Automatic Detection and Language Identification of Multilingual Documents

Marco Lui[♡♣], **Jey Han Lau**[♠] **and Timothy Baldwin**[♡♣]
[♡] Department of Computing and Information Systems
The University of Melbourne
[♣] NICTA Victoria Research Laboratory
[♠] Department of Philosophy
King's College London
mhlui@unimelb.edu.au, jeyhan.lau@gmail.com, tb@ldwin.net

Abstract

Language identification is the task of automatically detecting the language(s) present in a document based on the content of the document. In this work, we address the problem of detecting documents that contain text from more than one language (*multilingual* documents). We introduce a method that is able to detect that a document is multilingual, identify the languages present, and estimate their relative proportions. We demonstrate the effectiveness of our method over synthetic data, as well as real-world multilingual documents collected from the web.

1 Introduction

Language identification is the task of automatically detecting the language(s) present in a document based on the content of the document. Language identification techniques commonly assume that every document is written in one of a closed set of known languages for which there is training data, and is thus formulated as the task of selecting the most likely language from the set of training languages. In this work, we remove this monolingual assumption, and address the problem of language identification in documents that may contain text from more than one language from the candidate set. We propose a method that concurrently detects that a document is multilingual, and estimates the proportion of the document that is written in each language.

Detecting multilingual documents has a variety of applications. Most natural language processing techniques presuppose monolingual input data, so inclusion of data in foreign languages introduces noise, and can degrade the performance of NLP systems (Alex et al., 2007; Cook and Lui, 2012). Automatic detection of multilingual documents can be used as a pre-filtering step to improve the quality of input data. Detecting multilingual documents is also important for acquiring linguistic data from the web (Scannell, 2007; Abney and Bird, 2010), and has applications in mining bilingual texts for statistical machine translation from online resources (Resnik, 1999; Nie et al., 1999; Ling et al., 2013). There has been particular interest in extracting text resources for low-density languages from multilingual web pages containing both the low-density language and another language such as English (Yamaguchi and Tanaka-Ishii, 2012; King and Abney, 2013). King and Abney (2013, p1118) specifically mention the need for an automatic method "to examine a multilingual document, and with high accuracy, list the languages that are present in the document".

We introduce a method that is able to detect multilingual documents, and simultaneously identify each language present as well as estimate the proportion of the document written in that language. We achieve this with a probabilistic mixture model, using a document representation developed for monolingual language identification (Lui and Baldwin, 2011). The model posits that each document is generated as samples from an unknown mixture of languages from the training set. We introduce a Gibbs sampler to map samples to languages for any given set of languages, and use this to select the set of languages that maximizes the posterior probability of the document.

Transactions of the Association for Computational Linguistics, 2 (2014) 27–40. Action Editor: Kristina Toutanova.
Submitted 1/2013; Revised 7/2013; Published 2/2014.

Our method is able to learn a language identifier for multilingual documents from monolingual training data. This is an important property as there are no standard corpora of multilingual documents available, whereas corpora of monolingual documents are readily available for a reasonably large number of languages (Lui and Baldwin, 2011). We demonstrate the effectiveness of our method empirically, firstly by evaluating it on synthetic datasets drawn from Wikipedia data, and then by applying it to real-world data, showing that we are able to identify multilingual documents in targeted web crawls of minority languages (King and Abney, 2013).

Our main contributions are: (1) we present a method for identifying multilingual documents, the languages contained therein and the relative proportion of the document in each language; (2) we show that our method outperforms state-of-the-art methods for language identification in multilingual documents; (3) we show that our method is able to estimate the proportion of the document in each language to a high degree of accuracy; and (4) we show that our method is able to identify multilingual documents in real-world data.

2 Background

Most language identification research focuses on language identification for *monolingual* documents (Hughes et al., 2006). In monolingual LangID, the task is to assign each document D a unique language $L_i \in L$. Some work has reported near-perfect accuracy for language identification of large documents in a small number of languages (Cavnar and Trenkle, 1994; McNamee, 2005). However, in order to attain such accuracy, a large number of simplifying assumptions have to be made (Hughes et al., 2006; Baldwin and Lui, 2010a). In this work, we tackle the assumption that each document is monolingual, i.e. it contains text from a single language.

In language identification, documents are modeled as a stream of characters (Cavnar and Trenkle, 1994; Kikui, 1996), often approximated by the corresponding stream of bytes (Kruengkrai et al., 2005; Baldwin and Lui, 2010a) for robustness over variable character encodings. In this work, we follow Baldwin and Lui (2010a) in training a single model for languages that naturally use multiple encodings (e.g. UTF8, Big5 and GB encodings for Chinese), as issues of encoding are not the focus of this research.

The document representation used for language identification generally involves estimating the relative distributions of particular byte sequences, selected such that their distributions differ between languages. In some cases the relevant sequences may be externally specified, such as function words and common suffixes (Giguet, 1995) or grammatical word classes (Dueire Lins and Gonçalves, 2004), though they are more frequently learned from labeled data (Cavnar and Trenkle, 1994; Grefenstette, 1995; Prager, 1999a; Lui and Baldwin, 2011).

Learning algorithms applied to language identification fall into two general categories: Bayesian classifiers and nearest-prototype (Rocchio-style) classifiers. Bayesian approaches include Markov processes (Dunning, 1994), naive Bayes methods (Grefenstette, 1995; Lui and Baldwin, 2011; Tiedemann and Ljubešić, 2012), and compressive models (Teahan, 2000). The nearest-prototype methods vary primarily in the distance measure used, including measures based on rank order statistics (Cavnar and Trenkle, 1994), information theory (Baldwin and Lui, 2010a), string kernels (Kruengkrai et al., 2005) and vector space models (Prager, 1999a; McNamee, 2005).

