Language Determination: Natural Language Processing from Scanned Document Images

Penelope Sibun & A. Lawrence Spitz
Fuji Xerox Palo Alto Laboratory
3400 Hillview Avenue
Palo Alto, CA 94304 USA
sibun@pal.xerox.com

Abstract

Many documents are available to a computer only as images from paper. However, most natural language processing systems expect their input as character-coded text, which may be difficult or expensive to extract accurately from the page. We describe a method for converting a document image into *character shape codes* and *word shape tokens*. We believe that this representation, which is both cheap and robust, is sufficient for many NLP tasks. In this paper, we show that the representation is sufficient for determining which of 23 languages the document is written in, using only a small number of features, with greater than 90% accuracy overall.

1 Introduction

Computational linguists work with texts. Computational linguistic applications range from natural language understanding to information retrieval to machine translation. Such systems usually assume the language of the text that is being processed. However, as corpora become larger and more diverse this assumption becomes less warranted. Attention is now turning to the issue of determining the language or languages of a text before further processing is done. Several sources of information for language determination have been tried: short words (Kulikowski 1991, Ingle 1976); n-grams of words (Batchelder 1992); n-grams of characters (Cavner & Trenkle 1994); diacritics and special characters (Beesley 1988, Newman 1987); syllable characteristics (Mustonen 1965); morphology and syntax (Ziegler 1991). Each of these approaches is promising although none is completely accurate. More fundamentally, many rely on relatively large amounts of text data and all rely on data in the form of character codes (e.g., ASCII).

In today's world of text-based information, however, not all sources of text will be character coded. Many documents such as incoming faxes, patent applications, and office memos are only accessible on paper. Processes such as Optical Character Recognition (OCR) have been developed for mapping paper documents into character-coded text.

However, for applications like OCR, it is desirable to know the language a document is in before trying to decode its characters. There appears to be a fundamental Catch-22: natural language processing systems want to be able to work automatically with arbitrary documents, many of which may be available only on paper, and in the process, they minimally need to know which language or languages are present. The algorithms cited above can determine a document's language, but they require a character-coded representation of the text. OCR can produce such a representation, but OCR does not work well unless the language(s) of the document are known. So how can the language of a paper document be determined?

We have developed a method which reliably determines the language or languages of a document image. In this paper, we discuss Roman-alphabet languages such as English, Polish, and Swahili; see Spitz (1994) for a discussion of the determination of Asian-script languages. Our method finesses the problems inherent in mapping from an image to a character-coded representation: we map instead from the image to a *shape-based representation*. The basal representation is the *character shape code* of which there are a small number. These shape codes are aggregated into *word shape tokens* which are delimited by white space. From examining these word shape tokens we can determine the language of the document. An example of the transformation from character codes to character shape codes is shown in figure 1.

Character codes

```
Confidence in the international
monetary system was shaky enough be-
fore last week's action.
```

Character shape codes

```
AxxAAxxxx ix AAx ixAxxxxAixxxA
xxxxAxxg xgxAxx xxx xAxAg xxxxgA Ax-
Axxx AxxA xxxA'x xxAixx.
```

Figure 1: Character code representation and character shape code representation.

The shape-based representation of a document is proving to be a remarkably rich source of information. While our initial goal has been to use it for language identification, in support of downstream OCR pro-

cesses, we are finding that this representation may itself be sufficient for natural language applications such as document indexing and content characterization (see Nakayama (this volume), Sibun & Farrar 1994). We find these indications exciting because OCR is an expensive, slow, and often inaccurate process, especially in the presence of printing and scanning artifacts such as broken or touching characters or skew or curvature of text lines. Thus, if our technique allows natural language processing systems to apply OCR selectively or to side-step OCR entirely, such systems will become faster, less expensive, and more robust.

In this paper, we first explain the background of our system that constructs character shape codes and word shape tokens from a document image. We next describe our method for language determination from this shape-based representation, and demonstrate our approach using only the three languages English, French, and German. We then describe an automated version of this process that allows us to apply our techniques to an arbitrary set of languages and show its performance on 23 Roman-alphabet languages.

