Spectral analysis and text processing over the Computer Science literature: patterns and discoveries

Rosa V. E. Quille, Caetano Traina Jr., Jose F. Rodrigues Jr.
Instituto de Ciências Matemáticas e de Computação - USP
Avenida Trabalhador são-carlense, 400 - Centro
CEP: 13566-590 - São Carlos - SP
{encinas,caetano,junio}@icmc.usp.br

ABSTRACT

We defend the thesis that the use of text analytics can boost the results of analyses based on Singular Value Decomposition (SVD). To demonstrate our supposition, first we model the Digital Bibliography & Library Project (DBLP) as a relational schema; over this schema we use text analytics applied to the terms extracted from the titles of the articles. Then, we apply SVD on the relationships defined between these terms, publication vehicles, and authors; accordingly, we were able to identify the more representative communities and the more active authors relating them to the most meaningful terms and topics found in their respective publications. The results were semantically dense and concise, also leading to performance gains.

Categories and Subject Descriptors

G.1.3 [**Numerical Analysis**]: Numerical Linear Algebra-Singular Value Decomposition; H.2.8 [**Database Applications**]: Data mining

Keywords

DBLP, relational data, data analysis, matrix factorization, singular value decomposition

1. INTRODUCTION

We consider the Digital Bibliography & Library Project (DBLP) as a relational database over which we define an extensive analytical process based on text and linear-algebra analytical techniques. We model DBLP as a relational schema in which its entities (authors, events, vehicles, and terms) correspond to nodes of a graph representation, and its relationships correspond to edges. We use text analytics techniques to process the terms found on the articles' titles; then, we use Singular Value Decomposition (SVD), a powerful algebraic technique – also known as spectral decomposition – for matrix (graph) analysis.

2. RELATED WORK

SAC'14 March 24-38, 2014, Gyeongju, Korea

Singular value decomposition (spectral analysis) has been explored in many scenarios. In the work of Prakash *et al.* [6], the authors obtain the SVD from Laplacian representations, and then plot the vectors of the first matrix of the decomposition against themselves in what they call EE-plots. They found that such plots are very informative in what concerns communities. In another work, Kang *et al.* [1] explain how to calculate SVD's from billion-scale graphs, an operation that has several performance issues. To do so, they use MapReduce and Hadoop technologies for parallel distributed processing.

Leting *et al.* [7] perform spectral analysis over signed graphs. They found that such graphs have specific properties; in special, they found that when a graph is signed, its communities are more clearly observed, even if the connections among them tend to increase. Maruhashi and Faloutsos [5] introduce EigenDiagnostics, an algorithm that calculates and combines several spectral measures to spot patterns in graph-represented data. In another work, Kim *et al.* [2] apply spectral analysis over blog data (or blogosphere). They found a set of outstanding communities derived from the relationships drawn from blogs and posts; also, they were able to characterize and interpret the communities based on the key terms used to compose the posts.

Our work differs from former proposals as we combine text analytics with spectral analysis to gain deeper insight from the computer science literature – we aim at identifying main publication vehicles, authoritative authors, and prevalent communities, among other facts. In our experiments, we demonstrate that by refining the data before the SVD factorization, it is possible to pronouncedly improve the data analysis.

3. SINGULAR VALUE DECOMPOSITION

Technique SVD is a matrix factorization method widely used in applications such as signal processing and statistics [8]. Given an $A_{n\times m}$ matrix, let U be an nxr matrix whose columns are the singular vectors orthogonal to Λ, and V^T be the rxm matrix whose columns are the singular vectors orthogonal to Λ; then A can be defined by equality 1.

$$A_{n\times m} = U_{n\times r}\Lambda_{r\times r}V^T_{r\times m} = \sum_{i=1}^{rank(A)} (\lambda_i u_i \otimes v_i) \qquad (1)$$

where the symbol $\otimes$ is the outer product of two given vectors, and $rank(A)$ is the *rank* of matrix A. The *rank* of a matrix is the number of linearly independent rows (or columns) in it; thus,$rank(A) \leq \min\{m,n\}$.