Language identification has been applied in domains such as USENET messages (Cavnar and Trenkle, 1994), web pages (Kikui, 1996; Martins and Silva, 2005; Liu and Liang, 2008), web search queries (Ceylan and Kim, 2009; Bosca and Dini, 2010), mining the web for bilingual text (Resnik, 1999; Nie et al., 1999), building minority language corpora (Ghani et al., 2004; Scannell, 2007; Bergsma et al., 2012) as well as a large-scale database of Interlinear Glossed Text (Xia et al., 2010), and the construction of a large-scale multilingual web crawl (Callan and Hoy, 2009).

2.1 Multilingual Documents

Language identification over documents that contain text from more than one language has been identified as an open research question (Hughes et al., 2006). Common examples of multilingual documents are web pages that contain excerpts from another language, and documents from multilingual organizations such as the European Union.

	English	French	Italian	German	Dutch	Japanese
character	the␣	pour	␣di␣	␣auf	voo	は
byte	74 68 65 20	70 6F 75 7	20 64 69 20	20 61 75 66	76 6F 6	E3 81 AF

Table 1: Examples of per-language byte sequences selected by information gain.

The Australiasian Language Technology Workshop 2010 hosted a shared task where participants were required to predict the language(s) present in a held-out test set containing monolingual and bilingual documents (Baldwin and Lui, 2010b). The dataset was prepared using data from Wikipedia, and bilingual documents were produced using a segment from a page in one language, and a segment from the same page in another language. We use the dataset from this shared task for our initial experiments.

To the authors' knowledge, the only other work to directly tackle identification of multiple languages and their relative proportions in a single document is the LINGUINI system (Prager, 1999a). The system is based on a vector space model, and cosine similarity between a feature vector for the test document and a feature vector for each language L_i, computed as the sum of feature vectors for all the documents for language L_i in the training data. The elements in the feature vectors are frequency counts over byte n-grams ($2 \leq n \leq 5$) and words. Language identification for multilingual documents is performed through the use of *virtual mixed languages*. Prager (1999a) shows how to construct vectors representative of particular combinations of languages independent of the relative proportions, and proposes a method for choosing combinations of languages to consider for any given document.

Language identification in multilingual documents could also be performed by application of supervised language segmentation algorithms. Given a system that can segment a document into labeled monolingual segments, we can then extract the languages present as well as the relative proportion of text in each language. Several methods for supervised language segmentation have been proposed. Teahan (2000) proposed a system based on text compression that identifies multilingual documents by first segmenting the text into monolingual blocks. Rehurek and Kolkus (2009) perform language segmentation by computing a relevance score between terms and languages, smoothing across adjoining terms and finally identifying points of transition between high and low relevance, which are interpreted as boundaries between languages. Yamaguchi and Tanaka-Ishii (2012) use a minimum description length approach, embedding a compressive model to compute the description length of text segments in each language. They present a linear-time dynamic programming solution to optimize the location of segment boundaries and language labels.

3 Methodology

Language identification for multilingual documents is a multi-label classification task, in which a document can be mapped onto any number of labels from a closed set. In the remainder of this paper, we denote the set of all languages by L. We denote a document D which contains languages L_x and L_y as $D \rightarrow \{L_x, L_y\}$, where $L_x, L_y \in L$. We denote a document that does not contain a language L_x by $D \rightarrow \{\overline{L_x}\}$, though we generally omit all the languages not contained in the document for brevity. We denote classifier output using $\triangleright$; e.g. $D \triangleright \{L_a, L_b\}$ indicates that document D has been predicted to contain text in languages L_a and L_b.

3.1 Document Representation and Feature Selection

We represent each document D as a frequency distribution over byte n-gram sequences such as those in Table 1. Each document is converted into a vector where each entry counts the number of times a particular byte n-gram is present in the document. This is analogous to a bag-of-words model, where the vocabulary of "words" is a set of byte sequences that has been selected to distinguish between languages.

The exact set of features is selected from the training data using Information Gain (IG), an information-theoretic metric developed as a splitting criterion for decision trees (Quinlan, 1993). IG-based feature selection combined with a naive Bayes classifier has been shown to be particularly effective for language identification (Lui and Baldwin, 2011).

3.2 Generative Mixture Models

Generative mixture models are popular for text modeling tasks where a mixture of influences governs the content of a document, such as in multi-label document classification (McCallum, 1999; Ramage et al., 2009), and topic modeling (Blei et al., 2003). Such models normally assume full exchangeability between tokens (i.e. the bag-of-words assumption), and label each token with a single discrete label.

Multi-label text classification, topic modeling and our model for language identification in multilingual documents share the same fundamental representation of the latent structure of a document. Each label is modeled with a probability distribution over tokens, and each document is modeled as a probabilistic mixture of labels. As presented in Griffiths and Steyvers (2004), the probability of the i^{th} token (w_i) given a set of T labels $z_1 \cdots z_T$ is modeled as:

$$P(w_i) = \sum_{j=1}^{T} P(w_i|z_i = j)P(z_i = j) \qquad (1)$$

The set of tokens w is the document itself, which in all cases is observed. In the case of topic modeling, the tokens are words and the labels are topics, and z is latent. Whereas topic modeling is generally unsupervised, multi-label text classification is a supervised text modeling task, where the labels are a set of pre-defined categories (such as RUBBER, IRON-STEEL, TRADE, etc. in the popular Reuters-21578 data set (Lewis, 1997)), and the tokens are individual words in documents. z is still latent, but constrained in the training data (i.e. documents are labeled but the individual words are not). Some approaches to labeling unseen documents require that z for the training data be inferred, and methods for doing this include an application of the Expectation-Maximization (EM) algorithm (McCallum, 1999) and Labeled LDA (Ramage et al., 2009).

