Comparative Linguistics via Language Modeling

Scott Drellishak
Department of Linguistics
University of Washington
Box 35430
Seattle, WA 98195-4340
sfd@u.washington.edu

Abstract

Natural languages are related to each other genetically. Traditionally, such genetic relationships have been demonstrated using the comparative method. I describe a software system that implements an alternative method for detecting such relationships using simple statistical language modeling.

1 Introduction

The historical processes of change that affect natural languages result in genetic relationships between those languages. Speakers of a single language may become divided into multiple isolated speech communities, each of which subsequently develops a different new language. Well-known examples include the Indo-European languages, all of which are believed to derive from a language spoken several thousand years ago in Central Asia, and the Romance languages, all of which derive from the Vulgar Latin of the later Roman Empire. In the field of linguistics, such genetic relationships have traditionally been demonstrated using the comparative method, in which the lexicons of two languages are compared to find cognates, then regular sound changes are proposed to account for the observed differences between these words.

In this paper describes the development and implementation of an alternative method for detecting genetic relationships between languages. The method involves computing, using statistical language modeling techniques, a metric of textual (and, it is hoped, genetic) similarity between pairs of single-language texts. Pairs of texts that are more similar, it is hypothesized, ought to be more closely related genetically than pairs that are less similar. The metric of similarity described in §3 is based on an *n*-gram model of the distribution of individual letters in each text.

2 Data

In order to compare texts using statistical modeling, it is necessary that they be in the same writing system, or nearly so. Consider the case of a pair of texts, one of which is written in the Latin alphabet, the other in Japanese *kanji*. Statistically modeling the Latin text based on *kanji* character frequencies, or vice-versa, would produce a probability of zero (modulo smoothing) because the character sets do not overlap at all. In order to avoid this problem, all texts were selected from languages written in some variant of the Latin alphabet. Furthermore, it was necessary that the results of the method being developed could be compared against some standard; therefore, languages were chosen from families whose genetic relationships are well-understood. In particular, most of the languages chosen are part of the Indo-European language family, with a few additional languages from geographically adjacent language families such as Finno-Ugric and Turkic included for comparison.

Two corpora of texts were used in this project. The first contains works of prose fiction, either novels or, where novels were not available, stage plays. Because texts that are out of copyright are freely available on the Internet, works from a 100-year window centered around 1900 A.D. were se-

French:	Jules Verne. 1908. *Le pilote du Danube*. www.gutenberg.org/files/11484/11484-8.txt
Italian:	Cletto Arrighi. 1890. *Nana a Milano*. www.gutenberg.org/dirs/etext05/8nnml10.txt
Portuguese:	Camilo Castelo Branco. 1866. *A Queda d'um Anjo*. www.gutenberg.org/files/17927/17927-8.txt
Romanian:	Liviu Rebreanu. 1920. *Ion*. ro.wikisource.org/wiki/Liviu_Rebreanu
Spanish:	Concha Espina. 1909. *La Niña de Luzmela*. www.gutenberg.org/files/11657/11657-8.txt
English:	H. G. Wells. 1898. *The War of the Worlds*. www.gutenberg.org/files/36/36.txt
Dutch:	Nicolaas Beets. 1839. *Camera Obscura*. www.gutenberg.org/files/15975/15975-8.txt
German:	Hermann Hesse. 1922. *Siddhartha*. www.gutenberg.org/dirs/etext01/8sidd12.txt
Danish:	Herman Bang. 1896. *Ludvigsbakke*. www.gutenberg.org/files/10829/10829-8.txt
Norwegian:	Henrik Ibsen. 1884. *Vildanden*. www.gutenberg.org/files/13041/13041-8.txt
Swedish:	Laura Fitinghoff. 1907. *Barnen ifran Frostmofjaellet*. www.gutenberg.org/dirs/etext06/8bifr10.txt
Czech:	Karel Čapek. 1921. *R.U.R.* www.gutenberg.org/files/13083/13083-0.txt
Polish:	Bruno Schulz. 1934. *Sklepy cynamonowe*. www.gutenberg.org/dirs/etext05/sklep10.txt
Hungarian:	Ferenc Donászy. 1906. *Az arany szalamandra*. www.gutenberg.org/files/18365/18365-0.txt
Finnish:	Aleksis Kivi. 1870. *Seitsemän veljestä*. www.gutenberg.org/files/11940/11940-8.txt

Figure 1: The "novels" corpus

lected whenever possible. The complete list of texts in this data set, hereafter referred to as the "novels" data set, is shown in Figure 1. Two subsets of the novels corpus, as described below, were used in development, and the whole corpus was used for testing.

The second corpus of texts consisted of the text of the Universal Declaration of Human Rights (O.H.C.H.R. 2006) in 49 different languages. This data set (hereafter referred to as the "UDHR" data set) was used only in final testing. The length of each text in the UDHR data set was significantly shorter than the length of each text in the novels set. This resulted in a sizable reduction in training times, which were often quite long with texts in the novels data set. In spite of the reduced size of the data, however, the results of the system were not degraded.

