Learning a Monolingual Language Model from a Multilingual Text Database

Rayid Ghani
Center for Automated Learning and Discovery
School of Computer Science
Carnegie Mellon University
Pittsburgh, PA 15213 USA
Rayid.Ghani@cs.cmu.edu

Rosie Jones
Language Technologies Institute
School of Computer Science
Carnegie Mellon University
Pittsburgh, PA 15213 USA
Rosie.Jones@cs.cmu.edu

ABSTRACT

Language models are of importance in speech recognition, document classification, and database selection algorithms. Traditionally language models are learned from corpora specifically acquired for the purpose. Increasingly, however, there is interest in constructing language models for specific languages from heterogeneous sources such as the web. Query-based sampling has been shown to be effective for gauging the content of monolingual heterogeneous databases. We propose evaluating an extension to this approach by considering the case of learning a monolingual language model from a multilingual database, and extensions to the query-based sampling algorithm to handle this case. We test our approach on a corpus collected from the WWW and show that our proposed methods perform accurately and efficiently for learning a language model of Tagalog, when these documents are only 2.5% of the documents in a collection.

1. INTRODUCTION

Language Models have been used in many domains including database selection, speech recognition, optical character recognition, handwriting recognition [8], machine translation [1], and spelling correction [7].

In the domain of database selection, language models occupy a central position. Database selection algorithms use language models for each database and access the database using the model. Various studies have been performed on the optimal choice of features that a good language model should contain, but in general a language model describes the words that occur in a database, and frequency information indicating how often each term occurs [2]. In natural language tasks, a language model is usually formulated as a probability distribution $p(s)$ over strings of words s that attempts to reflect how frequently a string occurs in a language. The most widely used language models are n-gram models. In this paper, we construct unigram language models which assume that all words in a document are independent of each other and for a document d that is composed of words w_1, w_2, w_3 ,..., w_n, $p(d)$ can be expressed as $p(s) = p(w_1)p(w_2)...p(w_n)=\prod_{i=1}^{n} p(w_i)$. While this is a simplification, a unigram language model captures frequency information which is useful in a first understanding of the contents of a corpus but this approach could be easily extended to building an n-gram language model.

Traditionally, these models are constructed by approximating the word probabilities by maximum likelihood estimation (MLE) from a corpus of a specific language consisting of millions of words and then smoothing the probabilities using various techniques [3]. Currently, given a seed document (or small set of documents) in a particular language that is a minority in a database, there does not exist an easy way to collect a corpus of that language in an efficient manner (other than crawling the entire web and running all pages through a filter). In this paper, we attempt to build a corpus and hence a unigram language model for a minority language in a multilingual corpus. One obvious application of our work would be to incorporate our sampling strategies in a spider that uses reinforcement learning to crawl the web and collect documents in the desired language efficiently [10]. Our task would be relatively straightforward if (1) we had complete and free access to the multilingual corpus or database, (2) each document only contained words from a single language and (3) each document in the database came with a label specifying its language. Our task would then be reduced to the more common problem of building language models from a corpus specific for this purpose.

We treat the World Wide Web as our multilingual corpus and build a language model for a language that accounts for less than 3% of the documents in the database. The specific language we choose for the purposes of our experiments is Tagalog, the national language of the Phillipines. Since we cannot have complete access to the WWW, and access through a search engine is time-intensive, and every page on the WWW does not come labeled with the language the document was written in, we cannot apply traditional language modelling techniques to our database.

Instead, we use the approach introduced by Callan et al. [2] which uses query-based sampling to acquire language models from multiple databases. Query-based sampling is motivated by the fact that word occurrences follow a highly skewed distribution, with a few words occurring very often, and most words occurring rarely. In the light of evidence suggesting that the important vocabulary words occur frequently in a database [5, 9, 13], it is probable that these words might be acquired by sampling. Callan et al. show that if queries can be run and documents retrieved, then it is possible to sample the contents of each database in a way that will produce an accurate language model for the database.

