2006

Automatic Extraction of Keywords and Co-occurrence Keyword Sets

Mong-Hang Vo
San Jose State University

AUTOMATIC EXTRACTION

OF

KEYWORDS AND CO-OCCURRENCE

KEYWORD SETS

A Project Report

Presented to

The Faculty of the Department of Computer Science

San Jose State University

In Partial Fulfillment

of the Requirements for the Degree

Master of Science

by

Mong-Hang Vo

November 2006

APPROVED FOR THE DEPARTMENT OF COMPUTER SCIENCE

Dr. Tsau Young Lin

Dr. Suneuy Kim

Dr. Robert Chun

APPROVED FOR THE UNIVERSITY

Dr. Tsau Young Lin

Dr. Suneuy Kim

Dr. Robert Chun

ABSTRACT

AUTOMATIC EXTRACTION OF KEYWORDS AND CO-OCCURRENCE KEYWORD SETS

by Mong-Hang Vo

Internet search has become an essential part of almost everyone's daily life and work. To make wise personal and business decisions in a timely fashion, one must access the most relevant information efficiently. Because the amount of information on the Internet is enormous, it is important that a search engine ranks the information appropriately when it presents search results to users. Latent Semantic Indexing (LSI) addresses relevance ranking based on how significant a search word is in each document.

Some innovative approaches of computing higher dimensional LSI (HD-LSI) were explored in this project. In traditional LSI, the term frequency-inverse document frequency (TFIDF) is calculated based on how significant a single word is in a document. The goal of this project is to generalize LSI to higher dimensions regarding the traditional LSI as the one-dimensional special case.

A benefit of the project is to enable a search engine to rank documents based on the special meaning of multi-word phrases, such as "wall street," which is captured by a two-dimensional LSI method. Another benefit of the project is the reusable Java software components that compute HD-LSI and store the indexes into a relational database, from which many types of applications can access the HD-LSI data. The software components may be reused for studying the proximity of semantics among documents in high dimensional space in future research.

Besides the software engineering aspect, this project contributes to computer science by studying the different approaches to HD-LSI computation. In particular, the dimensional trends in each case were analyzed.

ACKNOWLEDGEMENTS

Foremost, my many thanks to my advisor Dr. T. Y. Lin for his invaluable insights and knowledgeable guidance, without which I would have certainly fallen into hopeless black holes and never been able to complete this project.

Moreover, I offer my deepest appreciation to Dr. Suneuy Kim and Dr. Robert Chun for participating in my project committee. Finally, I want to give special thanks to Dr. Cay Horstmann, for his support as a graduate coordinator of the computer science department.

Table of Contents

1. Introduction 10

2. Requirements 10

2.1 Project Scope 10

3. Design 10

3.1 Software Architecture 10
3.1.1 Document Preprocessor 11
3.1.2 Computation Unit of LSI for One Keyword (LSI1) 13
3.1.3 Computation Unit of LSI for Two Keywords (LSI2) 14
3.1.4 Computation Unit of LSI for Three or More Keywords (LSI3, LSI4, …) 17
3.2 Database Schemas 18

4. Implementation 22

4.1 Main Objective 22
4.2 Data Structure 22
4.3 Algorithm 22
4.3.1 Implementation of N(ti, dj) Computation 22
4.3.2 Performance Consideration 23
4.3.3 Maintainability Consideration 24
4.4 Programming Language 24

5. Deployment 24

5.1 Overview of Testing 25
5.2 Testing Requirements 25
5.2.1 Hardware Requirements 25
5.2.2 Software Requirements 25
5.3 Test cases 25
5.3.1 Document Preprocessor Test 25
5.3.2 Sparse Matrix Test 25
5.4 Program Run Dependencies 28

6. Analysis 29

6.1 Method 1 - Use TFIDF as a Threshold and N(ti, dj) as an Integer 29
6.1.1 LSI1 29
6.1.2 LSI2 31
6.1.3 LSI3 33
6.1.4 LSI4 34
6.1.5 Dimensional Trends 36
6.2 Method 2 - Use TFIDF as a Threshold and N(ti, dj) as a Fraction 37
6.2.1 LSI1 37
6.2.2 LSI2 38
6.2.3 LSI3 39
6.2.4 LSI4 41
6.2.5 Dimensional Trends 42
6.3 Method 3 - Use Document Frequency as a Threshold and N(ti, dj) as a Fraction 43
6.3.1 LSI1 43
6.3.2 LSI2 44
6.3.3 LSI3 46
6.3.4 LSI4 47
6.3.5 Dimensional Trends 49
6.4 Method 4 – Refined Method 2 by Removing Stop Words during Document Preprocessing 49

6.4.1 LSI1 50
6.4.2 LSI2 51
6.4.3 LSI3 52
6.4.4 LSI4 53
6.4.5 Dimensional Trends 55

7. Conclusion 56

8. Appendices 57

8.1 Appendix A – Stop List 57
8.2 Appendix B – Database Samples 57
8.2.1 Method 1 58
8.2.2 Method 2 62
8.2.3 Method 3 66
8.2.4 Method 4 70
8.3 Appendix C – References 74

List of Figures

Figure 1. Overview of The Design 11
Figure 2. Document Preprocessor and Inverted Table 12
Figure 3. Computation Unit of LSI for One Keyword (LSI1) 13
Figure 4. Preparation Phase of LSI for Two Keywords (PreLSI2) 14
Figure 5. Main Computation Unit of LSI for Two Keywords (LSI2) 15
Figure 6. UML Class Diagram 16
Figure 7. Preparation Phase of LSI for Three Keywords (Pre-LSI 3) 17
Figure 8. Main Computation Unit of LSI for Three Keywords (LSI 3) 18
Figure 9. Improved Algorithm for LSI Computation 24
Figure 10. Unit Tests Developed under the JUnit Framework Executed by Eclipse 27
Figure 11. Program Run Dependencies 28
Figure 12. The Size of the Inverted Table Generated from Document Preprocessor 29
Figure 13. Dimensional Trends of Method 1 36
Figure 14. Dimensional Trends of Method 2 43
Figure 15. Dimensional Trends of Method 3 49
Figure 16. Dimensional Trends of Method 4 55
Figure 17. Method1: LSI1 58
Figure 18. Method1: LSI2 59
Figure 19. Method1: LSI3 60
Figure 20. Method1: LSI4 61
Figure 21. Method2: LSI1 62
Figure 22. Method2: LSI2 63
Figure 23. Method2: LSI3 64
Figure 24. Method2: LSI4 65
Figure 25. Method3: LSI1 66
Figure 26. Method3: LSI2 67
Figure 27. Method3: LSI3 68
Figure 28. Method3: LSI4 69
Figure 29. Method4: LSI1 70
Figure 30. Method4: LSI2 71
Figure 31. Method4: LSI3 72
Figure 32. Method 4: LSI4 73

List of Tables

Table 1. Inverted Table Schema 18
Table 2. LSI 1 Table Schema 19
Table 3. Reduced Inverted2 Table Schema 19
Table 4. LSI 2 Table Schema 20
Table 5. Reduced Inverted3 Table Schema 20
Table 6. LSI 3 Table Schema 21
Table 7. LSI 4 Table Schema 21
Table 8. Method1: LSI1 Analysis 30
Table 9. Method1: LSI2 Analysis 32
Table 10. Method1: LSI3 Analysis 34
Table 11. Method1: LSI4 Analysis 36
Table 12. Method2: LSI1 Analysis 38
Table 13. Method2: LSI2 Analysis 39
Table 14. Method2: LSI3 Analysis 40
Table 15. Method2: LSI4 Analysis 42
Table 16. Method3: LSI1 Analysis 44
Table 17. Method3: LSI2 Analysis 45
Table 18. Method3: LSI3 Analysis 47
Table 19. Method3: LSI4 Analysis 49
Table 20. Method4: LSI1 Analysis 51
Table 21. Method4: LSI2 Analysis 52
Table 22. Method4: LSI3 Analysis 53
Table 23. Method4: LSI4 Analysis 55

1. Introduction

Popular Web search services have substantially changed the way we get information for work and life. Data mining technologies, such as those adopted by Amazon.com, have changed how we interact with one another in a large community. Therefore, an important question is whether we can apply advanced data mining technologies on improving Web search services. There are a number of efforts in making text search take advantage of meaning or semantics instead of merely relying on keywords [C97].

This project explores several approaches of high dimensional latent semantic indexing (HD-LSI) techniques. We will view traditional LSI as one-dimensional LSI [W05a]. This project will explore "LSI of co-occurring keywords." The project will begin a formal study, which consists of two parts: software engineering and empirical study. The goal of the software engineering part of the project is to develop a generic framework and a testable system that facilitate the extension of LSI to higher dimension. The goal of the empirical part of the project is to explore and analyze several approaches of computing HD-LSI using the system developed in the first part of the project.

2. Requirements

2.1 Project Scope

The goals of this project are as follows.

- To design software that computes HD-LSI.
- To design experiment to compare four methods for computing HD-LSI. Method 1 uses tfidf as a threshold to limit the size of input to HD-LSI computation. Method 2 differs from Method 1 in that a denominator is introduced in calculation of N(ti, dj), which is an important factor in the calculation of tfidf. In Method 2, the denominator is the total number of all types of tokens in document dj. Method 3 differs from Method 2 only on how the input sizes of multi-dimensional LSI calculation. Method 3 uses document frequency (DF) as the threshold to pick up only the terms with high enough DF to feed into the computation of LSI2 to reduce its input size. Method 4 differs from Method 2 in that it discards common function words (sometimes known as stop words) during document preprocessing.

The main scope of the project includes:

- Processing the entire collection of 16330 documents from University of California, Irvine Knowledge Discovery in Databases (UCI KDD) [U06]. Because LSI computation involves the entire collection of documents, the larger the collection, the more meaningful the result. The same is true for HD-LSI.
- Designing a database schema to store documents, terms, occurrence locations, and LSI information.
- Writing Java programs that process text documents and access an Oracle database via Java Database Connectivity (JDBC).
- Setting up a computer and deploying both the Java programs and the Oracle database on the computer to process a large set of text documents to produce scientific results.