The model that we propose for language identification in multilingual documents is similar to multi-label text classification. In the framework of Equation 1, each per-token label z_i is a language and the vocabulary of tokens is not given by words but rather by specific byte sequences (Section 3.1). The key difference with multi-label text classification is that we use monolingual (i.e. mono-label) training data. Hence, z is effectively observed for the training data (since all tokens must share the same label). To infer z for unlabeled documents, we utilize a Gibbs sampler, closely related to that proposed by Griffiths and Steyvers (2004) for LDA. The sampling probability for a label z_i for token w in a document d is:

$$P(z_i = j|z_{-i}, w) \propto \phi_j^{(w)} \cdot \theta_j^{(d)} \qquad (2)$$
$$\phi_j^{(w)} = P(w_i|z_i = j, z_{-i}, w_{-i})$$
$$\theta_j^{(d)} = P(z_i = j|z_{-i})$$

In the LDA model, $\theta_j^{(d)}$ is assumed to have a Dirichlet distribution with hyperparameter α, and the word distribution for each topic $\phi_j^{(w)}$ is also assumed to have a Dirichlet distribution with hyperparameter β. Griffiths (2002) describes a generative model for LDA where both $\phi_j^{(w)}$ and $\theta_j^{(d)}$ are inferred from the output of a Gibbs sampler. In our method, we estimate $\phi_j^{(w)}$ using maximum likelihood estimation (MLE) from the training data. Estimating $\phi_j^{(w)}$ through MLE is equivalent to a multinomial Naive Bayes model (McCallum and Nigam, 1998):

$$\hat{\phi}_j^{(w)} = \frac{n_j^{(w)} + \beta}{n_j^{(.)} + W\beta} \qquad (3)$$

where $n_j^{(w)}$ is the number of times word w occurs with label j, and $n_j^{(.)}$ is the total number of words that occur with label j. By setting β to 1, we obtain standard Laplacian smoothing. Hence, only $\hat{\theta}_j^{(d)}$ is updated at each step in the Gibbs sampler:

$$\hat{\theta}_j^{(d)} = \frac{n_{-i,j}^{(d)} + \alpha}{n_{-i}^{(d)} + T\alpha} \qquad (4)$$

where $n_{-i,j}^{(d)}$ is the number of tokens in document d that are currently mapped to language j, and $n_{-i}^{(d)}$ is the total number of tokens in document d. In both cases, the current assignment of z_i is excluded from the count. T is the number of languages (i.e. the size of the label set). For simplicity, we set α to 0. We note that in the LDA model, α and β influence the sparsity of the solution, and so it may be possible to tune these parameters for our model as well. We leave this as an avenue for further research.

3.3 Language Identification in Multilingual Documents

The model described in Section 3.2 can be used to compute the most likely distribution to have generated an unlabeled document over a given set of languages for which we have monolingual training data, by letting the set of terms w be the byte n-gram sequences we selected using per-language information gain (Section 3.1), and allowing the labels z to range over the set of all languages L. Using training data, we compute $\hat{\phi}_j^{(w)}$ (Equation 3), and then we infer $P(L_j|D)$ for each $L_j \in L$ for the unlabeled document, by running the Gibbs sampler until the samples for z_i converge and then tabulating z_i over the whole d and normalizing by $|d|$. Naively, we could identify the languages present in the document by $D \triangleright \{L_x \text{ if } \exists(z_i = L_x|D)\}$, but closely-related languages tend to have similar frequency distributions over byte n-gram features, and hence it is likely that some tokens will be incorrectly mapped to a language that is similar to the "correct" language.

We address this issue by finding the subset of languages λ from the training set L that maximizes $P(\lambda|D)$ (a similar approach is taken in McCallum (1999)). Through an application of Bayes' theorem, $P(\lambda|D) \propto P(D|\lambda) \cdot P(\lambda)$, noting that $P(D)$ is a normalizing constant and can be dropped. We assume that $P(\lambda)$ is constant (i.e. any subset of languages is equally likely, a reasonable assumption in the absence of other evidence), and hence maximize $P(D|\lambda)$. For any given $D = w_1 \cdots w_n$ and λ, we infer $P(D|\lambda)$ from the output of the Gibbs sampler:

$$P(D|\lambda) = \prod_{i=1}^{N} P(w_i|\lambda) \quad (5)$$

$$= \prod_{i=1}^{N} \sum_{j \in \lambda} P(w_i|z_i = j) P(z_i = j) \quad (6)$$

where both $P(w_i|z_i = j)$ and $P(z_i = j)$ are estimated by their maximum likelihood estimates.

In practice, exhaustive evaluation of the powerset of L is prohibitively expensive, and so we greedily approximate the optimal λ using Algorithm 1. In essence, we initially rank all the candidate languages by computing the most likely distribution over the full set of candidate languages. Then, for each of the top-N languages in turn, we consider whether to add it to λ. λ is initialized with L_u, a dummy language with a uniform distribution over terms (i.e. $P(w|L_u) = \frac{1}{|w|}$). A language is added if it improves $P(D|\lambda)$ by at least t. The threshold t is required to suppress the addition of spurious classes. Adding languages gives the model additional freedom to fit parameters, and so will generally increase $P(D|\lambda)$. In the limit case, adding a completely irrelevant language will result in no tokens being mapped to the a language, and so the model will be no worse than without the language. The threshold t is thus used to control "how much" improvement is required before including the new language in λ.

Algorithm 1 $DetectLang(L, D)$

```
L_N ← top-N z ∈ L by P(z|D)
λ ← {L_u}
for each L_t ∈ L_N do
    λ' ← λ ∪ L_t
    if P(D|λ) + t < P(D|λ') then
        λ ← λ'
    end if
end for
λ ← λ \ {L_u}
return D ▷ λ
```

3.4 Benchmark Approaches

We compare our approach to two methods for language identification in multilingual documents: (1) the *virtual mixed languages* approach (Prager, 1999a); and (2) the text segmentation approach (Yamaguchi and Tanaka-Ishii, 2012).