We extend query-based sampling and compare several variants of importance sampling for creating a language model of Tagalog from a corpus created by gathering web pages from the WWW. We run experiments with several sampling strategies, evaluating the language models created. We find that a simple approach can be used to iteratively build up a language model of the minority language. By selecting a query term with probability proportional to its frequency in the collection built so far, we can quickly obtain documents which give us useful vocabulary and term frequency information. We used a language model constructed on the entire set of possible minority language documents to evaluate those constructed by sampling.

2. MODEL

We refer to the minority language that we are building a language model of as M. The set of documents in a language other than M we call O. We refer to unique strings in our vocabulary as *words* and occurrences of these *words* as *terms*. The unigram language model we construct is a multinomial over the words and we use the terms multinomial and unigram language model interchangeably.

The model we build is a multinomial over all the words in the minority language M. As we are building a unigram language model, we only use single words for our features. We assign unseen words a probability of zero, and do not perform any smoothing. Though smoothing is a common feature when constructing language models, we believe that since the number of relevant documents in our database is very small, we can acquire most of the vocabulary by sampling and thus do not need to rely heavily on smoothing. It would be quite straightforward to add smoothing to our approach should the need arise in a particular implementation.

To evaluate the language model created by sampling, we acquire a set of documents labeled as belonging to the language M. We calculate the true parameters of the multinomial over words for M by taking the maximum likelihood estimate of the probabilities (i.e. by using the raw frequency statistics to calculate the probabilities, without smoothing.) It is expected that the word frequencies for the words in M would follow the Zipf distribution as is the case with most languages [2]. The multinomial language model estimated in this way from our entire database we will call LM_M. The vocabulary defined by this multinomial we will call v_M. The multinomial constructed from O may or may not follow a Zipf distribution, since this collection may span multiple languages. We will primarily be interested in the vocabulary v_M which we expect to intersect only very sparsely with that of v_O.

3. METHODOLOGY

We start with an example document from M (the set of documents in the target minority language), and one from O (the set of other documents). We then build a language model for M' (the current sample of M), which we will call $LM_{M'}$, as well as $LM_{O'}$ based on the sample O'. Based on these, we create a one or two-word search query, and retrieve a document to add to the corpus. We filter (classify) the document retrieved using the current language models, into either the minority or other language class, then iterate. To evaluate corpus construction, we build a unigram language model over the entire set of possible target minority documents M. We call this the true model, as it represents the knowledge we would have about M if we sampled all possible M documents in the collection. Its language model is written LM_M.

3.1 General Algorithm

1. Select one seed document from each of M and O.
2. Build language models for M' ($LM_{M'}$) and O' ($LM_{O'}$) from the seed documents.
3. Sample a document from the database.
4. Use the language filter to decide whether to add the new document to the list of documents in M, or those in O.
5. Update the language models for $LM_{M'}$ and $LM_{O'}$.
6. If the stopping criterion has not been reached, go back to Step 2.

The two important steps for our method are (3) and (4). We discuss the various sampling strategies used in the next section, and the filter we use to decide whether a document is in Tagalog is discussed later. In step (3), we do separate runs of our experiments by performing sampling both with and without replacement. Sampling with replacement simulates duplicates which occur on the web with reasonable frequency, avoids the need for duplicate detection, and simplifies the underlying statistical sampling model. Also, for a small corpus, sampling with replacement is prone to the danger of sampling the same documents multiple times and the resulting language model would then be far from the true one. Sampling without replacment imitates using a search engine to query the web and remove all duplicates of the same web page. This removes the possibility of sampling a document more than once. We did not perform extensive experimentation with stopping criteria, but found that there were detectable plateaus in the amount of information acquired with each sample.

3.2 Sampling Methodologies

Our goal is to sample a representative variety of examples of documents in Tagalog with a minimum number of queries. Our approach can rely on the high-dimensionality of the problem, along with the fact that most dimensions (vocabulary) from the two models are not shared and for that reason, we use variants of importance sampling.