3. Design

3.1 Software Architecture

Figure 1 shows the software architecture of this project that consists of the following major components:

- The document preprocessor.
- The computation unit of LSI for one keyword (LSI1).
- The computation unit of LSI for two keywords (LSI2).

- The computation units of higher dimensional LSI for *n* keywords (LSI*n*) work similarly. The object-oriented framework is designed to facilitate the extension of LSI to any dimension. The author has implemented the computation units up to dimension four in this project.

The document preprocessor reads the entire set of documents from a file system and stores the resulting inverted files as tables in a database. LSI1 takes the inverted tables from the database to calculate the LSI and stores the LSI into a different table in the same database. LSI2 takes as its input the Cartesian product of the inverted tables filtered with LSI to reduce its size. It then calculates the LSI for each pair of terms and stores them into a different table in the database. LSI3 and higher dimensional LSI modules work in a way similar to LSI2.

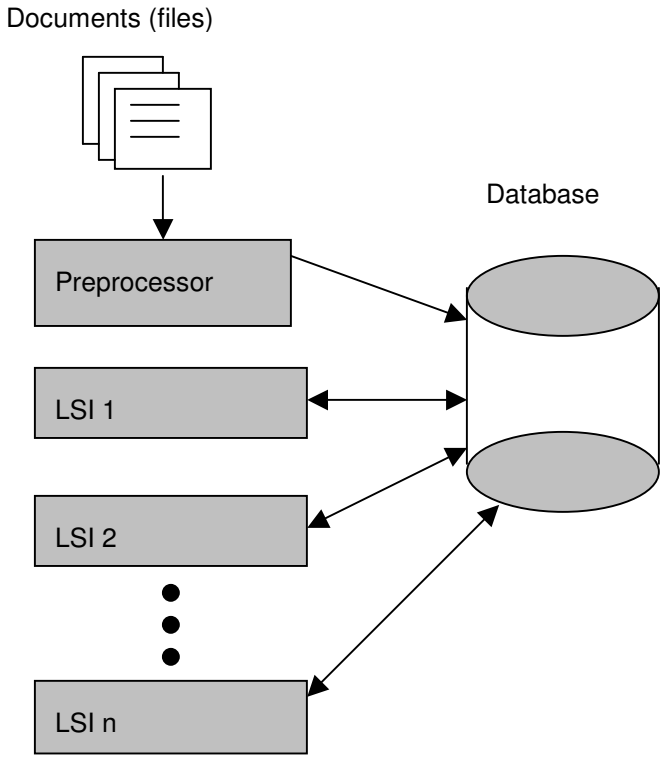

Figure 1. Overview of The Design

3.1.1 Document Preprocessor

The document preprocessor reads text documents from a file system. It preprocesses each document by extending the analyzer of a popular open-source software package, Apache Lucene. The analysis involved in document preprocessing consists of the following steps:

1. Tokenization is the preprocessing step for dividing a document into terms or words. The step is performed by the LetterTokenizer of Lucene, which uses the Java built-in method Character.isLetter(char) to determine whether a character is a letter or not. Any non-letter character is regarded as a separator between terms.
2. Canonicalization is the preprocessing step for reducing different forms of the same term into a single representation for accurate comparison between terms. Canonicalization involves lowercasing and stemming.

a. Lowercasing converts each letter of each term into lower case. For example, “Apple” and “apple” are both converted to “apple” so that the system regards them as being the same term regardless of whether the term appears at the beginning or somewhere in the middle of a sentence.

b. Stemming removes inflectional morphemes from each term. For example, “apples” is converted to the same representation as “apple” so that the system regards both “apples” and “apple” as being the same term regardless of whether the term is in its plural or singular form. Lucene uses Porter’s Algorithm for stemming.

For each term in each document, the document preprocessor canonicalizes the term and associates it with the document in which the term appears by inserting a row into inverted tables in an Oracle database. In the same row, the document preprocessor also records the location or position in which the term appears in the document.

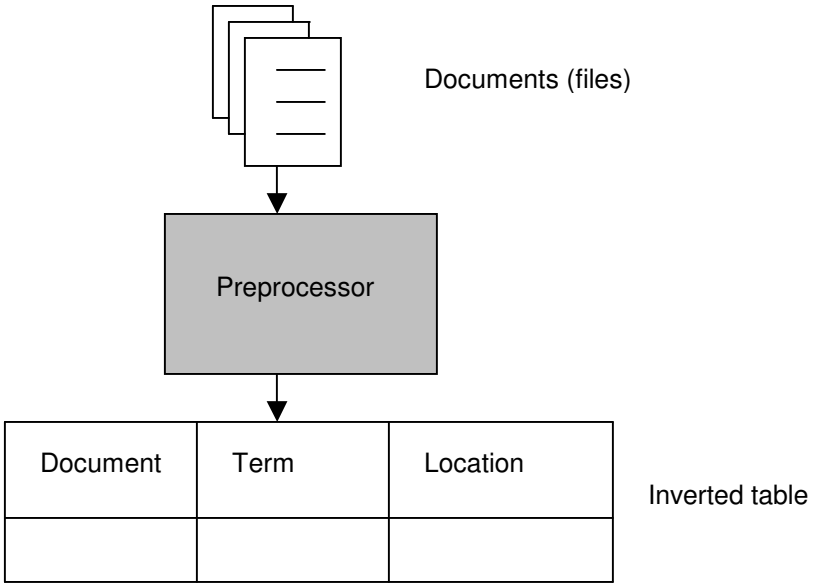

Figure 2. Document Preprocessor and Inverted Table

3.1.2 Computation Unit of LSI for One Keyword (LSI1)

LSI1 reads the inverted tables. It computes LSI for each term in each document. It then inserts its results into the LSI1 table in the database. The LSI1 table contains the information on how significant each term in each document. For example, if the term "apple" is very significant in Document #2, the tuple *<"apple", 2, lsi1>* will have a very large value for its *lsi1*. The author implemented the entire algorithm of LSI calculation in Java, and the algorithm and data structures will be described in Section 4 below.

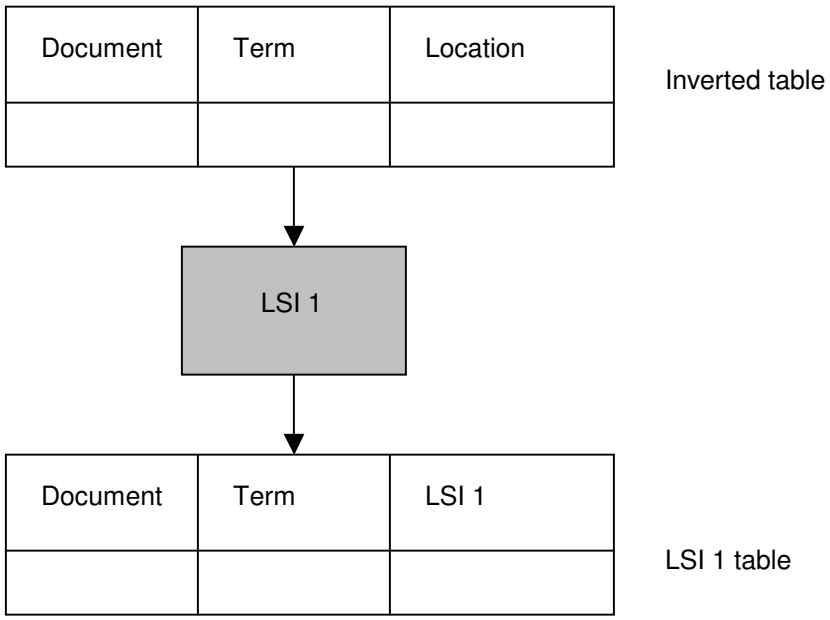

Figure 3. Computation Unit of LSI for One Keyword (LSI1)

3.1.3 Computation Unit of LSI for Two Keywords (LSI2)

The goal of LSI2 is to find out how significant each pair of tokens in each document. In theory, the input of LSI2 is the Cartesian product between the inverted table and itself. Since the inverted table has more than 5 million rows, the Cartesian product would have more than 2.5×10^{13} rows. Since the author did not have access to the computing machinery required for processing this enormous number of rows, the size of the inverted table needed to be reduced. The preparation phase of LSI2 (PreLSI2) was created for this purpose.

During the preparation phase of LSI2 computation, a subset of the inverted table is copied into a "reduced" inverted table. The resulting reduced inverted table contains only the terms that pass the criterion defined by the method of computation, to be described in Section 6. The reduced inverted table joins with itself forming the inverted table for LSI2 ("inverted table2").