We use SVD for two reason: (1) **Detect communities and outstanding elements**, according to which, after the singular-factorization, the largest singular values will correspond to subsets (communities) in which the elements both of the rows and of the columns interact more intensely; also, the largest values found in the singular vectors of matrices $U_{n\times r}$ and $V^T_{r\times m}$ will correspond to line and column elements that are highly active, spotting outstanding elements for the sake of analysis; and (2) **Dimensionality Reduction**, DBLP data has a high dimensionality in number of rows and columns; however, its *intrinsic dimensionality* is not that high, as many authors are inactive. It means that many dimensions are redundant in respect to the most significant data; for this reason we use technique *low-rank approximation* in order to restrict data to its most significant elements. This technique is given by:

$$A_{n\times m} \approx U_{n\times r}\tilde{\Lambda}_{r\times r}V^T_{r\times m} \quad (2)$$

where $\tilde{\Lambda}$ contains only the largest singular values, with the others being replaced by zero.

4. METHODOLOGY

Here we present an overview of the seven steps of our methodology: (1) Pre-processing and cleaning, (2) Modeling, (3) Relational transformation, (4) Selection, (5) Processing SVD, (6) Analysis - SVD, and (7) Interpretation/Evaluation – as illustrated in Figure 1. In the following sections, we provide details of each step.

4.1 Pre-processing and cleaning

DBLP data is full of redundancy, lack of conformity, non-homogeneity, and noise. Therefore, before we can start processing it, we must clean it up using multiple techniques applied to the papers' titles, names of authors, events, and vehicles. Our cleaning step includes the following techniques: (a) ASCII conversion of characters; (b) tokenization, the first step for text preparation in areas as natural language processing (NLP) and information retrieval (IR); (c) removal of stopwords, words that appear with high frequency in text sentences, but that have no content that can help in interpretation; and (d) stemming of the terms of the papers' titles, the combination of different forms of a word in a word representative joint, the *stem*.

4.2 Data modeling and relational transformation

Since DBLP is available in XML semi-structured format, it is not readily adequate to be represented as a graph; XML demands intense parsing operations, and does not support aggregation for the task of weighting the edges of the graph representation. Therefore, we firstly described DBLP as an entity-relationship model, further transforming it into a relational database. The model contains many-to-many relationships that describe the same information as that comprised by weighted graphs. This property is the focus of our analysis by means of SVD algebra.

4.3 Data selection

As presented in Table 1, the main entities of the database are Author and Article; the essence of DBLP's literature orbits this two entities. The former corresponds to 1,054,199 instances and the latter corresponds to 1,801,576 instances. These numbers are by far too large for algebraic processing, therefore we filtered them out by analyzing their distribution.

Table 1: Entities involved in our analysis

Entity	Number of entities
Authors	1,054,199
Articles	1,801,576
Events	3,050
Vehicles	4,262 (1,137 journals and 3,125 conferences)

4.3.1 Authors selection

One thing about DBLP is that it is heavily unbalanced in respect to its authors' production in number of articles. To verify this aspect closely, we plotted the Authors-Articles distribution in Figure 2(a). The plot shows a long-tail distribution in which the majority of authors has no more than 22 articles - more precisely this portion corresponds to 1,016,354 authors, or $\sim$ 96% of the instances. DBLP has also over 80 authors with more than 300 articles, and a champion author with over 600 articles. The plot clearly depicts a power-law distribution according to which the number of authors (y) having a certain number of articles (x) varies as a power of the number of articles, that is:

$$y \propto 10^6 x^{-2,06} \quad (3)$$

If we look close, this distribution is an instance of the Pareto Principle, or 80-20 rule, as $\sim$ 24% of the authors are responsible for $\sim$ 76% of the articles; while $\sim$ 76% of the authors are responsible for only $\sim$ 24% of the articles – see Figure 2(b). This leaded us to restrict our analysis to the 24% more prolific authors, that is, the 255,455 authors with 4 or more articles.

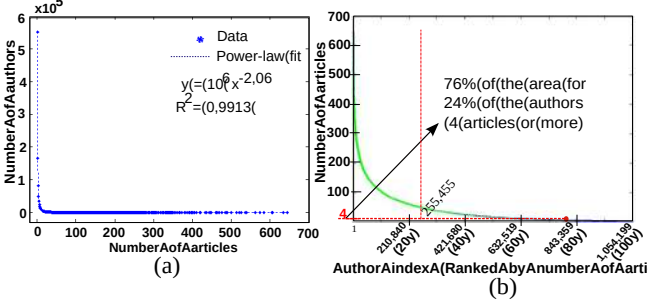

Figure 2: Authors-Articles distribution. (a) Number of articles $\times$ Number of authors. (b) Rank-plot on Author index $\times$ Number of articles.