2 Character shape codes and word shape tokens

Our determinations about document characteristics are made neither on the raw image[1] nor on the character codes by which the document can be represented. The determinations are made on a shape-based representation built of a novel component, the character shape code (Spitz 1993).

Four horizontal lines define the boundaries of three significant zones on each text line (see figure 2). The area between the bottom and the baseline is the descender zone; the area between the baseline and the top of such characters as x is the x zone; and the area between the x-height level and the top is the ascender zone.

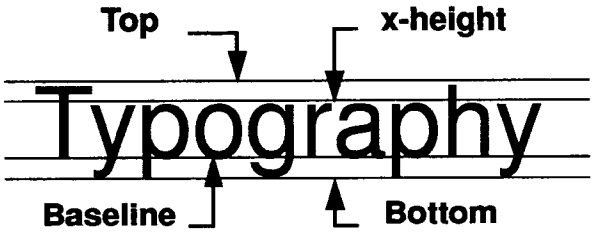

Figure 2: A text image showing the text line parameter positions: Top, x-height, Baseline and Bottom.

Characterizations of the number of connected components in a character cell and, in some instances, their aspect ratios, contribute to the coding. Thus most characters can be readily mapped from their positions relative to the baseline and x-height to a small number of distinct codes (see figure 3).[2]

Character shape code	Character
A	A-Zbdfhkltß0-9#$&/@|
x	acemnorsuvwxz
i	iáàâéèêîôûùñ
g	gpqyç
j	j
U	äëïöüÖÜ

Figure 3: Character shape codes.

2.1 Typesetting effects

Typesetters use different conventions. For example, in German text ü may be set as ue and ß may be set ss. Therefore, there may be several-to-one mappings of typeset information to character shape codes, since ü maps to U and ue to xx.

If this shape mapping can be done from document images, it can more trivially be accomplished from character-coded documents (e.g., ASCII, ISO-Latin-1, JIS, Unicode), providing, of course, that the method of encoding is known.

2.2 Computational complexity

Our approach takes on a much less difficult problem than does OCR. There is no need to investigate the fine structure of character images, the number of classes is small, and measurements are largely independent of font or typeface. As a result, the process of classifying text into character shape codes and aggregating those codes into word shape tokens is two to three orders of magnitude faster than current OCR technology.

3 Language determination

We have found that we can readily distinguish the language of a document for 23 Roman-alphabet (mostly European) languages from a relatively small text. This technique exploits the high frequency of short words in such languages and the diversity of their word shape token representations.

In this section, we describe our method for determining a document's language from the shape-based representation derived from the image (some of this

1. Document images may be obtained by scanning of paper documents, by retrieval from a document image database, or by digital rendering of a high level representation of the document.

2. This paper adopts the following conventions: `monospaced` to represent input characters, **boldface** to represent the character shape codes (**A**, **x**, **i**, **g**, **j**, **U**), and sans-serif to represent typographic conventions.

work has been reported in Nakayama & Spitz 1993). Our system learns how to discriminate a set of languages; then, for any input document, the system determines to which language it belongs. Our method uses the statistical technique of Linear Discriminate Analysis (LDA). First, we demonstrate the method using a hand-selected set of distinguishing features for a small set of languages. In section 4, we describe our process for automating the selection of distinguishing features across an arbitrary number of languages, and show the results on a corpus that includes documents from 23 languages.

Our initial set of discriminable languages comprised English, French, and German. To ascertain the set of discriminating features, we built a training corpus of approximately 15 scanned images of one-page documents for each language. We tokenized these images following the procedure described in section 2. This resulted in 7621 tokens from English, 6826 tokens from French, and 5472 tokens from German. We then ranked the frequency of word shape tokens across each corpus and noted the ten most frequent tokens. By comparing these top ten word shape tokens for each of the languages, we were able to select one per language that was both frequent in that language and less frequent in the other languages. Intuitively, each of these tokens is characteristic of its language; therefore, we call these *characteristic* tokens (see figure 4). The characteristic token for English is AAx; AAx constitutes 7% of the tokens in the English corpus and is quite rare in the others. In the German corpus, Aix is not the most frequent token: xx, xxA, Aix, and xxx each make up about 3% of the corpus while Axx constitutes 6%. However, of the five, only Aix is rare in the other languages. While Ax is frequent in all three corpora, it is overwhelmingly frequent in French, where it makes up 11% of the tokens (vs. 4% for English and 2% for German). These differences in the distribution of the characteristic tokens in the three corpora are sufficient for LDA to correctly identify each language almost every time (see figure 5).[3] The documents are from the training corpus: by a process called cross-validation, each was removed from the training corpus one at a time and classified based on the discriminating results from training on the rest of the corpus.

