[*First International Conference on Language Resources & Evaluation*, Granada, Spain, 28-30 May 1998]

Trainer Beware: Corpora for Language/Encoding Identification

Florence Reeder
Mitre Corporation
McLean Virginia, USA
[freeder@mitre.org]

Abstract

Training-based approaches to language processing require corpora. For example, corpora are being used for lexicon development, spelling correction and machine translation. Typically, one wants the corpora to reflect the type of data that is to be handled by the given system. The problem is that the real-world data is frequently noisy and can introduce problems in training-based approaches. The question, then, is if one should "clean up" the data before training and if so, how much? We have faced this very dilemma in the training and use of language and encoding identification algorithms. We will first discuss the problem of language and encoding identification. Then, we will describe the problems faced by our system and our initial attempts at handling these questions. Finally, we will examine the results of the exploration with some recommendations for researchers dealing with corpora-based techniques.

Introduction

Language and encoding identification is the starting point for foreign language document processing. Without knowing the language, the encoding and by extension, the fonts necessary to manipulate a document, a user can easily give up on trying to use a document. Many users have faced the frustration of finding a Web-site without being able to read the information it contains. Similarly, automated tools such as search engines need to be able to detect the language and encoding of a document to be able to effectively search against a document space.

Language/Encoding Identification

Language identification is the process of finding the source language of a given document, such as English, Russian, Chinese or French. While this is generally something that users can determine by knowing the source of an e-mail message, a novice user could have difficulty in figuring out the distinctions between closely related languages. A user or automated processing tool must know the language of a given document to handle it with the appropriate tools, but this is not the only piece of information required for effective document handling. Users and automated tools must also determine the character encoding set of a document. An encoding is a computer mapping of the character set of a language to numeric codes which the computer uses for such processes as comparing two words for similarity. Adams (1993) has documented some of the problems of encoding schemes in internationalization when he describes the "vast proliferation of 'standard' character encodings" as a hindrance to internationalization. For instance, there are 3 commonly used Russian 8-bit encodings (ISO-8859-5, CP-1251 and KOI-8). Kikui (1996) describes a list of encodings for Chinese, Japanese and Korean. The range of possibilities is widened further by the proliferation of seven-bit transliterations and transcriptions of languages. A transliteration is when a character is represented through another character or group of characters. A common example is the quoted transliteration of German where umlauts are represented by double-quotes as shown in Figure 1. A transcription is a phonetic representation of characters. Figure 2 shows an example of this for German text. While movement towards Unicode and ISO 10646 are a step towards eliminating this problem (Adams, 1993), the wide range of existing systems will cause identification problems for the foreseeable future. Additionally, the variability of these two standards has not sufficiently converged to allow for quick and easy language determination.

* Als Fazit werden Markierungen fu"r eine erneuerte Beruf(ung)spastoral,
* Aufbau eines Freundeskreises: Zu Altersgenossen beiderlei Geschlechts werden neue, tiefere Beziehungen hergestellt.
* Zum Thema Entwicklungsaufgaben vgl. R. Oerter/Eva Dreher, Jugendalter, in: R. O"rter/L.Montada (Hrsg.), ...

Figure 1: ASCII transliteration of German text

* Als Fazit werden Markierungen fuer eine erneuerte Beruf(ung)spastoral,
* Aufbau eines Freundeskreises: Zu Altersgenossen beiderlei Geschlechts werden neue, tiefere Beziehungen hergestellt.
* Zum Thema Entwicklungsaufgaben vgl. R. Oerter/Eva Dreher, Jugendalter, in: R. Oerter/L.Montada (Hrsg.), ...

Figure 2: ASCII transcription of German text

Like many natural language processing tools, a translation engine requires that the document to be translated is in the same encoding and language for which the engine was designed. A document which is represented in a transliteration when sent to a system expecting an ISO encoding will not be able to be translated. It is desirable, therefore, to verify the language and encoding of a document before the document is sent to the translation process. Unfortunately, when detection of the language is combined with detection of the encoding, the problem becomes very complex. Finding the language for a known encoding or the encoding for a known language is more straightforward than finding both for a particular document.

