Language Identification in Web Pages

Bruno Martins and Mário J. Silva
Faculdade de Ciências Universidade de Lisboa
1749-016 Lisboa, Portugal
{bmartins,mjs}@xldb.di.fc.ul.pt

ABSTRACT

This paper discusses the problem of automatically identifying the language of a given Web document. Previous experiments in language guessing focused on analyzing "coherent" text sentences, whereas this work was validated on texts from the Web, often presenting harder problems. Our language "guessing" software uses a well-known *n*-gram based algorithm, complemented with heuristics and a new similarity measure. Both fast and robust, the software has been in use for the past two years, as part of a crawler for a search engine. Experiments show that it achieves very high accuracy in discriminating different languages on Web pages.

1. INTRODUCTION

Language identification has become increasingly important, as more and more textual data is making its way on-line. When processing multilingual document collections, appropriate language annotations can be used to bootstrap shallow machine translation, parts-of-speech tagging, topic labeling or information retrieval [18]. Automatic "language guessing" systems have been described in the past, achieving very high accuracy [24]. However, text from the Web is considerably different [1], motivating us to revisit the problem. Difficulties introduced by the Web domain are generally related to the "noisier" nature of the text, including things like spelling errors, multilingual documents or small amounts of text.

In this paper, we describe a system to automatically identify the language of Web pages. It implements the *n*-gram based algorithm originally presented by Cavnar and Trenkle, and described as very fast, precise and tolerant to errors [6]. We also complemented the original algorithm with a more efficient similarity measure, together with heuristics to better handle Web data. We use this software in the context of a Portuguese Web search engine (`www.tumba.pt`). Language information is used to decide if a document crawled from a top-level domain other than ".PT" should be indexed. We are only interested in documents from domains other than ".PT" if they are written in Portuguese.

The rest of this document details the language identification system and the most important issues faced during its development.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.
SAC'05 March 13-17, 2005, Santa Fe, New Mexico, USA
Copyright 2005 ACM 1-58113-964-0/05/0003 ...$5.00.

2. RELATED WORK

Sibun and Reynar provided a good survey on existing techniques for language identification [24]. A variety of features have been used as discriminators. These include the presence of particular characters [28, 22], the presence of particular words [14, 26] and word classes [18], the presence of particular character *n*-grams [6, 11], or others [25]. In this context, *n*-grams refer to *n*-character contiguous slices of a longer string.

The *n*-gram method proposed by Canvar and Trenkle [6] seems to be the most promising approach. It is not only very efficient, as it also presents all the properties we deem most desirable: it performs well on brief passages of text, requires minimal amounts of training data, it is very robust to textual errors, and is fast and relatively easy to implement. In the published evaluation results, their method achieved 99.8% precision on discriminating among 8 different languages, misclassifying only 7 articles out of 3478. Furthermore, the algorithm was also reported to exhibit interesting generalization behaviors. For instance, when trained on English, French and Spanish, it tends strongly to classify German as English, therefore confirming the historical basis of English as a Germanic language.

Dunning's method [11], which involves *n*-gram statistics and Markov models, is also reported to perform very well – 99.9% of accuracy in discriminating two moderately-related languages, English and Spanish. It assumes that language can be modeled by a low order Markov process generating strings, and then uses Bayesian decision rules to select which of two phenomena caused a particular observation.

Other methods based on *n*-gram statistics have also been reported. For instance relative entropy, also known as Kullback-Leibler distance, was used by Sibun and Reynar [24]. Damashek used a model that computed dot-products of frequency vectors [10].

Besides applications in the language identification problem, it is interesting to note that *n*-gram based methods have also been applied to other problems involving text processing [20]. Previous studies report they perform very well in the extreme cases of document categorization according to topic [6] or the analysis of genetic sequences [11].