	English		French		German	
Token	**Rank**	Word	**Rank**	Word	**Rank**	Word
AAx	1	the				
xA	2	of	7			
Ax	3	to	1	la	6	
ix	4	is			10	
xxA	5	and			3	auf
xx	6		2	en	2	an
Axx	9		3	les	1	der
xxx	8		4	aux	5	wer
gxx			5	pas		
Aix					4	die

Figure 4: Most frequent word shape tokens in English, French and German: the top five for each language are shown; rankings of these are shown for the other languages when they fall in the top ten; shading indicates the characteristic token for each language; and common words that map to the top five tokens for each language are shown.

		Language assigned		
		English	**French**	**German**
Language of document	**English**	13		
	French		17	1
	German	1		14

Figure 5: Number of documents from each corpus assigned to each language.

It may be noted that each of the top five word shape tokens in each of the English, French, and German corpora is a mapping of closed class words such as determiners, conjunctions, and pronouns. This is not surprising, since closed class words are frequent in European languages. Of course, other words map to these word shape tokens too. For example, in English, the word `flu` maps to AAx. However the overwhelming proportion of AAx tokens in the English corpus are mappings of `the`. Since `the` is such a common word in English, we can expect AAx to be characteristic of any shape-level representation of an English document. Similar situations obtain in the other languages.

While it may seem fortuitous that in English AAx is virtually always a mapping of `the`, unique word shape tokens are more common in Roman-alphabet languages than one might suppose. We mapped an English lexicon of surface forms into word shape tokens and discovered that 20% of the resulting word shape tokens were unique; examples include the surface forms `apple` and `apples`.

4 Automated language determination

In the previous section, we discussed the selection of discriminating word shape tokens by hand. We now describe our method for automating this process. We have been able to use this technique to discover a discriminating set of tokens for a large fraction of the languages written in the Roman alphabet. We initially tested this automated technique by recapitulating our

3. In the case of the German document that was misclassified, examination of the image reveals that, due to printing and scanning artifacts, many characters are artifactually touching each other.

work done by hand in discriminating English, French, and German. We then applied the technique to a 755-document corpus comprising 23 languages.

4.1 The automated method

While it is easy to hand-select a single discriminating token for each of a few languages, the task becomes more complex as the number of languages grows. Further, a single feature per language may no longer be sufficient; a profile, or vector of features, for each language would be more robust.

For the automated method, a corpus for each of the languages is scanned and tokenized, and the tokens are sorted by frequency. The n most frequent tokens for each corpus are selected. We apply stepwise discriminant analysis, a variant of LDA, to this token set: variables are selected one by one according to their ability to discriminate between languages. The optimal value of n has not yet been determined. We need to gather enough discriminating tokens to characterize the languages as completely as possible. However, if we use too many, the accuracy of the classification may actually be degraded; further, relatively uncommon tokens may improve performance on test data but may not work well in general. As we discuss below, $n = 5$ suffices for three languages, but may not be optimal for 23.

There are several considerations for ensuring that this process is robust. The size of the corpus for each language must be sufficiently large in terms of both the number of documents and the total number of word shape tokens. The number of documents must be large enough to enable the LDA testing procedure to systematically eliminate some of them for cross validation without skewing the overall characteristics of the corpus. The number of word shape tokens must be large enough to be reflective of the language in which the documents are written to allow for accurate comparison between languages. A further consideration is that the number of discriminating tokens used by the LDA system should be considerably smaller than the number of documents.

For our initial test we selected the five most frequent word shape tokens from each of English, French, and German; this formed a set of ten tokens (because of overlap between corpora). Using stepwise discriminant analysis, the system found the best way to use the tokens by selecting the single token that was most discriminating and then for each of the remaining tokens adding the next most discriminating tokens given the ones that had already been selected. This resulted in a ranking of nine discriminating tokens (Ax, xA, ix, Aix, Axx, xx, AAx, xxA, xxx). The tenth was not found to improve the reliability of the discrimination; in fact accuracy peaked at four tokens.