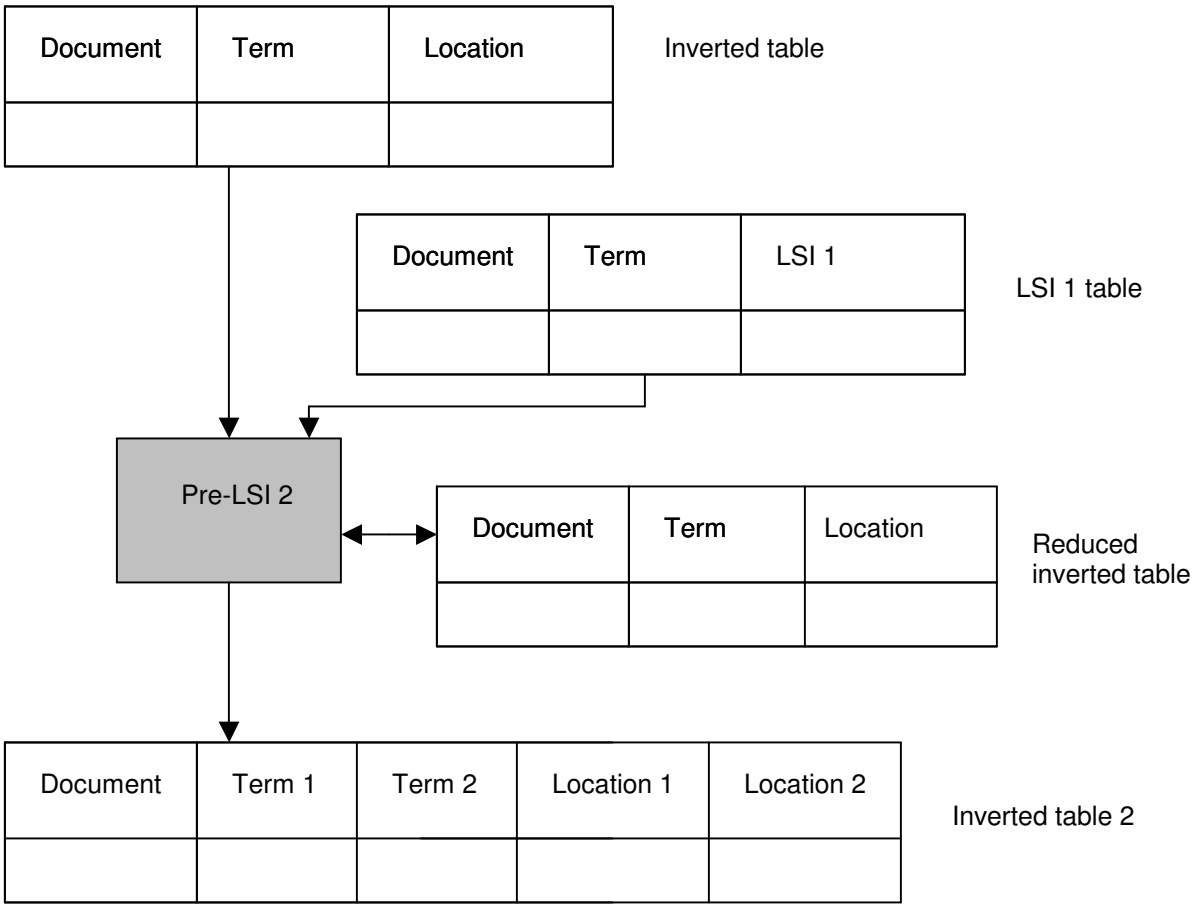

Figure 4. Preparation Phase of LSI for Two Keywords (PreLSI2)

The main computation unit LSI2 takes inverted table 2 as input to compute the LSI for each pair of terms in inverted table 2. The algorithm and data structure for the computation will be described in Section 4 below.

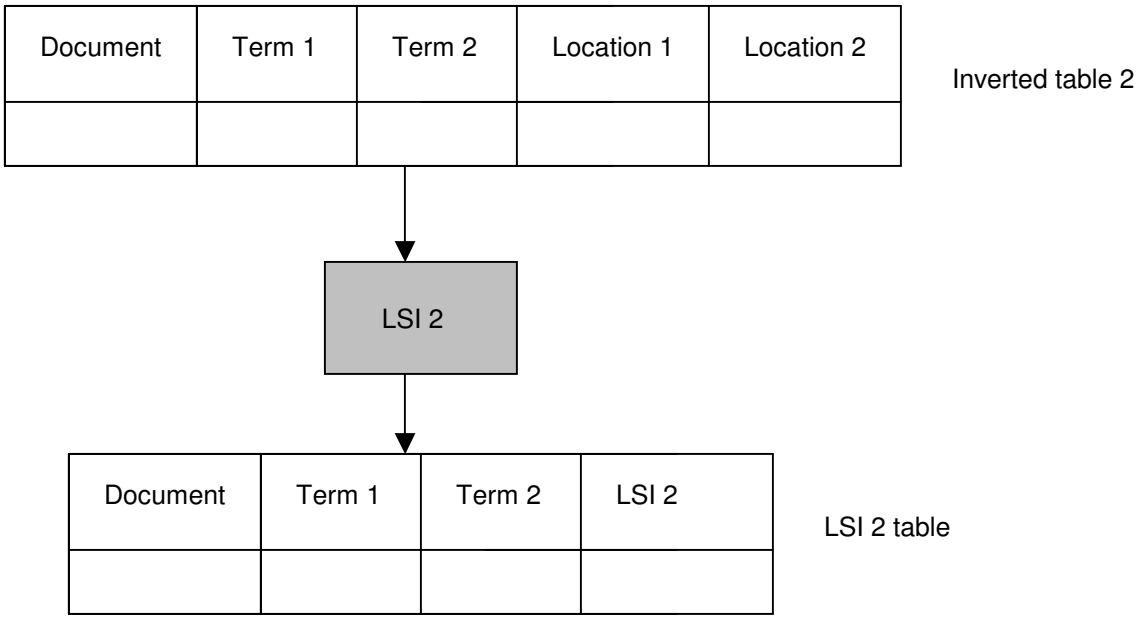

Figure 5. Main Computation Unit of LSI for Two Keywords (LSI2)

Since LSI2 shares the same algorithm and data structure as LSI1, LSI2 simply "extends" LSI1 using the object-oriented inheritance mechanism provided by Java, as shown in Figure 6. Because of the use of inheritance, the source code that implements the LSI algorithm and the necessary sparse matrix data structure that supports it is reused rather than being duplicated. The corresponding database access modules are reused in a similar way. For example, as shown in Figure 6, DbForLSI2 extends DbForLSI1. This allows all the reusable source code to be in DbForLSI1. The reusable source code accesses Oracle database using Java Database Connectivity (JDBC), and it does not need to reappear in DbForLSI2. The only source code that needs to be in DbForLSI2 is two SQL statements that are specific to two-dimensional LSI. The two SQL statements are invoked during the execution of the common LSI algorithm by polymorphism in Java.

Using the same object-oriented programming framework, the author implemented LSI3 and LSI4 with little additional source code. In the future, higher dimensional LSI Java classes can be made to be automatically generated from this framework.

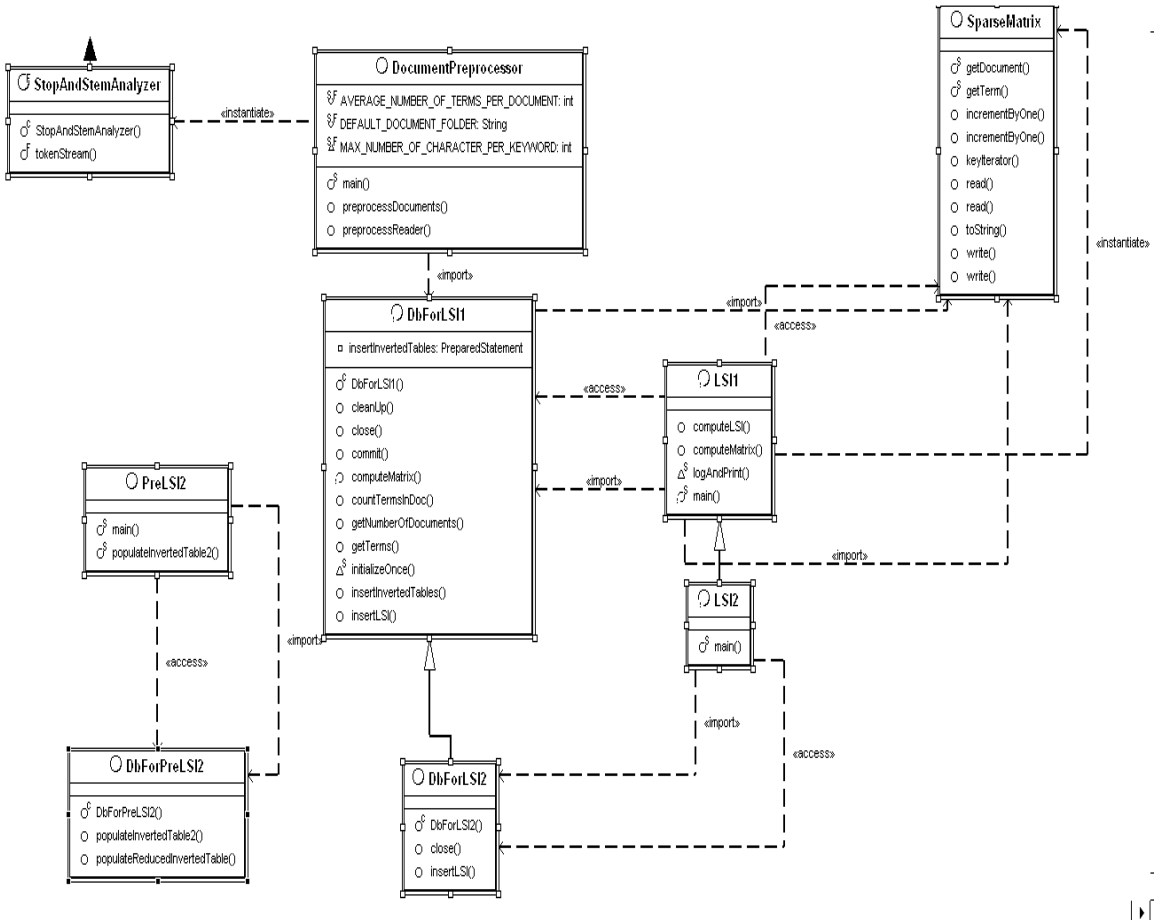

Figure 6. UML Class Diagram

LSI3, LSI4, ... LSI*n* (not shown in Figure 6) are all sub-classes of LSI1 similar to LSI2. DbForLSI3, DbForLSI4, ... DbForLSI*n* are all sub-classes of DbForLSI1 similar to DbForLSI2.

3.1.4 Computation Unit of LSI for Three or More Keywords (LSI3, LSI4, …)

The computation of LSI3 is very similar to that of LSI2. During the preparation phase, the reduced inverted table joins with itself three times to forming the inverted table for LSI3 ("inverted table3").