3. N-GRAMS IN TEXT CATEGORIZATION ACCORDING TO LANGUAGE

In this work, we propose to assign language labels to textual strings, using an approach based on the statistical characterization of text in terms of its constituent *n*-grams. The key benefit that *n*-gram based matching provides derives from its very nature: since every string is decomposed into small parts, any errors that are present tend to affect only a limited number of those parts, leaving the remainder intact. This is particularly interesting to our problem of language identification in Web pages. Tolerance to spelling

and grammatical errors is of crucial importance on the Web, as the quality of the documents varies considerably. If we count *n*-grams that are common to two strings, we get a measure of their similarity that is resistant to a wide variety of textual errors.

The definition of *n*-grams of characters also does not explicitly or implicitly require the specification of a separator, a problem that occurs for words [13]. For instance, text tokenization, stemming and/or lemmatization, which are relatively easy in English, become much more difficult for languages such as German, notable for the extensive use of compound words [4]. If *n*-grams of characters are used instead of words as the basic unit of information, there is simply no need to recognize words, and therefore no need for morphological and lexical analysis. Related forms of a word (e.g advance, advanced, advancing) also have a lot in common when viewed as sets of *n*-grams, and can therefore be appropriately treated.

Our approach to the language identification problem is based on the *n*-gram analysis method proposed by Cavnar and Trenkle [6]. Their technique is conceptually simple and achieves good performance, even with relatively small training sets (50Kb of training text is more than enough) or short strings to be classified (e.g. having more than 300 characters is enough for optimal performance).

Essentially, the categorization method involves constructing a probabilistic model (or "profile") based on character *n*-grams of different lengths, one for each language in the classification set The most frequent items in the profiles will reflect common letter combinations (i.e. "th", "ão") and morphemes ("the", "op"). Classification of an unknown string of text is performed by selecting the model most likely to have generated it, using an easily calculated distance measure between the model generated for the string to be classified and the language profiles.

In our system, we use *n*-grams of multiple lengths simultaneously, with *n* ranging from 1 to 5. The rationale for using unigrams in our system (in contrast to the original proposal by Cavnar and Trenkle) is directly related to diacritic characters, which we think are good clues in guessing the language of a document. We also append blanks to the beginning and ending of the string, in order to help with matching beginning-of-word and ending-of-word situations. Using the underscore character to represent blanks, the string "TUMBA!" would be composed of the following *n*-grams:

unigrams: _, T, U, M, B, A, !, _

bigrams: _T, TU, UM, MB, BA, A!, !_

trigrams: _TU, TUM, UMB, MBA, BA!, A!_, !__

quadgrams: _TUM, TUMB, UMBA, MBA!, BA!_, A!__, !___

quintgrams: _TUMB, TUMBA, UMBA!, MBA!_, BA!__, A!___, !____

The extracted *n*-grams are hashed into a table, storing the number of occurrences in the text. They are then sorted in descending order by their occurrence frequency. As reported by Cavnar and Trenkle, the top 400 or so *n*-grams are highly correlated to the language. We use these top 400 to build the profiles.

To measure the minimum distance between profiles, Cavnar and Trenkle used a simple rank-order statistic, obtained by calculating how far "out-of-rank" an *n*-gram in one profile is from its ranking position in the other – see Figure 1, borrowed from the original article. For each *n*-gram in the document profile, they would find its counterpart in the class profile, and then calculate how far out of place it is. The sum of all these values gives the measure, and the class profile corresponding to the lowest value is finally selected.

In this work we argue for the use of a new, more efficient, similarity measure. In the past, Lin has investigated the theoretical basis of similarity, deriving the general form of an information theoretic

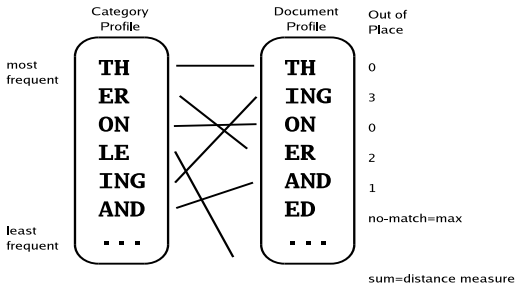


Figure 1: Computing similarity between two *n*-gram profiles.