The texts in both corpora come from a variety of languages for which a variety of encoding schemes exist. In order to enable cross-linguistic comparison, all texts were converted into the Unicode UTF-8 encoding, in which each character is represented either by a single 7-bit character or a sequence of 8-bit characters.

3 Methods and Development

The statistical technique used in the method described here was *n*-gram modeling of letter distributions. To accomplish this, each text was preprocessed: all characters were changed to lower case, all non-letter characters were removed, and each orthographic word was placed on a single line with its characters separated by spaces. Training of *n*-gram models was then accomplished using the SRILM toolkit (Stolcke 2002), treating each word in the original text as a "sentence" of individual letters.

3.1 *n*-gram Models

The algorithm used for detecting the genetic relationships in a set of languages was as follows. First, each text was preprocessed. Next, an *n*-gram model was trained on each text using SRILM's `ngram-count` program. Using these *n*-gram models, a matrix of perplexity values was calculated by using SRILM's `ngram` program to apply each model to every other text in the data set. The two texts with the lowest perplexity value were then merged into a single text. This process was repeated until only a single text remained. The sequence of merge operations that occur during this process form, bottom up, a binary-branching tree containing all the languages in the data set, which can be compared to the known genetic relationships of the languages for evaluation.

SRILM's tools provide a wide variety of options for building *n*-gram models. Various values of these options were explored in the development of the system, including various *n*-gram orders and smoothing techniques. Because the algorithm described above involves choosing the lowest perplexity, small variations in the perplexity matrix did not affect the output of the system. In particular, varying the smoothing method did not alter the perplexity values enough to affect the final output tree; therefore, SRILM's default smoothing (Good-Turing) was used.

	Romance	Indo-European
orth.	3: (((pt es) it) fr) ro	5: (pl ((en ro) (pt es))) cs
plain	4: (((pt es) it) fr) ro	5: ((pl cs) ((pt es) ro)) en
arch.	6: fr (((es pt) it) ro)	4: en ((pl ((es pt) ro)) cs)
C+V	3: fr (ro ((pt it) es))	3: ((pl cs) ((pt es) ro)) en

Table 1: *n*-gram development results

Another technique that did affect the output tree was filtering the texts to reduce their character "vocabulary", producing more vocabulary overlap between languages. Three different filtered versions of each text were created, each more reduced than the last: a version in which all letters with diacritics replaced with the corresponding simple letter (e.g. *é* with *e*, *č* with *c*), a version in which each letter was replaced with an archiphoneme from the set {*vowel*, *semi-vowel*, *liquid*, *nasal*, *fricative*, *other obstruent*}, and a version reduced to only two symbols: consonants and vowels.

3.2 *n*-gram Model Development

Two subsets of the novels corpus were used for development. The Romance data set, which was intended to improve detection of close genetic relationships, included Portuguese, Spanish, French, Italian, and Romanian texts. The Indo-European data set, intended to include a mixture of close and distant relationships, included Portuguese, Spanish, Romanian, English, Czech, and Polish. For each of the four kinds of filtering describe in §3.1, the *n*-gram order was increased until the output tree stopped changing. The results are reported in Table 1, each cell of which contains the order at which the *n*-gram model converged on a final tree and the tree itself, represented by ISO-639-1 language codes in nested parentheses.

For the Romance data set, the best results, in which the only error is the swapping of French and Italian, were achieved using orthography and plain letters. For the Indo-European set, the best results were for plain letters and consonants and vowels (both of which produced trees with no errors). The more closely-related languages, therefore, seem to share particular details of orthography, while more distant relationships are better modeled in a simplified character vocabulary. From Table 1, it can be seen that, across all filtered versions of both dev sets, the best results can be achieved using a model on plain letters with an *n*-gram order of five.

3.3 Factored Language Models

After the system was working with simple *n*-gram models, an attempt was made to improve it using Factored Language Models (FLMs) as described by Bilmes and Kirchhoff (2003). In an FLM, each word (or letter, in this case) has several different features, and the model includes a backoff path that defines how probabilities are calculated when one or more of the features are not available in the test data. In this project, there were four features on each letter corresponding to the filtered versions described in §3.2: the original orthographic letter, the letter without diacritics, the archiphoneme, and the consonant or vowel. As the order and number of features in FLM increases, the number of possible backoff paths quickly becomes large and training becomes computationally expensive.

3.4 FLM Development

To efficiently train FLMs, I used GA-FLM (Duh 2004, Duh & Kirchhoff 2004). GA-FLM's algorithm works by dividing the data into training and development data, then repeatedly mutating the backoff graph and smoothing parameters of the model, searching for improved results on the dev data. Unfortunately, this algorithm is only imperfectly adaptable to the system being developed. The power of an FLM for this application is that there might be, somewhere in the large-dimensional backoff graph space, a graph simultaneously well-suited to modeling both nearby genetic relationships and more distant ones. GA-FLM, however, optimizes a graph for a single development text. What is needed instead is an algorithm that, given a matrix of language pairs, optimizes a backoff graph to produce low perplexities on closely-related pairs and high perplexities on distantly-related pairs.

