Labeling the Languages of Words in Mixed-Language Documents using Weakly Supervised Methods

Ben King
Department of EECS
University of Michigan
Ann Arbor, MI
benking@umich.edu

Steven Abney
Department of Linguistics
University of Michigan
Ann Arbor, MI
abney@umich.edu

Abstract

In this paper we consider the problem of labeling the languages of words in mixed-language documents. This problem is approached in a weakly supervised fashion, as a sequence labeling problem with monolingual text samples for training data. Among the approaches evaluated, a conditional random field model trained with generalized expectation criteria was the most accurate and performed consistently as the amount of training data was varied.

1 Introduction

Language identification is a well-studied problem (Hughes et al., 2006), but it is typically only studied in its canonical text-classification formulation, identifying a document's language given sample texts from a few different languages. But there are several other interesting and useful formulations of the problem that have received relatively little attention. Here, we focus on the problem of labeling the languages of individual words within a multilingual document. To our knowledge, this is the first paper to specifically address this problem.

Our own motivation for studying this problem stems from issues encountered while attempting to build language resources for minority languages. In trying to extend parts of Kevin Scannell's Crúbadán project (Scannell, 2007), which automatically builds minority language corpora from the Web, we found that the majority of webpages that contain text in a minority language also contain text in other languages. Since Scannell's method builds these corpora by bootstrapping from the pages that were retrieved, the corpus-building process can go disastrously wrong without accounting for this problem. And any resources, such as a lexicon, created from the corpus will also be incorrect.

In this paper, we explore techniques for performing language identification at the word level in mixed language documents. Our results show that one can do better than independent word language classification, as there are clues in a word's context: words of one language are frequently surrounded by words in the same language, and many documents have patterns that may be marked by the presence of certain words or punctuation. The methods in this paper also outperform sentence-level language identification, which is too coarse to capture most of the shifts between language.

To evaluate our methods, we collected and manually annotated a corpus of over 250,000 words of bilingual (though mostly non-parallel) text from the web. After running several different weakly-supervised learning methods, we found that a conditional random field model trained with generalized expectation criteria is the most accurate and performs quite consistently as the amount of training data is varied.

In section 2, we review the related work. In section 3, we define the task and describe the data and its annotation. Because the task of language identification for individual words has not been explicitly studied in the literature, and because of its importance to the overall task, we examine the features and methods that work best for independent word language identification in section 4. We begin to ex-

Proceedings of NAACL-HLT 2013, pages 1110–1119,
Atlanta, Georgia, 9–14 June 2013.

amine the larger problem of labeling the language of words in context in section 5 by describing our methods. In section 6, we describe the evaluation and present the results. We present our error analysis in section 7 and conclude in section 8.

2 Related Work

Language identification is one of the older NLP problems (Beesley, 1988), especially in regards to spoken language (House and Neuburg, 1977), and has received a fair share of attention through the years (Hughes et al., 2006). In its standard formulation, language identification assumes monolingual documents and attempts to classify each document according to its language from some closed set of known languages.

Many approaches have been proposed, such as Markov models (Dunning, 1994), Monte Carlo methods (Poutsma, 2002), and more recently support vector machines with string kernels, but nearly all approaches use the n-gram features first suggested by (Cavnar and Trenkle, 1994). Performance of language identification is generally very high with large documents, usually in excess of 99% accuracy, but Xia et al. (2009) mention that current methods still can perform quite poorly when the class of potential languages is very large or the texts to be classified are very short.

This paper attempts to address three of the ongoing issues specifically mentioned by Hughes et al. (2006) in their survey of textual language identification: supporting minority languages, sparse or impoverished training data, and multilingual documents.

A number of methods have been proposed in recent years to apply to the problems of unsupervised and weakly-supervised learning. Excluding self- and co-training methods, these methods can be categorized into two broad classes: those which bootstrap from a small number of tokens (sometimes called prototypes) (Collins and Singer, 1999; Haghighi and Klein, 2006), and those which impose constraints on the underlying unsupervised learning problem (Chang et al., 2007; Bellare et al., 2009; Druck et al., 2008; Ganchev et al., 2010).

