Developing Language-tagged Corpora for Code-switching Tweets

Suraj Maharjan[1] and **Elizabeth Blair**[2] and **Steven Bethard**[2] and **Thamar Solorio**[1]
[1]Department of Computer Science
University of Houston
Houston, TX, 77004
smahajan2@uh.edu solorio@cs.uh.edu
[2]Department of Computer and Information Sciences
University of Alabama at Birmingham
Birmingham, AL, 35294
{eablair, bethard}uab.edu

Abstract

Code-switching, where a speaker switches between languages mid-utterance, is frequently used by multilingual populations worldwide. Despite its prevalence, limited effort has been devoted to develop computational approaches or even basic linguistic resources to support research into the processing of such mixed-language data. We present a user-centric approach to collecting code-switched utterances from social media posts, and develop language universal guidelines for the annotation of code-switched data. We also present results for several baseline language identification models on our corpora and demonstrate that language identification in code-switched text is a difficult task that calls for deeper investigation.

1 Introduction

A common phenomenon among multilingual speakers is *code-switching*, that is, switching between languages within a single context (Lipski, 1978). Code-switching can occur on a sentence-by-sentence basis, known as intersentential code-switching, as well as between words within a single sentence, known as intrasentential code-switching (Poplack, 1980).

Developing technology that can process this kind of mixed language data is important for a number of different sectors. In the fight against organized crime, human and drug trafficking smugglers travel between Mexico and the United States, and processing the mixed Spanish-English data that accompanies this trafficking could yield more actionable intelligence for law enforcement. In the service industry, companies like LENA[1] analyze child language to provide parents with a variety of metrics on child development, and their language processing tools must be taught to handle the language mixing common to bilingual children. And in data mining applications, companies like Dataminr[2] want to transform Twitter into actionable signals, and ignoring the multilingual portion of the world's population represents significant lost business opportunities.

In this paper, we describe our efforts in the development and annotation of corpora containing code-switched data in written form for two language pairs: Spanish-English and Nepali-English. These two language combinations are well suited for research in code switching: Spanish-English as an example of a large multilingual minority population (in the United States), and Nepali-English is an example of a population that is almost entirely multilingual. In both cases the two languages are written using the same Latin script. This is true for Nepali, even though Devanagari is its official script, because the education system in Nepal teaches typing only for English, so for digital content like social media it is common for Nepalese speakers to type using English characters.

We chose Twitter as the source of our data as the informal nature of tweets makes them a more natural source for code-switching phenomena. Many researchers have turned to Twitter as a source of data for research (i.e. (Roberts et al., 2012; Reyes et al., 2013; Tomlinson et al., 2014; Kong et al., 2014; Temnikova et al., 2014; Williams and Katz, 2012)). Typically, collecting Twitter data is a straightforward

[1] http://www.lenafoundation.org/

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

Proceedings of LAW IX - The 9th Linguistic Annotation Workshop, pages 72–84,
Denver, Colorado, June 5, 2015. ©2015 Association for Computational Linguistics

process involving the Twitter API, specifying the desired language, and a set of keywords or hash tags. For example, in the research on user intentions some of the hash tags used include: #mygoal, #iwon, #madskills, #imapro, #dowhatisay, #kissmyfeet, #proud. A similar process was followed by all of the previous work listed above. However, Twitter allows only one desired language to be specified, and no simple keywords exist for finding code-switched tweets. Searching for the words "code-switching" or "Spanglish" would have resulted in unnatural data, where the users were aware of the linguistic phenomenon, rather than the spontaneous use of more than one language that we seek. This is akin to research in cyberbullying, where data collection on Twitter using hash tags or keywords like #bully or #cyberbullying does not result in the actual bullying tweets (Dinakar et al., 2011). We present here a strategy to locate the right data in Twitter. We hope other researchers whose data needs cannot be met by simple keyword search can benefit from our lessons learned.

After collecting a sufficient amount of data with code-switching, we set out on the task of annotating the data using a combination of in-lab and crowdsourcing annotations. We develop a set of annotation guidelines that can be used for Twitter data and any language combination. The design of these annotation standards reflects the unique needs of mixed language data and the goal of supporting research in linguistic and sociolinguistic aspects of code-switching, as well as research in statistical methods for the automated processing of code-switching. Therefore, the annotations are theory agnostic, and follow a pragmatic definition of code-switching.

Finally, to show that the processing of code-switching text requires further advancement of our NLP technology, we present a case study in language identification with our corpora. Language identification of monolingual text has been considered a solved problem for some time now (McNamee, 2005) and even in Twitter the problem has been shown to be tractable when annotated data is available (Bergsma et al., 2012). However, as we demonstrate in this paper, when code-switching is present, the performance of state-of-the-art systems is not on par with that of monolingual sources. We predict that the difficulty increases for deeper and higher-level NLP tasks. In fact, Solorio and Liu (2008b) have shown already that part-of-speech tagging performance in code-switching data is also lagging behind that observed in monolingual sources.

2 Related Work

Although code-switching has not been investigated as deeply as monolingual text in the natural language processing field, there has been some work on the topic. An earlier example is the work by Joshi (Joshi, 1982), where he proposes a system that can help to parse and generate code-switching sentences. His approach is based on the matrix language-embedded language formalism and although the paper has a good justification it lacks an empirical evaluation supporting the proposed model. A few more recent examples of work in NLP and code-switching are the methods examined by Solorio and Liu that include developing a better part of speech tagging approach for code-switching text (Solorio and Liu, 2008b) and identifying potential code-switching points within text (Solorio and Liu, 2008a). In each of these projects, however, code-switching data was scarce, coming primarily from conversations. Because of complications with traditional evaluation measures, the code-switching point detection project used a new evaluation method, in which artificial code-switched content was generated and compared with genuine content (Solorio and Liu, 2008a).