Importance Sampling [11] is useful when $T(X)$, the true word distribution, cannot be sampled directly but still needs to be estimated. Instead a distribution $S(X)$ that is easy to sample from and approximates the true distribution is used. The process comprises of two steps:

1. Drawing samples $x_1,...,x_m$ from the approximate distribution $S(X)$

2. Approximating $T(X)$ by weighting the samples x_i according to $w_i = T(x_i)/S(x_i)$, that is approximating the overall distribution by weighting according to local measurements of it.

We use several strategies including random sampling and adaptive query-based sampling for step 1 of the importance sampling. Our query is a projection onto one or more dimensions of the feature space. Adaptive Sampling has also been used in other areas including ocean forecasting [4] and environmental monitoring [12] and has been shown to perform well for tasks in which sampling is expensive, the target class is rare, and target examples are clustered together in feature space.

For the second step of importance sampling, we evaluate each sample (document) using $T(x_i)$ (a language filter) and set the corresponding w_i to 1 if the document belongs to the target language and 0 otherwise. This process downweights the samples which are outside the target language and increases the importance of the others.

We will build a corpus of documents in language M by sampling documents from the entire database D. A random selection of documents from D will not suffice since most would not be from M. Applying a language filter would allow us to construct the corpus of those from M, but only very slowly. A more efficient approach is to ensure that most of the documents we examine are from M. The intersection in vocabulary of M and O is very small. Thus, selecting documents with vocabulary in LM_M and not in LM_O is likely to give us documents in language M. This is the basis for our methods for sampling. Note that all query-based sampling methods we employ are followed by a language filter described in Section 3.3. Thus it is not imperative that a sampling method choose documents in M at every sampling iteration. However, the more frequently it does, the faster and more efficient corpus creation will be.

Table 1 gives an overview of the query-based sampling methodologies in our experiments. We use uniform random sampling as a baseline for our more "intelligent" sampling techniques. Since only 2.5% of the documents in our experimental corpus belong to M, we expect a document picked at random to probably belong to O. This also serves as motivation for our problem since if a crawler is deployed on the WWW to sample pages at random, it is very unlikely that it is going to find enough web pages in Tagalog to build a corpus for an accurate language model in a reasonable amount of time.

When picking the most-frequent word according to $LM_{M'}$ ($w_{maxP_{M'}}$) or $LM_{O'}$ ($w_{maxP_{O'}}$) we simply take the word seen most frequently in the sample of M' or O' constructed so far, that is, the most probable word according to the maximum likelihood estimated language model for the current sample. For M' this is given by $w_{maxP_{M'}} = argmax_{w_i} P(w_i|LM_{M'})$.

Table 1: Query construction methodologies. $w_{maxP_{M'}} = argmax_{w_i}P(w_i|LM_{M'})$ is the most probable word according to the language model for the current sample M'; similarly for $w_{maxP_{O'}}$. These correspond to the words most frequently seen in the sampled corpora constructed so far. $w_{randP_{M'}}$ is a word chosen randomly, with probability proportional to its frequency in the current sample M'; similarly for $w_{randP_{O'}}$.

Query Method	Include word	Exclude word	Sample Query
random			
most-frequent	$w_{maxP_{M'}}$		"+sa"
unigram	$w_{randP_{M'}}$		"+mga"
most-frequent-exclude	$w_{maxP_{M'}}$	$w_{maxP_{O'}}$	"+sa −de"
unigram-exclude-most-frequent	$w_{randP_{M'}}$	$w_{maxP_{O'}}$	"+ang −de"
unigram-exclude-unigram	$w_{randP_{M'}}$	$w_{randP_{O'}}$	"+kanyang −more"

In order to pick a random word according to the language model, we generate a number $u \sim Uniform[0, 1]$. We then pick the word according to u and the cumulative distribution function (CDF) of the unigram $LM_{M'}$ or $LM_{O'}$. This means that more frequently seen words in our sample so far are more likely to be selected, but that even rarely seen words have a small chance of being selected. Words unseen in the sample cannot be selected in this way.