Solutions to Language Identification Problems

Like many language processing problems, solutions to the language/encoding identification problem can be knowledge-based, corpora-based or a combination of both techniques. Knowledge-based techniques study the language characteristics and the coding characteristics of a given language and design simple tests to take advantage of these. For instance, one might use a lexicon and match words against the lexicon or one might devise a simple article test. Grefenstette (1996) evaluated an approach where the top, common, short words (such as articles and function words) were checked for. A statistical profile for these words was built and compared against a sample document. Similarly, Kikui (1994) presented a system which combines simple statistics with knowledge-based heuristics. Heuristics included looking for specialized header information and specialized character sequences to narrow the search space to a particular coding family or language family. Once the language family was discerned, statistical solutions are utilized to make the final determination.

An example of a corpora-based solution to the language identification problem is that described by Grefenstette (1996). For this system, the designers acquired the ECI CD-ROM for Danish, Dutch, English, French, German, Italian, Norwegian, Portuguese, Spanish and Swedish. The algorithm itself handles the pre-processing data by marking word boundaries and appending and pre-pending underscores to words. A simple probability is then calculated for any tri-graphs appearing more than 100 times. A probability profile is calculated for new documents and matched against the trained language signatures for documents. The language is determined by a close match of the signatures. Additionally, commercial systems are becoming available[1] which utilize corpora.

Corpora Approaches

Because we wanted to have an encoding recognition capability that detected more encodings than were commercially available, we decided to look at utilizing corpora approaches to extend existing capabilities. The lure of a corpora approach is that the designers of these algorithms advertise that little background knowledge is required to effectively utilize the algorithms. The problem seems simple and straight-forward: you want to recognize languages and encodings in documents, you collect a set of examples of these languages and encodings and then you train the algorithm to recognize these. Evaluating existing algorithms is equally simple: you pull together a set of documents and send these documents to the given system. If the system correctly reports the language and encoding, you declare success. In looking at this problem, the algorithm developers promised relatively easy success.

Our experience is that this is not the case. The lessons that we learned document the ways in which it is not true. We will first describe the algorithm we used and then give a brief list of the problems we had in trying to utilize a corpus approach for a real-world problem.

The algorithm we utilized is an n-graph approach which was available as a government-off-the-shelf (GOTS) tool. This algorithm, Acquaintance, (Huffman, 1996) was originally designed for text retrieval, yet its abilities to categorize like documents made it a candidate for language and encoding identification. It works by building a representative n-graph signature for each category to be detected. An n-graph is defined as a character sequence (in this instance, 5). New data is compared against the trained signatures. Since the algorithm demonstrated a resistance to garbled text and a graceful degradation in the presence of error-filled text (Huffman, 1996), we believed that the work of assembling a corpus for real-world data would be handled effectively.

Applying Research Algorithms to Real World Data

The first question we addressed was the question of how the corpora-based approaches would identify languages and encodings for documents that are "sloppy." The answer is that performance can degrade when non-language information is introduced into a document. Because of the results of this testing and because we wanted to add new languages and encodings to the detection process, we decided to train new detection algorithms. It is during this process that we learned much about the dangers of corpora-based approaches to language processing.

Encouraged by the results of the work previously described, we anticipated the ability to apply corpora-based techniques to our problem. Our system is a shell around commercial off-the-shelf (COTS) translation engines. The system provides pre- and post-processing for documents such as encoding conversions and spelling correction. Users generally know the language or language family of a document, but the encoding is a constant source of confusion and error.

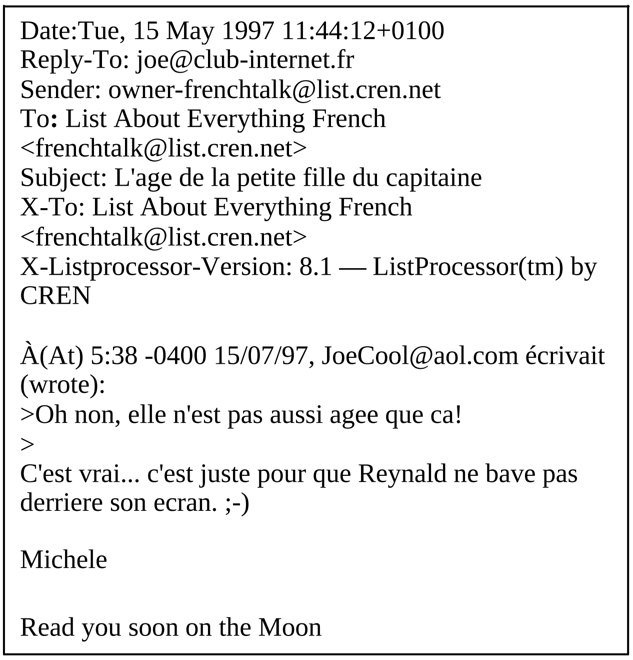
Due to its ready availability, we selected the Acquaintance algorithm to explore the problems of language and encoding identification. When applied to the TREC problem, the designers of the algorithm included pre-processing that stripped Standard Graphical Markup Language (SGML) tags from the data. The processing also dropped all "non-alphabetic" characters and transformed everything to lower-case. This scheme is much like the scheme used by Grefenstette and almost immediately posed problems for language and encoding recognition.