measure [17]. Experiments have shown that the proposed approach outperforms other popular similarity measures [3, 17]. Under the assumption that the probability of an *n*-gram occurring in a string is independent of other *n*-grams, Lin proposed the following formula to calculate the similarity between two strings x and y, given their constituent *n*-grams $ng(x)$ and $ng(y)$ respectively:

$$sim(x,y) = \frac{2*\sum_{t\in ng(x)\cap ng(y)} \log P(t)}{\sum_{t\in ng(x)} \log P(t) + \sum_{t\in ng(y)} \log P(t)}$$

The term independence assumption is very common on text analysis. It is untrue for words, and wildly untrue for *n*-grams, as adjacent *n*-grams share all but one letter. Nonetheless, the similarity metric does not appear to suffer from this unrealistic assumption.

Jiang and Conranth proposed a similar formula, using the same elements as Lin in a different way [16]:

$$sim(x,y) = 2*\sum_{t\in ng(x)\cap ng(y)} \log P(t) - (\sum_{t\in ng(x)} \log P(t) + \sum_{t\in ng(y)} \log P(t))$$

Their measure captures distance, the inverse of similarity. Previous experiments suggested this different arithmetic combination of the same terms does indeed yield better results [5].

4. HEURISTICS FOR THE CATEGORIZATION OF WEB DOCUMENTS

Although the *n*-gram approach is quite simple and reported to work very well, even on small strings, some issues have to be addressed when applying it to Web documents. These include:

1. Extract the text, the markup information, and meta-data.
2. Use meta-data information, if available and valid.
3. Filter common or "automatically generated" strings.
4. Weight *n*-grams according to HTML markup.
5. Handle situations where we possibly have insufficient data.
6. Handle multilingualism and the "hard to decide" cases.

The need to remove tags and comments from HTML documents before performing the categorization, as well as handling HTML character internationalization issues, is obvious. However, dealing with the large variety and diversity of information sources over the Web introduces challenges. Having a robust parser capable of tolerating common errors associated with malformed HTML documents is a very important aspect of dealing with Web data.

Some HTML pages include a language meta-tag, specifying its language. However, many HTML editing tools automatically set its value to English by default. As a result, meta-tag information is

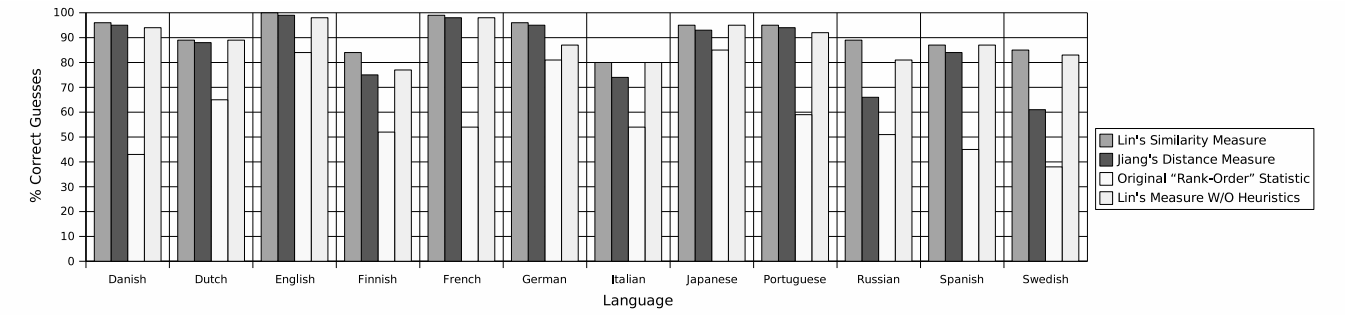


Figure 2: Results for the Language Identification Algorithm In Different Settings.

not always reliable. There is also no uniform way of specifying the language on the meta-tag (for instance Portuguese can be specified as pt, pt-pt, por, portugues, pt-portuguese, etc.) and therefore automatic tools are not always able to process this data. In our case, if we succeed in matching the information on the language meta-tag with the names of a set of known languages, and if the value provided corresponds to a language other than English, we return it instead of the language inferred by the *n*-gram classification system.

