4. **Available Information Assumption**: Language and encoding identification can be very simple under the ideal conditions. If we assume that all the documents are in HTML or XML etc., that the languages and encodings have been specified by using the relevant codes everywhere and that only those languages and encodings are possible that can be specified using these standard codes, then we merely need to write a program to use this information. The solution can also depend on the availability of other sources of information, *e.g.*, the list of function words or of characteristic unique words for every language encoding pair.

3. Languages and Encodings

Support for many languages with speakers numbering more than 10 million, *e.g.* the languages of the Indian sub-continent, has been mostly non-existent on computers. The 'encodings' used for these languages (except nowadays Unicode) are not recognized by the operating systems or Web authoring tools or even by the HTML standards. And they cannot be, to the satisfaction of all, because there is no exhaustive commonly agreed upon list of languages of India, let alone that of encodings. What is a language for some is just a dialect for others. And for every language (or dialect), there are numerous very different encodings. This, in simple words, means that the Web pages written in Indian languages almost always contain 'wrong' encoding in the meta-tags, simply because there is no HTML code for the encoding used by the author of the Web page.

A problem closely associated with language identification is that of encoding identification. That this is also a research problem may not be evident at first if we think only in terms of standard encodings like ASCII, ISO-8859-1, UTF-8 etc. An examination of multilingual documents will show that, from the language processing point of view, we have to understand the term 'encoding' a bit differently. It does not just mean the encodings recognised by operating systems or by standards organizations. 'Encoding' is a scheme used to **encode** the text of a particular language. It is a mapping from the letters (or other units) of a script to numbers in a range (say, 0 to 255). Thus defined, the term includes unrecognised encodings, which are quite common in many parts of the world. These encodings may be transliteration schemes, but not always.

Language and encoding are closely interconnected in such a way that if we could identify the language, we would most probably have also identified the encoding, or vice-versa. It could be argued that they can be identified separately and that identification of one may help the other. Previous work and our own experience, however, indicates that considering language-encoding pair as one unit might be a reasonable way to solve the problem. Under certain situations we could, of course, have prior information about one of these. This can be used by the identification system to make a better prediction.

For our experiments, we have not used any prior information about language or encoding, even if it could have been obtained just by looking at the byte format, *e.g.*, in the case of Unicode encodings like UTF-8, UTF-16, etc. We did this to evaluate our system only for the cases where such prior information is not available. For identifying language-encoding pairs in all the cases, we have used only a statistical method, as explained later. The synthetically created test data (as it was too expensive to 'annotate' real life multilingual data) for evaluation is adequate for the assumptions we have made, but for different assumptions we will need to create such data differently.

4. Related Work

There is a long history of work on language identification. In fact, it was one of the first language processing problems for which a statistical approach was used. Ingle (1976) used a list of short characteristic words in various languages and matched the words in the test data with this list. Such unique strings based methods were meant for human translators.

The earliest approaches used for automatic language identification were based on that same idea of unique strings. They were called 'translator approaches'. Beesley's (1988) automatic language identifier for online texts used mathematical language models originally developed for breaking ciphers. These models basically relied on orthographic features like characteristic letter sequences and frequencies for each language.

Some of the methods were similar to n-gram based text categorization (Cavnar and Trenkle 1994) which calculates and compares profiles of n-gram frequencies. Cavnar also proposed that the top 300 or so n-grams are almost always highly correlated with the language, while the lower ranked n-grams give more specific indication about the text, namely the topic. Many approaches similar to Cavnar's have been tried, the main difference being in the distance measure used. The similarity measures tried for language identification include mutual information or relative entropy, also called Kullback-Leibler distance (Sibun and Reynar 1996), cross entropy (Teahan and Harper 2001), and mutual or symmetric cross entropy (Singh 2006).

Giguet (1995) relied upon grammatically correct words instead of the most common words. He used the knowledge about the alphabet and the word morphology via *syllabation*. He tried this method for tagging sentences in a document with the language name, *i.e.*, dealing with multilingual documents.

Johnson's method (Stephen 1993) was based on characteristic 'common words' of each language. This method assumes unique words for each language. In practice, the test string might not contain any unique words.

Cavnar's method, combined with some heuristics, was used by Kikui (1996) to identify languages as well as encodings for a multilingual text. He relied on known mappings between languages and encodings and treated East Asian languages differently from West European languages.