In the past, most language identification research has been done at the document level. Some researchers, however, have developed methods to identify languages within multilingual documents (Singh and Gorla, 2007; Nguyen and Doğruöz, 2013; King and Abney, 2013). Their test data comes from a variety of sources, including web pages, bilingual forum posts, and jumbled data from monolingual sources, but none of them are trained on code-switched data, opting instead for a monolingual training set per language. This could prove to be a problem when working on code-switched data, particularly in shorter samples such as social media data, as the code-switching context is not present in training material.

One system tackled both the problems of code-switching and social media in language and code-switched status identification (Lignos and Marcus, 2013). Lignos and Marcus gathered millions of monolingual tweets in both English and Spanish in order to model the two languages and used crowdsourcing to annotate tens of thousands of Spanish tweets, approximately 11% of which contained code-switched content. This system was able to achieve 96.9% word-level accuracy and a 0.936 F-measure in identifying code-switched tweets.

The issue still stands that relatively little code-switching data, such as that used in Lignos and Marcus' research, is readily available. Even in their data, the percentage of code-switched tweets was barely over a tenth of the total test data. There have been other corpora built, particularly for other language pairs such as Mandarin-English (Li et al., 2012; Lyu et al., 2010), but the amount of data available and the percentage of code-switching present are not up to the standards of other areas of the natural language processing field. With this in mind, we sought to provide corpora for multiple language pairs, each with a better distribution of code-switching. In this paper we discuss the process we followed for two language pairs and our current efforts are targeted to grow the number of language pairs collected and annotated.

3 Corpus Creation

Developing the corpus involved two steps: locating code-switching tweets and using crowdsourcing to annotate them for language and an assortment of other tags. A small portion of these annotations were reviewed by in-lab annotators to measure agreement and gauge the quality of the crowdsourced data.

All token-level annotations were done according to a set of guidelines provided to all annotators and presented in this paper as Appendix A. There we show the guidelines specific for Spanish-English. For Nepali-English only a small customization of examples was needed. The tags they could select from were Lang1 (English), Lang2 (Spanish or Nepali), Named Entity, Ambiguous, Mixed, or Other. Words that exist in both languages, such as 'me' or 'no', were disambiguated using context if possible; if not, they were assigned the Ambiguous tag. The Mixed tag was reserved for words that contained portions of multiple languages, such as 'snap*chateame*' which contains both English and Spanish content. Anything that did not fall into these categories, such as other languages, gibberish, Twitter user handles, URLs, emoticons, symbols, and punctuation, were given the label Other. Hashtags were annotated according to the text following the # symbol. Slang, misspellings, and abbreviations were labeled according to the word(s) they represented.

3.1 Locating Code-Switched Data

Although locating code-switched tweets was not initially one of the bigger concerns of this project, it developed into quite an interesting problem. To refrain from biasing the data set towards particular words or phrases, we did not wish to use keyword-based search in order to obtain tweets. Our method of gathering data therefore became finding users who code-switched often and pulling their tweet histories. For Nepali, we searched for users that constantly switched between Nepali and English. An initial set of users was easily found via a collaborator from Nepal who has ties with many Nepali-English bilingual users on Twitter. We then looked for users mentioned in their tweets and checked to see if they too, were frequently code-switching. Eventually, we identified 42 frequent code-switchers and collected nearly 2000 tweets each from them. We filtered out all the retweets and tweets with urls.

For Spanish-English, however, locating code-switching users was difficult as we had no Spanish speaking collaborators with ties to a code-switching Twitter community. We first used Twitter's recent tweet search API to find tweets using English terms (taken from the most frequent English words in the Bangor Miami Corpus[3]) and restricted to tweets that Twitter's language detection identified as Spanish and that were sent from areas close to California and Texas. Results from this search were passed to in-lab annotators for token-level annotations according to the annotation guidelines. Code-switching ratios were low in this data set, so we ran a new search for tweets from the same geographical regions that Twitter identified as English containing the Spanish words that were most frequent in the results of the first search (ignoring ambiguous and stop words). Results from both searches were filtered to remove extremely similar tweets, spam tweets such as news and automatic posts from other social media sources, retweets, and tweets containing URLs (which were

[3] http://www.language-archives.org/item/oai:talkbank.org:BilingBank-Bangor-Miami

particularly prone to spam). We then pulled the first 50 tweets of each of the 135 most frequent users from the combined search results. These were annotated in-lab at the tweet level for code-switched content. Any users with fewer than three code-switching tweets were discarded, resulting in 44 users.

A small portion of this data, 1163 tweets distributed evenly among the 44 users, was two-way annotated in-lab at the token level, and used as quality control data for CrowdFlower annotation (see section 3.2). We used the resulting annotations to identify the frequency of code-switching for each user. All available tweets were pulled from the nine users with the highest code-switching frequency, and tweets from the next thirteen users were used to fill in up to 14,000 tweets.