Another problem concerns the fact that strings like "This page uses frames" or "Made for Internet Explorer" are very frequent and do not necessarily mean that a given Web resource has written content in the English language. For instance, in a large crawl of the Portuguese Web (about 3.5 million documents), we found the two strings above occurring about 35.000 times. We keep a small dictionary of such sentences, which are filtered in a pre-processing stage. Common words in the Internet, such as "applet" or "java", are also ignored through this process, as they occur all over the global Web, independently of language.

Pages containing tables with numeric data are also very common. To better handle these cases, we ignore all *n*-grams composed of only numeric and/or white space characters. *N*-grams are also weighted according to markup information. HTML defines a set of fields to which text in a document can be assigned. Terms appearing in different fields intuitively have different importance [9, 15] (i.e. text from the title of the page should, in principle, be more important). In our system, these fields serve as multiplicative factors for the frequency of *n*-grams within their scope. More specifically:

- *n*-grams in the title are counted three times.
- *n*-grams in descriptive meta-tags are counted twice.

Another heuristic concerns pages with very little text. When the text extracted from a Web page has less than 40 characters, we simply assign the document an "unknown language" label. This way, we make a small trade of recall for precision. Some documents will not be classified at all, but we have better chances of assigning a correct label.

Finally, multilingual documents are also very common on the Web. This constitutes a problem if we wish to assign the full content of a given document to one single language. For instance home-pages for people with foreign names are very common over the Portuguese Web, at least inside large institutional sites. To better handle this, we use a simple heuristic: when a document is neither classified as Portuguese or English (the two most common languages in our document collection) the algorithm is very likely making a mistake. In these cases, we try to re-apply the *n*-gram algorithm, weighting the largest continuous text block in the document (blocks are identified in the HTML parsing stage, taking in account the markup information) as three times more important than the normal text. The rationale for this is that the longest block will very likely correspond to a good description of the page, possibly in its main language.

5. EXPERIMENTS

For our experimental scenario, we used language profiles for 23 different languages, constructed from textual information extracted from newsgroups and the Web. Since we could not find an appropriate collection of Web pages to use as the "golden standard", we also organized a test collection, manually assigning HTML documents to one of twelve different languages (Danish, Dutch, English, Finnish, French, German, Italian, Japanese, Portuguese, Russian, Spanish and Swedish). The total number of documents in the collection is 6000, with 500 documents for each language. The pages in the collection were crawled from sites like on-line newspapers or Web portals. These generally contain a large number of pages in the same language, therefore making it easier to build the test collection and check it for errors in the language assignment process.

With this document collection, we tested the language identification system in different settings, using the three different similarity measures and the heuristics described above. Results are shown on Figure 2. Lin's measure was consistently the most accurate. On average, it outperformed the original "rank-order" measure proposed by Cavnar and Treckle by 32%. Results also confirm that the proposed heuristics improve the accuracy of the system. Using Lin's similarity measure without the heuristics results in a decrease in performance of about 2.8%. In the future, we plan on conducting statistical significance tests, in order to estimate a confidence degree in the reported improvements. This is particularly important to evaluate the benefit introduced by our heuristics, since the reported improvement is much smaller than in the case of the similarity measures.

Table 1 details the results obtained in the best setting for the system (Lin's measure with the heuristics). Although the reported values are good enough for the system to be used effectively, they are lower than those available in other studies with similar systems. We believe that this is mainly due to the much noisier nature of the texts being processed. The generalization behavior of the classifier was also quite notorious. For instance, similar European northern languages were many times confused. In detecting only the Portuguese documents, the system achieves 99% accuracy, with 92% of precision and 95% of recall – see Table 2.