Constraint-based weakly supervised learning has been applied to some sequence labeling problems, through such methods as contrastive estimation (Smith and Eisner, 2005), generalized expectation criteria (Mann and McCallum, 2008), alternating projections (Singh et al., 2010), and posterior regularization (Ganchev et al., 2010).

Perhaps the work that is most similar to this work is the study of *code-switching* within NLP literature. Most of the work done has been on automatically identifying code-switch points (Joshi, 1982; Solorio and Liu, 2008). The problem of identifying language in the presence of code-switching has seen the most attention in the realm of speech processing (Chu et al., 2007; Lyu and Lyu, 2008), among many others. Though code-switching has been well-studied linguistically, it is only one possible reason to explain why a document contains multiple languages, and is actually one of the less common causes observed in our corpus. For that reason, we approach this problem more generally, assuming no specific generative process behind multilingual text.

3 Task Definition

The task we describe in this paper is a sequence labeling problem, labeling a word in running text according to the language to which it belongs. In the interest of being able to produce reliable human annotations, we limit ourselves to texts with exactly two languages represented, though the techniques developed in this paper would certainly be applicable to documents with more than two languages. The two languages represented in the paper are known *a priori* by the labeler and the only training data available to the labeler is a small amount of sample text in each of the two languages represented.

In most NLP sequence labeling problems, the researchers can safely assume that each sequence (but not each item in the sequence) is independent and identically distributed (*iid*) according to some underlying distribution common to all the documents. For example, it is safe to assume that a sentence drawn from WSJ section 23 can be labeled by a model trained on the other sections. With the task of this paper we cannot assume that sequences from different documents are *iid*, (*e.g.* One document may have 90% of its words in Basque, while another only has 20%), but we do make the simplifying as-

sumption that sequences within the same document are *iid*.

Because of this difference, the labeler is presented each document separately and must label its words independently of any other document. And the training data for this task is not in the form of labeled sequences. Rather, the models in this task are given two monolingual example texts which are used only to learn a model for individual instances. Any sequential dependencies between words must be bootstrapped from the document. It is this aspect of the problem that makes it well-suited for weakly-supervised learning.

It is worth considering whether this problem is best approached at the word level, or if perhaps sentence- or paragraph-level language identification would suffice for this task. In those cases, we could easily segment the text at the sentence or paragraph level and feed those segments to an existing language identifier. To answer this question we segmented our corpus into sentences by splitting at every period, exclamation point, or question mark (an overly agressive approximation of sentence segmentation). Even if every sentence was given the correct majority label under this sentence segmentation, the maximum possible word-level accuracy that a sentence-level classifier could achieve is 85.8%, and even though this number reflects quite optimistic conditions, it is still much lower than the methods of this paper are able to achieve.

3.1 Evaluation Data

To build a corpus of mixed language documents, we used the BootCat tool (Baroni and Bernardini, 2004) seeded with words from a minority language. BootCat is designed to automatically collect webpages on a specific topic by repeatedly searching for keywords from a topic-specific set of seed words. We found that this method works equally well for languages as for topics, when seeded with words from a specific language. Once BootCat returned a collection of documents, we manually identified documents from the set that contained text in both the target language and in English, but did not contain text in any other languages. Since the problem becomes trivial when the languages do not share a character set, we limited ourselves to languages with a Latin orthography.

Language	# words	Language	# words
Azerbaijani	4114	Lingala	1359
Banjar	10485	Lombard	18512
Basque	5488	Malagasy	6779
Cebuano	17994	Nahuatl	1133
Chippewa	15721	Ojibwa	24974
Cornish	2284	Oromo	28636
Croatian	17318	Pular	3648
Czech	886	Serbian	2457
Faroese	8307	Slovak	8403
Fulfulde	458	Somali	11613
Hausa	2899	Sotho	8198
Hungarian	9598	Tswana	879
Igbo	11828	Uzbek	43
Kiribati	2187	Yoruba	4845
Kurdish	531	Zulu	20783

Table 1: Languages present in the corpus and their number of words before separating out English text.

We found that there was an important balance to be struck concerning the popularity of a language. If a language is not spoken widely enough, then there is little chance of finding any text in that language on the Web. Conversely if a language is too widely spoken, then it is difficult to find mixed-language pages for it. The list of languages present in the corpus and the number of words in each language reflects this balance as seen in Table 1.