We tried to extract some demographic characteristics of the users in our corpora. As the Twitter API does not give the gender information of users, we manually checked their profiles and used their names and profile pictures to identify their gender. Even with this method, we could not determine the gender for two Spanish-English and two Nepali-English users. The rest of the users were split almost evenly for Spanish-English (9 male and 11 females), while in the Nepali-English data we have 15 males and 6 females. Twitter also provides information about geographical location of the users. Our Spanish-English users came from Eastern, Central, Pacific, Mountain (US & Canada) timezones whereas all users for Nepali-English came from Kathmandu as per the Twitter API.

For the purpose of system development, testing and benchmarking, we divided the corpora into train and test sets. For Spanish-English the training set has 11,400 and the test set has 3,014 tweets. The Nepali-English corpus was split into 9,993 tweets for training and 2,874 tweets for testing. Table 1 shows the distribution of the six different tags across the training and test datasets for both Nepali-English and Spanish-English. As can be inferred from the table, the concentration of Lang1, Lang2 and Other tags is much higher than NE, Ambiguous and Mixed tags for both language pairs.

The Twitter users in each set (training vs. test) are disjoint to ensure that systems would not be overfitting to the idiosyncrasies of particular users. The split was designed to maintain the same balanced distribution of tweet content in both sets.

Table 1: Distribution of tags across training and test datasets.

	Nep-En (%)		**Es-En(%)**	
Tag	**Training**	**Test**	**Training**	**Test**
Lang1	31.14	19.76	54.78	43.28
Lang2	41.56	49.1	23.52	30.34
Mixed	0.08	0.60	0.04	0.03
NE	2.73	4.19	2.07	2.22
Ambiguous	0.09	-	0.24	0.12
Other	24.41	26.35	19.34	24.02

3.2 Crowdsourcing Annotations

In order to efficiently annotate the large amount of tweets needed for the corpus, we used the crowdsourcing platform CrowdFlower. This platform, similar to the Amazon Mechanical Turk (AMT) service, provides access to a community of crowdsourced workers who are willing to complete small tasks for relatively low pay. CrowdFlower differs from AMT in offering additional quality control services.

To gather the annotations, we created CrowdFlower tasks at the word level for each tweet. One task in the CrowdFlower interface consisted of the selected word designated within the full tweet in order to provide context. Following the recommendations in (Callison-Burch and Dredze, 2010), the tweet was made into an image displaying the text with the selected token highlighted by a yellow box. This was done in order to prevent users from simply copying the text into a language detection program. Underneath the image was a question asking them to select the correct annotation for the word using radio buttons listing each annotation category. There was also an optional comments section where they could leave a note about the question. To speed up the process and save money, words in the Other category that could be automatically detected (Twitter user handles, URLs, emoticons, symbols and punctuation) were excluded from CrowdFlower annotation.

Instructions for the job were provided to the workers at the beginning of each page of tasks. We provided a basic description of the job and how to interpret each portion of the task. After that, we gave a link to a PDF of a slightly modified, CrowdFlower-friendly version of the annotation guidelines provided to the in-lab annotators. The guidelines gave a description of the overall job and of each label, along with examples. There was also a section in the job's instructions containing a few key notes, such as how to handle named entities and slang.

We gave 15 tasks at a time to each crowdsourced worker. They were paid $0.03 for each fifteen-task page they completed. Payment was only given if the users met the strict quality controls built into the platform. CrowdFlower takes gold-annotated tasks along with the blank tasks and uses that gold to test workers as the job runs, removing the burden from the job organizers. To begin the job, workers must obtain at least 70% accuracy on an eight-question quiz made of the gold tasks. If they pass the quiz, they begin work on the task proper, but gold is continuously woven into their work. If they fall below the 70% threshold, their work is removed from the total data to avoid contamination and they are not paid for the low-quality annotations. Following the suggestions of (Zhai et al., 2013), we added gold equal to 20% of the job's tasks. To avoid additional negative results, we also limited workers to those from the United States, Argentina, Chile, Colombia, Mexico, and Peru for the Spanish-English corpus and Nepal and Bhutan for the Nepali-English corpus.

A few pilot jobs were run for each language pair on 100-tweet samples, using tweets that had already been annotated in-lab – three-way for Spanish-English and two-way for Nepali-English – to judge the accuracy of CrowdFlower workers' results. Analysis of the agreement allowed for improvement of the guidelines, particularly in the named entity and ambiguous categories, as well as confirmation that three-way CrowdFlower annotation provided acceptable results at the current payment scheme. The pilots also showed that CrowdFlower's aggregated results outperformed majority and trust-weighted voting schemes, so they were used in the final work.

The 14,000 Spanish-English tweets collected in section 3.1 were run through CrowdFlower in batches of 2000 tweets. All batches used the same set of gold tasks, which consisted of the 1163 tweets annotated two-way in the lab. Because we were unsure whether workers were reading the PDF instructions, we changed the instruction scheme for one of these jobs. The new scheme moved the label descriptions inline, where the workers could read them without clicking away. The PDF link was still provided to give them access to the examples.

The Nepali data, which was found to have a higher concentration of code-switching tweets during the in-lab annotations, was simply run in two 5,000 tweet batches and one 3,000 tweet batch. The gold data for quality control of this task contained 1,000 tweets that were annotated by two in-lab annotators.