When a query matches multiple documents, we pick one at random with uniform probability.

3.3 Document Filtering/Language Identification

Our sampling methodology assumes that we have an oracle to tell us whether a sampled document belongs to M (whether the document is in Tagalog), and thus whether to use it in building our next estimated language model. This oracle could be built using several different methodologies, two such being document filtering and language identification. Both of these topics have received attention from researchers for a long time and the purpose of this paper is not to conduct an in-depth study of which of these techniques is better or even to evaluate several techniques for our task. We construct a filter that just evaluates which distribution the document under consideration is most likely to have been generated from (M or O) and assigns it the corresponding label. We do not use priors on the classes, but merely the vocabulary from the current estimated model. Our filter is updated at every iteration of the experiment.

The algorithm is as follows:

- Count how many occurrences of words in the sampled document d are of words in the vocabulary in $LM_{M'}$.
- Count how many occurrences of words in d are of words in the vocabulary in $LM_{O'}$.
- Assign the document to M' or O' according to which score is higher.

Note that this filtering technique corresponds to a statistical model in which classes M' and O' are given uniform priors, and each word in their vocabularies is equally likely to be generated, i.e. also has a uniform probability. We are finding the maximum likelihood class to have generated the document under these simplifying assumptions.

State-of-the-art techniques in language identification [6] and document filtering augment such techniques with character-based trigram statistics and class priors. As we will discuss in section 6, our simple language filter was adequate, except when the seed document contained an unusually small number of words.

4. EVALUATION METRICS

We will take the multinomial LM_M (the multinomial estimated from all documents in language M in our database) to be the true multinomial distribution for M.

We evaluate the language model constructed using various sampling strategies by comparing it to the true language model given by the Tagalog documents in our database. The measures we use to evaluate our language model are discussed in this section.

4.1 Kullback Leibler Divergence

We measure the similarity between the true unigram lannguage model and the one constructed via our sampling by Kullback-Leibler (KL) divergence - a measure from information theory that represents the dissimilarity between two probability distributions, also called relative entropy. Let P and Q be two distributions over X. Q can be thought of as the learned model and P is the true model and X is the set of terms in the vocabulary. Then the Kullback-Leibler divergence is expressed as:

$$KL(P\|Q) = \sum_{i=1}^{n} Q(w_i) log \frac{Q(w_i)}{P(w_i)} \qquad (1)$$

where KL(P || Q) >= 0, and KL(P || Q) = 0 iff P and Q are equal. The Kullback-Leibler divergence can be considered as a kind of a distance between the two probability densities, though it is not a distance metric because it is not symmetric.

4.2 Percentage of Vocabulary Learned

This metric measures the proportion of the terms in the actual vocabulary that are found in the learned vocabulary. We can expect this measure to be low when we start our sampling (close to zero) and as we sample relevant documents in M, the language model should cover more of the terms found in the true vocabulary. The Percentage of Vocabulary learned (Per) metric gives equal importance to all the terms in the vocabulary and thus is not a good match for text data because of the skewed distribution of terms in a corpus. According to Callan et al. [2], about 75% of the vocabulary of a text database are words that occur 3 times or less.

4.3 Cumulative Term Frequency (Ctf) Ratio

A better measure for the quality of the learned vocabulary is the Ctf ratio which gives a weight to each term that is proportional to its importance in the database. Ctf measures the proportion of database term occurrences that are covered by terms in the learned language model. For a learned vocabulary V' and actual vocabulary V, and f_i and f_j both frequency counts over the whole minority corpus, the Ctf ratio is

$$\frac{\sum_{i \in V'} f_i}{\sum_{i \in V} f_i} \qquad (2)$$

5. DATA

Our dataset consisted of 16,537 documents collected from the web. This was broken down into 498 documents in Tagalog and 16,039 documents in other languages. The other language documents were in a mixture of Brazilian Portuguese and English.