We compared the performance of the automated system with that using the hand-selected tokens. When the top three automatically-selected tokens were used, performance was comparable to that of the three hand-selected tokens. Interestingly, there is no overlap in the misclassification of documents. Using four automatically-selected tokens, the system classified all but one document correctly (see figure 6).

		Language Assigned		
		English	French	German
Language of document	English	13		
	French		18	
	German	1		14

Figure 6: Assignment of documents to language using four automatically-selected discriminating tokens.

4.2 Automated determination for many languages

We have constructed a database of 755 one-page documents in 23 languages including virtually every European language written in the Roman alphabet. There are 18 Indo-European languages: Afrikaans, Croatian, Czech/Slovak[4], Danish, Dutch, English, French, Gaelic, German, Icelandic, Italian, Norwegian, Polish, Portuguese, Rumanian, Spanish, Swedish, and Welsh. There are two Uralic languages: Finnish and Hungarian. Finally, we include three languages from disparate families: Turkish, Swahili, and Vietnamese.

To construct a set of discriminating features, we selected the five most frequent word shape tokens from each language. Because of overlap, this resulted in 23 tokens. Some of these discriminating tokens have a high frequency across languages; in fact, xx appears in the top five of 22 of the languages we examined. However, even when we consider 23 languages, there are eight tokens appearing in the top five of one language which do not appear in the top five of any others. (This does not mean of course, that these tokens do not appear in other languages at all, but simply that they are relatively much less frequent.) The 23 tokens comprise the set (x, xx, xxx, xxxx, i, ix, xi, xix, A, AAx, Ax, AxA, AxAx, Axx, Axxx, xA, xxA, Ai, Aix, g, gx, xg, xxg, jx).

As before, we used LDA to build a statistical model of the language categorizations, and by cross validation tested the accuracy of the model (see figure 7). Our overall accuracy is better than 90%, while the accuracy for individual languages varies between 100% and 75%, with an outlier of 44% for Czech/Slovak. Examination of misclassifications proves somewhat instructive, as can be seen in the confusion matrix in figure 8. For example, Dutch and Afrikaans are closely related languages, and the only error in either language is the categorization of one Afrikaans document as Dutch. Among the five

4. We initially considered Czech and Slovak as separate languages, but this yielded worse results than combining them. We feel our decision was legitimate because "Slovak is similar enough to Czech to be considered by some as merely a dialect" despite "the existence of slightly different alphabets, as well as distinct literatures" (Katzner 1986, p 91).

Romance languages - French, Italian, Spanish, Portuguese, Rumanian - nine of the ten classification errors are within that language family. For the Scandinavian language family - Danish, Norwegian, Swedish, and Icelandic - the pattern is less clear. Two Norwegian documents are classified as Icelandic, but the three other errors in that family are classifications outside of the family.

Language	abbr	Acc (%)	Language	abbr	Acc (%)
Afrikaans	af	97	Italian	it	95
Croatian	cr	100	Norwegian	no	95
Czech/Slovak	cs	44	Polish	po	100
Danish	da	96	Portuguese	pt	96
Dutch	du	100	Rumanian	ru	93
English	en	95	Spanish	sp	95
Finnish	fi	75	Swahili	sa	97
French	fr	92	Swedush	sw	98
Gaelic	ga	86	Turkish	tu	93
German	ge	97	Vietnamese	vi	100
Hungarian	hu	94	Welsh	we	97
Icelandic	ic	96			

Figure 7: Language detection accuracy. The abbreviations shown are used as indices in figure 8.

Croatian, Czech/Slovak, and Polish are all Slavic languages; Hungarian and Finnish are related to each other but not to any other European languages. However, there is a large cluster of errors within the set of these five languages. Most of these errors are for Czech/Slovak documents; in fact, Czech/Slovak was recognized far less accurately than any other language and it is unclear why. It may be the case that many of these documents are of poor quality. Seventeen of the 69 errors seem to be random; while we are working to reduce such errors, it is unlikely that we can eliminate them entirely. It is possible that 23 discriminating tokens is not sufficient; since the accuracy has been improved by the addition of each new token, adding several more may continue the improvement.