3.3 Review and Agreement

To judge the validity of the CrowdFlower annotations, one-way in-lab review was performed on small segments of the crowdsourced results. 1,000 tweets were reviewed from jobs using the PDF instruction scheme and another 500 were reviewed from the job using the inline instruction scheme. Inter-annotator agreement measures were calculated between the original and reviewed annotations for each scheme. The measures used were observed agreement, Fleiss multi-π, and Cohen multi-κ (Artstein and Poesio, 2008) calculated for the full data set, as well as observed agreement per annotation category.

The CrowdFlower annotation results' agreement with the in-lab review was above expectations. All three overall agreement measures were at or above 0.9. At the category level, agreement was high for the simpler categories, such as Lang1, Lang2, and Other, but dipped considerably for the more complicated ones such as named entities. This is consistent with the error analysis done by King and Abney (2013), where the most frequent source of error was named entities. Ambiguous and Mixed made up only approximately one tenth of a percent each of the total annotations given, so the agreements on these are unreliable. Named entities, at three to five percent of the data, show a more reliable result.

There was little difference, at most 0.01, in the annotation agreement between the jobs using PDF and inline instruction schemes. It is unlikely that this small difference in agreement is indicative of a useful difference in annotation quality. Optional customer experience surveys provided to workers by CrowdFlower after task completion showed slightly more happiness with pay and test questions when using the inline instructions, even though neither of these factors changed between jobs. It is possible that although performance is unchanged, worker satisfaction may be higher when using inline instructions instead of linking to an external PDF.

4 Benchmark Systems

To show the shortcomings of state-of-the-art systems on code-switched social media text, two benchmark systems for language identification were run on the annotated corpora. The first was a simple dictionary approach, while the second was a state-of-the-art word-level language identification system designed for multilingual documents (King and Abney, 2013).

The two systems were evaluated on their performance in language identification at the word level and identifying code-switching at the tweet level. Performance was measured using accuracy, precision, recall, and F-measure. A tweet was marked as code-switching only if it contained at least one label for each language.

4.1 Language Identification with Dictionaries

The dictionary approach was designed as the simplest possible system for language identification using the collected training data. The lowercase form of all of the words in the training data were split into separate lexicons based on their tag. Hashtags had the # removed and the text was included as a word.

The system only assigned language tags (Lang1, or Lang2) and Other. If the lowercase form of a word appeared in one lexicon but not the other, it assigned that lexicon's language. If the word was a Twitter user handle, URL, emoticon, symbol or punctuation, it assigned the Other category. Otherwise, if the word existed in both or neither lexicons, it assigned the majority language from the training data.

4.2 Language Identification with CRFs

The state-of-the-art language identification system of King and Abney (2013) was designed for word-level annotation on multilingual documents, and was thus a suitable choice for our task. This weakly supervised system uses Conditional Random Fields (CRF) with Generalized Expectation (GE) criteria (Mann and McCallum, 2008). The system itself was provided by the authors, so no reimplementation was necessary.

The CRF GE language id system requires samples of monolingual text from each language as training data. The English and Spanish training sets were pulled from Twitter searches in the Texas and California areas for consistency, using Twitter's language identification along with the language's stop words as queries in order to get reliable results. Equal amounts, approximately 10MB each, of data were collected for each language. Monolingual Nepali tweets in Roman script were harder to find. The Twitter API only allows to search for tweets using Devanagari. So, we looked for other sources of Romanized Nepali text, such as song lyrics websites, news websites etc. We crawled nearly 1.2MB of song lyrics from song lyrics websites. However, this was not enough. Hence we returned to Twitter to identify users who tweet in Nepali by using Devanagari script. We collected the remaining 9MB of data (117,806 tweets) from these users and then transliterated them to Roman Script by using our Devanagari to Roman transliteration script[4].

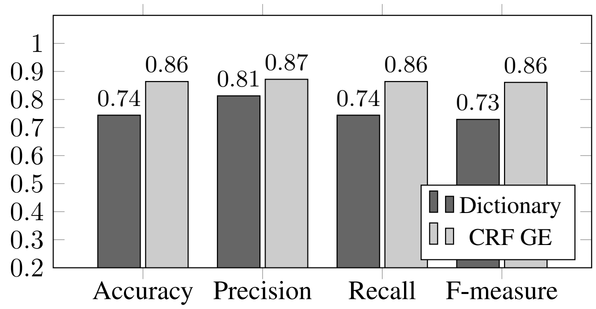


Figure 1: Benchmark system performance at the word level on Spanish-English language tags.

The training data was gathered into a single file per language and fed into the CRF GE system. Then each test tweet was input to the system for prediction. We removed from the tweet tokens with a hash tag, emoticons and tokens of the type @username.

5 Benchmark Results

To provide a fair comparison of the benchmark systems we only evaluate prediction performance for the words labeled with Lang1 or Lang2 in the gold data, as the benchmark systems were not designed for named entities, ambiguous words or mixed words. We report results using the familiar metrics accuracy, precision, recall, and F-measure. The results are shown in Figures 1 and 2. For Spanish-English, both systems performed well under the state of the art from Lignos and Marcus who obtained 96.9% word-level

[4] The script can be downloaded from `http://www2.cs.uh.edu/~suraj/scripts/devnagari2roman.py`

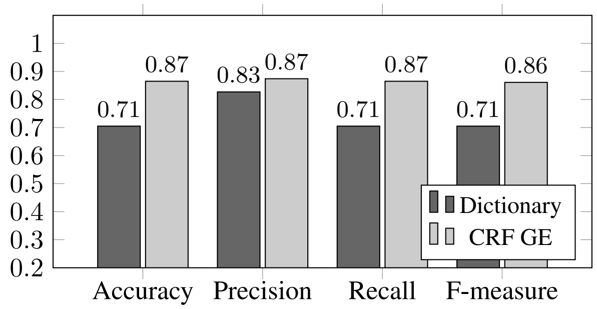


Figure 2: Benchmark system performance at the word level on Nepali-English language tags.