Prager (1999a) describes LINGUINI, a language identifier based on the vector-space model commonly used in text classification and information retrieval. The document representation used by Prager (1999a) is a vector of counts across a set of character sequences. Prager (1999a) selects the feature set based on a TFIDF-like approach. Terms with occurrence count $m < n \times k$ are rejected, where m is the number of times the term occurs in the training data (the TF component), n is the number of languages in which the term occurred (the IDF component, where "document" is replaced with "language"), and k is a parameter to control the overall number of terms selected. In Prager (1999a), the value of k is reported to be optimal in the region 0.3 to 0.5. In practice,

the value of k indirectly controls the number of features selected. Values of k are not comparable across datasets as m is not normalized for the size of the training data, so in this work we do not report the values of k and instead directly select the top-N features, weighted by $\frac{m}{n}$. In LINGUINI, each language is modeled as a single pseudo-document, obtained by concatenating all the training data for the given language. A document is then classified according to the vector with which it has the smallest angle; this is implemented by finding the language vector with the highest cosine with the document vector.

Prager (1999a) also proposes an extension to the approach to allow identification of bilingual documents, and suggests how this may be generalized to any number of languages in a document. The gist of the method is simple: for any given pair of languages, the projection of a document vector onto the hyperplane containing the language vectors of the two languages gives the mixture proportions of the two languages that minimizes the angle with the document vector. Prager (1999a) terms this projection a *virtual mixed language* (VML), and shows how to find the angle between the document vector and the VML. If this angle is less than that between the document vector and any individual language vector, the document is labeled as bilingual in the two languages from which the mixed vector was derived. The practical difficulty presented by this approach is that exhaustively evaluating all possible combinations of languages is prohibitively expensive. Prager (1999a) addresses this by arguing that in multilingual documents, "the individual component languages will be close to d (the document vector) – probably closer than most or all other languages". Hence, language mixtures are only considered for combinations of the top m languages.

Prager (1999a) shows how to obtain the mixture coefficients for bilingual VMLs, arguing that the process generalizes. Prager (1999b) includes the coefficients for 3-language VMLs, which are much more complex than the 2-language variants. Using a computer algebra system, we verified the analytic forms of the coefficients in the 3-language VML. We also attempted to obtain an analytic form for the coefficients in a 4-language VML, but these were too complex for the computer algebra system to compute. Thus, our evaluation of the VML approach proposed by Prager (1999a) is limited to 3-language VMLs. Neither Prager (1999a) nor Prager (1999b) include an empirical evaluation over multilingual documents, so to the best of our knowledge this paper is the first empirical evaluation of the method on multilingual documents. As no reference implementation of this method is available, we have produced our own implementation, which we have made freely available.[1]

The other benchmark we consider in this paper is the method for text segmentation by language proposed by Yamaguchi and Tanaka-Ishii (2012) (hereafter referred to as SEGLANG). The actual task addressed by Yamaguchi and Tanaka-Ishii (2012) is to divide a document into monolingual segments. This is formulated as the task of segmenting a document $D = x_1, \cdots, x_{|D|}$ (where x_i denotes the i^{th} character of D and $|D|$ is the length of the document) by finding a list of boundaries $B = [B_1, \cdots, B_{|B|}]$ where each B_i indicates the location of a language boundary as an offset from the start of the document, resulting in a list of segments $X = [X_0, \cdots, X_{|B|}]$. For each segment X_i, the system predicts L_i, the language associated with the segment, producing a list of labellings $L = [L_0, \cdots, L_{|B|}]$, with the constraint that adjacent elements in L must differ. Yamaguchi and Tanaka-Ishii (2012) solve the problem of determining X and L for an unlabeled text using a method based on minimum description length. They present a dynamic programming solution to this problem, and analyze a number of parameters that affect the overall accuracy of the system. Given this method to determine X and L, it is then trivial to label an unlabeled document according to $D \triangleright \{L_x \text{ if } \exists L_x \in L\}$, and the length of each segment in X can then be used to determine the proportions of the document that are in each language. In this work, we use a reference implementation of SEGLANG kindly provided to us by the authors.

Using the text segmentation approach of SEGLANG to detect multilingual documents differs from LINGUINI and our method primarily in that LINGUINI and our method fragment the document into small sequences of bytes, and discard information about the relative order of the fragments. This is in contrast to SEGLANG, where this information

[1] https://github.com/saffsd/linguini.py

System	$\mathcal{P}_M$	$\mathcal{R}_M$	$\mathcal{F}_M$	$\mathcal{P}_\mu$	$\mathcal{R}_\mu$	$\mathcal{F}_\mu$
Benchmark	.497	.467	.464	.833	.826	.829
Winner	.718	.703	.699	.932	.931	.932
SEGLANG	.801	.810	.784	.866	.946	.905
LINGUINI	.616	.535	.513	.713	.688	.700
Our method	.753	.771	.748	.945	.922	.933

Table 2: Results on the ALTW2010 dataset. "Benchmark" is the benchmark system proposed by the shared task organizers. "Winner" is the highest-$\mathcal{F}_\mu$ system submitted to the shared task.

is utilized in the sequential prediction of labels for consecutive segments of text, and is thus able to make better use of the locality of text (since there are likely to be monolingual blocks of text in any given multilingual document). The disadvantage of this is that the underlying model becomes more complex and hence more computationally expensive, as we observe in Section 5.

3.5 Evaluation

We seek to evaluate the ability of each method: (1) to correctly identify the language(s) present in each test document; and (2) for multilingual documents, to estimate the relative proportion of the document written in each language. In the first instance, this is a classification problem, and the standard notions of precision ($\mathcal{P}$), recall ($\mathcal{R}$) and F-score ($\mathcal{F}$) apply. Consistent with previous work in language identification, we report both the document-level *micro-average*, as well as the language-level *macro-average*. For consistency with Baldwin and Lui (2010a), the macro-averaged F-score we report is the average of the per-class F-scores, rather than the harmonic mean of the macro-averaged precision and recall; as such, it is possible for the F-score to not fall between the precision and recall values. As is common practice, we compute the F-score for $\beta = 1$, giving equal importance to precision and recall.[2] We tested the difference in performance for statistical significance using an approximate randomization procedure (Yeh, 2000) with 10000 iterations. Within each table of results (Tables 2, 3 and 4), all differences between systems are statistically significant at a $p < 0.05$ level.

