Combinatorial Optimization and Genetic Algorithms in Text Selection for Corpus-Based Historical Linguistics Research

John D. Sundquist, Andrew Rothwell
Purdue University
{jsundqui, arothwel@purdue.edu}

Abstract

This paper describes a case study in which a genetic algorithm is used for text selection from a corpus that contains texts of widely varying word counts from different genres that are spread out over various time periods. The application of the genetic algorithm was effective in selecting an optimal set of texts and in decreasing computation time. The implications of the findings are discussed in light of diachronic, corpus-based research studies.

1 Introduction

A common challenge in corpus-based diachronic linguistic research is obtaining a balanced and representative set of texts for quantificational analysis when the only historical corpora that are available from earlier periods of a language contain texts with widely varying lengths from different genres that are spread out over long periods of time. Diachronic studies that take into account type and token frequencies and the interaction of both for certain linguistic phenomena, for instance, face challenges in dealing with sample size differences and a lack of balance across genres and word count from various time intervals (Biber, 1993). Moreover, normalization of frequencies to account for sample size differences often fails to avoid skewed results in the analysis of longitudinal data (Tweedie and Baayen, 2011). These issues can be described as the corpus linguistic version of a combinatorial optimization problem in finding an ideal set of texts with a similar total number of words per time period that is balanced across genre and word count per text. We propose a way to alleviate this problem by means of a genetic fitness algorithm. Using a 406-million-word historical corpus of texts as part of a case study, we demonstrate how the algorithm effectively selects a set of texts with an optimal distribution that is balanced across genres and word count for various time intervals.

2 Background

In corpus-based historical linguistics research, many analyses of frequency of certain linguistic phenomena require an even balance of texts across genres (Davies, 2012). Moreover, any diachronic investigation of type and token frequencies faces additional difficulties in dealing with significant sample size effects (Koplenig, 2015). Furthermore, clustering of texts from some years with gaps in others causes additional complications for obtaining a balanced and representative sample of language in any statistical analysis of longitudinal data (Moisl, 2015).

There are many available historical corpora for which these issues pose methodological challenges. For example, the Corpus of Historical American English (COHA) from Davies (2010) includes texts that represent four different genres, namely, magazines, fiction, non-fiction, and newspapers. COHA consists 115,000 individual texts that range in word count from 300 words to over 300,000 words, spread out over 20 different decades between 1810 and 2010. Moreover, word counts per decade in COHA range from 1.1 million words from texts in the 1810s to 29 million words from those in the 2000s. In addition, while COHA includes texts from each year between 1810 and 2009, texts are not evenly distributed in each year: there are many short texts from some years but single, long texts from other years. In sum, texts vary by word count,

genre, and decade; decades and years vary greatly by total word count and the number of available texts. This common corpus design issue poses challenges if one wishes to maintain balance and representativeness at the same time as taking into consideration equal sample sizes per decade with an even distribution of texts of varying lengths.

2.1 Combinatorial Optimization

This issue is reminiscent of combinatorial optimization problems in which an optimal solution must be identified from a finite set of possible solutions (Papadimitriou and Steiglitz, 1998). Such issues arise, for instance, in supply chain optimization (Eskandarpour, Majid, et al. 2015) or logistics (Sbihi and Eglese, 2007), in which attempts are made to allocate a finite set of resources in the most efficient combination.

2.2 Genetic Algorithms

One solution to optimization problems in various domains is the application of genetic algorithms (GAs). For instance, GAs have been used for optimization in business portfolios (Chang et al, 2009; Oh, 2005), electric power dispatch in engineering (Abido, 2006), and processor scheduling tasks in computer science (Hou et al, 1994). GAs attempt to find optimal solutions in a search space using properties of natural selection, including "gene mutation" and "breeding" (Goldberg, 2006). To simulate this natural selection, a fitness function is created (Binitha and Sathya, 2012). The fitness function can then be used to rank individuals during breeding and allow higher ranked individuals to reproduce more than others. The next generation then goes through a similar process. This process attempts to identify incremental improvements in each generation.

2.3 The Fitness Function in the Linguistic Domain

In the linguistic domain, the selection optimization problem can be mathematically represented follows:

$$\min\big(f(x)\big)\ where\ f(x) = \sigma_{(\text{sources per year})} + \sigma_{(\text{sources per genre})} + \sigma_{(\text{word count per year})} + \sigma_{(\text{word count per genre})} + |\text{total word count - optimal word count}|$$