Typically, research systems utilize existing corpora from organizations such as the Linguistic Data Consortium (LDC) or the European Language Resource Association (ELRA) to train their algorithms. The most ready source of data appears to be newspaper texts and news wire feeds. Additionally, this data is frequently SGML

[1] INSO Corporation and ALIS Technologies are two such providers. (Reeder, 1998).

tagged to promote multiple uses and has been validated by independent sources.

Unlike this information, the data that we must process is not the clean, well-formed data generally used in research. On the World Wide Web (WWW), data is characterized by a mix of encodings, documents containing multiple languages, documents with Hypertext Markup Language (HTML) tags, ASCII art in signature blocks and other irregularities. Flanagan (1996) describes this type of data when she discusses work being done at CompuServe for automated translation of on-line bulletin boards, e-mail and search results. "Online text is characterized by great variability. It ranges from informal and highly stylized forum messages to business letters and technical texts. ... Forum messages are often hurriedly written, or written to evoke the personality of the writer, and can contain numerous spelling, punctuation and grammar errors." Figure 3 shows a typical e-mail message with two encoding schemes, mixed languages and headers. All of these serve to hinder performance of training-based approaches. The text in Figure 3 also demonstrates short documents. If the text is small (under 100 characters), detection is very difficult. Notice that in this case, the number of characters which would be identified as English (the header information) far exceeds the body of the message.



Date:Tue, 15 May 1997 11:44:12+0100
Reply-To: joe@club-internet.fr
Sender: owner-frenchtalk@list.cren.net
To: List About Everything French
<frenchtalk@list.cren.net>
Subject: L'age de la petite fille du capitaine
X-To: List About Everything French
<frenchtalk@list.cren.net>
X-Listprocessor-Version: 8.1 — ListProcessor(tm) by
CREN

À(At) 5:38 -0400 15/07/97, JoeCool@aol.com écrivait
(wrote):
>Oh non, elle n'est pas aussi agee que ca!
>
C'est vrai... c'est juste pour que Reynald ne bave pas
derriere son ecran. ;-)

Michele

Read you soon on the Moon

Figure 3: Mixed encodings and languages

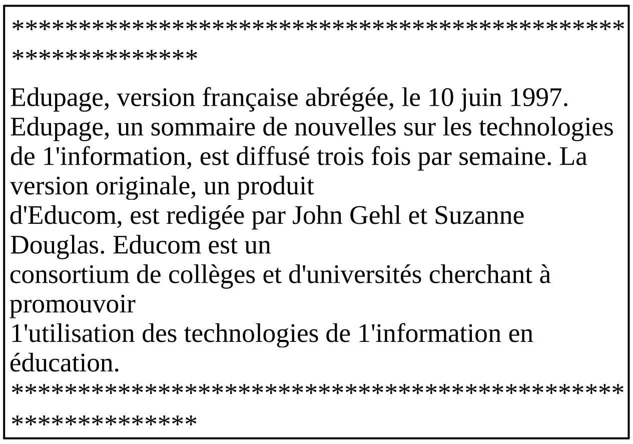
Results

Our initial experiments focused on French, German and Spanish documents while recognizing the ISO-8859-1 (ISO1) encoding and one or two ASCII-based transliterations for each language. Our initial corpus was primarily newspaper texts, SGML-tagged. We were recording success rates of over 95% on the corpus, but when we applied this to the real-world data, performance dropped to the low 80% range. This varied across languages with German retaining higher accuracy. After re-training on cleaned data, the accuracy improved to above 95%. In doing the cleaning, we addressed some of the punctuation problem. By changing the definition of punctuation, we increased accuracy from 97% to 98.6%. Yet to be accounted for, however, is how to handle the unidentified documents when nearly 90% of these are incorrectly identified because they are too small (under 100 characters) or contain a mix of languages and encodings.