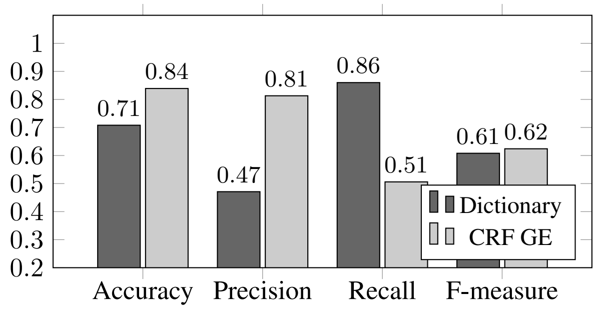


Figure 3: Performance at the tweet level on Spanish-English code-switching detection.

accuracy and a 0.936 F-measure in identifying code-switched tweets for multilingual documents, with the dictionary at 74% accuracy and the CRF GE at 86%. We observe a similar result for Nepali-English, with the dictionary at 70% and the CRF GE at 87%. These unsurprising results show that even systems designed to receive more than one language as input assume longer monolingual contexts. But spontaneous code switching does not obey these patterns.

The tweet level analysis seen in Figures 3 and 4 show that for Spanish-English, performance is on par with the token-level results, while for Nepali-English the dictionary system outperforms the CRF GE model with an accuracy of 90%. The strength of the dictionary system for Nepali-English may be due to the smaller word and character overlap between these languages.

6 Analysis

Since the Dictionary Approach considers only the tokens and ignores the context, tokens that are spelled the same way in both English and another language are often mislabeled. This is the case for words like

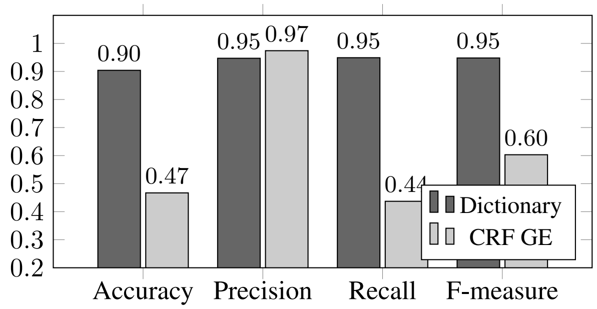


Figure 4: Performance at the tweet level on Nepali-English code-switching detection.

man , *gate* and *din* that in Nepali mean like, date and day, respectively, and words like *me*, *red*. Also, as expected, language identification fails in the case of infrequent, unseen and misspelled tokens, such as *comrade*, *yuss*(yes), *b-lated*, and *vokamanchey*(hungry men). Another source of error for Nepali-English is that there is no standard Romanized spelling for Nepali words. People just use whatever sounds phonetically similar. For example, in Nepali the word for pain may be written as *pidaa*, *peeda*, or *pida*.

The CRF GE failed to detect small code-switched content embedded inside large monolingual segments. We observed many cases of single English words in Nepali context classified as Nepali. We believe that these misclassifications might be occurring due to the underlying sequential model of the CRF GE that relies on larger contexts.

In another analysis, we computed the overlap of words and n-grams (2-5) between each pair of languages, Nepali-English and Spanish-English, in the training datasets. Our goal was to quantify the overlap of lexical items in each code-switching language pair. Our assumption is that higher overlap represents a more challenging task for the language identification task. Table 2 shows percentages of common tokens between languages. Bigrams show similar overlap in each language pair, but as the n-grams become larger, the overlap between Spanish-English is considerably larger than that for Nepali-English.

A natural question from our data is if both bilingual communities have similar linguistic behaviors. This study requires a deeper syntactic analysis of the samples and we leave this for future research. But a simple exploratory analysis can consider the most frequent items used in English, their common lan-

Table 2: N-gram overlap across language pairs.

Tokens	Nep-Eng (%)	Span-Eng (%)
words	1.39	3.54
char -2	52.01	52.21
char -3	33.36	40.36
char -4	12.66	21.31
char -5	3.43	9.00

guage. We looked at the most frequent English words and found that both communities use similar English words while code-switching. These words mainly include function words (*the*, *to*, *yo*, *he*, *she*, *and*) and abbreviations used in social media (*lol*, *lmao*, *idk*). Stop words are the most commonly occurring words even in monolingual texts, so it is no surprise that they appear here too. In the case of abbreviations, some of them such as *lol* and *lmao* have become social media lingo rather than abbreviations of English words and thus cross language barriers.

Figure 5 shows the learning curve for Spanish-English and Nepali-English training dataset using the Dictionary Approach. For this experiment, we divided the training data into 80:20 ratio. The 20% of the training data was used for cross validation. We gradually increased the training data and computed error on training as well as cross validation dataset. The graphs show that adding more data is likely to improve the performance as cross validation error seems to be decreasing with addition of more lexicons. This experiment justifies our investment on annotating more data using Crowdflower.