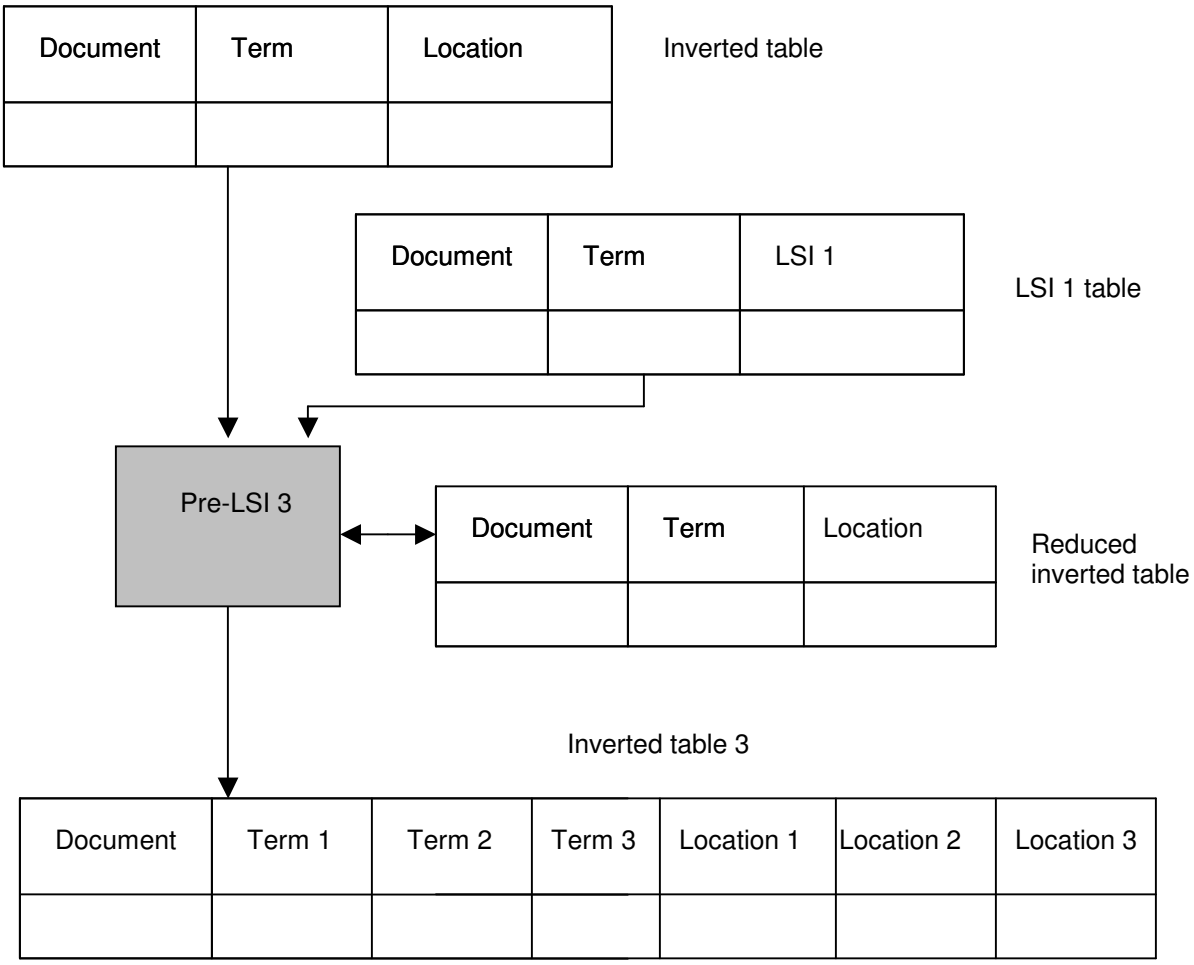

Figure 7. Preparation Phase of LSI for Three Keywords (Pre-LSI 3)

The main computation unit LSI3 takes inverted table 3 as input to compute the LSI for each tuple of terms in inverted table 3. The algorithm and data structure for the computation are the same as those for LSI2. The framework is general enough to be extended to handle LSI4 and higher dimensions. As an initial experiment, the author has extended the framework to implement the calculation of high dimensional LSI up to LSI4.

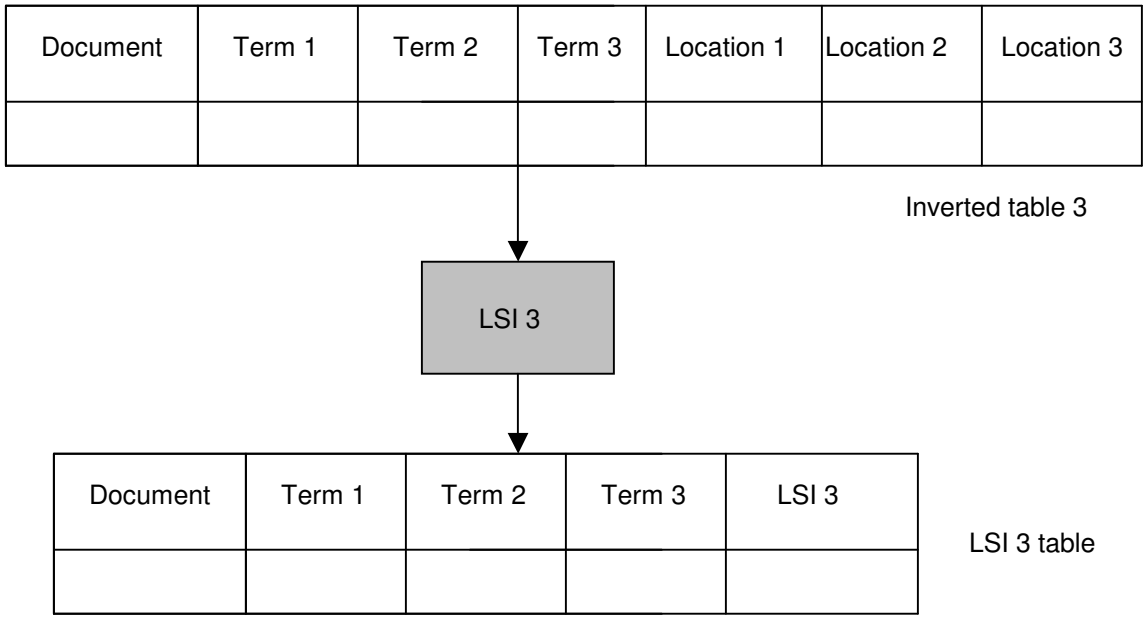

Figure 8. Main Computation Unit of LSI for Three Keywords (LSI 3)

3.2 Database Schemas

The document preprocessor reads 16,330 text documents downloaded from UCI KDD. Each term in each document is stored into one big inverted table in the database. Table 1 shows the schema for the inverted table.

Name	Null?	Type
DOCUMENT	NOT NULL	VARCHAR2(6)
TERM	NOT NULL	VARCHAR2(18)
LOCATION	NOT NULL	NUMBER(5)

Table 1. Inverted Table Schema

For each document, a value of LSI1 for each term is stored in the database. Table 2 shows the schema for the LSI1 table. The larger is the value of LSI1, the more important is the term in that document.

Name	Null?	Type
DOCUMENT	NOT NULL	VARCHAR2(6)
TERM	NOT NULL	VARCHAR2(18)
LSI1	NOT NULL	FLOAT(126)

Table 2. LSI 1 Table Schema

The reduced inverted table, which is a subset of the inverted table and contains only the terms that have high LSI1 values, joins with itself forming the inverted table for LSI2 ("inverted table2").

Name	Null?	Type
DOCUMENT	NOT NULL	VARCHAR2(6)
TERM1	NOT NULL	VARCHAR2(18)
TERM2	NOT NULL	VARCHAR2(18)
LOCATION1	NOT NULL	NUMBER(5)
LOCATION2	NOT NULL	NUMBER(5)

Table 3. Reduced Inverted2 Table Schema

Similarly, for each document, a value of LSI2 for each high-LSI1 term-pair is stored in the database. Below is the schema for the LSI2 table. The larger is the value of LSI2, the more important is the pair of terms in that document.

Name	Null?	Type
DOCUMENT	NOT NULL	VARCHAR2(6)
TERM1	NOT NULL	VARCHAR2(18)
TERM2	NOT NULL	VARCHAR2(18)
LSI2	NOT NULL	FLOAT(126)

Table 4. LSI 2 Table Schema

Similarly, during the preparation phase of LSI4's computation, the reduced inverted3 table, which is a subset of the inverted table2 and contains only the terms that have high LSI2 values, joins with itself three times to forming the inverted table for LSI3 ("inverted table3").

Name	Null?	Type
DOCUMENT	NOT NULL	VARCHAR2(6)
TERM1	NOT NULL	VARCHAR2(18)
TERM2	NOT NULL	VARCHAR2(18)
TERM3	NOT NULL	VARCHAR2(18)
LOCATION1	NOT NULL	NUMBER(5)
LOCATION2	NOT NULL	NUMBER(5)
LOCATION3	NOT NULL	NUMBER(5)

Table 5. Reduced Inverted3 Table Schema

For each document, a value of LSI3 for each term is stored in the database. Table 3 shows the schema for the LSI3 table.

Name	Null?	Type
DOCUMENT	NOT NULL	VARCHAR2(6)
TERM1	NOT NULL	VARCHAR2(18)
TERM2	NOT NULL	VARCHAR2(18)
TERM3	NOT NULL	VARCHAR2(18)
LSI3	NOT NULL	FLOAT(126)

Table 6. LSI 3 Table Schema

Similarly, for each document, a value of LSI4 for each 4-tuple of terms is stored in the database. Below is the schema for the LSI4 table.

Name	Null?	Type
DOCUMENT	NOT NULL	VARCHAR2(6)
TERM1	NOT NULL	VARCHAR2(18)
TERM2	NOT NULL	VARCHAR2(18)
TERM3	NOT NULL	VARCHAR2(18)
TERM4	NOT NULL	VARCHAR2(18)
LSI4	NOT NULL	FLOAT(126)

Table 7. LSI 4 Table Schema

4. Implementation

This section describes the implementation details of the computation of TFIDF: its objective, data structure, algorithm, and programming language.