Lessons Learned

The first lesson we learned was that data can be mislabeled. In particular, we found that data that was labeled as Portuguese, but was Spanish. The data had been categorized according to place of origin and was assumed to be in a single language. The organizers of the data had not validated the consistency of the data. It is especially important, then, when tasking those poor graduate students to collect language examples for training and testing, to ensure that their data is validated by language experts.

Next arose the problem of the uniformity of the data. The data of news articles tended to be written in one particular style. Additionally, it was relatively well-formed data. When we applied the training set to real-world data, we found much that was not well-formed. Thus, it is important that the corpora accurately reflect the type of data to be recognized. For instance, a corpus that contained a significant number of stock quotes and baseball box-scores such as in newspaper corpora, caused the recognition algorithm to believe that any document containing a large proportion of numbers was in the language containing the stock quotes. We then addressed the issue of whether or not to utilize tagged data as such. Allowing the tags to remain in the training set introduced yet another problem. When we looked at the n-graphs generated by the training program, we realized that among the highest scoring n-graphs often was the name of the language of the article contained in the tags. This caused recognition to be high in the training set, but low in the general data.



****
****

Edupage, version française abrégée, le 10 juin 1997.
Edupage, un sommaire de nouvelles sur les technologies
de 1'information, est diffusé trois fois par semaine. La
version originale, un produit
d'Educom, est redigée par John Gehl et Suzanne
Douglas. Educom est un
consortium de collèges et d'universités cherchant à
promouvoir
1'utilisation des technologies de 1'information en
éducation.
****
****

Figure 4: Repeated Header with ASCII Art

Tag removal alerted us to another problem - that either documents needed to be scrubbed before training and recognition or that the algorithms needed to account for features such as ASCII art. Figure 4 shows a standard header that could skew the statistics when using this data to train language examples. Because transliteration schemes use unusual characters and different character sets use different ranges of numbers to reflect word boundaries and punctuation, it becomes difficult to scrub documents. For example, blindly removing the "~" character from documents could cause the system to miss the Spanish transliteration which substitutes this character for "**ñ**".

After working with a data set that was more standardized, we attempted to gather a month's worth of documents which had been provided for translation and catalog these. This has proven to be a costly and time-consuming undertaking. It is worth examining more closely the questions of effective corpora design.

Future Work

We are now addressing questions such as how much data is sufficient to accurately represent the entire search space? This is somewhat dependent on the algorithm, although the ability to transform similar documents in a language to any encoding is very helpful in designing a complete corpus.

Since statistical approaches tend to degrade when input data is sparse, how much can we expect from these algorithms? Cleaning documents sometimes yielded exemplars which were less than 100 characters long. These cannot generally be used for training and are missed by many recognition algorithms. We hope to have more realistic answers to these questions soon, but preliminary results indicate that more is better for training sets and that there are data size thresholds below which statistical approaches are not the most optimal choice.

Finally, we will continue to examine commercial solutions and training options. Algorithmic improvements, hybrid approaches to the recognition problems and better training data should enhance solutions to this problem.

Conclusion

The problem of collecting a corpus for statistical training algorithms remains. We caution future researchers to carefully analyze the corpus before applying algorithms. While this detracts from the advantages of a corpus-based approach, applying appropriate domain knowledge to the corpus design will provide a better product long-term. We also recommend that researchers and users of these algorithms be keenly aware of algorithm limitations when analyzing the corpus.

References

Adams, G. (1993). Internationalization and Character Set Standards. *Standard View,* 1(1), 31 - 39.

Flanagan, M. (1996). Two Years Online: Experiences, Challenges and Trends. In *Expanding MT Horizons: Proceedings of the Second Conference of the Association for Machine Translation in the Americas,* (pp. 192-197). Washington, DC: AMTA.

Grefenstette, G. (1995). Comparing Two Language Identification Schemes. In *Proceedings of the Third International Conference on Statistical Analysis of Textual Data.* Also from http://www.rxrc.xerox.com/research/mett.

Huffman, S. (1996). Acquaintance: Language-Independent Document Categorization by N-Grams. In *The Fourth Text Retrieval Conference (TREC-4),* (pp. 359 - 371). Gaithersburg, MD: National Institute of Standards and Technology.

Kikui, G. (1996). Identifying the Coding System and Language of On-line Documents on the Internet. In *Proceedings of the Sixteenth International Conference on Computational Linguistics* (pp. 652 - 657). Copenhagen, Denmark.

Reeder, F. (1998). SILC Review. In *Multilingual Communications and Technology,* (9)2 pp. 20-21