For researchers who wish to make use this data, the set of annotations used in this paper is available from the first author's website[1].

3.2 Annotation

Before the human annotators were presented with the mixed-language documents fetched by BootCat, the documents were first stripped of all HTML markup, converted to Unicode, and had HTML escape sequences replaced with the proper Unicode characters. Documents that had any encoding errors (*e.g.* original page used a mixture of encodings) were excluded from the corpus.

[1] `http://www-personal.umich.edu/~benking/resources/mixed-language-annotations-release-v1.0.tgz`

ENG:	because of LUTARU.Thank you ntate T.T! Sevice...
SOT:	Retselisitsoemonethi ekare jwale hotla sebetswa ...
ENG:	Lesotho is heading 4 development #big-ups Mr ...
SOT:	Basotho bare monoana hao its'upe.
ENG:	Just do the job and lets see what you are made ...
SOT:	Malerato Mokoena Ntate Thabane, molimo ...
ENG:	It is God who reigns and if God is seen in your ...
SOT:	Mathabo Letsie http://www.facebook.com/taole. ...
ENG:	As Zuma did he should introduce a way of we can ...
SOT:	Msekhotho Matona a rona ha a hlomamisoe, re ...

Table 2: An example of text from an annotated English-Sotho web page.

Since there are many different reasons that the language in a document may change (*e.g.* code-switching, change of authors, borrowing) and many variations thereof, we attempted to create a broad set of annotation rules that would cover many cases, rather than writing a large number of very specific rules. In cases when the language use was ambiguous, the annotators were instructed simply to make their best guess. Table 2 shows an example of an annotated document.

Generally, only well-digested English loanwords and borrowings were to be marked as belonging to the foreign language. If a word appeared in the context of both languages, it was permissible for that word to receive different labels at different times, depending on its context.

Ordinary proper names (like "John Williams" or "Chicago") were to be marked as belonging to the language of the context in which they appear. This rule also applied to abbreviations (like "FIFA" or "BBC"). The exception to this rule was proper names composed of common nouns (like "Stairway to Heaven" or "American Red Cross") and to abbreviations that spelled out English words, which were to be marked as belonging to the language of the words they were composed of.

The annotators were instructed not to assign labels to numbers or punctuation, but they were allowed to use numbers as punctuation as clues for assigning other labels.

3.3 Human Agreement

To verify that the annotation rules were reasonable and led to a problem that could potentially be solved by a computer, we had each of the annotators mark up a small shared set of a few hundred words from each of eight documents, in order to measure the inter-annotator agreement.

The average actual agreement was 0.988, with 0.5 agreement expected by chance for a kappa of 0.975.

Language	# words	Language	# words
Azerbaijani	211	Lingala	1816
Banjar	450	Lombard	2955
Basque	1378	Malagasy	4038
Cebuano	1898	Nahuatl	3544
Chippewa	92	Ojibwa	167
Cornish	2096	Oromo	1443
Croatian	1505	Pular	1285
Czech	1503	Serbian	1515
English	16469	Slovak	1504
Faroese	1585	Somali	1871
Fulfulde	1097	Sotho	2154
Hausa	2677	Tswana	2191
Hungarian	1541	Uzbek	1533
Igbo	2079	Yoruba	2454
Kiribati	1891	Zulu	1075
Kurdish	1674		

Table 3: Number of total words of training data for each language.

3.4 Training Data

Following Scannell (2007), we collected small monolingual samples of 643 languages from four sources: the Universal Declaration of Human Rights[2], non-English Wikipedias[3], the Jehovah's Witnesses website[4], and the Rosetta project (Landsbergen, 1989).

Only 30 of these languages ended up being used in experiments. Table 3 shows the sizes of the monolingual samples of the languages used in this paper.

[2]The Universal Declaration of Human Rights is a document created by the United Nations and translated into many languages. As of February 2011 there were 365 versions available from `http://www.unicode.org/udhr/`

[3]As of February 2011, there were 113 Wikipedias in different languages. Current versions of Wikipedia can be accessed from `http://meta.wikimedia.org/wiki/List_of_Wikipedias`

[4]As of February 2011, there were 310 versions of the site available at `http://www.watchtower.org`

They range from 92 for Chippewa to 16469 for English. Most of the languages have between 1300 and 1600 words in their example text. To attempt to mitigate variation caused by the sizes of these language samples, we sample an equal number of words with replacement from each of English and a second language to create the training data.