4.3 Discussion of methodology

While LDA has proved adequate, there are some drawbacks to this technique. We are somewhat disappointed by the system's accuracy. Examination of token frequencies suggests that the profiles for each language are distinct enough that 90% should be a lower bound on classification accuracy. However, for several languages the accuracy was much lower, and for many more it was not much better than 90%. A more troubling problem is the instability of the model. When we add or delete languages, overall accuracy fluctuates between 80% and 93%. This suggests that removing a language affects the typical distribution across all languages, which should not be the case. It is difficult to identify the underlying causes of both of these observations. Finally, the results of LDA are difficult to interpret. All these considerations suggest that LDA may not be the best technique to use. Therefore, we are exploring alternative statistical models, such as classification trees, to find an approach that is more robust for our task.

5 Comparison with other methods

It is difficult for us to compare our approach to other methods of language determination. Most sources we have found are simply guides for librarians or translators. For example, Ingle (1976) found that the presence or absence of specific one- or two-character words suffices to distinguish among 17 Roman-alphabet languages. There are several implemented systems, some of which report on their accuracy, but none is addressing exactly the same problem as ours: all work from character-coded text. However, it is useful to get a ballpark estimate of the accuracy to be expected of character-based systems.

Batchelder (1992) trained neural networks to recognize 3-6 character words from 10 languages. While her networks had high accuracy in recognizing words from the training set, their best-case performance on untrained words was 53%, thus making accurate determination of a document's language highly unlikely.

Cavner and Trenkle (1994) used n-grams of characters for n = 1 to 5. Their task was not language determination *per se*, but determining to which country's newsgroup (in the netnews soc.culture hierarchy) a document belonged. In each newsgroup, the documents were written in either English or other language(s). For documents longer than 300 characters, the system determined the correct newsgroup with 97% accuracy when using the 100 most frequent n-grams. These results are good, but the technique should be tested on a set of documents for which the languages are known and the topics are varied.

Kulikowski (1991) used a semi-automatic method to determine a profile of frequent 2-3 character words for nine languages. He claims at least 95% accuracy for determining that a single-language document is in one of the nine languages or in none of them. Unfortunately he does not expand on this claim. Henrich (1989) used criteria such as language-specific word-boundary character sequences and common short words to determine the language of sentences in English, French, or German.

Mustonen (1965) used discriminant analysis to distinguish English, Swedish, and Finnish words. His system, which used 43 discriminating features, such as particular letters and syllable types, performed with 76% accuracy. This relatively poor performance is probably due to the data being isolated words rather than documents, though it may also be due to overfitting of the test data by too many features (see section 4.1).

We would like to emphasize that our statistics on word shape token distribution across the various lan-

	Detected Language																						Error Total
	en	ge	du	af	fr	it	sp	pt	ru	da	no	se	ic	ga	we	cr	cs	pl	hu	fi	tu	sa	vi	
en	36	1			1																			2
ge		29												1										1
du			28																					0
af			1	29																				1
fr					23			1						1										2
it						35		1	1															2
sp							39	2																2
pt							1	25																1
ru						3			38															3
da					1					25														1
no											39		2											2
se												40								1				1
ic													23							1				1
ga						2			2				1	31										5
we															30							1		1
cr																33								0
cs									1							6	24	16	3	5				31
pl																		28						0
hu																		2	29					2
fi																	6		2	24				8
tu							1													1	28			2
sa																					1	37		1
vi																							13	0
	0	1	1	0	2	5	2	4	4	0	0	0	3	2	0	6	6	18	3	9	1	1	0	69

Figure 8: Confusion matrix showing detection accuracy between languages. Numbers on the major diagonal indicate the number of correct classifications for each language. Numbers off the diagonal indicate classification errors.

guages are generated entirely from scanned images of text. We feel this is important because the text whose language we are trying to identify should not be systematically different in any way from the texts from which the discriminate analysis was generated. For example, typographic conventions such as a ligature between a vowel and an acute accent (as in characters like à) cause the character shape code recognizer to classify these characters as A. However, if we were working from encoded on-line corpora we would "know" that such a character should be classified as i.

