Dimension Reduction: A Powerful Principle for Automatically Finding Concepts in Unstructured Data

Holger Bast

Max-Planck-Institut für Informatik
66123 Saarbrücken, Germany
bast@mpi-sb.mpg.de

Abstract

Dimension reduction techniques have been a successful avenue for automatically extracting the "concepts" underlying unstructured data, a task that naturally arises in fields as diverse as information retrieval, image processing, social science, etc. It is surprising how much can be achieved for this task using only the raw data itself, without resorting to any additional knowledge or intelligence. We will survey the most important schemes contributed from the various communities to date, by commenting on the following aspects: optimization techniques, the role of normalizations, setting the parameters, computing time, quality of results, and the integration of external knowledge.

1 Introduction

Information is made up of units, but humans are usually interested more in the bigger picture. When the author was recently searching the web for "nonnegative matrix factorization", he was actually interested in papers on that *topic*, not in literal occurrences of that phrase. A digital photograph is made up of millions of pixels, but a human's interest usually lies more in higher-order features like shapes etc. An opinion survey's immediate result are counts, but the desire is usually to learn about trends.

In this short paper, we will survey methods for extracting such concepts *automatically*, from representations of data as (real-valued) matrices. A suitable such matrix for a collection of text documents would have each document correspond to a column and each word correspond to a row, with a particular entry specifying, for example, how often that word occurs in that document. In image processing, a good representation for some of the methods that follow would be to have a column for each pixel with rows representing features like coordinates, color, intensity, etc. For an arbitrary collection of objects, if equipped with a measure for pairwise similarity, a generic representation would be the (square) matrix of all pairwise similarities.

The basic idea, which we refer to as *dimension reduction*, is to approximate a given such $m \times n$ matrix M by a matrix of a given rank k, typically much smaller than both m and n. In other words, the goal is to find an $m \times k$ matrix C and an $k \times n$ matrix M' such that the product $C \cdot M'$, which is a matrix of rank (at most) k, is a good approximation to the original matrix M. Calling the columns of C *concepts*, each column of M is then approximated by a linear combination of these concepts, with the k coefficients given by the corresponding column of M'. We remark that some methods only compute M' explicitely, but neither C nor $C \cdot M'$. [1]

The dimension reduction idea has been applied in a large variety of contexts, and it is truly amazing how much can be achieved for the apparently intelligence-demanding task of extracting underlying concepts that make sense to a human, by approaches that are completely ignorant of any world knowledge, relying only on the raw data itself (there are, of course, limits; we come to this in Section 2.6).

The sheer mass of papers published on dimension reduction schemes is overwhelming. Moreover, contributions come from quite different communitites (information retrieval, image processing, machine learning, artifical intelligence, theoretical computer science, mathematics, even social sciences), each with their own peculiar terminology and way of looking at things. What adds to this confusing complexity is that some papers focus more on so-called *clustering*, where each object is assigned to exactly one concept, while others also consider what could be called *soft clustering*, where documents can be fractionally assigned to several concepts. These problems are actually more closely related than is generally realized; the way we described them here should give a first hint.

[1] It is also worth noting that while all the methods considered for this paper compute a decomposition based on standard matrix multiplication, that is, involving $+$ and $\cdot$, other operations, for example, involving $\vee$ and $\wedge$, would also make sense for many applications. The mathematics, however, usually becomes harder to deal with then.

2 Brief Survey

In the following sections we will survey this large body of research by viewing it from various angles: optimization techniques, the role of normalizations, setting the parameters, computing time, quality of results, and the integration of external knowledge. Citations will be in an exemplary (not comprehensive) fashion, with an emphasis on recent results and our own ongoing research on the topic.

2.1 Optimization techniques

The majority of dimension reduction schemes, and indeed all considered for this survey, use one of the following two basic optimization techniques.

In the one technique, widely known as *spectral analysis*, for a given matrix and some k, the eigenvectors pertaining to the k largest eigenvalues are computed; this corresponds to finding that k-dimensional subspace from which the objects in the original space have minimal total euclidean distance. The other fundamental technique is to use the so-called *expectation maximization (EM)* principle [9] or a variant of it; this finds a probability distribution with few degrees of freedom (namely, the concepts) which generates the given data with maximum likelihood.

The spectral methods are typically easier to implement and faster to compute, but suffer from the fact that minimum total euclidean distance is rarely a realistic objective. More realistic (probabilistic) data models usually lead to EM-based methods. These, however, are generally slower, cf. Section 2.4, and since EM is an iterative local search algorithm, there is always the risk of getting stuck in a local minimum.

2.2 Normalizations

Different methods use different normalizations — of the rows of the matrix or of its columns, by L_1, L_2, or L_∞ norm, by centering (subtracting the mean), or any combination of these. This apparent detail turns out to make a big difference in practice. We give only two striking examples here, pertaining to the basic *latent semantic indexing (LSI)* technique [8], which was the first successful application of the dimension reduction idea in information retrieval.

In [18] we compared two of LSI's widely used variants and showed that for any given number of terms and documents, a corpus and query can be constructed such that the ranking produced by the one variant is *the complete reverse* of the ranking produced by the other variant. In practice, the effect is usually not as extreme, of course, but it clearly shows.