We removed HTML mark-up and punctuation, converted capitalized letters to lower-case, and considered any string of remaining characters to be a word. The Tagalog subcollection has a vocabulary of 35,482 words, with a total of 281,379 term occurrences.

To find out how much of our Tagalog vocabulary intersected with English, we used the file `/usr/dict/words` which contains English words for spelling programs and can be found on all UNIX SYSTEMS. `/usr/dict/words` has a vocabulary of 45402 words. Of these, 5393 vocabulary items appeared in our Tagalog documents, accounting for 41,277 occurrences. The most frequent of these was "at", at 5659 terms, which is the Tagalog word for "and". The next most frequent was "the", which at 860 occurred on average twice per document. Tagalog documents contained some "internet English" (phrases such as "site provided as a service supported by...", "..update site regularly to...", and so words such as "page", "www" and some English terms occurred in the multinomial. In total, 28,000 Tagalog vocabulary items were unique to Tagalog in our dataset.

6. RESULTS

In this section, we present the results of our experiments comparing various query-based sampling strategies. We show that sampling using a query term chosen with probability proportional to its frequency in the current sample successfully finds more documents in the target langauge. This approach also gives a good coverage of vocabulary and terms. We also observed that is not important to the variance of results whether sampling is done with or without replacement. At a basic level, a sampling method which samples a large proportion of minority-language documents can be said to be a successful sampling algorithm, as it succeeds in finding a large number of documents in the target language.

Figure 1 shows the number of document sampled in Tagalog versus the total number of documents sampled, averaged

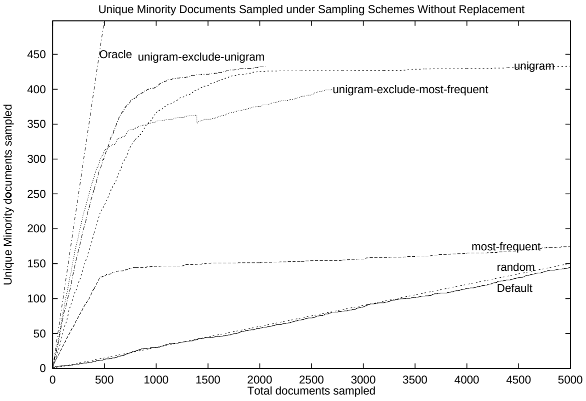

Figure 1: Random selection adds documents to the corpus and language model in the target minority language very slowly, relative to the number of samples taken at any point. Sampling documents containing a term chosen with probability proportional to its frequency in the corpus constructed so far has a near optimal rate of sampling documents in the target language.

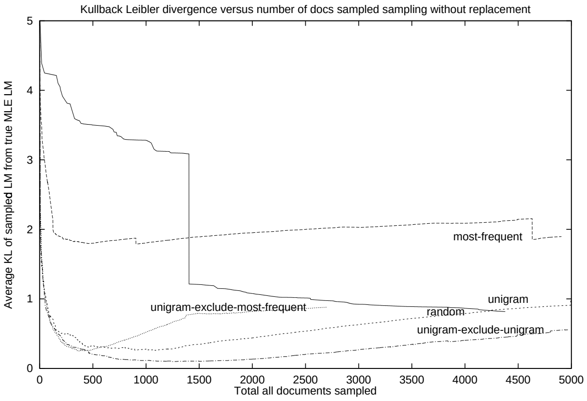

Figure 2: Sampling according the `multinomial-exclude multinomial` gives consistently better KL values than the other methods.