To evaluate the predictions of the relative proportions of a document D written in each detected language L_i, we compare the topic proportion predicted by our model to the gold-standard proportion, measured as a byte ratio as follows:

$$gs(L_i|D) = \frac{\text{length of } L_i \text{ part of } D \text{ in bytes}}{\text{length of } D \text{ in bytes}} \quad (7)$$

We report the correlation between predicted and actual proportions in terms of Pearson's r coefficient. We also report the mean absolute error (MAE) over all document–language pairs.

4 Experiments on ALTW2010

Our first experiment utilizes the ALTW2010 shared task dataset (Baldwin and Lui, 2010b), a synthetic dataset of 10000 bilingual documents[3] generated from Wikipedia data, introduced in the ALTW2010 shared task,[4] The dataset is organized into training, development and test partitions. Following standard machine learning practice, we train each system using the training partition, and tune parameters using the development partition. We then report macro and micro-averaged precision, recall and F-score on the test partition, using the tuned parameters.

The results on the ALTW2010 shared task dataset are summarized in Table 2. Each of the three systems we compare was re-trained using the training data provided for the shared task, with a slight difference: in the shared task, participants were provided with multilingual training documents, but the systems targeted in this research require monolingual training data. We thus split the training documents into monolingual segments using the metadata provided with the dataset. The metadata was only published after completion of the task and was not available to task participants. For comparison, we have included the benchmark results published by the shared task organizers, as well as the score attained by the winning entry (Tran et al., 2010).

[2] Intuitively, it may seem that the maximal precision and recall should be achieved when precision and recall are balanced. However, because of the multi-label nature of the task and variable number of labels assigned to a given document by our models, it is theoretically possible and indeed common in our results for the maximal macro-averaged F-score to be achieved when macro-averaged precision and recall are not balanced.

[3] With a small number of monolingual documents, formed by randomly selecting the two languages for a given document independently, leaving the possibility of the same two languages being selected.

[4] http://comp.mq.edu.au/programming/task_description/

We tune the parameters for each system using the development partition of the dataset, and report results on the test partition. For LINGUINI, there is a single parameter k to be tuned: the number of features per language. We tested values between 10000 and 50000, and selected 46000 features as the optimal value. For our method, there are two parameters to be tuned: (1) the number of features selected for each language, and (2) the threshold t for including a language. We tested features-per-language counts between 30 and 150, and found that adding features beyond 70 per language had minimal effect. We tested values of the threshold t from 0.01 to 0.15, and found the best value was 0.14. For SEGLANG, we introduce a threshold t on the minimum proportion of a document (measured in bytes) that must be labeled by a language before that language is included in the output set. This was done because our initial experiments indicate that SEGLANG tends to over-produce labels. Using the development data, we found the best value of t was 0.10.

We find that of the three systems tested, two outperform the winning entry to the shared task. This is more evident in the macro-averaged results than in the micro-averaged results. In micro-averaged terms, our method is the best performer, whereas on the macro-average, SEGLANG has the highest F-score. This suggests that our method does well on higher-density languages (relative to the ALTW2010 dataset), and poorly on lower-density languages. This also accounts for the higher micro-averaged precision but lower micro-averaged recall for our method as compared to SEGLANG. The improved macro-average F-score of SEGLANG comes at a much higher computational cost, which increases dramatically as the number of languages is increased. In our testing on a 16-core workstation, SEGLANG took almost 24 hours to process the ALTW2010 shared task test data, compared to 2 minutes for our method and 40 seconds for LINGUINI. As such, SEGLANG is poorly suited to detecting multilingual documents where a large number of candidate languages is considered.

The ALTW2010 dataset is an excellent starting point for this research, but it predominantly contains bilingual documents, making it difficult to assess the ability of systems to distinguish multilingual documents from monolingual ones. Furthermore, we are unable to use it to assess the ability of systems to detect more than 2 languages in a document. To address these shortcomings, we construct a new dataset in a similar vein. The dataset and experiments performed on it are described in the next section.

5 Experiments on WIKIPEDIAMULTI

To fully test the capabilities of our model, we generated WIKIPEDIAMULTI, a dataset that contains a mixture of monolingual and multilingual documents. To allow for replicability of our results and to facilitate research in language identification, we have made the dataset publicly available.[5] WIKIPEDIAMULTI is generated using excerpts from the mediawiki sources of Wikipedia pages downloaded from the Wikimedia foundation.[6] The dumps we used are from July–August 2010.

To generate WIKIPEDIAMULTI, we first normalized the raw mediawiki documents. Mediawiki documents typically contain one paragraph per line, interspersed with structural elements. We filtered each document to remove all structural elements, and only kept documents that exceeded 2500 bytes after normalization. This yielded a collection of around 500,000 documents in 156 languages. From this initial document set (hereafter referred to as WIKICONTENT), we only retained languages that had more than 1000 documents (44 languages), and generated documents for WIKIPEDIAMULTI as follows:

1. randomly select the number of languages K ($1 \leq K \leq 5$)
2. randomly select a set of K languages $S = \{L_i \in L \text{ for } i = 1 \cdots K\}$ without replacement
3. randomly select a document for each $L_i \in S$ from WIKICONTENT without replacement
4. take the top $\frac{1}{K}$ lines of the document
5. join the K sections into a single document.

As a result of the procedure, the relative proportion of each language in a multilingual document tends not to be uniform, as it is conditioned on the length of the original document from which it was sourced, independent of the other $K-1$ for the other languages that it was combined with. Overall, the average document length is 5500 bytes (standard deviation = 3800 bytes). Due to rounding up in taking

[5] http://www.csse.unimelb.edu.au/~tim/

[6] http://dumps.wikimedia.org

System	$\mathcal{P}_M$	$\mathcal{R}_M$	$\mathcal{F}_M$	$\mathcal{P}_\mu$	$\mathcal{R}_\mu$	$\mathcal{F}_\mu$
SEGLANG	.809	.975	.875	.771	.975	.861
LINGUINI	.853	.772	.802	.838	.774	.805
Our method	.962	.954	.957	.963	.955	.959

Table 3: Results on the WIKIPEDIAMULTI dataset.

the top $\frac{1}{k}$ lines (step 4), documents with higher K tend to be longer (6200 bytes for $K = 5$ vs 5100 bytes for $K = 1$).