In [13] it was found that on a large document collection, standard LSI performs worse than basic text-matching, which the authors attribute to the unproportionally large weight LSI gives to frequent terms. To remedy this, they suggested a particular additional normalization(!). But when comparing this new variant to standard LSI and to basic text-matching, they found that on each of the three different collections they considered, a different method came out as the (clear) winner.

Another issue in this context is that there are numerous instances in the literature where two methods, which from the given descriptions look pretty different, can actually be shown to be identical up to normalization. We come back to this important point in the concluding Section 3.

2.3 Setting Parameters

Every dimension reduction scheme comes with one or more parameters (one is always the number of concepts), the appropriate setting of which, just like for the normalizations, is essential for obtaining high-quality results. For most methods there is no guidance for this choice except empirical evidence. In some methods averaging over different runs with different parameters makes sense [12]. A few methods have implicit a quantification of how well a parameter worked, so they can just try out values systematically [14].

Unfortunately (or fortunately for those looking for research problems), there is hardly any theory for computing a provably good parameter setting for a given problem instance. Efron [11] surveys a number of statistically well-founded procedures for computing a good value k for the number of concepts, without any performance guarantee however. A first result of that kind is given in [5], where for a special type of query a formula is given which provably computes the optimal dimension for such queries, i.e., the dimension where the precision peaks.

2.4 Computing time

The spectral methods are usually based on some variant of the Lanczos algorithm [6]. For computing k eigenvectors (which give the k concepts), the bulk of the running time of Lanczos and related algorithms is spent on computing $O(k)$ matrix-vector products [6]. This gives a total running time of $O(nz \cdot k)$, where nz is the number of nonzero entries in the given (typically sparse) matrix.

One iteration of the EM-based methods requires computation proportional to the amount of the given data and to the number of concepts. Usually, sparseness of the given data matrix can be exploited also here, in which case we have a running time of $O(nz \cdot k)$ per iteration. A few dozen iterations are usually sufficient, but even then there is a tangible performance gap to the spectral methods.

Neither of these bounds imposes a principal limit on the use of dimension reduction schemes in practice, not even for huge amounts of data. The EM-based *probabilistic latent semantic indexing (PLSI)* scheme of [12], for example, powers a search engine for the fairly large MEDLINE database; you can try it at `http://www.nlm.nih.gov/medlineplus/searchtips.html`.

On the other hand, what is peculiar about *each* dimension reduction scheme, is that the theoretical optimum which all the heavy computation is aimed at, is definitely not the desired optimum. We already mentioned that eucledian distance (in the vector space spanned by the columns or rows of the given matrix) is rarely a meaningful measure in concept-extraction tasks. One is tempted to say that the spectral methods work so well in such a wide variety of contexts *not because* they minimize certain eucledian distances but rather *despite* of this fact. And the EM-based methods are indeed principally not run until convergence to avoid "overoptimization" effects. There may hence well be other objectives, which are at least as realistic but easier to optimize.

Steps in that direction have been taken in [15] and [4], where an attempt is made to explain the entries of the reduced (by spectral methods) term-term affinity matrix directly by co-occurrence information like distances and number of paths between terms in the co-occurrence graph (where each term is a vertex, and there is an edge between two vertices, if the two terms co-occur in at least one document).

If the goal is merely to reduce computational cost, an idea is to *sample* from the full data, optimize on the sample, and then extrapolate. For spectral analysis, there has been theoretical work in that direction by [10] and [1].

2.5 Quality of results

An assessment of dimension reduction schemes is difficult, not only theoretically but even empirically, for the following reason. There is (for good reason) no mathematical definition of what a good concept is, and ultimately a human must assess the quality of a given solution. However, the fully automatic dimension reduction approach is especially interesting for huge amounts of data, which no human can ever sift manually; for example, how to obtain the set of *all* relevant web pages — let's assume there were an objective relevance criterion here — for a query on "non-negative matrix factorization" (at the time of this writing, Google was indexing 4,285,199,774 pages)? A must-read in that context is [7].

The lack of a formal problem definition is of course a major obstacle also for solid theoretical work. Theory still has its place here, however, since simple models can very well give insights on whether a method does basic things right or not. A nice result in that vein was given in [17], where it is proven that under a number of well-defined and reasonable assumptions, a simple spectral clustering algorithm finds the "true" clusters. For soft clustering, a comparable result has been given in [2].

In [3] we implemented a tool which allows for an intuitive, interactive evaluation of the performance of a dimension reduction scheme on an arbitrary collection of text documents. This tool permits insight into *how* a method brought about a certain result, and why it succeeded in certain aspects and failed in others; issues that remain completely obscured behind the typical performance figures: either carefully selected examples or averages over a large number of very heterogeneous problem instances. Currently, only the PLSI scheme of [12] is implemented, but we are working on an interface that enables a "plug-in" of arbitrary schemes. The tool can be downloaded from the web [3], and it will be presented at the workshop.