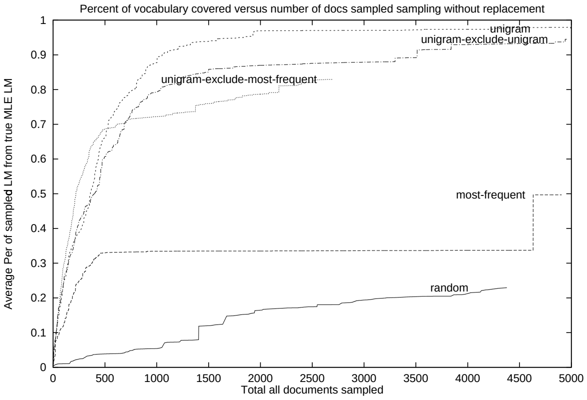

Figure 3: Consistently using the most frequent term as part of the query leads to algorithm convergence with a smaller percent of possible vocabulary terms seen. This reflects the mixed and overlapping vocabulary between the minority language and other languages in the corpus.

over three runs of each algorithm with different seed documents.The "Oracle" sampling rate, in which every sampled document is in our target minority language is simply the line $y = x$. Since we are sampling without replacement, the number of Tagalog documents sampled cannot exceed the total number in our collection. Also shown is the default sampling rate, the expected number of minority documents to be sampled, given a particular sample size. This is the line $y = x * 498/16547$. Note that random sampling hugs the default line, as is to be expected. The `multinomial` and `multinomial-exclude-multinomial` methods sample significantly greater than the expected number of minority language documents. `multinomial-exclude-most-frequent` also samples effectively and levels off at the same point as the other two.

Note that `most-frequent` flattens out at just under 150 documents, failing to find the full range of possible minority-language documents. As it finds a local maximum in the space of sample queries, it does not perform any extensive search to find new vocabulary items. By contrast,the other methods cover a variety of documents relatively quickly, and continue to explore the space of possible vocabulary more extensively.

As we can observe from Figures 2, 3, and 4, all the sampling schemes give sub-collections which model the true language model for Tagalog with increasing accuracy as more documents are added. However, as the number of documents sampled exceeds the pool of available target documents, the accuracy drops off. This is an important phenomenon to be aware of when sampling for very rare languages. We did not implement stopping criteria in these experiments, though a natural metric to use would be intersection of vocabulary in the minority and other models, which increases as saturation is reached.

This pattern is reflected in the decreasing KL divergence and increasing Percentage Learned (Figure 3) and Ctf values (Figure 4). It is interesting to note that the schemes using the multinomial outperform the rest of them throughout the experiments. As the number of documents sampled increases, `multinomial-exclude-multinomial` performs better than the others.

All results we have shown were under the condition of sampling without replacement. Since our sampled corpus at each iteration was a relatively large proportion of the total Tagalog sub-corpus size, we would expect the variance in our estimate to be higher if we sample with replacement. However, sampling with replacement is algorithmically simpler, since we need not explicitly check whether a document has been seen already, nor perform duplicate detection. Note that neither approach gives an independent identically distributed sample of documents since the document sampled at each iteration is dependent on the set of documents seen so far.

Figures 5 and 6 show three runs with different initial Taga-

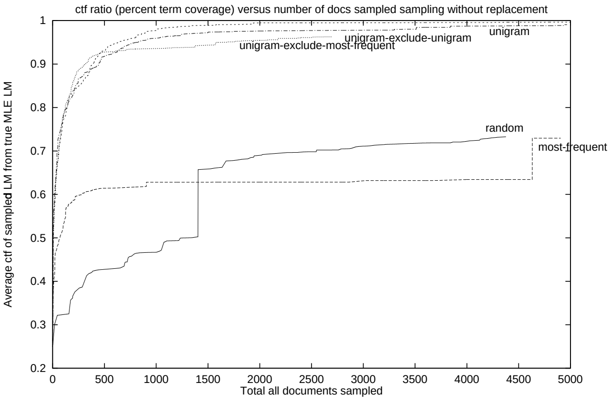

Figure 4: In term coverage ratio, the multinomial sampling techniques perform comparably, and better than techniques incorporating the most frequent term.