4.1 Main Objective

The key computation of latent semantic indexing is to calculate term frequency-inverse document frequency (TFIDF) based on the formulae below [W05b].

TFIDF(term_i, document_j) = tf(ti; dj) log |Tr|/|Tr(ti)|

where *Tr(ti)*= the number of documents in *Tr* in which *ti* occurs at least once.

$$\mathrm{tf}(\mathrm{ti};\mathrm{dj}) = \begin{cases} 1+\log(\mathrm{N}(\mathrm{ti};\mathrm{dj})) & \text{if } \mathrm{N}(\mathrm{ti};\mathrm{dj})>0 \\ 0 & \text{otherwise} \end{cases}$$

N(ti, dj) = the frequency of *ti* in *dj*.

As shown above, the computation of TFIDF requires the computation of the important matrix $N(ti, dj)$. Conceptually, *N(ti, dj)* is a huge matrix, which spans the two-dimensional Cartesian space of terms and documents.

4.2 Data Structure

As shown in Figure 6, the SparseMatrix data structure is instantiated and accessed by LSI1, which is the common implementation of all higher dimensional LSI Java classes.

To represent this sparse matrix efficiently in physical memory, the author uses a Java Tree Map to represent $N(ti, dj)$ with the key order *dj* and then *ti*. A Map in Java is simply a set of key-value pairs. Given a key, a map returns the value that is associated with that key.

If the value of a particular key is zero, the algorithm does not store the key-value pair in the map. If a key does not exist in the map, the algorithm returns zero as the default value of that key. Therefore, the tree map represents the sparse matrix efficiently in physical memory.

4.3 Algorithm

Although the formula of TFIDF appears to be straightforward, several issues are taken into consideration during the implementation of the computation unit.

4.3.1 Implementation of N(ti, dj) Computation

The sparse matrix N(ti, dj) is populated as the program scans through an inverted table. In Method 1 of the analysis, to be described in Section 6, the entry of N(ti, dj) is incremented by 1 per occurrence of ti in dj. In the other methods, it is incremented by 1/(the number of terms in dj) per occurrence of ti in dj. The resulting N(ti, dj) may be less than 1, and therefore 1 + log(N(ti, dj)) may be a negative number. When the inverse document frequency (always non-negative) is multiplied by a negative number, the resulting tfidf would not be useful. To avoid this undesirable situation, a large enough coefficient is added in the calculation of term frequency before taking the logorithm of N(ti, dj) to ensure that the logorithm is always non-negative.

4.3.2 Performance Consideration

To take advantage of the efficient representation of sparse matrix, I have redesigned the algorithm mainly for performance. Consider the implementation that directly implements the formulas for TFIDF.

```
For each document dj,

   For each term ti,
           if N(ti; dj) > 0, then
                 tf(ti; dj) = 1 + log(N(ti; dj))
           else
                 tf(ti; dj) = 0

           TFIDF(ti; dj) = tf(ti; dj) log/ |Tr|/|Tr(ti)|
```

There are severe performance problem with this straightforward implementation because of excessive unnecessary iterations and database access. There were 16330 documents in the UCI KDD corpus, which contains 88867 different terms. The above straightforward implementation would require 1.5 billion (16330 x 88867) iterations. However, the entire inverted table contains only 6.3 million rows. Therefore, only 6.3 million iterations are required in the optimal implementation.[1] The straightforward implementation also prohibits Tr(ti) from being efficiently computed in the same loop as the TFIDF computation.

4.3.2.1 Improved Algorithm

The author has designed and implemented an algorithm that has the number of iterations equal to the number of rows in the inverted table. Furthermore, it computes Tr(ti) efficiently in the same loop as the TFIDF computation.

```
1:     sql = getComputeMatrixSql();

       try
       {
          resultSet = stmt.executeQuery(sql);
          int rowNumber = 0;
2:        Set incrementedTrtiForCurrentDocument = new HashSet();
          String previousDocumentId = null;
          while (resultSet.next())
          {
             String documentId =  resultSet.getString(1);
             String term = resultSet.getString(2);
             int denominator = resultSet.getInt(3);

             Ntidj.incrementByOneOverDenominator(term,   documentId,   (double)
denominator);

3:           if (! documentId.equals(previousDocumentId))
                // Now it is a different document
                incrementedTrtiForCurrentDocument.clear();
             previousDocumentId = documentId;

             // Increment Trti only if it has not been
```

[1] Although the numbers here ignore the fact that short documents were removed, the performance problem is severe regardless of whether short documents are removed.

```
            // incremented for the current document.
            if (! incrementedTrtiForCurrentDocument.contains(term))
            {
               Trti.incrementByOne(term);
               incrementedTrtiForCurrentDocument.add(term);
            }

            if (fivePercent == 0 || (rowNumber % fivePercent == 0))
               // Print progress indicator
               System.out.print(String.valueOf((short) (rowNumber * 100.0
                     / totalNumberOfRows + .5))
                     + "% ");
            rowNumber++;
         }
         System.out.println();

      }
```

Figure 9. Improved Algorithm for LSI Computation

Line 1 polymorphically gets the SQL statement that selects the rows from the inverted table ordered by document ID. Line 2 instantiates a HashSet for computing Tr(ti) efficiently. The set remembers the terms that have incremented Tr(ti) for the current document. The algorithm loops through each row of the inverted table in the order of document ID. Whenever it detects a new document ID, it empties out the HashSet to ensure that Tr(ti) is computed correctly, as shown in Line 3.

4.3.3 Maintainability Consideration

Because of the object-oriented design as shown in Figure 6, the improved algorithm for LSI computation shown in Figure 9 automatically benefits any higher dimensional LSI computation without duplicating the source code. The polymorphism in Java allows each higher dimensional LSI database module to implement its own SQL statement to be returned by the method getComputeMatrixSql(). This allows the same efficient Java code to be shared by different dimensional calculations with each dimension having its own SQL for database access.

4.4 Programming Language

The main criteria of choosing the Java programming language of this project are:

- Cross-platform independent
- Easy to maintain
- Productive IDEs (Eclipse and NetBeans)
- Unit test using JUnit test framework

Java works across many operating systems, including Unix and Windows, without much modification of source code. It also has the industry-wide standard database interface JDBC, which allows a program to use database software from many vendors and open-source communities, such as Oracle, DB2, and MySQL. Because of the popularity of Java in both industry and academia, many useful tools and frameworks have been developed to improve the productivity of programmers and testers. Eclipse/NetBeans and JUnit are good examples.

5. Deployment

This section describes the deployment aspects of the project. It describes the testing considerations for each component and the dependencies among the components.

5.1 Overview of Testing

For unit testing, the author has developed test cases using the JUnit test framework [J01] because many open source and commercial tools support the framework.

5.2 Testing Requirements

This section describes the hardware and the software requirements for testing.

5.2.1 Hardware Requirements

This project currently has been tested on an HP AMD64 3700 PC with the Windows XP Media Center Edition. The computation of LSI for two-keyword pairs is too resource intensive for a typical personal computer to handle. Therefore, the current LSI2 program uses a very high threshold (LSI1=10.0) to filter out most of the possible two-keyword pairs from its input.

To calculate LSI for two-keyword pairs for an input of a good size, the program will need to be deployed onto a server machine.

5.2.2 Software Requirements

The following software needs to be installed on a windows machine for further developing the system.

- Java Development Kit 1.5.0+
- Java Unit test framework (JUnit 3.0+). [J01]
- Oracle9i Enterprise Edition Release 9.2.0+
- Apache Lucene 2.0.0+. [A06]

5.3 Test cases

The system consists of several components, and each component performs a specific function. The following sections describe how each the unit test verifies the function of each component.

5.3.1 Document Preprocessor Test

Given a test input line of text, the preprocessor is expected to perform the following tasks correctly: tokenizing, lowercasing, stemming, and stop-word filtering.

5.3.2 Sparse Matrix Test

The test case ensures that the data structure behaves as expected. It verifies the functionality by performing the following tests:

- Test writing and then reading: write a value into the matrix, read it back, and expect the same value.
- Test reading the default value: read a non-existing entry and expect that the matrix returns zero.
- Test incrementing by one: Write a value into the matrix, increment the value by one in the matrix, read it back, and expect the value to be one greater than the original one.

Two articles were chosen from the Wall Street Journal as the input documents for the test cases.

5.3.2.1 Test LSI 1

The test case ensures that the computation behaves as expected. It verifies the functionality by performing the following tests:

- Test the normal case with a keyword “hedge,” which appears five times in the first document, but does not appear in the second documents.

- TestZero: a real word that does not appear in a particular document
- Test NaN: “garbage1874650*#”

5.3.2.2 Test LSI 2

The test case ensures that the computation behaves as expected. It verifies the functionality by performing the following tests:

- Test the normal case with a keyword “hedge fund,” which appears four times in the first document, but does not appear in the second documents.
- TestZero: a real word that does not appear in a particular document
- Test NaN: “garbage1874650*# garbage1874650*#”

5.3.2.3 Test LSI 3

The test case ensures that the computation behaves as expected. It verifies the functionality by performing the following tests:

- Test the normal case with a keyword “service oriented architecture”
- TestZero: a real word that does not appear in a particular document
- Test NaN: “garbage1874650*# garbage1874650*# garbage1874650*#”

Figure 10 shows the results of a successful execution of all the unit tests in the Eclipse integrated development environment. All the test cases were developed under the JUnit framework, which provides a standard way that facilitates the integration and testing efforts.