As standard deviation shows the spread of data from the mean, it is a useful measure to determine if a year or genre is overrepresented in the data. The reason both the standard deviation per number of sources and per total word count are included in the fitness function is to take into account the effects in expression of large word count sources compared to small word count sources. A theoretical optimal selection would produce a score of 0 for a selection with exactly 6 million words that also has an equal number of sources and word counts per year and genre. For example, for a selection of the 1930s, we calculated f (x) as follows:

number of sources per year {1930: 31, 1931: 265,1932: 149,1933: 102,1934: 44,1935: 52,1936: 158,1937: 232,1938: 491,1939: 425}

number of sources per genre {"NF": 65, "FIC" : 148,"MAG" : 1736}

number of words per year {1930: 298006, 1931: 1420474,1932: 633197, 1933: 662985,1934: 189769,1935: 254002,1936: 638916,1937: 763989,1938: 630819,1939: 511100,}

number of words per genre {"NF": 2001868, "FIC": 2001372,"MAG" : 2000017}

$\sigma_{(\text{sources per year})} = 159.5357$

$\sigma_{(\text{sources per genre})} = 941.7071$

$\sigma_{(\text{word count per year})} = 349638.6488$

$\sigma_{(\text{word count per genre})} = 958.14421322332$

|total word count - optimal word count| = 3257

Fitness score is: 354077.6885

3 Methodology

The goal of this project was to come up with a possible solution to the optimization problem and to test the effectiveness of this solution. As a case study, we used COHA because of its convenient division into decades, its broad range of word counts of individual texts, and its variety of genres. The project involved four steps: establishing text selection criteria, manually selecting texts that meet these criteria as best as possible, implementing a genetic algorithm to select texts, and comparing the results of the algorithm with those of manual selection.

3.1 Selection Criteria

Criteria for text selection were based on five categories: word count per decade, word count

per genre, the number of texts per genre, the number of texts per year, and word count per year. We determined that the optimal total word count per decade in COHA was 6 million words, based on the largest set of available texts in the corpus from the earliest decade in this study (1820s). The total word count of this decade in COHA is 6.9 million words, and 6 million was an even number divisible by three (i.e., the number of genres represented in this decade in COHA). Secondly, we determined that each decade should contain texts from magazines, fiction, and non-fiction, and that each genre's word count should be 2 million. Because the decades between 1820 and 1860 do not contain the fourth genre in later decades of COHA (newspapers), we excluded this type of genre from the analysis in order to maintain balance throughout the entire corpus. Thirdly, we aimed to have each year in the corpus represented; no single year between 1820 and 2009 should lack texts from one of the three genres. Lastly, we aimed to find a relatively even distribution of word count per year. This certified that no single year would have a significantly higher or lower word count than others.

3.2 Manual Text Selection

The second step in this process involved the manual selection of texts. Using the chronological source list in COHA that contains the year, word count, and genre of each text, we attempted to fit the criteria listed above as best as possible by handpicking texts. Using the criterion of two million words per genre as the starting point, we randomly selected texts from that genre from each year until the total reached as close to two million words as possible.

3.3 Application of Genetic Algorithm

Next, we set up the genetic algorithm for text selection. For each decade, we calculated a fitness score based on the number of sources in COHA per year and per genre, and the number of words per year and per genre, as described in the previous section.

The algorithm involves *n* binary vector individuals of length *k* which represent possible selections of texts to include in a balanced sample size. These individuals undergo three phases; combinational breeding, mutation, fitness evaluation. N randomized binary vector individuals of length k are created at startup. The individuals are then sorted based on fitness before the first breeding occurs. Two individuals A and B are selected to breed. A is selected from the top 33% of all individuals and B is selected at random from the set of all individuals. A and B are combined at a randomly chosen splice point to create two new individuals C and D. New individuals C contain all genes from A before the splice point and the remaining genes from B. D receives all genes from B before the splice point and the remaining from A. The top performer from the generation is also copied into the new generation to preserve positive mutation trends. Next, mutation takes place on newly created individuals. The total number of words in a genre different than the optimal number of words in a genre is calculated. In our case, this was two million words per genre. Then, iterating over the vector, each gene is turned on or off with a probability of 1/n, if flipping the value of the gene will cause the vector to become closer to the optimal number of words. After mutation is completed, the generation is evaluated for fitness based on the previously discussed fitness function. The individuals are then sorted by fitness score in descending order. Breeding then continues until a certain number of generations have been created and evaluated.[1]

4 Results

In Table 1, comparative results are presented on the fitness scores of 10 runs of the GA using 64 individuals and 1,000 generations along with the fitness score for the manual selection. The lower the fitness score, the more optimal the selection of texts. The minimum score is presented from the genetic algorithm as well as the average (mean) fitness score over ten runs.

The GA was able to produce at least one set of texts for each decade that was more optimal than the handpicked set. In some cases, the

[1] Code for the algorithm is available here: https://github.com/corpus-based-research-lab

minimum GA-produced score was as much as 82% or 83% better, as in the 1930s or 1940s. While the GA selected a more optimal set of texts in all decades at least once, the average score was not always better than the score for manual selection. In the 1950s, 1960s, 1970s, and 2000s, for example, the manual selection was actually better than the mean fitness score on ten runs of the GA.