Lang	Dan	Dut	Eng	Fin	Fre	Ger	Ita	Jap	Por	Rus	Spa	Swe
Chi	0	0	0	0	0	1	0	6	1	12	0	0
Dan	480	48	0	6	0	3	5	0	0	12	2	22
Dut	2	447	0	0	2	0	0	0	0	0	0	0
Eng	5	3	499	31	1	6	12	4	10	37	24	17
Fin	0	0	0	421	0	0	0	0	0	0	0	0
Fre	0	0	0	4	495	1	4	0	0	0	2	0
Ger	1	0	0	7	0	482	12	0	9	16	9	0
Ice	1	0	0	0	0	0	0	0	0	0	0	0
Ita	0	0	0	1	0	0	403	0	0	0	0	0
Jap	0	0	0	0	0	0	0	475	0	0	0	0
Por	0	0	0	0	0	0	20	0	475	6	15	0
Rus	0	0	0	0	0	0	0	0	0	444	0	0
Spa	0	0	0	0	0	2	42	0	0	0	435	1
Swe	0	0	0	16	0	0	0	0	0	2	0	425
Unkown	11	2	1	14	2	5	2	15	5	1	13	35
#Correct	480	447	499	421	495	482	403	475	475	444	435	425
%Correct	96%	89%	100%	84%	99%	96%	80%	95%	95%	89%	87%	85%

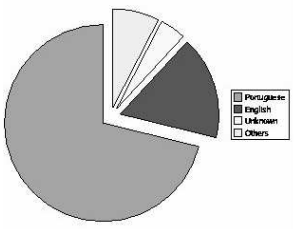
Table 1: Results for the Language Identification Algorithm Using Lin's Similarity Measure.

	Portuguese Docs	Non-Portuguese Docs
Assigned as Portuguese	475	41
Assigned as Non-Portuguese	25	5459

Table 2: Discriminating Portuguese Documents.

In our testing, we worked with a Pentium IV 2.66GHz server with 896 MB main memory, running RedHat Linux 9.0 and with the Java Development Kit version 1.4.2 installed (the *n*-gram categorization software was implemented in Java). For the 6000 documents in our test collection, and for each setting of the language guessing system, the total time for loading language profiles from disk, loading and parsing HTML, extracting the text, and classifying it according to language, was less than twenty minutes, corresponding to about five documents per second.

Also interesting are the results of running this algorithm on a large collection of about 3.5 million pages from the Portuguese Web, as indexed by our search engine – see Figure 5. Given the high accuracy of our language guessing system, we can state that a significant portion of the pages hosted under the ".PT" domain is actually written in foreign languages, especially English.



Language	# Documents
Portuguese	(71%) 2334389
English	(17%) 552761
unknown	(4%)142509
German	(3%) 108277
Spanish	(1%) 41813
French	(1%) 21411
Romanian	(0%) 12555
Polish	(0%) 11561
Swedish	(0%) 5320
Italian	(0%) 4454
others	(1%) 36841
Norwegian	(0%) 2951
	(100%) 3274842

Figure 3: Language distribution on the Portuguese Web.

6. LIMITATIONS

Our experiments confirm that the more similar two languages are, the harder it becomes to statistically discriminate them. The generalization behavior of the classification approach can be interesting in some tasks, but we ideally would like to discriminate very similar languages, or even slight variations of the same language.

The main problem we have in the application of this system lays in its inability to discriminate Portuguese documents from Brazilian ones (we expect similar problems in discriminating English texts originating from the UK or from USA, or documents from different French speaking countries). On a small test, our system indeed achieved poor precision in these cases, and this is a practical necessity for our Web crawler, since we do not want to harvest all pages from Brazil. Even humans can have difficulties in performing the task [27], so the problem will always be hard to solve using only the textual information from the documents.

In our search engine, we currently ignore resources from the ".BR" domain, but in the future, we would like to find a more elegant approach to this problem. Instead of just using the most frequent *n*-grams for a language, these cases could be further disambiguated using *n*-grams that although do not appear frequently are very specific to a given language. For instance, the characters "ä" or "ö" are used on Brazilian words but not on Portuguese ones.