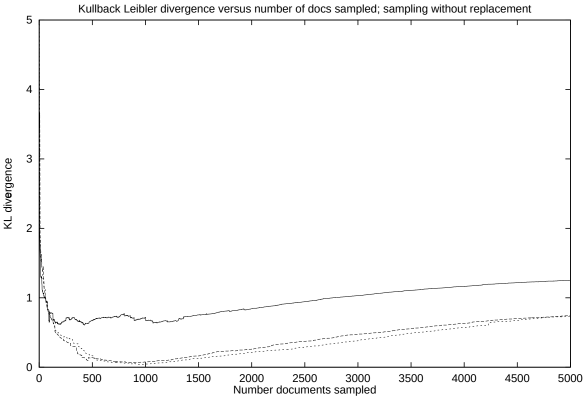

Figure 5: Sampling without replacement shows relatively small variance across three runs with different initial Tagalog documents and random seeds.

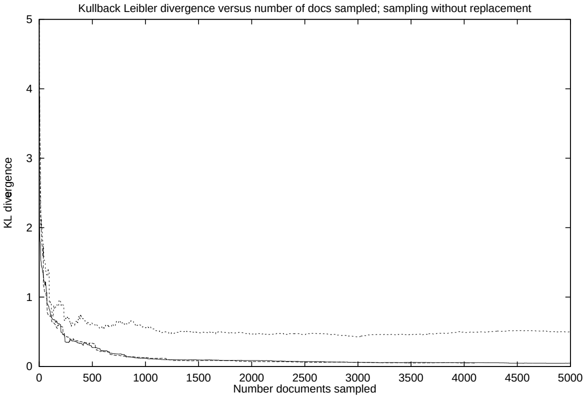

Figure 6: Sampling with replacement also shows relatively small variance across three runs with different initial Tagalog documents and random seeds.

log documents and random seeds, for the cases of sampling without replacement, and sampling with replacement respectively. The variance is not substantially different in either case. Thus it may be adequate to perform the algorithmically simpler operation of sampling with replacement for corpus and language model construction.

Not shown here are the results of an experiment in which we deliberately chose a very unusual Tagalog document as our seed document. It contained only two distinct words. In this case, our language filter assigned the first sample Tagalog document to the Other class, and then never recovered. Thus for degenerate seed documents our filtering can break down, and fail to allow the sampling algorithm to perform adequately. We do not expect this to be the case in general.

7. DISCUSSION

An interesting question is whether our techniques give us a random selection of documents from language M. Consider the following simplified model:

- documents in M are generated by sampling with replacement according to a multinomial over unigrams
- unigrams in M are conditionally independent given the language M

In this special case, the method of selecting any term unique to M, and then selecting a random document containing that word will indeed give us a random selection of documents in M. If we can reliably filter, we do not even need to select a term unique to M.

Now replace the constraint that all words are conditionally independent given M with the following condition:

- there exists a set of pair-wise conditionally independent terms [optionally unique to M] which span M; i.e.

$$\begin{aligned}\exists S &= \{s_1, ...s_k : \forall d \epsilon M, \exists s_i \epsilon d \\ &\quad AND \\ &\quad \forall i, j, p(s_i, s_j | d \epsilon M) = p(s_i | d \epsilon M) * p(s_j | d \epsilon M)\}\end{aligned}$$

then sampling from these terms would ensure a representative sample of documents from M.

The latter model is actually quite a reasonable one. Function words are frequently considered to be of little use in information retrieval tasks, as they are thought to occur relatively uniformly among documents. Thus a set of conditionally independent terms may consist of function words.

We found that the most frequent five words (in their lower case form) from our corpus of Tagalog ("sa", "ng", "ang", "na", "at") cover 393 of our 498 documents, while adding the next five words ("mga", "ay", "a", "ti", "ni") covers 418 documents.

