}$
23: $\quad$ **end for**
24: **end for**
25: **return** M^{accept}

`langid.py` (Lui and Baldwin, 2012): an n-gram feature set selected using data from multiple sources, combined with a multinomial naive Bayes classifier.

`CLD2` (McCandless, 2010): the language identifier embedded in the Chrome web browser;[2] it uses a naive Bayes classifier and script-specific tokenization strategies.

`LangDetect` (Nakatani, 2010): a naive Bayes classifier, using a character n-gram based representation without feature selection, with a set of normalization heuristics to improve accuracy.

`LDIG` (Nakatani, 2012): a Twitter-specific LangID tool, which uses a document representation based on tries, combined with normalization

[2] `http://www.google.com/chrome`

heuristics and Bayesian classification, trained on Twitter data.

whatlang (Brown, 2013): a vector-space model with per-feature weighting over character n-grams.

YALI (Majliš, 2012): computes a per-language score using the relative frequency of a set of byte n-grams selected by term frequency.

TextCat (Scheelen, 2003); an implementation of Cavnar and Trenkle (1994), which uses an ad-hoc rank-order statistic over character n-grams.

MSR-LID (Goldszmidt et al., 2013): based on rank-order statistics over character n-grams, and Spearman's ρ to measure correlation. Twitter-specific training data is acquired through a bootstrapping approach. We use the 49-language model provided by the authors, and the best parameters reported in the paper.

We investigated the performance of the systems using manually-labeled datasets of Twitter messages (Table 2), including the ZHENJA set described in Section 2.1.[3] We find that all the systems tested perform well on TROMP, with the exception of TextCat. CARTER covers a very similar set of languages to TROMP, yet all systems consistently perform worse on it. This suggests that TROMP is biased towards messages that LangID systems are likely to identify correctly (also observed by Goldszmidt et al. (2013)). This is due in part to the post-processing applied to the messages, but also suggests a bias in how messages were selected. LDIG is the best performer on TROMP and CARTER, albeit falling slightly short of the 99.1% accuracy reported by the author (Nakatani, 2012). However, it is only trained on 17 languages and thus is not able to fully support BERGSMA and ZHENJA, and so we cannot draw any conclusions on whether the method will generalize well to more languages. The system that supports the most languages by far is whatlang, but as a result its accuracy on Twitter messages suffers. Manual analysis suggests this is due to Twitter-specific "noise" tipping the model in favor of lower-density languages. On BERGSMA, LangDetect is the best performer, likely due to its specific heuristics for distinguishing certain language pairs (Nakatani, 2010), which happen to be present in the BERGSMA dataset. Overall, in their off-the-shelf configuration, only three systems (langid.py, CLD2, LangDetect) perform consistently well on LangID of Twitter messages. Even so, the macro-averaged F-Scores observed were as low as 83%, indicating that whilst performance is good, the problem of LangID of Twitter messages is far from solved.

Given that the set of languages covered and accuracy varies between systems, we investigated a simple voting-based approach to combining the predictions. For each dataset, we considered all combinations of 3, 5, and 7 systems, combining the predictions using a simple majority vote. The single-best combination for each dataset is reported in Table 3. In all cases, the macro-averaged F-score is improved upon, showing the effectiveness of the voting approach. Hence, for purposes of LangID in Algorithm 1, we chose to use a majority-vote ensemble of langid.py, CLD2 and LangDetect, a combination that generally performs well on all datasets.[4] Where all 3 systems disagree, the message is labeled as unknown, which does not count as a separate language for determining if a user is multilingual, mitigating the risk of wrongly rejecting a monolingual user due to misclassifying a particular message. This ensemble is hereafter referred to as VOTING.