The WIKIPEDIAMULTI dataset contains training, development and test partitions. The training partition consists of 5000 monolingual (i.e. $K = 1$) documents. The development partition consists of 5000 documents, 1000 documents for each value of K where $1 \leq K \leq 5$. The test partition contains 200 documents for each K, for a total of 1000 documents. There is no overlap between any of the partitions.

5.1 Results over WIKIPEDIAMULTI

We trained each system using the monolingual training partition, and tuned parameters using the development partition. For LINGUINI, we tested feature counts between 10000 and 50000, and found that the effect was relatively small. We thus use 10000 features as the optimum value. For SEGLANG, we tested values for threshold t between 0.01 and 0.20, and found that the maximal macro-averaged F-score is attained when $t = 0.06$. Finally, for our method we tested features-per-language counts between 30 and 130 and found the best performance with 120 features per language, although the actual effect of varying this value is rather small. We tested values of the threshold t for adding an extra language to λ from 0.01 to 0.15, and found that the best results were attained when $t = 0.02$.

The results of evaluating each system on the test partition are summarized in Table 3. In this evaluation, our method clearly outperforms both SEGLANG and LINGUINI. The results on WIKIPEDIAMULTI and ALTW2010 are difficult to compare directly due to the different compositions of the two datasets. ALTW2010 is predominantly bilingual, whereas WIKIPEDIAMULTI contains documents with text in 1–5 languages. Furthermore, the average document in ALTW2010 is half the length of that in WIKIPEDIAMULTI. Overall, we observe that SEGLANG has a tendency to over-label (despite the introduction of the t parameter to reduce this effect), evidenced by high recall but lower precision. LINGUINI is inherently limited in that it is only able to detect up to 3 languages per document, causing recall to suffer on WIKIPEDIAMULTI. However, it also tends to always output 3 languages, regardless of the actual number of languages in the document, hurting precision. Furthermore, even on ALTW2010 it has lower recall than the other two systems.

6 Estimating Language Proportions

In addition to detecting multiple languages within a document, our method also estimates the relative proportions of the document that are written in each language. This information may be useful for detecting documents that are candidate bitexts for training machine translation systems, since we may expect languages in the document to be present in equal proportions. It also allows us to identify the predominant language of a document.

A core element of our model of a document is a distribution over a set of labels. Since each label corresponds to a language, as a first approximation, we take the probability mass associated with each label as a direct estimate of the proportion of the document written in that language. We examine the results for predicting the language proportions in the test partition of WIKIPEDIAMULTI. Mapping label distributions directly to language proportions produces excellent results, with a Pearson's r value of 0.863 and an MAE of 0.108.

Although labels have a one-to-one correspondence with languages, the label distribution does not actually correspond directly to the language proportion, because the distribution estimates the proportion of byte n-gram sequences associated with a label and not the proportion of bytes directly. The same number of bytes in different languages can produce different numbers of n-gram sequences, because after feature selection not all n-gram sequences are retained in the feature set. Hereafter, we refer to each n-gram sequence as a *token*, and the average number of tokens produced per byte of text as the *token emission rate*.

We estimate the per-language token emission rate (Figure 1) using the training partition of WIKIPEDIAMULTI. To improve our estimate of the language proportions, we correct our label distribution

Original text `the␣cat␣in␣the␣hat`

n-gram features

`he␣` : 2	`the␣` : 2
`␣hat` : 1	`␣in␣` : 1
`␣th` : 1	`␣the` : 1
`hat` : 1	`he␣c` : 1
`in␣t` : 1	`n␣th` : 1

Emission rate $\frac{\#\text{bytes}}{\#\text{tokens}} = \frac{18}{12} = 1.5$ bytes/token

Figure 1: Example of calculating n-gram emission rate for a text string.

using estimates of the per-language token emission rate R_{L_i} in bytes per token for $L_i \in L$. Assume that a document D of length $|D|$ is estimated to contain K languages in proportions P_i for $i = 1 \cdots K$. The corrected estimate for the proportion of L_i is:

$$Prop(L_i) = \frac{P_i \times R_{L_i}}{\sum_{j=1}^{K} (P_j \times R_{L_j})} \quad (8)$$

Note that the $|D|$ term is common to the numerator and denominator and has thus been eliminated.

This correction improves our estimates of language proportions. After correction, the Pearson's r rises to 0.981, and the MAE is reduced to 0.024. The improvement is most noticeable for language–document pairs where the proportion of the document in the given language is about 0.5 (Figure 2).