7 Discussion

Upon reviewing the size and content distribution of the corpora, we believe our attempt to generate sets of code-switching social media content was successful. Although code-switching does not make up the majority of the data, there is a strong balance between it and other types of data, such as the named entities, ambiguous and mixed words, and monolingual tweets of both languages. This blend provides additional data for the development of research systems and gives a more realistic sample of how Twitter users approach code-switching.

Finding code-switching tweets for Spanish-English required significant effort, but our approach led to a selection of data with an acceptable amount of code-switched content. Because we wanted to avoid the kind of bias caused by searching for particular words, heuristically filtering the data, or working with a single user, the process was difficult; it was, however, worth the effort to make sure that a system could not gain an unfair advantage by training on a particular user or set of repeated words.

If possible, as it was with the Nepali data set, finding a community that uses code-switching often appears to be the easiest method to obtain the data in bulk. If that is not available, however, searching for tweets in one language while querying for terms in another appears to be an effective way to locate such users. The small batch of in-lab work was enough to identify some users, but a larger set such as the first CrowdFlower job was much more effective at identifying the most useful users.

When developing more corpora, it would be ideal to find a way to identify users with higher code-switching concentrations in their timelines. One potential approach that could be addressed in future work is to look into the Twitter users that a code-switching user is following, as they may have a high probability of code-switching as well. If the percentage of code-switching tweets can be increased, it would allow for more flexibility when selecting data to include in the set, as well as potentially lowering annotation costs if a particular percentage of code-switching content is required.

In total, the CrowdFlower jobs cost $1,541.62 for Spanish-English and $1,636.81 for Nepali-English. The token-level costs come out to $0.0088 per token for Spanish-English and $0.0087 per token for Nepali-English. This is far less expensive than the same three-way annotations would have cost if done in-lab, and without these low rates, a data set of this size would not have been possible for the project. When combined with the exceptional inter-annotator agreement observed between the CrowdFlower results and the lab, it is evident that CrowdFlower's customization and quality control measures can provide inexpensive, high-quality annotations.

8 Conclusion

Code-switching is a prevalent, complex, and growing aspect of communication – particularly in social media – which will not disappear any time soon. To keep up with this trend, natural language processing research must consider code-switched text, not just monolingual sources. We have detailed the methods and issues behind the development of multiple code-switching corpora of Twitter data, providing a point from which more of this research can branch forth.

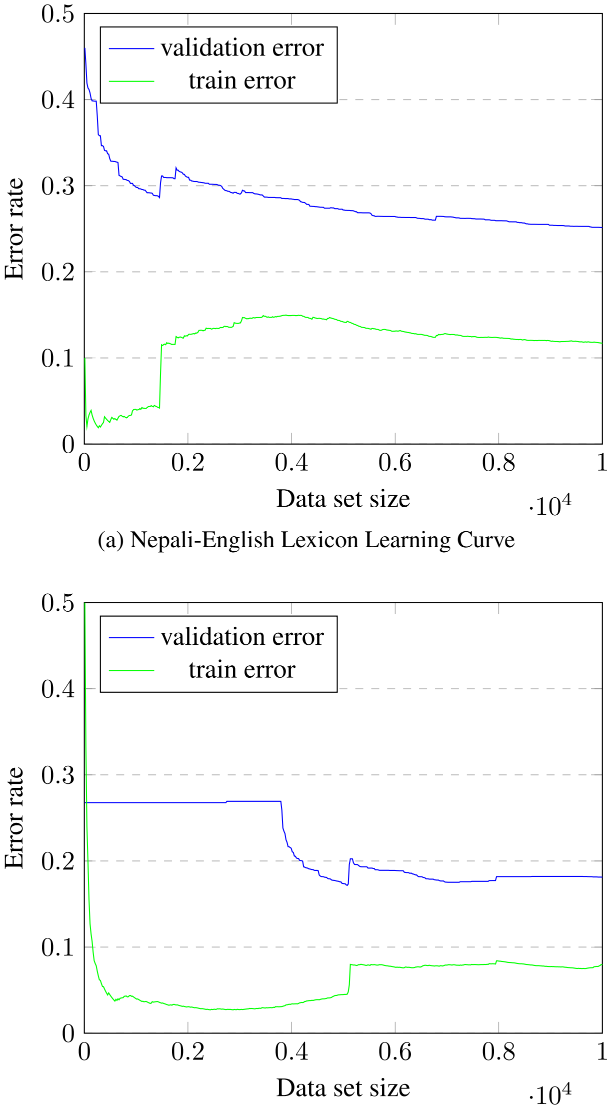
The corpora themselves can be useful to those seeking samples of data containing code-switched text, along with all of the noise that comes with social media data. The corpora contain a balance of code-switched text with monolingual and other types of data which have been tagged not only for the primary languages, but also for named entities, ambiguous and mixed words, and irrelevant characters. These annotations were primarily generated through crowd-sourcing, but their quality has been verified through high agreement with conventional, in-lab annotators.

We believe a major benefit of our research is the method of gathering and annotating the data, which we have described in detail, from the first steps of collection to the final review. Hopefully, these methods of searching for tweets and locating code-switching users can be helpful in the creation of data sets in broader scopes and additional language pairs. The approach to crowdsourcing via CrowdFlower that we have used has also provided us with good results and may be of use in further expansion. Potential improvements on these methods, such as gathering chains of followers for code-switching users on Twitter or attempting different instruction schemes on CrowdFlower, could provide even better results.