2.6 External Knowledge

Dimension reduction schemes come to their limit, when, abstractly speaking, the hints for the underlying concepts cannot be distinguished from the more random constellations in the data which carry no specific information. An example would be, in a document collection, a single rare word being the only but highly specific hint for a topic; phenomenons of that kind are actually very frequent. An important aspect of any dimension reduction scheme is therefore its ability to incorporate some form of external knowledge.

Most methods actually permit an ad hoc extension to this scenario. We here mention just one recent, more principled step in that direction. Kamvar et al. [14] showed how information of the kind "these two documents certainly (do not) belong to the same topic" can be plugged into their spectral dimension reduction scheme. The bottom line of their investigations was that the performance increases with the amount of such knowledge input *as well as* with the amount of the raw mere data.

3 Directions for Future Research

First, given the success of the dimension reduction idea and the mass of papers published on it, it would be highly desirable to have a general framework or taxonomy, which in particular would highlight the commonalities of the various schemes and where they differ. We already mentioned the various instances of apparently different schemes, which a closer look reveals to be identical up to normalization. Another issue are researchers from different communitites doing very closely related work under different names, and in complete ignorance of each other (e.g., the works of [16] and [12], and their respective successors).

Second, there is an obvious need for a better theoretical underpinning, in particular for an understanding of the effects of apparently minor details like normalizations and the setting of parameters, which to a high degree influence the actual performance. It seems a bit weird to invest large amounts of effort in complicated algorithms and/or analyses, when some completely heuristic parameter setting makes all the difference in practise.

Third, one of the most important and also most interesting directions of research, in the opinion of the author, is the seamless integration of external knowledge. Doing away with such knowledge altogether puts unsurmountable suboptimal performance limits for most applications, but it seems that already very little such input is enough to boost the quality of the results. Much more insight is required into which part of the concept-extraction task a machine can in principle do, which part requires genuine knowledge, and how to combine the one with the other.

References

[1] D. Achlioptas and F. McSherry. Fast computation of low rank matrix approximations. In *Proceedings of the 33rd Annual ACM Symposium on Theory of Computing (STOC'01)*, pages 611–618, 2001.

[2] Y. Azar, A. Fiat, A. Karlin, F. McSherry, and J. Saia. Spectral analysis of data. In *Proceedings of the 33rd ACM Symposium on Theory of Computing (STOC'01)*, 2001.

[3] H. Bast, D. Brunotte, B. Grundmann, D. Fischer, C. Kayali, D. Tsesarskij, and I. Weber. A visualization tool for PLSI, 2004. Information and download at `http://www.mpi-sb.mpg.de/~bast/plsi`.

[4] H. Bast and R. Kenmogne. Explaining the values in the truncated term-term co-occurence matrix in latent semantic indexing, 2004. Master's thesis, ongoing work.

[5] H. Bast and D. Majumdar. The optimal dimension in latent semantic analysis, 2004. PhD thesis, ongoing work.

[6] M. Berry. Large scale singular value computations. *International Journal of Supercomputer Applications*, 6(1):13–49, 1992.

[7] D. Blair and M. Maron. An evaluation of retrieval effectiveness for a full-text document-retrieval system. *Communications of the ACM*, 28(3):289–299, 1985.

[8] S. C. Deerwester, S. T. Dumais, T. K. Landauer, G. W. Furnas, and R. A. Harshman. Indexing by latent semantic analysis. *Journal of the American Society of Information Science*, 41(6):391–407, 1990.

[9] A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society (B)*, 39:1–38, 1977.

[10] P. Drineas, A. Frieze, R. Kannan, S. Vempala, and V. Vinay. Clustering in large graphs and matrices. In *Proceedings of the 10th ACM Symposium on Discrete Algorithms (SODA'99)*, pages 291–299, 1999.

[11] M. Efron. *Eigenvalue-based Estimators for Optimal Dimensionality Reduction in Information Retrieval*. PhD thesis, University of North Carolina, Chapel Hill, 2003.

[12] T. Hofmann. Unsupervised learning by probabilistic latent semantic analysis. *Machine Learning Journal*, 42(1):177–196, 2001.

[13] P. Husbands, H. Simon, and C. H. Q. Ding. On the use of the singular value decomposition for text retrieval. *Computational Information Retrieval*, pages 145–156, 2001.

[14] S. D. Kamvar, D. Klein, and C. D. Manning. Spectral learning. In *18th International Joint Conference on Artificial Intelligence (IJCAI'03)*, August 2003.

[15] A. Kontostathis and W. M. Pottenger. Detecting patterns in the LSI term-term matrix. In *Proceedings of the Workshop on Foundations of Data Mining and Discovery, IEEE International Conference on Data Mining (ICDM'02)*, 2002.

[16] D. D. Lee and H. S. Seung. Learning the parts of objects by non-negative matrix factorization. *Nature*, 401:788–791, 1999.

[17] A. Ng, M. Jordan, and Y. Weiss. On spectral clustering: Analysis and an algorithm. In *Proceedings of the 15th Conference on Advances in Neural Information Processing Systems (NIPS'01)*, 2002.

[18] J. Parreira. Information retrieval by dimension reduction — a comparative study. Master's thesis, Universität des Saarlandes, Saarbrücken, 2003.