Decade	Manual selection fitness score	Best GA-produced fitness score over 10 runs	Mean GA-produced fitness score over 10 runs
1820s	1211446	1112004	1148684
1830s	193462	85548.02	165701
1840s	233466	106483.5	164453
1850s	331000	72554.43	145310
1860s	341023	159968.9	199040
1870s	214410	96212.33	152432
1880s	210788	74056.47	122584
1890s	288324	114985.2	155776
1900s	299384	105349.4	155031
1910s	220693	102385.3	128593
1920s	352332	251358.1	326632
1930s	354077	61688.79	313775
1940s	688920	126513.6	306132
1950s	288735	150477.6	374268
1960s	207570	71848.53	258747
1970s	212580	120438.4	378351
1980s	322409	84518.17	305222
1990s	351593	143293.5	207697
2000s	200915	130686	253250

Table 1: Fitness scores by decade

For the most part, however, the average of 10 runs of the GA was better than that of the manual selection. In 15 of the 19 decades, the mean fitness score produced by the GA was better than that of the manual selection. In some decades, such as the 1850s and 1940s, the average fitness score was as much as 56% better.

5 Discussion

The results indicate that a GA was useful in dealing with combinatorial optimization problems encountered in text selection. The GA outperformed manual text selection in at least one run in all decades. In most instances, the best fitness score produced by the GA was mostly a noticeable improvement on that of the manual selection.

In some cases, the overall mean score of all ten runs per decade was less optimal than that of the manual selection. This result occurred in more recent decades in the 20th century in COHA's set of texts. In these decades, the GA-produced mean scores were less optimal than those of the handpicked set. The reason for this lack of consistency over the ten runs is most likely the significantly larger number of texts in COHA from more recent decades. The 2000s, for instance, contain 13,906 different texts, while the 1830s contain only 712 unique texts. The larger search space requires more generations or individuals within each generation of the GA in order to obtain a better result.

In addition to the improvement of text selection with greater balance, an additional advantage gained by using a GA for combinatorial optimization is computation time. Manual selection of texts in the case of this corpus, for instance, took the authors several hours of research time. Applying the GA to help with this process greatly shortened the process. The time devoted to each decade varied by the number of texts to analyze, but the process was much faster than during manual selection. For a decade in COHA with fewer texts (e.g., 1820s), the GA produced results on average in 13.25 seconds; a larger set of texts (e.g., 2000s) took on average 216.35 seconds. These results were computed on a stock i5 2.6GHz 8Gb ram Microsoft Surface. In comparison to manual selection, the GA-produced optimization method saves significant time.

6 Conclusion

The implications of these findings are that a GA can be used to solve optimization issues in corpus studies that require a balanced, longitudinally-organized sets of texts with similar word counts from evenly distributed time periods. This case study was conducted on one particular corpus, although the methods are applicable across a variety of corpora with a range of designs. One of the limitations of the study was that the GA was inconsistent in producing optimal results in some decades with a significantly greater number of texts than in others, although the GA could be modified in the future by adding more generations and individuals. For the most part, however, findings indicate that this method is particularly useful in situations with corpora that include highly uneven-sized texts from a variety of genres and time periods.

References

Abido, Mohammad Ali. 2006. Multiobjective evolutionary algorithms for electric power dispatch problem. *IEEE Trans. on Evolutionary Computations*, 10(3):315-329.

Biber, Douglas. 1993. Representativeness in corpus design. *Literary and Linguistic Computing,* 8(4):243-57.

Binitha, S., and Sathya, S. S. 2012. A survey of bio inspired optimization algorithms. *International Journal of Soft Computing and Engineering*, *2*(2):137-151.

Chang, Tun-Jen, Sang-Chin Yang, and Kuang-Jung Chang. 2009. Portfolio optimization problems in different risk measures using genetic algorithm. *Expert Systems with Applications* 36(7):10529-10537.

Davies, Mark. 2010. *The Corpus of Historical American English*. http://corpus.byu.edu/coha

Davies, Mark. 2012. Expanding horizons in historical linguistics with the 400-million word Corpus of Historical American English. *Corpora*, *7*(2):121-157.

Goldberg, David E. 2006. *Genetic algorithms*. Pearson Education India.

Hou, Edwin S., Nirwan Ansari, and Hong Ren. 1994. A genetic algorithm for multiprocessor scheduling. *IEEE Transactions on parallel and distributed systems* 5(2):113-120.

Koplenig, Alexander. 2015. Using the parameters of the Zipf-Mandelbrot Law to measure diachronic, lexical, syntactical, and stylistic changes–a large-scale corpus analysis. *Corpus Linguistics and Linguistic Theory,* 14(1):1-34.

Moisl, Hermann. 2015. *Cluster analysis for corpus linguistics*. Walter de Gruyter, Berlin.

Oh, K. J., Kim, T. Y., & Min, S. (2005). Using genetic algorithm to support portfolio optimization for index fund management. *Expert Systems with Applications*, 28(2):371-379.

Papadimitriou, Christos H., and Kenneth Steiglitz. 2015. *Combinatorial optimization: algorithms and complexity*. Dover Publications, Mineola, New York.

Eskandarpour, Majid, Pierre Dejax, Joe Miemczyk, and Olivier Péton. 2015. Sustainable supply chain network design: an optimization-oriented review. *Omega*, 54:11-32.

Sbihi, Abdelkader, and Richard W. Eglese. 2007. Combinatorial optimization and green logistics. *4OR*, 5(2):99-116.

Tweedie, Fiona J., and R. Harald Baayen. How variable may a constant be? Measures of lexical richness in perspective. *Computers and the Humanities*, 32(5):323-352.