To build our final dataset, we collected all messages by active users from the 1% feed made available by Twitter over the course of 31 days, between 8 January 2012 and 7 February 2012. We deemed users active if they had posted at least 5 messages in a single day on at least 7 different days in the 31-day period we collected data for. This gave us a set of approximately 2M users. For each user, we partitioned their messages (RandomSplit in Algorithm 1) by selecting one day at random. All of the messages posted by the user on this day were treated as heldout data ($M_u^{heldout}$), and the remainder of the user's messages (M_u^{main}) were used to determine the language(s) spoken by the user. The day chosen was randomly selected per-user to avoid any bias that may be introduced by messages from a particular day or date. Of the active users, we identified 85.0% to be monolingual, covering a set of 65 languages. 50.6% of these users spoke English (en), 14.1% spoke Japanese (ja), and 13.0% spoke Portuguese (pt); this user-level

[3] We do not limit the comparison to languages supported by each system as this would bias evaluation towards systems that support few languages that are easy to discriminate.

[4] MSR-LID was excluded due to technical difficulties in applying it to a large collection of messages because of its oversized model.

Dataset	langid.py	CLD2	LangDetect	LDIG	whatlang	YALI	TextCat	MSR-LID
TROMP	0.983	0.972	0.959	0.986	0.950	0.911	0.814	0.983
CARTER	0.917	0.902	0.891	0.943	0.834	0.824	0.510	0.927
BERGSMA	0.847	0.911	0.923	*0.000*	0.719	*0.428*	*0.046*	*0.546*
ZHENJA	0.871	0.884	0.831	*0.315*	0.622	0.877	0.313	0.848

Table 2: Macro-averaged F-Score on manually-annotated Twitter datasets. *Italics* denotes results where the dataset contains languages not supported by the identifier.

Dataset	Single Best		Voting		3-System
	System	F-Score	Systems	F-Score	F-Score
TROMP	LDIG	0.986	CLD2, MSR-LID, LDIG	0.992	0.986
CARTER	LDIG	0.943	MSR-LID, langid.py, LDIG	0.948	0.927
BERGSMA	LangDetect	0.923	CLD2, LangDetect, langid.py	0.935	0.935
ZHENJA	CLD2	0.884	CLD2, MSR-LID, LDIG, YALI, langid.py	0.969	0.941

Table 3: System combination by majority voting. All combinations of 3, 5 and 7 systems were considered. For each dataset, we report the single-best system, the best combination, and F-score of the majority-vote combination of langid.py, CLD2 and LangDetect.

language distribution largely mirrors the message-level language distribution reported by Baldwin et al. (2013) and others. From this set of users, we randomly selected up to 100 users per language, leaving us with a pool of 26011 held-out messages from 2914 users. Manual inspection of these messages revealed a number of English messages mislabeled with another language, indicating that even predominantly monolingual users occasionally introduce English into their online communications. Such messages are generally entirely English, with code-switching (i.e. multiple languages in the same message) very rarely observed. In order to eliminate mislabeled messages, we applied all 8 systems to this pool of 26011 messages. Where at least 5 systems agree and the predicted language does not match the user's language, we discarded the message. Where 3 or 4 systems agree, we manually inspected the messages and eliminated those that were clearly mislabeled (this is the only manual step in the construction of this dataset). Overall, we retained 24220 messages (93.1%). From these, we sampled up to 5 messages per unique user, producing a final dataset of 14178 messages across 65 languages (hereafter referred to as the TWITUSER dataset).

3 Evaluating off-the-shelf language identifiers on Twitter

Given TWITUSER, our broad-coverage Twitter corpus, we return to the task of examining the performance of the off-the-shelf LangID systems we discussed in Section 2.2 (Table 4, left side). In terms of macro-averaged F-Score across the full set of 65 languages, CLD2 is the single best-performing system. Unlike langid.py and LangDetect, CLD2 does not always produce a prediction, and instead has an in-built threshold for it to output a prediction of "unknown". This is reflected in the elevated precision, at the expense of decreased recall and message-level accuracy. Systems like langid.py which always make a prediction have reduced precision, balanced by increased recall and message-level accuracy. As with the manually-annotated datasets, we experimented with a simple voting-based approach to combining multiple classifiers. We again experimented with all possible combinations of 3, 5 and 7 classifiers, and found that on TWITUSER, a majority-vote ensemble of CLD2, langid.py and LangDetect attains the best macro-averaged F-Score, and also outperforms any individual system on all of the metrics considered. We note that this is exactly the VOTING ensemble of Section 2.2, validating its choice as LangID(m) in Algorithm 1.