4.3.2 Terms selection

Articles carry more information than simply defining instances; all of them have a title composed of a set of semantic terms. The semantics of such terms can be used to interpret and correlate DBLP data with richer details because specific terms address specific areas and research interests. By considering the terms of the titles instead of the articles, we got two advantages: since many terms appear recurrently in the titles, there are 292,919 terms, much less than the number of articles; and, interpreting key terms is simpler than interpreting complex titles. The drawback is that many terms appear in a frequency high enough to prevent valuable interpretation (e.g., "complex" and "efficient"); therefore, in this

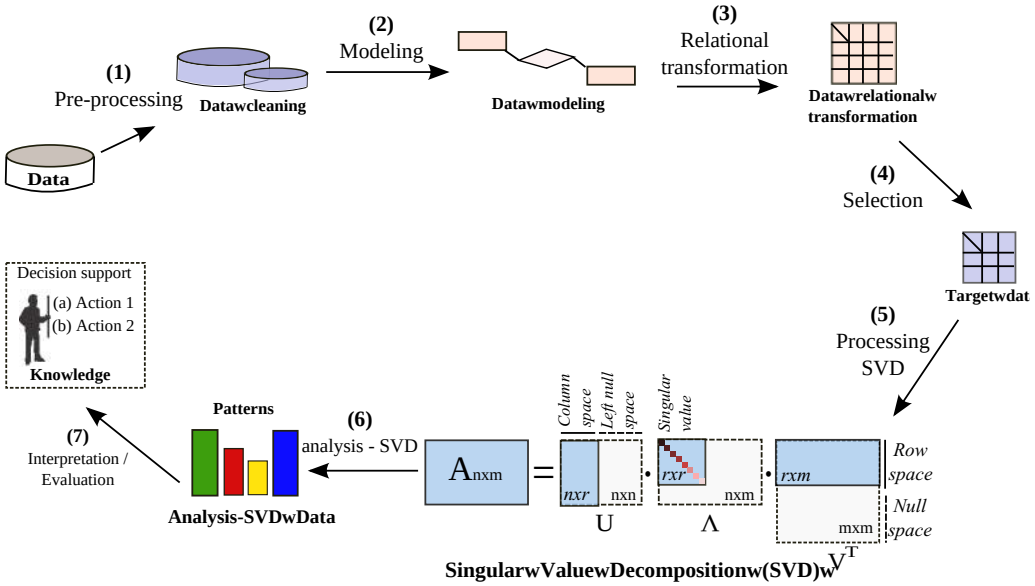

Figure 1: Diagrammatic representation of our analytical process.

step, we select the most relevant terms from the titles of the articles.

In order to choose the terms, we calculated their term frequency (TF); thus, we counted them and plotted the results. In Figure 3, we can see the most frequent terms x the term rank. There we can observe two things: (1) there are too many terms with very low frequency; these terms are considered irrelevant to semantic text analysis; (2) there are many terms with very high frequencies - these terms are also considered irrelevant.

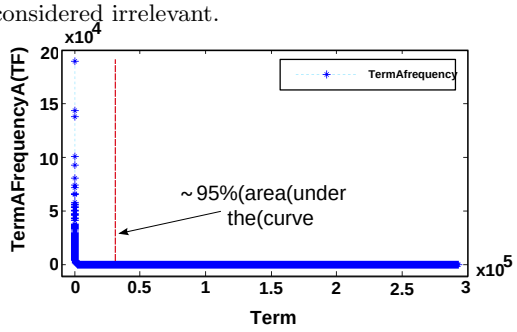

Figure 3: Ith-most frequent term x Term Frequency.

Hence, to restrict our space of terms, we interpreted the distribution of the terms according to Lunh's work [4] and Zipf's Law, which states that the most significative terms (TS) are those that are not too common, and neither those that are too rare; and, also, that the significance of the terms is given by the Normal Distribution centered on the mean (μ) of the TF-rank and standard deviation (σ), and given by equation 4:

$$TS(i) = f(r_i; \mu, \sigma) = \frac{1}{\sigma\sqrt{2\pi}} e^{-(r_i-\mu)^2/2\sigma^2} \qquad (4)$$

where r_i is the TF-rank of the term i, and $TS(i)$ is the significance value of the term i.