6 Conclusion

We have described our method for generating word shape tokens from images and have shown how this shape-level representation of the text can be used for important tasks such as determining the language or languages of a document. We have shown that the method can discriminate among 23 languages with high accuracy.

Since our approach is statistical, the more text our system sees in a document image, the more reliably it can determine the document's language. So far, we have not tried to determine the language of a document shorter than 27 words, and most of the documents we work with are a few hundred words long (2000-3000 characters). We are investigating the lower bound on the length of texts whose language we can reliably determine. In the ideal case we would be able to detect the presence of a very few words of a secondary language interpolated into a document predominated by another language.

In other work, we are using the shape-level representation as input to higher-level natural language processing systems for rudimentary content analysis. However, many sorts of information, particularly style characteristics, can be derived from the shape-level rep-

resentation directly. For instance, since the number of character shape codes extracted form a document is comparable to the number of characters, characterizations about word length in a shape-level representation apply as well to the character-coded version of the document. This word length characterization is not perfect: ligatures introduce some uncertainty. Additionally, braces, brackets, and parentheses which are typically set contiguous with words, are currently mapped to A, this will affect word length counts. We are refining the mapping to account for these delimiting characters.

Acknowledgments

We thank David Hull for his expertise in developing statistical methods and help in explaining them, Arlene Holloway for patiently scanning and processing most of our document image database and helping to analyze the results, Marti Hearst and Michael Berch for comments on drafts of this paper, and Jussi Karlgren for a last-minute reference.

Bibliography

Batchelder, Eleanor Olds, *A Learning Experience: Training an Artificial Neural Network to Discriminate Languages,* Unpublished Technical Report, 1992.

Beesley, Kenneth R., *Language Identifier: A Computer Program for Automatic Natural-Language Identification of On-line Text,* Language at Crossroads: Proceedings of the 29th Annual Conference of the American Translators Association, 12-16 Oct 1988, pp 47-54.

Cavner, William B. & Trenkle, John M., *N-Gram Based Text Categorization,* Proceedings of the Third Annual Symposium on Document Analysis and Information Retrieval, 11-13 April 1994, pp 161-169.

Henrich, Peter, *Language Identification for the Automatic Grapheme-to-Phoneme Conversion of Foreign Words in a German Text-to-Speech System,* Proceedings of Eurospeech 1989, European Speech Communication and Technology, Paris, Sept. 1989, pp 220-223.

Ingle, Norman C. *A Language Identification Table,* The Incorporated Linguist vol. 15 no. 4 pp 98-101, 1976.

Katzner, Kenneth, *The Languages of the World,* London: Routledge, 1986.

Kulikowski, Stan, *Using Short Words: A Language Identification Algorithm,* Unpublished Technical Report, 1991.

Mustonen, Seppo, *Multiple Discriminant Analysis in Linguistic Problems,* Statistical Methods in Linguistics, No. 4, Skriptor Fack, Stockholm, 1965, pp 37-44.

Nakayama, Takehiro & Spitz, A. Lawrence, *European Language Determination from Image,* Proceedings of the International Conference on Document Analysis and Recognition, 20-22 Oct 1993, pp 159-162.

Nakayama, Takehiro, *Modeling Content Identification from Document Images,* this volume.

Newman, Patricia, *Foreign Language Identification: First Step in the Translation Process,* Proceedings of the 28th Annual Conference of the American Translators Association, 8-11 October 1987, pp 509-516.

Sibun, Penelope & David S. Farrar, *Content Characterization Using Word Shape Tokens,* Procedings of the 15th International Conference on Computational Linguistics, Kyoto, Japan, 1994, pp 686-690.

Spitz, A. Lawrence, *Generalized Line, Word and Character Finding,* Progress in Image Analysis and processing III, Impedovo, Ed., World Scientific, 1993, pp 377-383.

Spitz, A. Lawrence, *Script and Language Determination from Document Images,* Proceedings of the Third Annual Symposium on Document Analysis and Information Retrieval, 11-13 April 1994, pp 229-235.

Ziegler, Douglas-Val, *The Automatic Identification of Languages Using Linguistic Recognition Signals.* Dissertation, State University of New York at Buffalo, 1991.