3.1 Adapting off-the-shelf LangID to Twitter

Tromp and Pechenizkiy (2011) propose to remove links, usernames, hashtags and smilies before attempting LangID, as they are Twitter specific. We experimented with applying this cleaning procedure to each message body before passing it to our off-the-shelf systems (Table 4, right side). For LDIG and MSR-LID, the results are exactly the same with and without cleaning. These two systems are specifically targeted at Twitter messages, and thus may include a similar normalization as

Tool	Without Cleaning				With Cleaning			
	P	R	F	Acc	P	R	F	Acc
langid.py	0.767	0.861	0.770	0.842	0.759	0.861	0.766	0.840
CLD2	0.852	0.814	0.806	0.775	0.866	0.823	0.820	0.780
LangDetect	0.618	0.680	0.626	0.839	0.623	0.687	0.634	0.854
LDIG	0.167	0.239	0.189	0.447	0.167	0.239	0.189	0.447
whatlang	0.749	0.655	0.663	0.624	0.739	0.667	0.663	0.623
YALI	0.441	0.564	0.438	0.710	0.449	0.560	0.443	0.705
TextCat	0.327	0.245	0.197	0.257	0.316	0.295	0.230	0.316
MSR-LID	0.533	0.609	0.536	0.848	0.533	0.609	0.536	0.848
VOTING	0.920	0.876	0.887	0.861	0.919	0.883	0.889	0.868

Table 4: Macro-averaged Precision/Recall/F-Score, as well as message-level accuracy for each system on TWITUSER. The right side of the table reports results after applying message-level cleaning (Tromp and Pechenizkiy, 2011).

part of their processing pipeline. This also suggests that the systems do not leverage this Twitter-specific content in making predictions. Other systems generally show a small improvement with cleaning, except for langid.py. The VOTING ensemble also benefits from cleaning, due to the improvement in two of its component classifiers (CLD2 and LangDetect). This cleaning procedure is trivial to implement, so despite the improvement being small, it may be worth implementing if adapting off-the-shelf language identifiers to Twitter messages.

Goldszmidt et al. (2013) suggest bootstrapping a Twitter-specific language identifier using an off-the-shelf language identifier and an unlabeled collection of Twitter messages. We tested this approach, using the 3 systems that provide tools to generate new models from labeled data (LangDetect, langid.py and TextCat). We constructed bootstrap collections by: (1) using the off-the-shelf tools to directly identify the language of messages; and (2) using Algorithm 1. Overall, the bootstrapped identifiers are not better than their off-the-shelf counterparts. For TextCat there is an increase in accuracy using bootstrapped models, but the accuracy of TextCat with bootstrapped models is still inferior to LangDetect and langid.py in their off-the-shelf configuration. For LangDetect, utilizing bootstrapped models does not always increase the accuracy of LangID of Twitter messages. Where it does help, the bootstrap collections that are effective vary with the target dataset. For langid.py, none of the bootstrapped models outperformed the off-the-shelf model. This suggests that for LangID, the same features that are predictive of language in other domains are equally applicable to Twitter messages, and that the cross-domain feature selection procedure proposed utilized by langid.py (Lui and Baldwin, 2011) is able to identify these features effectively.

Dataset	Period	Proportion
CARTER	Jan – Apr 2010	76.4%
BERGSMA	May 2007 – Feb 2012	92.2%
TWITUSER	Jan – Feb 2012	79.7%

Table 5: Proportion of messages from each dataset that were still accessible as of August 2013.