In our case $\mu = 2033.6$ and $\sigma = 412.1392$, what gives us the plot seen in Figure 4(a). In the plot, the terms near the left end are high frequency terms, which are generally too common to be significant; the terms near the far right are low frequency terms, too rare to be significant. Therefore, the most useful terms are in the mid-range, the core of the importance given by a Normal Distribution. We use the techniques of Liu and Hoeber [3] in order to algorithmically identify the best cut; the technique generated the curve seen in Figure 4(b), which corresponds to re-calculated importance of the terms after processing. Finally, we were left with the 4,061 most significant terms.

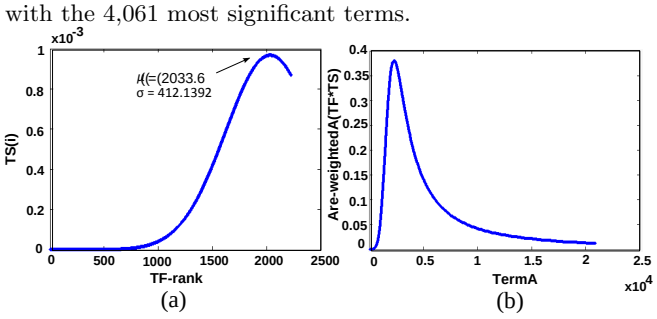

Figure 4: (a) TF-rank × Term significance (b) Term × Re-calculated importance of the terms (re-weighted).

5. EXPERIMENTS: SINGULAR VALUE DECOMPOSITION AND ANALYSIS

For our experiments, we use datasets of vehicles (all the conferences, workshops, and journals), authors, and terms, selected as described in subsections 4.3.1 and 4.3.2. With these sets of entities, we considered two experimental cases as summarized in Table 2.

In order to apply the Singular Value Decomposition, we represent, one at a time, each of the two experiment cases as nxm matrices A where n is the number of Terms and m is the number of Vehicles (cases 1) or Authors (case 2). This matrix was based on a bipartite graph defined by entities and relationships of our relational schema. After each pro-

Table 2: Dataset configurations used in the experiments.

case 1: Terms (Luhn) x Vehicles (Luhn)	Elements
Terms	4,061
Vehicles	3,014
Case 2: Terms (Luhn) x Authors	**Elements**
Terms	4,360
Authors	255,455

cessing, we ended up with three matrices, $U_{n\times r}$, $\Lambda_{r\times r}$, and $V^T_{r\times m}$, as predicted by the SVD theory – Equation 1.

5.1 Case 1: Terms (Luhn) x Vehicles (Luhn)

In this case, we selected the terms based on Luhn's theory, what caused the number of terms to be much lower. Though less terms, we ended up with terms that are not too frequent, neither too rare; that is, now we use the 4,096 most significant terms in contrast to the elevated number of 20,905 terms of the former experiment.

In this case, the denser semantic content led us to a bigger number of significant communities. Now, the singular values sum up to 80% and 90% only when we consider the 19 up to the 40 highest singular values – see Figure 5. This is a more intuitive result because there are many disciplines and interrelated disciplines in computer science; therefore, an elevated number of communities is expected. We proceeded the same way for selecting the most representative vehicles; with Luhn's theory we excluded the vehicles that were too prolific and those that had too few publications every year. From initial 4,254 vehicles, we went to 3,014.

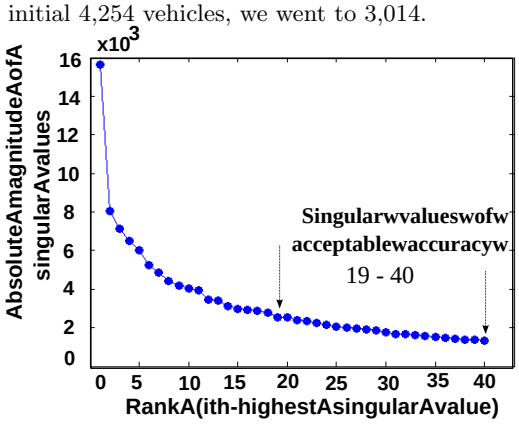

Figure 5: Scree-plot of the energy levels of the SVD for Terms (Luhn) x Vehicles.