7 Real-world Multilingual Documents

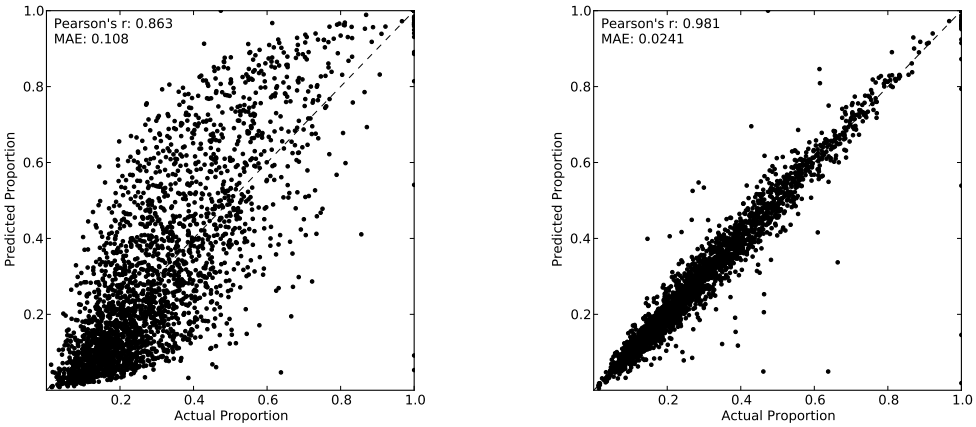
So far, we have demonstrated the effectiveness of our proposed approach using synthetic data. The results have been excellent, and in this section we validate the approach by applying it to a real-world task that has recently been discussed in the literature. Yamaguchi and Tanaka-Ishii (2012) and King and Abney (2013) both observe that in trying to gather linguistic data for "non-major" languages from the web, one challenge faced is that documents retrieved often contain sections in another language. SEGLANG (the solution of Yamaguchi and Tanaka-Ishii (2012)) concurrently detects multilingual documents and segments them by language, but the approach is computationally expensive and has a tendency to over-label (Section 5). On the other hand, the solution of King and Abney (2013) is incomplete, and they specifically mention the need for an automatic method "to examine a multilingual document, and with high accuracy, list the languages that are present in the document". In this section, we show that our method is able to fill this need. We make use of manually-annotated data kindly provided to us by Ben King, which consists of 149 documents containing 42 languages retrieved from the web using a set of targeted queries for low-density languages. Note that the dataset described in King and Abney (2013) was based on manual confirmation of the presence of English in addition to the low-density language of primary interest; our dataset contains these bilingual documents as well as monolingual documents in the low-density language of interest. Our purpose in this section is to investigate the ability of automatic systems to select this subset of bilingual documents. Specifically, given a collection of documents retrieved for a target language, the task is to identify the documents that contain text in English in addition to the target language. Thus, we re-train each system for each target language, using only training data for English and the target language. We reserve the data provided by Ben King for evaluation, and train our methods using data separately obtained from the Universal Declaration of Human Rights (UDHR). Where UDHR translations for a particular language were not available, we used data from Wikipedia or from a bible translation. Approximately 20–80 kB of data were used for each language. As we do not have suitable development data, we made use of the best parameters for each system from the experiments on WIKIPEDIAMULTI.

System	$\mathcal{P}$	$\mathcal{R}$	$\mathcal{F}$
Baseline	0.719	1.00	0.837
SEGLANG	0.779	0.991	0.872
LINGUINI	0.729	0.981	0.837
Our method	0.907	0.916	0.912

Table 4: Detection accuracy for English-language inclusion in web documents from targeted web crawls for low-density languages.

We find that all 3 systems are able to detect that each document contains the target language with 100% accuracy. However, systems vary in their ability to detect if a document also contains English in addition to the target language. The detection accuracy for English-language inclusion is summarized in Table 4.[7] For comparison, we include a heuristic baseline based on labeling all documents as contain-

[7] Note that Table 2 and Table 3 both report macro and micro-averaged results across a number of languages. In contrast Table 4 only reports results for English, and the values are not directly comparable to our earlier evaluation.



(a) without emission rate correction

(b) with emission rate correction

Figure 2: Scatterplot of the predicted vs. actual language proportions in a document for the test partition of WIKIPEDIAMULTI (predictions are from our method; each point corresponds to a document-language pair).

ing English. We find that, like the heuristic baseline, SEGLANG and LINGUINI both tend to overlabel documents, producing false positive labels of English, resulting in increased recall at the expense of precision. Our method produces less false positives (but slightly more false negatives). Overall, our method attains the best $\mathcal{F}$ for detecting English inclusions. Manual error analysis suggests that the false negatives for our method generally occur where a relatively small proportion of the document is written in English.

8 Future Work

Document segmentation by language could be accomplished by a combination of our method and the method of King and Abney (2013), which could be compared to the method of Yamaguchi and Tanaka-Ishii (2012) in the context of constructing corpora for low-density languages using the web. Another area we have identified in this paper is the tuning of the parameters α and β in our model (currently $\alpha = 0$ and $\beta = 1$), which may have some effect on the sparsity of the model.

Further work is required in dealing with cross-domain effects, to allow for "off-the-shelf" language identification in multilingual documents. Previous work has shown that it is possible to generate a document representation that is robust to variation across domains (Lui and Baldwin, 2011), and we intend to investigate if these results are also applicable to language identification in multilingual documents. Another open question is the extension of the generative mixture models to "unknown" language identification (i.e. eliminating the closed-world assumption (Hughes et al., 2006)), which may be possible through the use of non-parametric mixture models such as Hierarchical Dirichlet Processes (Teh et al., 2006).

9 Conclusion

We have presented a system for language identification in multilingual documents using a generative mixture model inspired by supervised topic modeling algorithms, combined with a document representation based on previous research in language identification for monolingual documents. We showed that the system outperforms alternative approaches from the literature on synthetic data, as well as on real-world data from related research on linguistic corpus creation for low-density languages using the web as a resource. We also showed that our system is able to accurately estimate the proportion of the document written in each of the languages identified. We have made a full reference implementation of our system freely available,[8] as well as the synthetic dataset prepared for this paper (Section 5), in order to facilitate the adoption of this technology and further research in this area.

[8] https://github.com/saffsd/polyglot

Acknowledgments

We thank Hiroshi Yamaguchi for making a reference implementation of SEGLANG available to us, and Ben King for providing us with a collection of real-world multilingual web documents. This work was substantially improved as a result of the insightful feedback received from the reviewers.

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

Steven Abney and Steven Bird. 2010. The human language project: building a universal corpus of the world's languages. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 88–97. Association for Computational Linguistics.

Beatrice Alex, Amit Dubey, and Frank Keller. 2007. Using foreign inclusion detection to improve parsing performance. In *Proceedings of the Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning 2007 (EMNLP-CoNLL 2007)*, pages 151–160, Prague, Czech Republic.

Timothy Baldwin and Marco Lui. 2010a. 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 and Marco Lui. 2010b. Multilingual language identification: ALTW 2010 shared task dataset. In *Proceedings of the Australasian Language Technology Workshop 2010 (ALTW 2010)*, pages 5–7, Melbourne, Australia.