Bontcheva et al. (2013) report positive results from the integration of LangID priors (Carter et al., 2013), but we did not experiment with them, as the calculation of priors is relatively expensive compared to the other adaptations we have considered, in terms of both run time and developer effort. Furthermore, there is a number of open issues that are likely to affect the effectiveness of the priors, such as the size and the scope of the message collection used to determine the prior. This is an interesting avenue of future work but is beyond the scope of this particular paper. However, we observe that priors based on user identity (e.g the "Blogger" prior) are likely to be artificially effective on TWITUSER, because the messages have been sampled from users that we have identified as monolingual.

3.2 Twitter API predictions

For CARTER, BERGSMA and TWITUSER, we have access to the original identifiers for each message, which use used to download the messages via the Twitter API.[5] Table 5 reports the proportion of each dataset that is still accessible as of August 2013. For the messages that we were able

[5] http://dev.twitter.com

to recover, the full response from the API now includes language predictions. We do not report quantitative results on the accuracy of the Twitter API predictions as the Twitter API terms of service forbid benchmarking ("You will not attempt ... to ... use or access the Twitter API ... for ... benchmarking or competitive purposes"). Furthermore, any results would be impossible to replicate: the set of messages that are accessible is likely to continue to decrease, and the accuracy of Twitter's predictions may vary as updates are made to the API.

Error analysis of the language predictions provided by the Twitter API shows that at the time of writing, for the languages supported the accuracy of the Twitter API is not substantially better than the best off-the-shelf language identifiers we examined in this paper. However, about a quarter of the languages present in TWITUSER are never offered as predictions. This has implications for the precision of LangID in other languages: one notable example is poor precision in Italian, due to some Romanian messages being identified as Italian (no messages are identified as Romanian). This suggests that caution must be taken in taking the language predictions offered by the Twitter API as goldstandard. The accuracy of the predictions is not perfect, and highlights the need for further research into improving the scope and accuracy of LangID for Twitter messages.

4 Conclusion

In this paper, we presented ZHENJA and TWITUSER, two novel datasets of language-labeled Twitter messages. ZHENJA is constructed using a conventional manual annotation approach, whereas TWITUSER is constructed using a novel mostly-automated method that leverages user identity. Using these new datasets alongside three previously-published datasets, we compared 8 off-the-shelf LangID systems over Twitter messages, and found that a simple majority vote across three specific systems (`CLD2`, `langid.py`, `LangDetect`) consistently outperforms any individual system. We also found that removing Twitter-specific content from messages generally improves the performance of off-the-shelf systems that have not been tuned to Twitter. We reported that the predictions provided by the Twitter API are not better than state-of-the-art off-the-shelf systems, and that a number of languages in use on Twitter appear to be unsupported by the Twitter API, underscoring the need for further research to broaden the scope and accuracy of language identification from Twitter messages.

Acknowledgments

NICTA is funded by the Australian Government as represented by the Department of Broadband, Communications and the Digital Economy and the Australian Research Council through the ICT Centre of Excellence program.

References

Timothy Baldwin and Marco Lui. 2010. Language identification: The long and the short of the matter. In *Proceedings of Human Language Technologies: The 11th Annual Conference of the North American Chapter of the Association for Computational Linguistics (NAACL HLT 2010)*, pages 229–237, Los Angeles, USA.

Timothy Baldwin, Paul Cook, Marco Lui, Andrew MacKinlay, and Li Wang. 2013. How noisy social media text, how diffrnt social media sources? In *Proceedings of the 6th International Joint Conference on Natural Language Processing (IJCNLP 2013)*, Nagoya, Japan.

Shane Bergsma, Paul McNamee, Mossaab Bagdouri, Clayton Fink, and Theresa Wilson. 2012. Language identification for creating language-specific Twitter collections. In *Proceedings the Second Workshop on Language in Social Media (LSM2012)*, pages 65–74, Montréal, Canada.