4 Word-level Language Classification

We shift our attention momentarily to a subproblem of the overall task: independent word-level language classification. While the task of language identification has been studied extensively at the document, sentence, and query level, little or no work has been done at the level of an individual word. For this reason, we feel it is prudent to formally evaluate the features and classifiers which perform most effectively at the task of word language classification (ignoring any sequential dependencies at this point).

4.1 Features

We used a logistic regression classifier to experiment with combinations of the following features: character unigrams, bigrams, trigrams, 4-grams, 5-grams, and the full word. For these experiments, the training data consisted of 1000 words sampled uniformly with replacement from the sample text in the appropriate languages. Table 4 shows the accuracies that the classifier achieved when using different sets of features averaged over 10 independent runs.

Features	Accuracy
Unigrams	0.8056
Bigrams	0.8783
Trigrams	0.8491
4-grams	0.7846
5-grams	0.6977
{1,2,3,4,5}-grams	0.8817
{1,2,3,4,5}-grams, word	0.8819

Table 4: Logistic regression accuracy when trained using varying features.

The use of all available features seems to be the best option, and we use the full set of features in all proceeding experiments. This result also concurs with the findigs of (Cavnar and Trenkle, 1994), who found 1-5-grams to be most effective for document language classification.

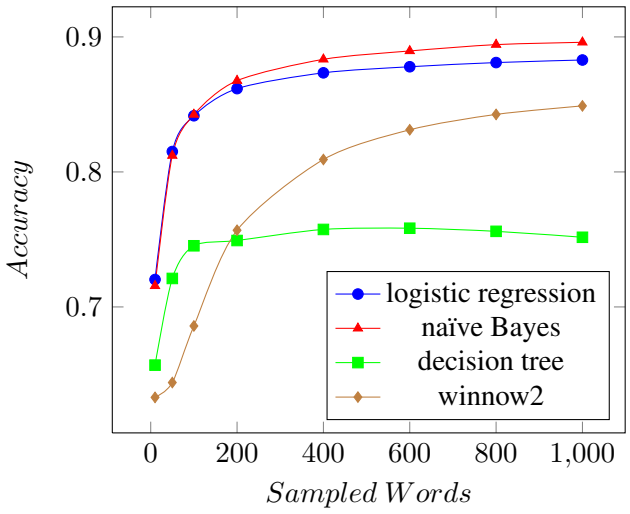

Figure 1: Learning curves for logistic regression, naïve Bayes, decision tree, and Winnow2 on the independent word classification problem as the number of sampled words in each training example changes from 10 to 1000.

4.2 Classifiers

Using all available features, we compare four MALLET (McCallum, 2002) classifiers: logistic regression, naïve Bayes, decision tree, and Winnow2. Figure 1 shows the learning curves for each classifier as the number of sampled words comprising each training example is varied from 10 to 1000.

Since a naïve Bayes classifier gave the best performance in most experiments, we use naïve Bayes as a representative word classifier for the rest of the paper.

5 Methods

Moving onto the main task of this paper, labeling *sequences* of words in documents according to their languages, we use this section to describe our methods.

Since training data for this task is limited and is of a different type than the evaluation data (labeled instances from monolingual example texts *vs.* labeled sequences from the multilingual document), we approach the problem with weakly- and semi-supervised methods.

The sequence labeling methods are presented with a few new sequence-relevant features, which are not applicable to independent word classification (since these features do not appear in the training data):

- a feature for the presence of each possible non-word character (punctuation or digit) between the previous and the current words
- a feature for the presence of each possible non-word character between the current and next words

In addition to independent word classification, which was covered in section 4, we also implement a conditional random field model trained with generalized expectation criteria, a hidden Markov model (HMM) trained with expectation maximization (EM), and a logistic regression model trained with generalized expectation criteria.

We had also considered that a semi-Markov CRF (Sarawagi and Cohen, 2004) could be useful if we could model segment lengths (a non-Markovian feature), but we found that gold-standard segment lengths did not seem to be distributed according to any canonical distribution, and we did not have a reliable way to estimate these segment lengths.