Another solution could involve complementing our method with hyper-link analysis and/or meta-data propagation [7, 19], as Brazilian Web documents are in principle more likely to point to other Brazilian documents than to Portuguese ones. Using the information in the Whois databases (information for IP address space allocations and assignments) could also prove beneficial for our Web harvesting problem, as Whois records contain information about the contact address of the persons responsible for the domains. However, Whois database information is generally not public.

7. CONCLUSIONS AND FUTURE WORK

This paper presented a language identification system based on a well known algorithm that measures similarity according to the prevalence of short letter sequences (*n*-grams). Because Web documents have special characteristics, the method was complemented with a set of heuristics, in order to better handle this information.

Although the system has already demonstrated good performance on a realistic setting, there is still room for further improvements. For instance, a better tuning of the profiles used to classify new documents could raise performance to values closer to the ones reported in other experiments involving *n*-gram based classification systems.

Using linkage information and the text from hypertext anchors could also provide improvements on the overall results. Previous experiments have concluded that hypertext anchors provide very good summaries of the target documents [2]. We could, for instance, associate all the text coming from in-link anchors to the

documents being pointed to, or even propagate text and/or meta-data using linkage information. However, this would not be appropriate to our Web crawling problem, as full linkage information is only available after all pages have been visited. The same could be said for a more advanced n-gram weighting scheme, requiring frequency statistics from the entire collection [23, 21].

We would also like to explore principled approaches for smoothing the data (i.e. Good-Turing [12]), in order to account for rare character sequences. Chen and Goodman provide a good survey on the subject [8]. A central notion to many smoothing techniques is the Good-Turing estimate, stating that for any n-gram that occurs r times, we should pretend that it occurs r^* times, where $r^* = (r+1)\frac{n_{r+1}}{n_r}$ and n_r is the number of n-grams that occur exactly r times in the training data.

Finally, our "language classification" system can also one-day be used for other classification problems (categorizing pages by topic, by author, etc.), as the n-gram method has also been previously reported as promising in many different tasks.

8. ACKNOWLEDGMENTS

Special thanks to our university colleagues for their comments on early drafts of this paper. Our gratitude goes also to the various members of Linguateca, for their valuable insights and suggestions. This research was partially supported by FCT - Fundação para a Ciência e Tecnologia, under grants POSI/SRI/40193/2001 (project XMLBase) and SFRH/BD/10757/2002 (FCT scholarship).

9. REFERENCES

[1] E. Amitay. Hypertext - the importance of being different. Master's thesis, Centre for Cognitive Science, Edinburgh University, 1997.

[2] E. Amitay. Using common hypertext links to identify the best phrasal description of target Web documents. In *Proceedings of the SIGIR-98 Post-Conference Workshop on Hypertext Information Retrieval for the Web*, 1998.

[3] J. A. Aslam and M. Frost. An information-theoretic measure for document similarity. In *Proceedings of SIGIR-03, the 26th annual international conference on Research and development in informaion retrieval*, pages 449–450. ACM Press, July 2003.

[4] I. Biskri and S. Delisle. Text classification and multilinguism: Getting at words via n-grams of characters. In *Proceedings of SCI-2002, 6th World Multiconference on Systemics, Cybernetics and Informatics*, volume 5, pages 110–115, July 2002.

[5] A. Budanitsky and G. Hirst. Semantic distance in WordNet: An experimental, application-oriented evaluation of five measures. In *Workshop on WordNet and Other Lexical Resources, 2nd meeting of the North American Chapter of the Association for Computational Linguistics (NAACL-2000)*, June 2001.

[6] W. B. Cavnar and J. M. Trenkle. N-gram-based text categorization. In *Proceedings of SDAIR-94, the 3rd Annual Symposium on Document Analysis and Information Retrieval*, pages 161–175, Las Vegas, Nevada, U.S.A, 1994.

[7] S. Chakrabarti, B. E. Dom, S. R. Kumar, P. Raghavan, S. Rajagopalan, A. Tomkins, D. Gibson, and J. Kleinberg. Mining the Web's link structure. *Computer*, 32(8):60–67, August 1999.