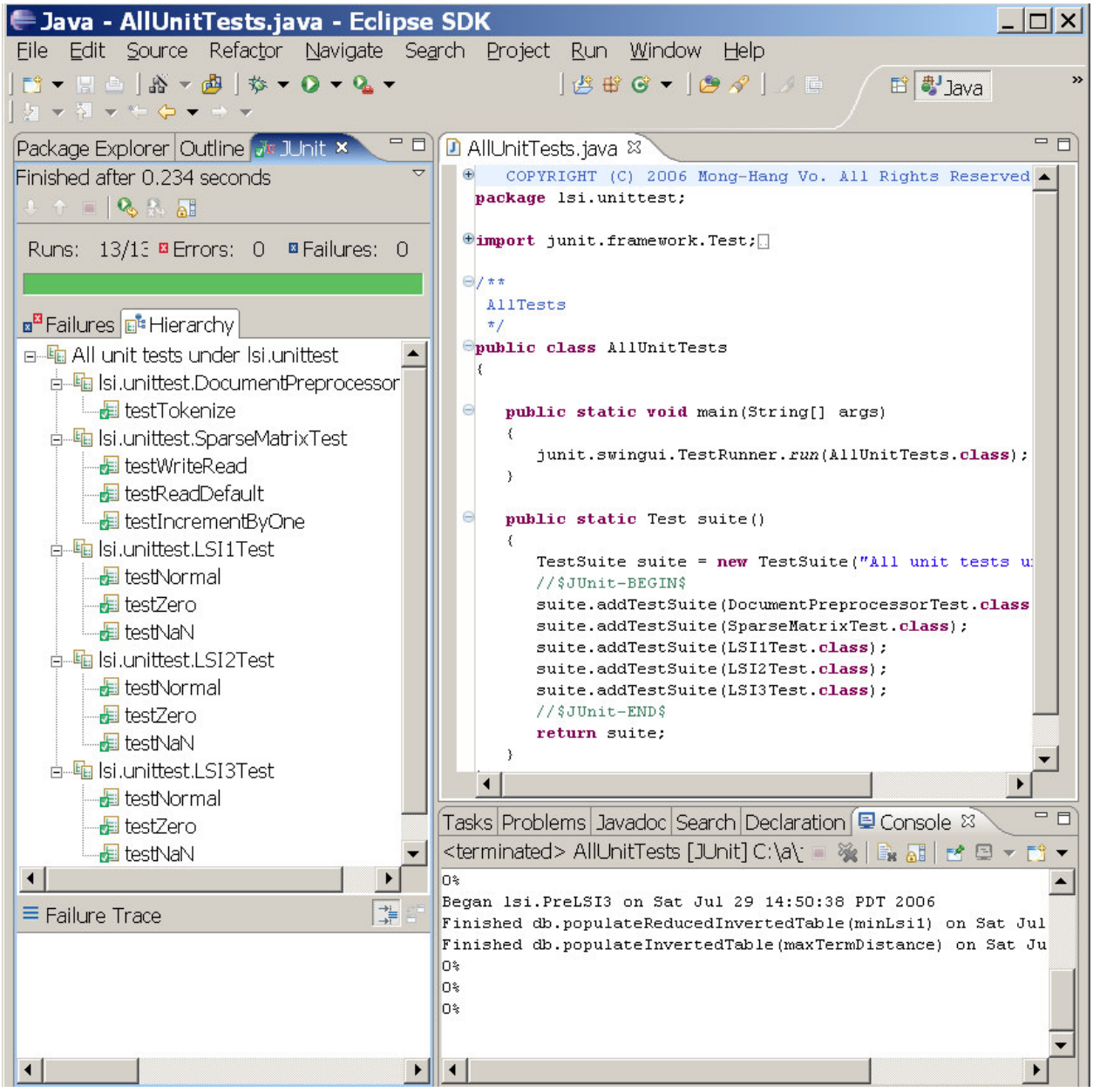

Figure 10. Unit Tests Developed under the JUnit Framework Executed by Eclipse

5.4 Program Run Dependencies

Although the author has implemented the LSI computation units up to 4 dimensions, the software system and framework can be extended to compute LSI of any number of dimensions. The DocumentPreprocessor and the single dimensional LSI (LSI1) computation have to be run only once.

Each of the higher dimensional LSI computations depends only on LSI1 and can be run in parallel independently. For any number of dimension *d*, PreLSI *d* must be run before LSI *d* because the PreLSI program produces the input for LSI *d* by reducing the size of the (conceptual) d-way Cartesian Product of the inverted table of LSI1. All unit tests have no dependencies.

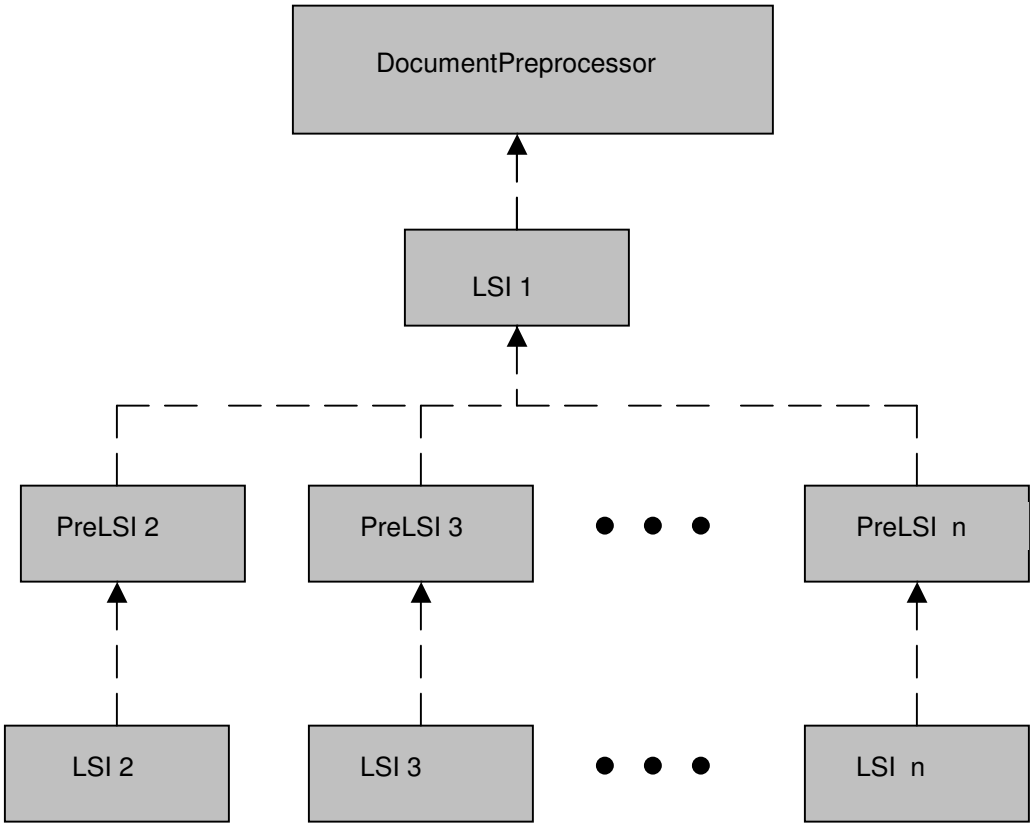

Figure 11. Program Run Dependencies

6. Analysis

Method 1, Method 2, and Method 3 have the same preprocessor step. The document preprocessor took 31 minutes to finish. The algorithm in Method 2 favors the short documents, and as a results, precision was severely impaired.

The author decided to remove short documents by implementing a Java utility (ClassifyLongAndShortDocs) that invokes the tokenizer of Lucene and removes the documents with less than 200 tokens. As a result, out of 16330 documents in UCI KDD, 5147 short documents were removed. To produce scientific results, all methods use the same set of 11183 documents. Before the author removed short documents, there were 6336032 rows in the inverted table in the Oracle database. After the author removed short documents, there were 5560671 rows in the inverted table in the Oracle database.

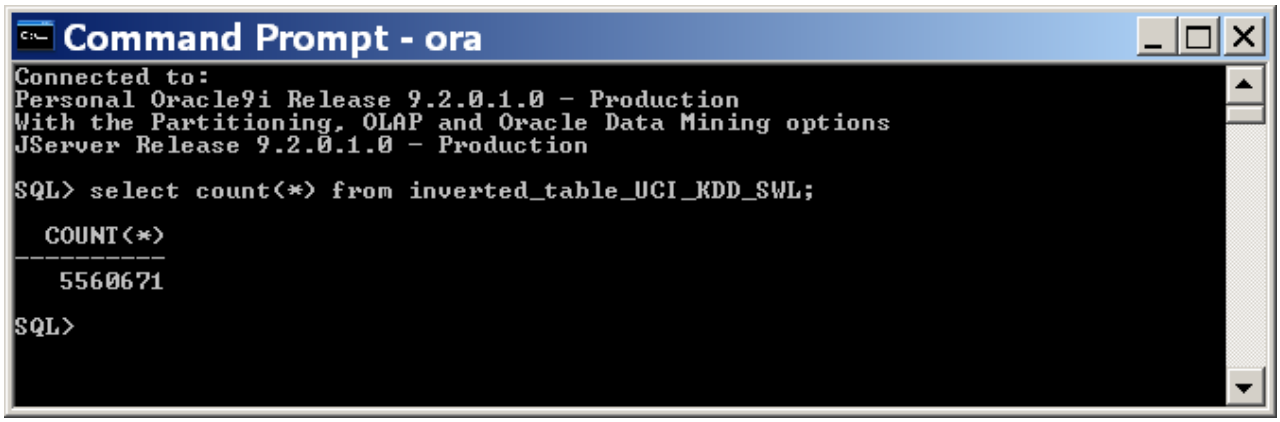

Figure 12. The Size of the Inverted Table Generated from Document Preprocessor

The corpus used in these experiment contains some uuencoded binary files, whose semantics cannot be captured by LSI. However, uuencoding is no longer popular nowadays. To reduce the effect of uuencoded binary files, the author examines only those tokens that have more than four characters because most tokens, such as "ax," in a typical uuencoded file are short. For all the methods and number of dimensions of LSI below, only tokens with more than four characters are analyzed.