We calculated a measure of the independence between pairs of the five most frequent words given these documents in M,

Table 2: Information between pairs of the five most frequent Tagalog words

score	word pair
0.233484137352623	na na
0.230802728988242	at at
0.229097111336914	ng ng
0.220177900356446	ang ang
0.202968500508056	sa sa
0.181622070611764	na ang
0.166364252189481	sa ang
0.163896550055644	sa na
0.157122465766681	ng na
0.154884598635506	na at
0.152658828083418	ng at
0.150372574635893	sa ng
0.142739633231722	at ang
0.142642860599023	sa at
0.140695311365946	ng ang

by calculating

$$P(w_1, w_2|d_M) - P(w_1|d_M) * P(w_2|d_M)$$

This gave the results in table 2.

Clearly none of these word-pairs are completely independent, but they are less correlated with each of the other words than with themselves.

8. FUTURE WORK

An interesting experiment would be to vary the number of documents in M present in the corpus. Since Tagalog web pages are very sparse, we could not obtain a larger sample for our corpus. Currently, we are in the process of collecting web pages in Slovenian, which has a much larger presence on the WWW, and plan to run similar sampling experiments while varying the number of Slovenian documents in our database. We also plan to vary the heterogeneity of the database and create a new corpus with documents from several closely related languages and varying the number of languages.

Since the aim of this paper was to investigate various query-based sampling strategies, we did not research in to the creation of a perfect document filter. Having access to an oracle that could label each document with its language would improve our results immensely. Another factor worth exploring would be instead of picking a document at random from the results returned by a query, pick a document with some desired features (e.g. long documents, documents containing most unique words). We would then expect to get the most "informative" documents for our task.

For the experiments reported in this paper, we select one document at every iteration to add to our corpus. An interesting variation of this would be to add multiple number of documents up to all the documents returned by each query. This would make the process of constructing the langauge model (and the corpus) faster but maybe at the cost of less coverage of terms.

9. ACKNOWLEDGEMENTS

This research has been supported in part by the DARPA under research contract F30602-97-1-0215, and by LIS under grant REC-9720374. We would like to thank John Lafferty, Steve Fienberg, Sean Slattery and Kamal Nigam for helpful comments.

10. REFERENCES

[1] P. Brown, S. D. Pietra, V. D. Pietra, and R. Mercer. The mathematics of statistical machine translation. *Computational Linguistics*, 19(2), 1993.

[2] J. Callan, M. Connell, and A. Du. Automatic discovery of language models for text databases. In *Proceedings of the 1999 ACM SIGMOD International Conference on Management of Data*, pages 479–490, Philadelphia, 1999. ACM.

[3] S. F. Chen and J. T. Goodman. An empirical study of smoothing techniques for language modeling. In *Proceedings of the 34th Annual Meeting of the Association for Computational Linguistics*, pages 310–318, 1996.

[4] T. Curtin, J. Bellingham, J. Catipovic, and D. Webb. Autonomous ocean sampling networks. *Oceanography*, 6(3):86–94, 1993.

[5] S. Dumais. Latent semantic indexing (LSI) and TREC-2. In D. K. Harma, editor, *The Second Text REtrieval Conference (TREC-2)*, pages 105–115. Gaithesburg, MD, 1994.

[6] T. Dunning. Statistical identification of language. Technical Report MCCS 94-273, New Mexico State University, 1994.

[7] A. R. Golding and D. Roth. A winnow-based approach to context-sensitive spelling correction. *Machine Learning*, 34(1-3):107–130, 1999.

[8] F. Jelinek. *Statistical Methods for Speech Recognition.* MIT Press, 1999.

[9] H. P. Luhn. The automatic creation of literature abstracts. *IBM Journal of Research and Development*, 2(2), 1958.

[10] J. Rennie and A. McCallum. Using reinforcement learning to spider the web efficiently. In *Proceedings of the 16th International Conference on Machine Learning*, 1999.

[11] M. A. Tanner. *Tools for Statistical Inference.* Springer-Verlag, 1996.

[12] S. K. Thompson and A. George. *Adaptive Sampling.* John Wiley and Sons, 1996.

[13] G. K. Zipf. *Human behavior and the principle of least effort: An introduction to human ecology.* Addison-Wesley, Cambridge MA, 1949.