The datasets can be downloaded from the site: `http://emnlp2014.org/workshops/CodeSwitch/call.html`

Acknowledgments

We thank our collaborators Mona Diab and Julia Hirschberg for their contributions to the general annotation guidelines as well as the in lab annotators and CrowdFlower contributors for their labelling efforts. This research was partially funded by the National Science Foundation under award No. 1462142.



(a) Nepali-English Lexicon Learning Curve

(b) Spanish-English Lexicon Learning Curve

Figure 5: Learning curve for Nepali-English and Spanish-English

Appendix A. Spanish-English Code-switching Annotations for Twitter

1. WORD-LEVEL ANOTATIONS

Tokens that **start with a @ character**, **urls**, **emoticons** or any token that does not contain any letters such as **punctuation marks** and **numbers** (examples: ♥, ! , -_-, □, •• >, @____)), and the **{symbol}** tokens should all be labeled as '**None of the above**'.

If a number represents a word in the sentence it should be labeled as the language of that word instead of 'None of the above'. An example is 'I like 2 party.', but not 'Meet me in 2 hours.'

For tokens beginning with a # tag consider them as a single token and label them according to the regular word level guidelines.

1.1. Language

For each word in the Source, identify whether it is **Spanish**, **English**, **Mixed**, **Other**, **Ambiguous**, or **NE** (for named entities, which are proper names that represent names of people, places, organizations, locations, movie titles, and song titles). Below is an example showing the correct tags (labels) for each token in the source.

Source	**Language**	**Source**	**Language**
i	English	Tuesdays	English
always	English	Around	English
tell	English	6	None of the above
him	English	pero	Spanish
to	English	it	English
sing	English	's	English
to	English	not	English
me	English	worth	English
pero	Spanish	it	English
nunca	Spanish		
quiere	Spanish		

Ambiguous words

Ambiguous words are words that, in context, could belong to either language. This can happen because words such as `red`, `a`, `doctor`, `me`, and `can` are valid words in both languages. However, every instance of such a word is not ambiguous – only those instances where there is not enough context to decide whether the word is being used as English or Spanish. The fragment on the left shows an example where a potentially ambiguous word, `me`, is not ambiguous because the context helps identify the language, while the example on the right shows a truly ambiguous word, `NO`, which could be in either English or Spanish. Note that typos and misspellings should be labeled with the corresponding language.

Source	**Language**	**Source**	**Language**
i	English	Johnny	NE
always	English	Depp	NE
tell	English	para	Spanish
him	English	Dr.	NE
to	English	Strange	NE
sing	English	?..	None of the above
to	English	**NO**	**Ambiguous**
me	**English**		
pero	Spanish		
nunca	Spanish		
quiere	Spanish		

Mixed words

Mixed words are words that are partially in one language and partially in another. This can occur when the first part of a word is in English and the second part is in Spanish, or vice versa. The mixed category should only be used if the word clearly has a portion in one language and another portion in a different language. It is not for words that could exist entirely in either language (see Ambiguous).

Source	**Language**
@Sof_1D17	None of the above
Ayy	Spanish
que	Spanish
pepe	NE
snapchateame	**Mixed**
el	Spanish
arreglo	Spanish

Named Entities (NE)

This is a difficult section. Please read carefully. NEs are proper names. Examples of NEs are names that refer to people, places, organizations, locations, movie titles, and song titles. Named entities are usually, **but not always**, capitalized, so capitalization can’t be the only criterion to distinguish them. **Named entities can be multiple words, including articles (see the examples).** Examples of NEs and their tags are shown below.

Source	**Language**	**Source**	**Language**	**Source**	**Language**
Mejor	Spanish	and	English	@username	None of the above
Vente	Spanish	I	English	it	English
para	Spanish	told	English	‘s	English
el	Spanish	her	English	on	English
West	**NE**	to	English	**telemundo**	**NE**
Coast	**NE**	record	English	**el**	**NE**
and	English	**La**	**NE**	**señor**	**NE**
visit	English	**Reina**	**NE**	**de**	**NE**
me	English	**del**	**NE**	**los**	**NE**
lol	English	**Sur**	**NE**	**cielos**	**NE**

Abbreviations

Abbreviations should be labeled according to the full word(s) they represent. Some examples are shown below.

Source	**Language**	**Source**	**Language**	**Source**	**Language**
Mr.	**English**	**lol**	**English**	jajaja	Spanish
Smith	NE	yeah	English	**ntc**	**Spanish**
was	Spanish	I	English	gracias	Spanish
quejandose	Spanish	hear	English	por	Spanish
como	Spanish	you	English	todo	Spanish
siempre	Spanish	wey	Spanish		

Other

Languages other than Spanish or English should be labeled as Other. This category includes gibberish and unintelligible words. The example on the left shows some content that is not in English or Spanish (it is in Portuguese). The example on the right is an example of gibberish.

Source	**Language**	**Source**	**Language**
eu	**Other**	**Zaaas**	**Other**
voto	**Other**	viejas	Spanish
por	**Other**	zorras	Spanish
um	**Other**		
mundo	**Other**		
onde	**Other**		

References

Ron Artstein and Massimo Poesio. 2008. Inter-coder agreement for computational linguistics. *Computational Linguistics*, 34(4):555–596, December.

Shane Bergsma, Paul McNamee, Mossaab Bagdouri, Clayton Fink, and Theresa Wilson. 2012. Language identification for creating language-specific twitter collections. In *Proceedings of the Second Workshop on Language in Social Media*, pages 65–74, Montréal, Canada, June. Association for Computational Linguistics.