5.1 Conditional Random Field Model trained with Generalized Expectation

Generalized expectation (GE) criteria (Druck et al., 2008) are terms added to the objective function of a learning algorithm which specify preferences for the learned model. When the model is a linear chain conditional random field (CRF) model, we can straightforwardly express these criteria in the objective function with a KL-divergence term between the expected values of the current model $\tilde{p}$ and the preferred model $\hat{p}$ (Mann and McCallum, 2008).

$$\mathcal{O}(\theta; D, U) = \sum_d \log p_\theta(y^{(d)}|x^{(d)}) - \frac{\sum_k \theta_k}{2\sigma^2} - \lambda D(\hat{p}||\tilde{p}_\theta)$$

Practically, to compute these expectations, we produce the smoothed MLE on the output label distribution for every feature observed in the training data. For example, the trigram "ter" may occur 27 times in the English sample text and 34 times in the other sample text, leading to an MLE of $\hat{p}(\texttt{eng}|\texttt{ter}) \approx 0.44$.

Because we do not expect the true marginal label distribution to be uniform (*i.e.* the document may not have equal numbers of words in each language), we first estimate the expected marginal label distribution by classifying each word in the document independently using naïve Bayes and taking the resulting counts of labels produced by the classifier as an MLE estimate for it: $\hat{p}(\texttt{eng})$ and $\hat{p}(\texttt{non})$.

We use these terms to bias the expected label distributions over each feature. Let $\mathcal{F}_{eng}$ and $\mathcal{F}_{non}$ respectively be the collections of all training data features with the two labels. For every label $l \in \mathcal{L} = \{\texttt{eng}, \texttt{non}\}$ and every feature $f \in \mathcal{F}_{eng} \cup \mathcal{F}_{non}$, we calculate

$$p(l|f) = \frac{count(f, \mathcal{F}_l) + \delta}{count(f, \bigcup_i \mathcal{F}_i) + \delta|\mathcal{L}|} \times \frac{\hat{p}(l)}{p_{uniform}(l)},$$

the biased maximum likelihood expected output label distribution. To avoid having $p(l|f) = 0$, which can cause the KL-divergence to be undefined, we perform additive smoothing with $\delta = 0.5$ on the counts before multiplying with the biasing term.

We use the implementation of CRF with GE criteria from MALLET (McCallum, 2002), which uses a gradient descent algorithm to optimize the objective function. (Mann and McCallum, 2008; Druck, 2011)

5.2 Hidden Markov Model trained with Expectation Maximization

A second method we used was a hidden Markov model (HMM) trained iteratively using the Expectation Maximization algorithm (Dempster et al., 1977). Here an HMM is preferable to a CRF because it is a generative model and therefore uses parameters with simple interpretations. In the case of an HMM, it is easy to estimate emission and transition probabilities using an external method and then set these directly.

To initialize the HMM, we use a uniform distribution for transition probabilities, and produce the emission probabilities by using a naïve Bayes classifier trained over the two small language samples.

In the expectation step, we simply pass the document through the HMM and record the labels it produces for each word in the document.

In the maximization step, we produce maximum-likelihood estimates for transition probabilities from the transitions between the labels produced. To estimate emission probabilities, we retrain a naïve Bayes classifier on the small language samples along the set of words from the document that were labeled as being in the respective language. We iterated this process until convergence, which usually took fewer than 10 iterations.

We additionally experimented with a naïve Bayes classifier trained by EM in the same fashion, except that it had no transition probabilities to update. This classifier's performance was almost identical to that of the GE-trained MaxEnt method mentioned in the following section, so we omit it from the results and analysis for that reason.

5.3 Logistic Regression trained with Generalized Expectation

GE criteria can also be straightforwardly applied to the weakly supervised training of logistic regression models. The special case where the constraints specified are over marginal label distributions, is called *label regularization*.

As with the CRF constraint creation, here we first use an ordinary supervised naïve Bayes classifier in order to estimate the marginal label distributions for the document, which can be used to create more accurate output label expectations that are biased to the marginal label distributions over all words in the document.

We use the MALLET implementation of a GE-trained logistic regression classifier, which optimizes the objective function using a gradient descent algorithm.