Kalina Bontcheva, Leon Derczynski, Adam Funk, Mark A. Greenwood, Diana Maynard, and Niraj Aswani. 2013. TwitIE: An open-source information extraction pipeline for microblog text. In *Proceedings of Recent Advances in Natural Language Processing (RANLP 2013)*, Hissar, Buglaria.

Ralf Brown. 2013. Selecting and weighting n-grams to identify 1100 languages. In *Proceedings of the 16th international conference on text, speech and dialogue (TSD 2013)*, Plzeň, Czech Republic.

Simon Carter, Wouter Weerkamp, and Manos Tsagkias. 2013. Microblog language identification: Overcoming the limitations of short, unedited and idiomatic text. *Language Resources and Evaluation*, pages 1–21.

William B. Cavnar and John M. Trenkle. 1994. N-gram-based text categorization. In *Proceedings of the Third Symposium on Document Analysis and Information Retrieval*, pages 161–175, Las Vegas, USA.

Hakan Ceylan and Yookyung Kim. 2009. Language identification of search engine queries. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP*, pages 1066–1074, Singapore.

Aron Culotta. 2010. Towards detecting influenza epidemics by analyzing Twitter messages. In *Proceedings of the KDD Workshop on Social Media Analytics*.

Marc Darnashek. 1995. Gauging similarity with n-grams: Language-independent categorization of text. *Science*, 267:843–848.

Rafael Dueire Lins and Paulo Gonçalves. 2004. Automatic language identification of written texts. In *Proceedings of the 2004 ACM Symposium on Applied Computing (SAC 2004)*, pages 1128–1133, Nicosia, Cyprus.

Ted Dunning. 1994. Statistical identification of language. Technical Report MCCS 940-273, Computing Research Laboratory, New Mexico State University.

Joseph L. Fleiss. 1971. Measuring nominal scale agreement among many raters. *Psychological Bulletin*, 76(5):378–382.

Emmanuel Giguet. 1995. Categorisation according to language: A step toward combining linguistic knowledge and statistical learning. In *Proceedings of the 4th International Workshop on Parsing Technologies (IWPT-1995)*, Prague, Czech Republic.

Kevin Gimpel, Nathan Schneider, Brendan O'Connor, Dipanjan Das, Daniel Mills, Jacob Eisenstein, Michael Heilman, Dani Yogatama, Jeffrey Flanigan, and Noah A. Smith. 2011. Part-of-speech tagging for Twitter: Annotation, features, and experiments. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies (ACL HLT 2011)*, pages 42–47, Portland, USA.

Moises Goldszmidt, Marc Najork, and Stelios Paparizos. 2013. Boot-strapping language identifiers for short colloquial postings. In *Proceedings of the European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases (ECMLPKDD 2013)*, Prague, Czech Republic.

Gregory Grefenstette. 1995. Comparing two language identification schemes. In *Proceedings of Analisi Statistica dei Dati Testuali (JADT)*, pages 263–268, Rome, Italy.

Harald Hammarstrom. 2007. A Fine-Grained Model for Language Identication. In *Proceedings of Improving Non English Web Searching (iNEWS07)*, pages 14–20.

Bo Han, Paul Cook, and Timothy Baldwin. 2013. Lexical normalization for social media text. *ACM Trans. Intell. Syst. Technol.*, 4(1):5:1–5:27, February.

Stephen Johnson. 1993. Solving the problem of language recognition. Technical report, School of Computer Studies, University of Leeds.

Ben King and Steven Abney. 2013. Labeling the languages of words in mixed-language documents using weakly supervised methods. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1110–1119, Atlanta, Georgia.

Balachander Krishnamurthy, Phillipa Gill, and Martin Arlitt. 2008. A few chirps about Twitter. In *Proceedings of the First Workshop on Online Social Networks (WOSN 2008)*, pages 19–24, Seattle, USA.