Here we use again the scores of the vectors v_r and u_r to identify the most important terms and vehicles. In Table 3, we present 6 sample communities along with their most important terms. One can observe a deeper specificity of the communities due to the higher semantic density used in the analysis. Furthermore, in Table 4, we review the most important vehicles for the sample communities C2 and C16. In community C2, the characterization of the community is mainly given by conf/icip (International Conference on Image Processing - IEEE ICIP), a traditional image processing conference that is organized annually since 1994. In community C16, in turn, there is a clear human-computer interaction profile, as indicated by vehicles conf/chi (Human Factors in Computing Systems - ACM SIGCHI) which was formed in 1982, and conf/hci (Human-Computer Interaction) founded in 1984. We can observe that an event is more important if it is older.

Table 3: Terms (Luhn) x Vehicles (Luhn) – most frequent terms in six communities.

Topic	Most frequent terms
C2: image processing and computer vision	imag, video, segment, recognit, detect
C5: software engineering and web	data, inform, softwar, web, manag
C9: bio and parallel computing	simul, protein, data, gene, parallel
C16: interaction and multimedia	fuzzi, interact, design, comput, video
C20: control systems and programing	control, comput, stabil, linear, program
C29: chemical data processing	molecular, chemic, structur, studi, calcul

Table 4: Terms (Luhn) x Vehicles (Luhn) – top six conference and journals in the second and sixteenth community.

Vehicles (C2)	Score	#Articles
conf/icip	0.5670	13427
journals/ieicet	0.1835	12106
conf/icra	0.1774	11694
conf/icc	0.1587	6439
conf/vtc	0.1427	6875
Vehicles (C16)	**Score**	**#Articles**
conf/chi	0.2997	6903
conf/hci	0.2580	6323
journals/fss	0.2017	2669
conf/fuzzie	0.1804	6323
journals/nar	0.1304	2419

In this second experiment, we could observe that the SVD processing was significantly improved by the selection of terms based on the semantic filtering. In this case, the communities were better characterized both in terms and in vehicles, providing an insightful panorama of the computer science research.

5.2 Case 2: Terms (Luhn) x Authors

In the last experiment, we joined authors and the terms of the titles of the papers published by these authors. The analysis, here, is supposed to indicate communities according to the collaboration in between researchers rather than the topics of vehicles. The SVD decomposition indicated a large number – over 200 – of significant singular values in λ_r as illustrated in Figure 6, an expected behavior since the communities of authors obey to research groups, geographical proximity, and affinity in the area of expertise. As a sample, we present two highly significant terms for 6 major communities in Table 5. Based on these terms, we were able to characterize each community and, also, to estimate how active each community is, what is expressed by the order in the table.

In Table 6, we used the score of matrix $V_{r\times m}$ in order to track the most active authors in communities C2 and C4; along with the score, we validated the profile of the authors by checking the number of citations of each one at Microsoft Academic Research repository. For example, in community C2 the most important author is Thomas S. Huang with

19,988 citations; he is in the top 6 of the Most Prolific DBLP Authors, having Pattern Recognition and Computer Vision as his primary research area. The second most important author is Wen Gao, with 16,336 citations, who is the top 4 most prolific author of DBLP; his research interests are Pattern Recognition, Computer Vision and Multimedia. In the same way, in community C4 the most authoritative author is Sudhakar M. Reddy, with 8,119 citations, who is the 573rd most prolific author of DBLP; his research interests are Distributed and Parallel Computing. The second author is Irith Pomeranz, with 5,563 citations, top 506 in DBLP, and Distributed and Parallel Computing as his research interests. These data demonstrated that the SVD method can be quite effective and interpretable. Moreover, this analysis is also applicable to other data domains with precision and interesting analytical properties.

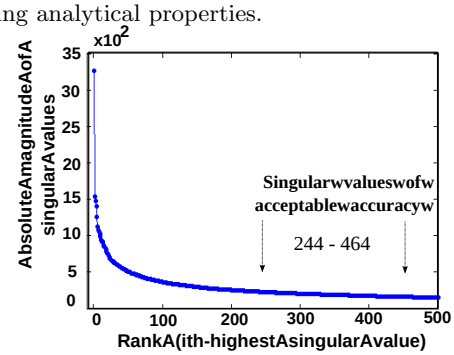

Figure 6: Scree-plot of the energy levels of the SVD for Terms (Luhn) x Authors.