5.4 Word-level Classification

Our fourth method served as a baseline and did not involve any sequence labeling, only independent classification of words. Since naïve Bayes was the best performer among word classification methods, we use that the representative of independent word classification methods. The implementation of the naïve Bayes classifier is from MALLET.

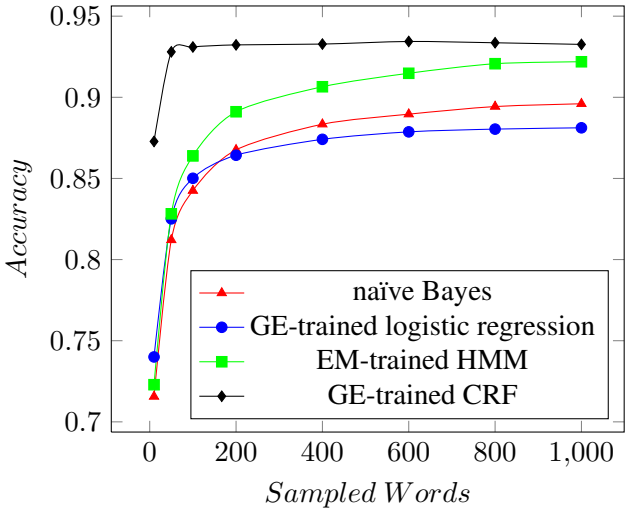

Figure 2: Learning curves for naïve Bayes, logistic regression trained with GE, HMM trained with EM, and CRF trained with GE as the number of sampled words in each training example changes from 10 to 1000.

We also implemented a self-trained CRF, initially trained on the output of this naïve Bayes classifier, and trained on its own output in subsequent iterations. This method was not able to consistently outperform the naïve Bayes classifier after any number of iterations.

6 Evaluation and Results

We evaluated each method using simple token-level accuracy, *i.e.* whether the correct label was assigned to a word in the document. Word boundaries were defined by punctuation or whitespace, and no tokens containing a digit were included. Figure 2 displays the accuracy for each method as the number of sampled words from each language example is varied from 10 to 1000.

In all the cases we tested, CRF trained with GE is clearly the most accurate option among the methods examined, though the EM-trained HMM seemed to be approaching a similar accuracy with large amounts of training data. With a slight edge in efficiency also in its favor, we think the GE+CRF approach, rather than EM+HMM, is the best approach for this problem because of its consistent performance across a wide range of training data sizes. In its favor, the EM+HMM approach has a slightly

lower variance in its performance across different files, though not at a statistically significant level.

Contrary to most of the results in (Mann and McCallum, 2010), a logistic regression classifier trained with GE did not outperform a standard supervised naïve Bayes classifier. We suspect that this is due to the different nature of this problem as compared to most other sequence labeling problems, with the classifier bootstrapping over a single document only. In the problems studied by Mann and McCallum, the GE-trained classifier was able to train over the entire training set, which was on average about 50,000 instances, far more than the number of words in the average document in this set (2,500).

7 Error Analysis

In order to analyze the types of mistakes that the models made we performed an error analysis on ten randomly selected files, looking at each mislabeled word and classifying the error according to its type. The results of this analysis are in Table 5. The three classes of errors are (1) named entity errors, when a named entity is given a label that does not match the label it was given in the original annotation, (2) shared word errors, when a word that could belong to either language is classified incorrectly, and (3) other, a case that covers all other types of errors.

Method	NE	SW	Other
GE+CRF	41%	10%	49%
EM+HMM	50%	14%	35%
GE+MaxEnt	37%	12%	51%
Naïve Bayes	42%	17%	40%

Table 5: Types of errors and their proportions among the different methods. NE stands for Named Entity, SW stands for Shared Word, and Other covers all other types of errors.

Our annotation rules for named entities specified that named entities should be given a label matching their context, but this was rather arbitrary, and not explicitly followed by any of the methods, which treat a named entity as if it was any other token. This was the one of most frequent types of error made by each of the methods and in our conclusion in section 8, we discuss ways to improve it.