6.1 Method 1 - Use TFIDF as a Threshold and N(ti, dj) as an Integer

In Method 1, N(ti, dj) is simply the number of times that token ti occurs in document dj.

6.1.1 LSI1

The author judged whether a term was significant in a document by examining the term in the context of the document. The 20 document-term pairs with the highest TFIDF were examined. In the cases where the author and the algorithm disagree, the author assumes that she is right and the algorithm is wrong.

As it turns out, the algorithm was right 18 times out of 20, which is 90%. In both of the error cases, the document was source code of a computer program. It took 30 minutes to finish the LSI1 computation. The following SQL query is used for selecting the results for analysis.

```
select * from lsi1_SWL_integer where length(term)>4 and lsi1 > 37.7 order by
lsi1 desc;
```

DOC	TERM	LSI1	SIGNIFICANT?	EXPLANATION
84286	elohim	51.923125	Yes	The document explores the question whether Robert

				Weiss is the only Orthodox Christian. It made many references to Elohim and Jehovah, where Elohim is the Father/God and Jehovah is the Son/Lord.
178571	stephanopoulo	45.460611	Yes	The document describes a press briefing by George Stephanopoulos.
76479	gayan	44.808429	Yes	The document is on the dispositions of three people, of which Gayane (Gaya) Vazgenovna Hakopian is one of them.
76479	zinaida	44.3846219	Yes	The document is on the dispositions of three people, of which Zinaida Poghosovna Hakopian is one of them.
14991	maxbyt	44.1652671	No	The document contains the assembly source code implementing the Lattice Gas based encryption algorithm. “MAXBYTE” is a symbolic constant that happens to occur many times in the document.
178314	stephanopoulo	43.9494823	Yes	The document describes a press briefing by George Stephanopoulos
179073	stephanopoulo	42.6479084	Yes	The document describes a press briefing by George Stephanopoulos
178898	reisman	41.0286224	Yes	The document describes Judith Reisman, who is prosecution's expert witness at the Mapplethorpe trial in Cincinnati.
59283	cesarean	40.7167397	Yes	This document is about Rates of Cesarean Delivery -- United States, 1991.
84286	mcconki	39.9085595	Yes	The article comments on many writings of Mcconki.
176936	bolshevik	39.8853719	Yes	The document is about American Bolshevik war.
59554	retinol	39.7085148	Yes	This document is about Vitamin A (Retinol) and infection.
84286	jehovah	39.5070241	Yes	The document is about God. The Father is "Jehovah".
176944	stephanopoulo	39.0801272	Yes	The document describes a press briefing by George Stephanopoulos.
84314	zarathushtra	39.0579779	Yes	The document is about ZARATHUSHTRA, founder of the religion know as Zoroastrianism or Mazdaism.
176936	falkland	38.5348479	Yes	The document describes "Falklands crisis".
38692	sphinx	38.1373116	Yes	This document is about SPHINX: Satellite Image Processing under X11. It is a subject of the email.
51151	enviroleagu	38.1373116	Yes	This document is about “EnviroLeague”, which is new youth movement. . It is a subject of the email.
66435	xclrp	37.703645	No	“Xclrp” is a variable in the C code.
83442	caligiuri	37.703645	Yes	This document is about David Caligiuri received one of The Advocate's homophobia rewaa awards: the A Prayer A Day Keeps the Lust Away citation.

Table 8. Method1: LSI1 Analysis

6.1.2 LSI2

The algorithm was right 17 times out of 20, which is 85%. In all three of the error cases, the document is a list of products, which is a semi-structured document. For instance, in one of such documents, the "version" field often just happens to come immediately before the "comment" field.

```
Version:    2.1
Comments:   General purpose, Notebook interface on Next, Mac,
            nice graphics.
```

As we can see, "version comment" is not a significant phrase in the document.

To reduce the size of the input to LSI2 computation, a threshold is set so that only terms with high enough TFIDF are used in LSI2 computation. The author chose 14 as the threshold in this case. It took 1 hour and 5 minutes to finish the computation of LSI2. The following SQL statement is used for reducing the size of the inverted table for LSI2 computation.

```
insert into reduced_inverted select distinct * from inverted_table where term
in (select term from lsi1 where lsi1 >= 10)
```

The following SQL query is used for selecting the results for analysis.

```
select * from lsi2_SWL_integer where length(term1)>4 and length(term2)>4 and
term1 != term2 and lsi2 > 35.0 order by lsi2 desc;
```

DOC	TERM1	TERM2	LSI2	SIGNIFICANT?	EXPLANATION
68012	window	microsoft	50.6268891	Yes	The document is on X Servers for DOS, Microsoft Windows, OS/2, etc.
176936	south	georgia	43.232274	Yes	The document is on the secret purpose of Falklands War, in which the military secret of South Georgia Island is significant.
39632	gamma	correct	42.9836686	Yes	The document is on gamma correction.
176960	senior	administr	42.5567375	Yes	The document is on a background briefing by senior administration officials.
176936	georgia	island	40.712587	Yes	The document is on the secret purpose of Falklands War, in which the military secret of South Georgia Island is significant.
54215	danger	ordnanc	38.5516976	Yes	The document is on Ohio House Bill 278, which expands the definition of dangerous ordnance.
68012	memori	mbyte	38.1373116	Yes	The document is on X Servers for DOS, Microsoft Windows, OS/2, etc. The phrase "Memory: ? Mbyetes" occurs many times indicating megabytes of memory are often a significant system requirement.
15590	version	comment	37.6927411	No	The document is a list of large integer arithmetic packages. The "version" field often just happens to come immediately before the "comment" field. "Version comment" is not a significant phrase in the

					document.
59125	smokeless	tobacco	37.6927411	Yes	The document is on various public health issues, one of which is the use of smokeless tobacco among adults.
176960	administr	offici	37.1385733	Yes	The document is on a background briefing by senior administration officials.
59126	cancer	center	37.1385733	Yes	The document is a health newsletter, in which NCI-Designated Cancer Centers are a significant topic.
15252	product	cipher	36.7431272	Yes	The document is an FAQ on product ciphers.
176936	rockefel	cartel	36.2666293	Yes	The document is on the secret purpose of Falklands War, in which Rockefeller cartel plays a significant role.
9956	paradox	engin	36.2666293	Yes	The document is on Borland/Microsoft database C libraries, in which the Paradox Engine is a major topic of discussion.
59283	cesarean	deliveri	35.73379	Yes	This document is about Rates of Cesarean Delivery.
68012	network	softwar	35.3090937	No	This document is on X Windows on the PC. The phase happens to be a field that repeats many times.
176936	secret	naval	35.1686381	Yes	This document on the secret purpose of Falklands War. Plan to unveil their secret weapons, especially their secret naval fleets.
59126	comprehens	cancer	35.1686381	Yes	This document describes comprehensive" cancer centers (28), which emphasize a Multidisciplinary approach to cancer research, patient care, and community outreach.
68012	price	latest	35.1686381	No	This document is on X Windows on the PC. The phase happens to be a field that repeats many times.
53663	ground	conductor	35.0166177	Yes	This document is about the equipment-grounding conductor.

Table 9. Method1: LSI2 Analysis

6.1.3 LSI3

The algorithm was correct 18 times out of 19, which is 95%. In the only error case, the “price” field often just happens to come immediately before the “latest version” field. An example of such is as follows:

```
Prices:
        $75.00
Latest Version:
        1.5.3
```

As we can see, “price latest version” is not a significant phrase in the document. The computation of LSI3 took about 1 hours and 11 minutes. The following SQL query is used for selecting the results for analysis.

```
select * from lsi3_SWL_integer where  length(term1)>4 and length(term2)>4 and
length(term3)>4 and term1 != term2 and term2 != term3 and term1 != term3 and
lsi3 > 27.4 order by lsi3 desc;
```

DOC	TERM1	TERM2	TERM3	LSI3	SIGNIFICANT?	EXPLANATION
176960	senior	administr	offici	42.5567375	Yes	The document is on a background briefing by senior administration officials.
176936	south	georgia	island	40.712587	Yes	The document is on the secret purpose of Falklands War, in which the military secret of South Georgia Island is significant.
59126	comprehens	cancer	center	35.1686381	Yes	The document is a health newsletter, in which comprehensive cancer centers are a significant topic.
68012	price	latest	version	35.1686381	No	The document is a list of platform-specific X servers. The “price” field often just happens to come immediately before the “latest version” field. “Price latest version” is not a significant phrase in the document.
53468	american	hockei	leagu	30.7871938	Yes	The document is an FAQ on hockey. The American Hockey League is significant.
10011	virtual	packet	driver	29.8050072	Yes	The document is on setting up a SLIP client under DOS and Windows, in which virtual packet drivers are significant.
59284	coronari	heart	diseas	29.8050072	Yes	The document is an FDA medical newsletter, in which coronary heart disease is a significant subject of discussion.
178918	holocaust	memori	council	29.320448	Yes	The document is on the U.S. Holocaust Memorial Museum, in which the Holocaust Memorial Council is significant
76071	holocaust	memori	council	29.320448	Yes	The document is on the U.S. Holocaust Memorial Museum, in

						which the Holocaust Memorial Council is significant
38658	sigkid	research	showcas	28.7070161	Yes	This document is about the SIGKids Research Showcase.
59207	kidnei	stone	format	28.7070161	Yes	This document describes how to prevent kidney stone formation.
61316	meteor	shower	maximum	28.7070161	Yes	This document is about the space calendar, which contains Meteor shower.
59323	experiment	doubl	blind	28.7070161	Yes	This document is about the Experimental Double-blind Study, "The efects of vitamin B6 Supplementation on premenstrual sysmptoms" Obstet.
61435	celsiu	degre	fahrenheit	28.4980162	Yes	This document is about solar system containing Celsius Fahrenheit and degrees.
176936	secret	naval	instal	27.4622164	Yes	This document on the secret purpose of Falklands War. Plan to unveil their secret weapons, especially their secret naval fleets.
76943	ghost	rider	appear	27.4622164	Yes	This document is about the comics (Ghost Rider).
61316	solar	longitud	degre	27.4622164	Yes	This document is about the space calendar, which contains solar longitude degrees.
52619	nilsson	calgari	flame	27.4622164	Yes	This document reports the stats of National Hockey League, and Kent Nilsson, Calgary Flames won many rounds.
61293	redesign	advisori	committe	27.4622164	Yes	This document is about a report on redesign team. Comment to Redesign Advisory Committee.