Table 5: Terms (Luhn) x Authors – most frequent terms in six communities.

Topic	Most frequent terms
C2: Pattern recognition and image retrieval	imag, recognit, video, learn, segment
C4: Graph Theory	graph, problem, algorithm, program, logic
C12: Image and Video Processing	web, servic, video, fuzzi, problem
C28: Intelligent Information and Database Systems	databas, mobil, fuzzi, queri, perform
C39: Visualization	optim, visual, mobil, semant, distribut
C50: Bioinformatics	structur, protein, predict, scheme, sequenc

6. CONCLUSIONS

We presented an extensive analytical process suitable for the relationships observed in relational databases. The proposed process departs from the representation of the references in between tuples (vertices) of the database as edges of a graph, and involves techniques of text analytics combined with Singular Value Decomposition. This way, our methodology is able to identify communities and outstanding elements according to the interrelationship that the elements of the database define in the context of the relational schema.

We performed experiments over the DBLP dataset in order to defend the thesis that the use of text analytics can boost the results of SVD analysis. Our results demonstrated that, with the aid of text-based techniques, the output of the SVD process is pronouncedly denser in terms of semantic and specificity. Our results are explained by the use of techniques that rely on Luhn's theory and on Zipf's law, what, for the DBLP domain, produced a semantically concentrated core of data to be processed before the SVD analysis; as consequence, we achieved significant performance gains as well.

Table 6: Terms (Luhn) x Authors – top six authoritative authors in the second and fourth communities.

Author (C2)	Score	#Articles
Thomas S. Huang	0.1115	605
Wen Gao	0.0840	606
HongJiang Zhang	0.0645	295
Chin-Chen Chang	0.0582	645
Edwin R. Hancock	0.0576	536
Barry L. Nelson	0.0552	255
Author (C4)	**Score**	**#Articles**
Sudhakar M. Reddy	0.0652	525
Irith Pomeranz	0.0636	448
Noga Alon	0.0525	443
Marek Karpinski	0.0467	268
Alan M. Frieze	0.04637	270
Wil M. P. van der Aalst	0.0424	280

Acknowledgements

This work received support from Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq-560104/2010-3), Fundação de Amparo à Pesquisa do Estado de São Paulo (FAPESP-2011/13724-1) and Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (Capes).

7. REFERENCES

[1] U. Kang, B. Meeder, and C. Faloutsos. Spectral analysis for billion-scale graphs: Discoveries and implementation. LNCS 6635, pages 13–25. 2011.

[2] S.-W. Kim, K.-N. Kim, C. Faloutsos, and J.-H. Lee. Spectral analysis of a blogosphere. In *CIKM*, pages 2145–2148. ACM, 2011.

[3] H. Liu and O. Hoeber. A luhn-inspired vector re-weighting approach for improving personalized web search. In *Web Intelligence and Intelligent Agent Technology*, pages 301–305. IEEE Press, 2011.

[4] H. P. Luhn. The automatic creation of literature abstracts. *IBM J. Res. Dev.*, pages 159–165, 1958.

[5] K. Maruhashi and C. Faloutsos. Eigendiagnostics: Spotting connection patterns and outliers in large graphs. In *IEEE ICDMW*, pages 1328 –1337, 2010.

[6] B. Prakash, M. Seshadri, A. Sridharan, S. Machiraju, and C. Faloutsos. Eigenspokes: Surprising patterns and scalable community chipping in large graphs. In *IEEE ICDM*, pages 290–295, 2009.

[7] L. Wu, X. Ying, X. Wu, A. Lu, and Z.-H. Zhou. Spectral analysis of k-balanced signed graphs. LNCS 6635, pages 1–12. Springer Berlin Heidelberg, 2011.

[8] D. Zheng, K. A. Hoo, and M. J. Piovoso. Low-order model identification of distributed parameter systems by a combination of singular value decomposition and the karhunen-loève expansion. *Industrial & Engineering Chemistry Research*, 41(6):1545–1556, 2002.