In a regression analysis to determine which factors had the greatest correlations with the GE-trained CRF performance, the estimated proportion of named entities in the document had by far the greatest correlation with CRF accuracy of anything we measured. Following that in decreasing order of correlation strength were the cosine similarity between English and the document's second language, the number of words in the monolingual example text (even though we sampled from it), and the average length of gold-standard monolingual sequences in the document.

The learning curve for GE-trained CRF in Figure 2 is somewhat atypical as far as most machine learning methods are concerned: performance is typically non-decreasing as more training data is made available.

We believe that the model is becoming over-constrained as more words are used to create the constraints. The GE method does not have a way to specify that some of the soft constraints (for the labels observed most frequently in the sample text) should be more important than other constraints (those observed less frequently). When we measure the KL-divergence between the label distributions predicted by the constraints and the true label distribution, we find that this divergence seems to reach its minimum value between 600 and 800 words, which is where the GE+CRF also seems to reach its maximum performance.

The step with a naïve Bayes classifier estimating the marginal label distribution ended up being quite important overall. Without it, the accuracy dropped by more than a full percentage point absolute. But the problem of inaccurate constraint estimation is one that needs further consideration. Some possible ways to address it may be to prune the constraints according to their frequency or perhaps according to a metric like entropy, or to vary the GE-criteria coefficient in the objective function in order to penalize the model less for varying from the expected model.

8 Conclusion

This paper addresses three of the ongoing issues specifically mentioned by Hughes et al. (2006) in their survey of textual language identification. Our approach is able to *support minority languages*; in

fact, almost all of the languages we tested on would be considered minority languages. We also address the issue of *sparse or impoverished training data*. Because we use weakly-supervised methods, we are able to successfully learn to recognize a language with as few as 10 words of training data[5]. The last and most obvious point we address is that of *multilingual documents*, which is the focus of the paper.

We present a weakly-supervised system for identifying the languages of individual words in mixed-language documents. We found that across a broad range of training data sizes, a CRF model trained with GE criteria is an accurate sequence classifier and is preferable to other methods for several reasons.

One major issue to be improved upon in future work is how named entities are handled. A straightforward way to approach this may be to create another label for named entities, which (for the purposes of evaluation) would be considered not to belong to any of the languages in the document. We could simply choose not to evaluate a system on the named entity tokens in a document. Alternatively, the problem of language-independent named entity recognition has received some attention in the past (Tjong Kim Sang and De Meulder, 2003), and it may be beneficial to incorporate such a system in a robust word-level language identification system.

Going forward, an issue that needs to be addressed with this method is its dependence on knowing the set of possible languages *a priori*. Because we don't see an easy way to adapt this method to accurately label words in documents from a possible set of thousands of languages when the document itself may only contain two or three languages, we would propose the following future work.

We propose a two-step approach to general word-level language identification. The first step would be to examine a multilingual document, and with high accuracy, list the languages that are present in the document. The second step would be identical to the approach described in this paper (but with the two-language restriction lifted), and would be responsible for labeling the languages of individual words, using the set of languages provided by the first step.

[5]With only 10 words of each language as training data, the CRF approach correctly labels 88% of words

References

Marco Baroni and Silvia Bernardini. 2004. Bootcat: Bootstrapping corpora and terms from the web. In *Proceedings of the Fourth International Conference on Langauge Resources and Evaluation (LREC 2004)*, volume 4, pages 1313–1316, Lisbon, Portugal.

Kenneth R. Beesley. 1988. Language identifier: A computer program for automatic natural-language identification of on-line text. In *Proceedings of the 29th Annual Conference of the American Translators Association*, volume 47, page 54.

Kedar Bellare, Gregory Druck, and Andrew McCallum. 2009. Alternating projections for learning with expectation constraints. In *Proceedings of the Twenty-Fifth Conference on Uncertainty in Artificial Intelligence*, pages 43–50. AUAI Press.

William B. Cavnar and John M. Trenkle. 1994. N-gram-based text categorization. In *Proceedings of the Third Annual Symposium on Document Analysis and Information (SDAIR 94)*, pages 161–175, Las Vegas, Nevada.

Ming-Wei Chang, Lev Ratinov, and Dan Roth. 2007. Guiding semi-supervision with constraint-driven learning. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, pages 280–287, Prague, Czech Republic, June. Association for Computational Linguistics.