Chris Callison-Burch and Mark Dredze. 2010. Creating speech and language data with amazon's mechanical turk. In *Proceedings of the NAACL HLT 2010 Workshop on Creating Speech and Language Data with Amazon's Mechanical Turk*, pages 1–12, Los Angeles, June. Association for Computational Linguistics.

Karthik Dinakar, Roi Reichart, and Henry Lieberman. 2011. Modeling the detection of textual cyberbullying. In *Workshop on the Social Mobile Web*.

A. Joshi. 1982. Processing of sentences with intrasentential code-switching. In Ján Horecký, editor, *COLING-82*, pages 145–150, Prague, July.

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, June. Association for Computational Linguistics.

Lingpeng Kong, Nathan Schneider, Swabha Swayamdipta, Archna Bhatia, Chris Dyer, and Noah A. Smith. 2014. A dependency parser for tweets. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1001–1012, Doha, Qatar, October. Association for Computational Linguistics.

Ying Li, Yue Yu, and Pascale Fung. 2012. A mandarin-english code-switching corpus. In Nicoletta Calzolari, Khalid Choukri, Thierry Declerck, Mehmet Uğur Doğan, Bente Maegaard, Joseph Mariani, Jan Odijk, and Stelios Piperidis, editors, *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC-2012)*, pages 2515–2519, Istanbul, Turkey, May. European Language Resources Association (ELRA). ACL Anthology Identifier: L12-1573.

Constantine Lignos and Mitch Marcus. 2013. Toward web-scale analysis of codeswitching. In *87th Annual Meeting of the Linguistic Society of America*.

J. Lipski. 1978. Code-switching and the problem of bilingual competence. In M. Paradis, editor, *Aspects of bilingualism*, pages 250–264. Hornbeam.

D.C. Lyu, T.P. Tan, E. Chng, and H. Li. 2010. Seame: a mandarin-english code-switching speech corpus in south-east asia. In *INTERSPEECH*, volume 10, pages 1986–1989.

S. Gideon Mann and Andrew McCallum. 2008. Generalized expectation criteria for semi-supervised learning of conditional random fields. In *Proceedings of ACL-08: HLT*, pages 870–878. Association for Computational Linguistics.

Paul McNamee. 2005. Language identification: A solved problem suitable for undergraduate instruction. *J. Comput. Sci. Coll.*, 20(3):94–101, February.

Dong Nguyen and A. Seza Doğruöz. 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, Washington, USA, October. Association for Computational Linguistics.

S. Poplack. 1980. Sometimes I'll start a sentence in Spanish y termino en español: toward a typology of code-switching. *Linguistics*, 18(7/8):581–618.

Antonio Reyes, Paolo Rosso, and Tony Veale. 2013. A multidimensional approach for detecting irony in twitter. *Language Resources and Evaluation*, 47(1):239–268.

Kirk Roberts, Michael A. Roach, Joseph Johnson, Josh Guthrie, and Sanda M. Harabagiu. 2012. Empatweet: Annotating and detecting emotions on twitter. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC-2012)*. European Language Resources Association (ELRA).

Anil Kumar Singh and Jagadeesh Gorla. 2007. Identification of languages and encodings in a multilingual document. In *Proceedings of ACL-SIGWAC's Web As Corpus3*, Belgium.

Thamar Solorio and Yang Liu. 2008a. Learning to predict code-switching points. In *Empirical Methods on Natural Language Processing, EMNLP-2008*, pages 973–981, Honolulu, Hawaii, October. Association for Computational Linguistics.

Thamar Solorio and Yang Liu. 2008b. Part-of-speech tagging for English-Spanish code-switched text. In *Empirical Methods on Natural Language Processing, EMNLP-2008*, pages 1051–1060, Honolulu, Hawaii, October. Association for Computational Linguistics.

Irina Temnikova, Andrea Varga, and Dogan Biyikli. 2014. Building a crisis management term resource for social media: The case of floods and protests. In Nicoletta Calzolari (Conference Chair), Khalid Choukri, Thierry Declerck, Hrafn Loftsson, Bente Maegaard, Joseph Mariani, Asuncion Moreno, Jan Odijk, and Stelios Piperidis, editors, *Proceedings of the Ninth International Conference on Language Resources and Evalu-*

ation (LREC'14), Reykjavik, Iceland, may. European Language Resources Association (ELRA).

Marc Tomlinson, David Bracewell, Wayne Krug, and David Hinote. 2014. #mygoal: Finding motivations on twitter. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC-2014)*. European Language Resources Association (ELRA).

Jennifer Williams and Graham Katz. 2012. A new twitter verb lexicon for natural language processing. In Nicoletta Calzolari (Conference Chair), Khalid Choukri, Thierry Declerck, Mehmet Uur Doan, Bente Maegaard, Joseph Mariani, Asuncion Moreno, Jan Odijk, and Stelios Piperidis, editors, *Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC'12)*, Istanbul, Turkey, may. European Language Resources Association (ELRA).

H. Zhai, T. Lingren, L. Deleger, Q. Li, M. Kaiser, L. Stoutenborough, and I. Solti. 2013. Web 2.0-based crowdsourcing for high-quality gold standard development in clinical natural processing. *Journal of Medical Internet Research*, 15(4). Retrieved May 15, 2014 from `http://www.ncbi.nlm.nih.gov/pmc/articles/PMC3636329/`.