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.

David M. Blei, Andrew Y. Ng, and Michael I. Jordan. 2003. Latent Dirichlet allocation. *Journal of Machine Learning Research*, 3:993–1022.

Alessio Bosca and Luca Dini. 2010. Language identification strategies for cross language information retrieval. In *Working Notes of the Cross Language Evaluation Forum (CLEF)*.

Jamie Callan and Mark Hoy, 2009. *ClueWeb09 Dataset*. Available at `http://boston.lti.cs.cmu.edu/Data/clueweb09/`.

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.

Paul Cook and Marco Lui. 2012. langid.py for better language modelling. In *Proceedings of the Australasian Language Technology Association Workshop 2012*, pages 107–112, Dunedin, New Zealand.

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.

Rayid Ghani, Rosie Jones, and Dunja Mladenic. 2004. Building minority language corpora by learning to generate web search queries. *Knowledge and Information Systems*, 7(1):56–83.

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.

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

Thomas L. Griffiths and Mark Steyvers. 2004. Finding scientific topics. *Proceedings of the National Academy of Sciences*, 101:5228–5235.

Thomas Griffiths. 2002. Gibbs sampling in the generative model of latent Dirichlet allocation. *Technical Report, Stanford University*.

Baden Hughes, Timothy Baldwin, Steven Bird, Jeremy Nicholson, and Andrew MacKinlay. 2006. Reconsidering language identification for written language resources. In *Proceedings of the 5th International Conference on Language Resources and Evaluation (LREC 2006)*, pages 485–488, Genoa, Italy.

Genitiro Kikui. 1996. Identifying the coding system and language of on-line documents on the internet. In *Proceedings of the 16th International Conference on Computational Linguistics (COLING '96)*, pages 652–657, Kyoto, Japan.

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.

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.

David D. Lewis. 1997. The Reuters-21578 data set. available at `http://www.daviddlewis.com/resources/testcollections/reuters21578/`.

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, August. Association for Computational Linguistics.

Jicheng Liu and Chunyan Liang. 2008. Text Categorization of Multilingual Web Pages in Specific Domain. In *Proceedings of the 12th Pacific-Asia Conference on Advances in Knowledge Discovery and Data Mining*, PAKDD'08, pages 938–944, Osaka, Japan.

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.

Bruno Martins and Mário J. Silva. 2005. Language identification in web pages. In *Proceedings of the 2005 ACM symposium on Applied computing*, pages 764–768, Santa Fe, USA.

Andrew McCallum and Kamal Nigam. 1998. A comparison of event models for Naive Bayes text classification. In *Proceedings of the AAAI-98 Workshop on Learning for Text Categorization*, pages Available as Technical Report WS–98–05, AAAI Press., Madison, USA.

Andrew Kachites McCallum. 1999. Multi-label text classification with a mixture model trained by EM. In *Proceedings of AAAI 99 Workshop on Text Learning*.

Paul McNamee. 2005. Language identification: a solved problem suitable for undergraduate instruction. *Journal of Computing Sciences in Colleges*, 20(3):94–101.

Jian-Yun Nie, Michel Simard, Pierre Isabelle, and Richard Durand. 1999. Cross-language information retrieval based on parallel texts and automatic mining of parallel texts from the web. In *Proceedings of 22nd International ACM-SIGIR Conference on Research and Development in Information Retrieval (SIGIR'99)*, pages 74–81, Berkeley, USA.

John M. Prager. 1999a. Linguini: language identification for multilingual documents. In *Proceedings the 32nd Annual Hawaii International Conference on Systems Sciences (HICSS-32)*, Maui, Hawaii.

John M. Prager. 1999b. Linguini: Language identification for multilingual documents. *Journal of Management Information Systems*, 16(3):71–101.

John Ross Quinlan. 1993. *C4.5: Programs for Machine Learning*. Morgan Kaufmann, San Mateo, USA.

Daniel Ramage, David Hall, Ramesh Nallapati, and Christopher D. Manning. 2009. Labeled LDA: A supervised topic model for credit attribution in multi-labeled corpora. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing (EMNLP 2009)*, pages 248–256, Singapore.

Radim Rehurek and Milan Kolkus. 2009. Language Identification on the Web: Extending the Dictionary Method. In *Proceedings of Computational Linguistics and Intelligent Text Processing, 10th International Conference (CICLing 2009)*, pages 357–368, Mexico City, Mexico.

Philip Resnik. 1999. Mining the Web for bilingual text. In *Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics*, pages 527–534, College Park, USA.

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*, pages 5–15, Louvain-la-Neuve, Belgium.

W. J. Teahan. 2000. Text Classification and Segmentation Using Minimum Cross-Entropy. In *Proceedings the 6th International Conference "Recherche d'Information Assistee par Ordinateur" (RIAO'00)*, pages 943–961, Paris, France.

Yee Whye Teh, Michael I. Jordan, Matthew J. Beal, and David M. Blei. 2006. Hierarchical Dirichlet processes. *Journal of the American Statistical Association*, 101:1566–1581.

Jörg Tiedemann and Nikola Ljubešić. 2012. Efficient discrimination between closely related languages. In *Proceedings of the 24th International Conference on Computational Linguistics (COLING 2012)*, pages 2619–2634, Mumbai, India.

Giang Binh Tran, Dat Ba Nguyen, and Bin Thanh Kieu. 2010. N-gram based approach for multilingual language identification. poster. available

at http://comp.mq.edu.au/programming/task_description/VILangTek.pdf.

Fei Xia, Carrie Lewis, and William D. Lewis. 2010. Language ID for a thousand languages. In *LSA Annual Meeting Extended Abstracts*, Baltimore,USA.

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.

Alexander Yeh. 2000. More accurate tests for the statistical significance of result differences. In *Proceedings of the 18th International Conference on Computational Linguistics (COLING 2000)*, pages 947–953, Saarbrücken, Germany.