Chyng-Leei Chu, Dau-cheng Lyu, and Ren-yuan Lyu. 2007. Language identification on code-switching speech. In *Proceedings of ROCLING*.

Michael Collins and Yoram Singer. 1999. Unsupervised models for named entity classification. In *Proceedings of the Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora*, pages 100–110.

Arthur P. Dempster, Nan M. Laird, and Donald B. Rubin. 1977. Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 1–38.

Gregory Druck, Gideon Mann, and Andrew McCallum. 2008. Learning from labeled features using generalized expectation criteria. In *Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval (SIGIR 2008)*, pages 595–602. ACM.

Gregory Druck. 2011. *Generalized Expectation Criteria for Lightly Supervised Learning*. Ph.D. thesis, University of Massachusetts Amherst.

Ted Dunning. 1994. Statistical identification of language. Technical report.

Kuzman Ganchev, João Graça, Jennifer Gillenwater, and Ben Taskar. 2010. Posterior regularization for structured latent variable models. *The Journal of Machine Learning Research*, 11:2001–2049.

Aria Haghighi and Dan Klein. 2006. Prototype-driven learning for sequence models. In *Proceedings of the Human Language Technology Conference of the NAACL, Main Conference*, pages 320–327, New York City, USA, June. Association for Computational Linguistics.

A.S. House and E.P. Neuburg. 1977. Toward automatic identification of the language of an utterance. i. preliminary methodological considerations. *The Journal of the Acoustical Society of America*, 62:708.

Baden Hughes, Timothy Baldwin, Steven Bird, Jeremy Nicholson, and Andrew MacKinlay. 2006. Reconsidering language identification for written language resources. In *Proc. International Conference on Language Resources and Evaluation*, pages 485–488.

Aravind K. Joshi. 1982. Processing of sentences with intra-sentential code-switching. In *Proceedings of the 9th conference on Computational linguistics-Volume 1*, pages 145–150. Academia Praha.

Jan Landsbergen. 1989. The rosetta project. pages 82–87, Munich, Germany.

Dau-Cheng Lyu and Ren-Yuan Lyu. 2008. Language identification on code-switching utterances using multiple cues. In *Ninth Annual Conference of the International Speech Communication Association*.

Gideon S. 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, Columbus, Ohio, June. Association for Computational Linguistics.

Gideon S. Mann and Andrew McCallum. 2010. Generalized expectation criteria for semi-supervised learning with weakly labeled data. *The Journal of Machine Learning Research*, pages 955–984.

Andrew McCallum. 2002. Mallet: A machine learning for language toolkit. http://mallet.cs.umass.edu.

Arjen Poutsma. 2002. Applying monte carlo techniques to language identification. *Language and Computers*, 45(1):179–189.

Sunita Sarawagi and William W. Cohen. 2004. Semimarkov conditional random fields for information extraction. *Advances in Neural Information Processing Systems (NIPS 2004)*, 17:1185–1192.

Kevin P. Scannell. 2007. The crúbadán project: Corpus building for under-resourced languages. In *Building and Exploring Web Corpora: Proceedings of the 3rd Web as Corpus Workshop*, volume 4, pages 5–15, Louvain-la-Neuve, Belgium.

Sameer Singh, Dustin Hillard, and Chris Leggetter. 2010. Minimally-supervised extraction of entities from text advertisements. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 73–81, Los Angeles, California, June. Association for Computational Linguistics.

Noah A. Smith and Jason Eisner. 2005. Contrastive estimation: Training log-linear models on unlabeled data. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL'05)*, pages 354–362, Ann Arbor, Michigan, June. Association for Computational Linguistics.

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

Erik F. Tjong Kim Sang and Fien De Meulder. 2003. Introduction to the conll-2003 shared task: Language-independent named entity recognition. In Walter Daelemans and Miles Osborne, editors, *Proceedings of the Seventh Conference on Natural Language Learning at HLT-NAACL 2003*, pages 142–147.

Fei Xia, William Lewis, and Hoifung Poon. 2009. Language ID in the context of harvesting language data off the web. In *Proceedings of the 12th Conference of the European Chapter of the ACL (EACL 2009)*, pages 870–878, Athens, Greece, March. Association for Computational Linguistics.