[8] S. F. Chen and J. Goodman. An empirical study of smoothing techniques for language modeling. In A. Joshi and M. Palmer, editors, *Proceedings of ACL-96, the 34th Annual Meeting of the Association for Computational Linguistics*, pages 310–318, San Francisco, 1996. Morgan Kaufmann Publishers.

[9] Y. S. M. Cutler and W. Meng. Using the structure of HTML documents to improve retrieval. In *Proceedings of the USENIX Symposium on Internet Technologies and Systems*, 1997.

[10] M. Darnashek. Gauging similarity with n-grams: language independent categorization of text. *Science*, 267(5199):843–848, 1995.

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

[12] I. J. Good. The population frequencies of species and the estimation of population parameters. *Biometrika*, 40:237–264, 1953.

[13] G. Grefenstette and P. Tapanainen. What is a word, what is a sentence? problems of tokenization. In *Proceedings of COMPLEX-94, the 3rd International Conference on Computational Lexicography*, pages 79–87, 1994.

[14] P. Henrich. Language identification for the automatic grapheme-to-phoneme conversion of foreign words in a german text-to-speech system. In *Proceedings of Eurospeech 1989, European Speech Communication and Technology*, pages 220–223, September 1989.

[15] C. Hill. Information space based on html structure. In E. M. Voorhees and D. K. Harman, editors, *Proceedings of TREC-9, the 9th Text REtrieval Conference*. Department of Commerce of National Institute of Standards and Technology, 2000.

[16] J. Y. Jiang and D. W. Conrath. Semantic similarity based on corpus statistics and lexical taxonomy. In *Proceedings of ROCLING-X, the ROCLING 1997 International Conference on Research on Computational Linguistics*, 1997.

[17] D. Lin. An information-theoretic definition of similarity. In *Proceedings of ICML-98, the 15th International Conference on Machine Learning*, pages 296–304. Morgan Kaufmann, San Francisco, CA, 1998.

[18] R. D. Lins and P. Gonçalves. Automatic language identification of written texts. In *Proceedings of SAC-2004, the 2004 ACM symposium on Applied computing*, pages 1128–1133. ACM Press, 2004.

[19] M. Marchiori. The limits of web metadata, and beyond. In *Proceedings of WWW-98, the 7th International World Wide Web Conference*, April 1998.

[20] P. McNamee and J. Mayfield. Character n-gram tokenization for european language text retrieval. *Information Retrieval*, 7, April 2004.

[21] E. Miller, D. Shen, J. Liu, and C. Nicholas. Performance and scalability of a large-scale n-gram based information retrieval system. *Journal of Digital Information*, 1(21), 2000.

[22] P. Newman. Foreign language identification – first step in the translation process. In K. Kummer, editor, *Proceedings of the 28th Annual Conference of the American Translators Association*, pages 509–516, 1987.

[23] C. Pearce and B. Rye. N-gram term weighting: A comparative analysis. Technical Report TR-R52-001-98, National Security Agency Technical, January 1998.

[24] P. Sibun and J. C. Reynar. Language identification: Examining the issues. In *Proceedings of SDAIR-96, the 5th Symposium on Document Analysis and Information Retrieval*, pages 125–135, 1996.

[25] P. Sibun and A. L. Spitz. Language determination: natural language processing from scanned document images. In *Proceedings of ANLP-94, the 4th conference on Applied natural language processing*, pages 15–21. Morgan Kaufmann Publishers Inc., 1994.

[26] C. Souter, G. Churcher, J. Hayes, J. Hughes, and S. Johnson. Natural language identification using corpus-based models. *Hermes Journal of Linguistics*, 13:183–203, 1994.

[27] L. Wittmann, T. Pêgo, and D. Santos. Português do Brasil e de Portugal: alguns contrastes. In *Actas do XI Encontro da Associação Portuguesa de Linguística*, pages 465–487, 1995.

[28] D.-V. Ziegler. *The Automatic Identification of Languages Using Linguistic Recognition Signals*. PhD thesis, State University of New York, 1991.