Canasai Kruengkrai, Prapass Srichaivattana, Virach Sornlertlamvanich, and Hitoshi Isahara. 2005. Language identification based on string kernels. In *Proceedings of the 5th International Symposium on Communications and Information Technologies (ISCIT-2005)*, pages 896–899, Beijing, China.

Vasileios Lampos, Tijl De Bie, and Nello Cristianini. 2010. Flu Detector – tracking epidemics on Twitter. In *Proceedings of the European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases (ECML PKDD 2010)*, pages 599–602, Barcelona, Spain.

Wang Ling, Guang Xiang, Chris Dyer, Alan Black, and Isabel Trancoso. 2013. Microblogs as parallel corpora. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 176–186, Sofia, Bulgaria.

Marco Lui and Timothy Baldwin. 2011. Cross-domain feature selection for language identification. In *Proceedings of the 5th International Joint Conference on Natural Language Processing (IJCNLP 2011)*, pages 553–561, Chiang Mai, Thailand.

Marco Lui and Timothy Baldwin. 2012. langid.py: An off-the-shelf language identification tool. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (ACL 2012) Demo Session*, pages 25–30, Jeju, Republic of Korea.

Martin Majliš. 2012. Yet another language identifier. In *Proceedings of the Student Research Workshop at the 13th Conference of the European Chapter of the Association for Computational Linguistics*, pages 46–54, Avignon, France.

Michael McCandless. 2010. Accuracy and performance of google's compact language detector. blog post. available at `http://blog.mikemccandless.com/2011/10/accuracy-and-performance-of-googles.html`.

Shuyo Nakatani. 2010. Language detection library (slides). `http://www.slideshare.net/shuyo/language-detection-library-for-java`. Retrieved on 21/06/2013.

Shuyo Nakatani. 2012. Short text language detection with infinity-gram. blog post. available at `http://shuyo.wordpress.com/2012/05/17/short-text-language-detection-with-infinity-gram/`.

Dong Nguyen and A. Seza Dogruoz. 2013. Word level language identification in online multilingual communication. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 857–862, Seattle, USA.

Sasa Petrović, Miles Osborne, and Victor Lavrenko. 2010. Streaming first story detection with application to twitter. In *Proceedings of Human Language Technologies: The 11th Annual Conference of the North American Chapter of the Association for Computational Linguistics (NAACL HLT 2010)*, pages 181–189, Los Angeles, USA.

Arne Roomann-Kurrik. 2013. Introducing new metadata fro tweets. blog post. available at `https://dev.twitter.com/blog/introducing-new-metadata-for-tweets`.

Takeshi Sakaki, Makoto Okazaki, and Yutaka Matsuo. 2010. Earthquake shakes Twitter users: real-time event detection by social sensors. In *Proceedings of the 19th International Conference on the World Wide Web (WWW 2010)*, pages 851–860, Raleigh, USA.

Frank Scheelen, 2003. *libtextcat*. Software available at `http://software.wise-guys.nl/libtextcat/`.

Shiladitya Sinha, Chris Dyer, Kevin Gimpel, and Noah A. Smith. 2013. Predicting the NFL using Twitter. In *Proceedings of the ECML/PKDD Workshop on Machine Learning and Data Mining for Sports Analytics*, Prague, Czech Republic.

Erik Tromp and Mykola Pechenizkiy. 2011. Graph-based n-gram language identification on short texts. In *Proceedings of Benelearn 2011*, pages 27–35, The Hague, Netherlands.

Tommi Vatanen, Jaakko J. Vayrynen, and Sami Virpioja. 2010. Language identification of short text segments with n-gram models. In *Proceedings of the 7th International Conference on Language Resources and Evaluation (LREC 2010)*, pages 3423–3430.

Hiroshi Yamaguchi and Kumiko Tanaka-Ishii. 2012. Text segmentation by language using minimum description length. In *Proceedings the 50th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 969–978, Jeju Island, Korea.