In an attempt to circumvent this restriction, I tried training FLMs using two kinds of development data. In the first, an FLM was trained for each single-language text using a held-out portion (10%) of the text as the target for training, which should produce an FLM that maximizes the probability of that single language. In the other, the development set contained a small fraction (0.1%) *every* language in the data set, in the hope that the resulting FLM would accommodate more distantly related languages. However, in neither case were the results of the system better than the *n*-gram

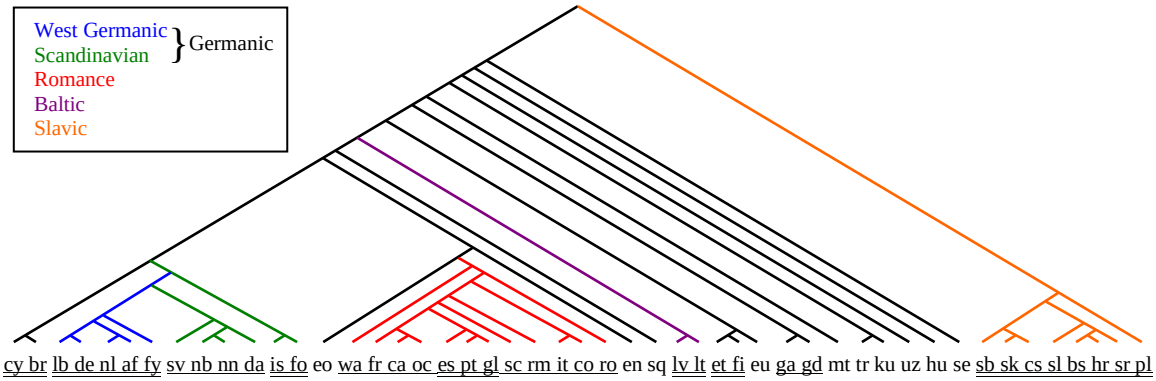

Figure 2: Final results on the "UDHR" corpus

model results; furthermore, the training was much slower, such that FLM orders of larger than three (with all four factors) took unmanageably long to train.

4 Results

The best results during development, then, were with an order-5 *n*-gram model on plain letters. The system was run with those settings across the 49-language UDHR data set. The resulting tree is shown in Figure 2[1]. Groups of languages in the correct relationships are double-underlined; related languages correctly grouped together, but with the wrong relationships, are single-underlined.

Compared to the known genetic relationships (Gordon 2005), much of this tree is remarkably correct. The Slavic languages are all together in the right relationships. The relationships for West Germanic are also correct. Scandinavian is nearly so, except that Nynorsk (nn) should be with the pair Icelandic (is) and Faroese (fo) (which should be within the family). The Romance languages are together, but misclassified—Romanian (ro) is correctly peripheral, but Walloon (wa) is closely related to French (fr). Similarly, Catalan (ca) and Occitan (oc) should close to Spanish (es).

Other parts of the tree are less correct. The Brythonic Celtic languages Welsh (cy) and Breton (br) are together, as are the Goidelic Celtic Irish (ga) and Scots Gaelic (gd), but the relationship between the two families was not detected. English (en) is oddly categorized as an outlying Romance language (as is Esperanto (eo)), perhaps because the register of the UDHR includes many Latinate words, or because of language contact between English and French. Another possible effect of contact is the close grouping of Turkish (tr) and the Indo-European language Kurdish (ku), which are unrelated but geographically adjacent. Some of the larger-scale relationships are wrong as well; the Slavic languages, for example, are shown as less similar to the Germanic and Romance languages than several non-Indo-European languages such as Maltese (mt), Finnish (fi), and Estonian (et).

[1] Sorbian, which has no ISO-639-1 code, is represented by sb.

5 Conclusions and Future Work

The method described here for automatically detecting genetic relationships between languages using simple *n*-gram models is quite successful with close relationships, but less so with more distant relationships. It seems possible that properly-trained FLMs could address this flaw, but currently available training algorithms first need to be enhanced to optimize a backoff graph for a target matrix of document pairs.

References

Gordon, Raymond G., Jr. (ed.). 2005. Ethnologue: Languages of the World, Fifteenth edition. Dallas, TX: SIL International.

Bilmes, Jeff and Katrin Kirchhoff. 2003. "Factored Language Models and Generalized Parallel Backoff". *Proceedings of HLT/NAACL*, pp. 4-6.

Duh, Kevin. 2004. "GA-FLM User's Manual".

Duh, Kevin and Katrin Kirchhoff. 2004. "Automatic Learning of Language Model Structure". UWEETR-2004-0014.

Stolcke, Andreas. 2002. "SRILM – An Extensible Language Modeling Toolkit". *Proceedings of ICSLP*, pp. 901-904.

Office of the High Commissioner for Human Rights. 2006. http://www.unhchr.ch/udhr/index.htm