Determining Relatedness of Textual Data

An Implementation Utilizing Dimension Reduction

Justin Lu

Advisor: Rafail Ostrovsky

Abstract

Determining how related pieces of textual data are to each other is a problem that has various important applications in data mining and searching. The goal is to be able to, given three arbitrary pieces of textual data, determine that the first pieces is more closely related to the second than the third (or vice versa.) Additionally we would prefer a space efficient representation of our data, otherwise we might as well use the original data directly.

In this project a small size representation for a particular datum was created, a signature of sorts. that allows it to be compared against other data to compare for relatedness. The main idea is to use dimension reduction as a tool to achieve this goal, which entails utilizing very large vectors as a system of representation. The project therefore naturally fell into two parts: finding a good representation and experimenting on it with dimension reduction.

A representation was made utilizing word commonality with data from the online encyclopedia, Wikipedia, which served as a source of real world information. Then, dimension reduction was utilized to shrink the data to a smaller size. The effectiveness of this method was then tested against merely using a smaller basis to achieve the same ends. The data shows that the dimension reduction method preserves information much better than random basis reduction alone.

Representation

Finding a good representation is important, if the system of representation produces poor results, it certainly will not become any better after undergoing dimension reduction. The representation

in this project makes use of a very large vector representation and word commonality. The the nth element of each vector corresponds to the number of words in common with the nth Wikipedia article.

The idea is that two pieces of data both containing words from similar articles in Wikipedia are more likely to be related to each other. To compare two representations, one simply finds the distance between the two vectors: the larger this distance is the less related they are.

Some adjustments are made to this to make it more feasible (see implementation). After implementation this was then tested and found anecdotally to produce good results, articles about similar things would be ranked closer together. This representation forms our canonical version, a sort of "best that can be gotten" from using word commonality and a large vector representation. Subsequent representations, specifically smaller ones, will be tested against this version to determine effectiveness.

Dimension Reduction

Dimension reduction was done based on the paper, *Fast Dimension Reduction Using Rademacher Series on Dual BCH Codes* by Ailon and Liberty. The method is based off the Johnson-Lindenstrauss property which gives a guarantee of preserving distances between points after embedding from a high dimension to a low dimension. This method involves using a transformation matrix (k by d, which embeds from the larger d to the smaller k) which is obtained by multiplying a code matrix, in which every row is also some row of the the Walsh-Hadamard matrix, by a randomized diagonal matrix. Some constants are also multiplied in for normalization purposes.

Implementation

To produce representations all the articles from Wikipedia were downloaded and analyzed. A hash table data structure was created wherein the keys are a particular word used in the articles and the values are the indices of all articles that contain that word. In doing so a representation for a piece of data can be quickly calculated by simply looking up each word and adding one to the appropriate index. The hash tables and any representations generated were stored as

marshaled (serialized) Ocaml data structures (hash tables and arrays respectively). Some adjustments are made to this: only using articles that are beyond a certain length, ignoring common words, having a relevancy threshold (only a common word number above a certain value is recorded), and having a threshold on the amount of words considered in each datum to be compared.

The dimension reduction was done by generating the matrices and applying them to the representations. First the BCH was generated using the formula given in the paper, this was done once and set aside for future use. The randomized diagonal matrix was then applied to the BCH matrix and the result is the desired transformation matrix.

Testing Methodology

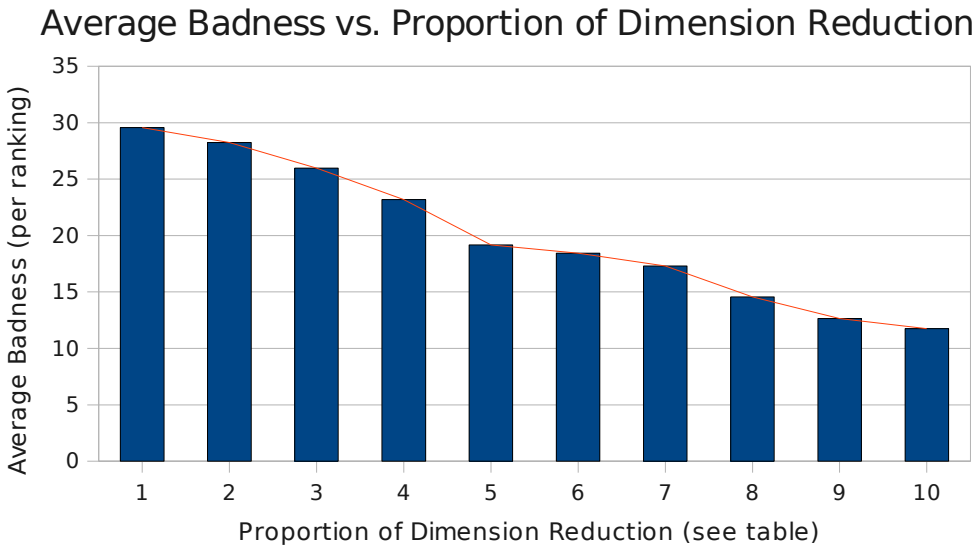
To test the efficacy of any particular representation one hundred random pieces of data were selected. The full vector representation of these data were made. Then five pieces were chosen at random and had their representations compared against all the other one hundred pieces. The articles were then sorted by how close they were to the piece in question, forming a ranking. This process is repeated with the representation to be tested. The rankings are then compared by looking at where a particular index is in the new representation and where it is in the old one. The difference between each of these is then added up and divided by the amount of indices total. This yields the average badness, that is, on average how far away each index is from where it should be for a particular representation.

The goal is to reduce from d, the original dimension, to k, the target dimension. The JLT dimension reduction method was then compared a against a simpler method of dimension reduction for this task. The simpler method is simply selecting a random subset of the basis and using that as the smaller representation. Furthermore tests were done utilizing a combination of the two, wherein the representation was first reduced by random basis subset selection to an intermediary representation with a basis, of size i, which is somewhere between k and d. Then this intermediary representation was further reduced via JLT to k, the target size. Different values of i were utilized ranging from k (full random basis reduction, no JLT dimension reduction) to d (no random basis reduction, full JLT dimension reduction) to show the superiority of the latter.

Test Results

k = 128 (target dimension)

d = 65536 (original dimension)



Number	Intermediary Basis (i)	Average Badness
1	128 (Full Random Basis Reduction)	29.56
2	256	28.25
3	512	25.98
4	1024	23.19
5	2048	19.16
6	4096	18.43
7	8192	17.31
8	16384	14.55
9	32768	12.65
10	65536 (Full JLT Dimension Reduction)	11.76

References

N. Ailon and B. Chazelle. Approximate nearest neighbors and the fast Johnson-Lindenstrauss transform. *In Proceedings of the 38st Annual Symposium on the Theory of Computing* (STOC), pages 557–563, Seattle, WA, 2006.

N. Ailon and E. Liberty. Fast dimension reduction using Rademacher series on dual BCH codes. *Proceedings of the Nineteenth Annual ACM-SIAM Symposium on Discrete Algorithms,* SODA 2008, San Francisco, California, USA, January 20-22, 2008. SIAM 2008