Table 10. Method1: LSI3 Analysis

6.1.4 LSI4

The algorithm was correct 16 times out of 20, which is 80%. There is no apparent common cause among the error cases. The computation of LSI4 took about 2 hours. The following SQL query is used for selecting the results for analysis.

```
select * from lsi4_SWL_integer where  length(term1)>4 and length(term2)>4 and
length(term3)>4 and length(term4)>4 and term1 != term2 and term2 != term3 and
term1 != term3 and term3 != term4 and lsi4 > 19.56 order by lsi4 desc;
```

DOC	TERM1	TERM2	TERM3	TERM4	LSI4	SIG?	EXPLANATION
59323	experiment	doubl	blind	studi	27.4622164	Yes	The document explores the question whether PMS can be prevented by a diet change. The experimental

							double-blind studies on various nutrients are important.
59207	prevent	kidnei	stone	format	26.0252007	Yes	The document is on how to prevent kidney stone formation.
178918	holocaust	memori	museum	newslett	24.0901005	Yes	The document describes a build problem of XView on SPARC Classic, in which the source file build/include/xview/notify.h has many compilation problems.
76071	holocaust	memori	museum	newslett	24.0901005	Yes	The document describes a build problem of XView on SPARC Classic, in which the source file build/include/xview/notify.h has many compilation problems.
104312	orang	counti	fairgnd	costa	22.2453941	Yes	The document is on the latest SoCal rides. Orange County Fairgnds, Costa Mesa. is a significant locat
67882	troubl	shoot	strang	error	22.2453941	Yes	The documents is an FAQ on OPEN LOOK GUI, and “Trouble Shooting: Strange Error Messages” is an important subject.
59435	huntington	medic	research	institut	22.2453941	Yes	The document is a press release from Huntington Medical Research Institutes.
60774	upper	atmospher	research	satellit	21.4588112	Yes	The document is on the ozone images taken from the Upper Atmosphere Research Satellite.
38778	siggraph	onlin	bibliographi	project	19.6237818	Yes	This document describes siggraph online bibliography project
10099	bjorn	myrland	sipaa	sintef	19.5635786	No	The phase is a part of email address in the email header.
178573	alcohol	cigarett	marijuana	cocain	19.5635786	Yes	This document is about “Drug Use Up At Younger Age”.
179054	foreign	intellig	advisori	board	19.5635786	Yes	The document describes a CLINTON: Press Briefing by Dee Dee Myers.
38658	sigkid	research	showcas	entri	19.5635786	Yes	This document is about the SIGKids Research Showcase.
54215	lawfulli	acquir	possess	carri	19.5635786	Yes	This document is about Ohio House Bill 278 (Sec.

							2923.1).
59122	highwai	traffic	safeti	administr	19.5635786	No	This document is about Medical Newsletter. The phase just occurs to be in the references.
76943	panther	havok	black	panther	19.5635786	Yes	This document is about comics.
76943	havok	black	panther	havok	19.5635786	Yes	This document is about comics.
76943	black	panther	havok	black	19.5635786	Yes	This document is about comics.
68012	higher	wollongong	pathwai	access	19.5635786	No	This document is on X Windows on the PC. The phase happens to be a field that repeats many times.
67107	graphic	displai	defaultscreen	graphic	19.5635786	No	This document describes how to get the actual size of memory for running computer programming. The phases are the parameters of the function.

Table 11. Method1: LSI4 Analysis

6.1.5 Dimensional Trends

Method 1 appears to produce very good precision, especially in higher dimensions. Moreover, its preciseness appears to be independent of the dimension. The method performs well no matter whether short documents are included in the analysis.

A potential drawback of Method 1 is that it seems to favor long documents. This is not an issue if the long document is full of content in the form of unstructured text because the latent semantic indexing works especially well when it has enough content to perform upon.

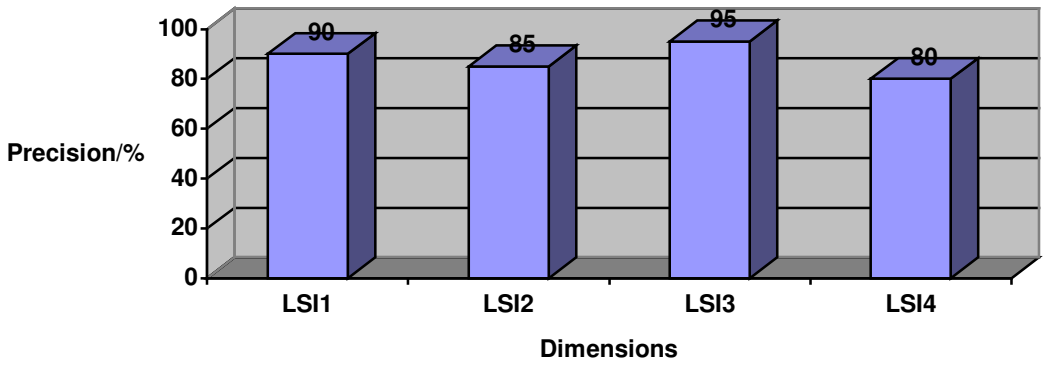

Figure 13. Dimensional Trends of Method 1

This long-document effect becomes an issue only when the long document is somewhat content-less. In the UCI KDD corpus, a common example of such a “content-less” long document is a uuencoded binary file. The corpus consists of Usenet newsgroup articles from the 80s and early 90s, during which it was a common practice to post uuencoded binary files on the Internet. Although these files have many tokens

when tokenized by Lucene, the LSI algorithm performs poorly in capturing the semantics or contents in them. Because (by the design of uuencoding scheme) uuencoded binary files appear to be ASCII to any algorithm, it poses a challenge to remove them from the corpus automatically. Fortunately, most tokens produced from uuencoded files are four-character or shorter. The author uses this four-character threshold to produce meaningful results from this experiment practically reducing the effect from uuencoded files.

While uuencoding has become less popular nowadays, another type of long documents will likely to continue to pose a challenge to LSI. They are semi-structured documents, which contain many repeated fields (name-value pairs with different values) embedding in ASCII files in non-standard ways. An example is as follows:

```
Prices:
        $75.00
Latest Version:
        1.5.3
```

In this example, because the combination "price latest version" just happens to occur many times in the documents, the TFIDF ends up to be very high from the LSI3 algorithm. However, "price latest version" is not a significant phrase in the document. The document was a list of platform-specific X servers. It remains a challenge to identify semi-structured documents or to improve the precision of LSI on them.

6.2 Method 2 - Use TFIDF as a Threshold and N(ti, dj) as a Fraction

Method 2 differs from Method 1 in that a denominator is introduced in calculation of N(ti, dj). In Method 2, the denominator is the total number of all types of tokens in document dj. The intent of introducing the denominator is to normalize N(ti, dj) so that long documents (those with many tokens) do not get higher values.

For TFIDF to meaningfully indicate the significance of a term in a document, both the TF and the IDF parts must have the same sign. For example, if TF was negative and IDF was positive, the product of TF and IDF would be meaningless. Since IDF is always non-negative, TF should be made non-negative too. With the denominator introduced in the calculation of N(ti, dj), TF (being 1 + log(N(ti, dj))) may be negative in some case. To solve this problem, a large enough constant coefficient is also introduced in the calculation of N(ti, dj). The author chose 40,000 as the coefficient because it is larger than the total number of tokens of the longest document.

6.2.1 LSI1

The algorithm was correct five out of 20 times, which is 25%. Most (14) of the errors are due to that the term is a part of an email address. If we removed email addresses from the documents (say by a regular expression), the algorithm would be correct 95% of the time ((14+5)/20 * 100% = 95%).

It took 18 minutes to finish the computation of LSI1. Let us examine the results.

```
select * from lsi1_UCI_KDD_SWL where length(term)>4 and lsi1 > 75.4 order by
lsi1 desc;
```

DOC	TERM	LSI1	SIGNIFICANT?	EXPLANATION
66435	xclrp	83.5038806	No	“Xclrp” is a variable in the C code.
60354	satam	78.7418875	No	“Satam” is a first name of the author and a part of email address.
38683	ilmenau	78.1819596	No	“Ilmenau” is a part of the email address.
60654	uswnvg	77.3382247	No	“Uswnvg” is a part of the email address.
68204	hardwarecolor	77.0815098	Yes	This document is about creating your own ColorMap,

				i.e. Lookup Table in X11 R4.
15927	anovak	76.8772605	No	“Anovak” is a part of the email address.
39620	dorsai	76.8229038	Yes	This document describes “dorsai”, which is a community-based service.
59059	spect	76.5953597	Yes	This document discusses questions about SPECT imaging.
59427	bracelet	76.3107344	Yes	This document is about Copper Bracelet.
53796	buhrow	76.1364835	No	“Buhrow” is a part of the email address and the author’s last name.
38289	ederveen	76.1268978	No	“Ederveen” is is an author’s last name and is a part of the email address.
60881	nsiad	76.0505645	Yes	This document is about NASP: Key Issues Facing the Program (31 Mar 92) GAO/T-NSIAD-92-26
67269	timessqr	76.0079001	No	“Timessqr” is a part of the email address.
52210	callan	75.8950739	No	“Callan” is an author’s last name and is a part of the email address.
51850	bucknel	75.8810662	No	“Bucknel” is an organization and is a part of the email address.
66987	ledoux	75.8810662	No	“Ledoux” is an author’s last name and is a part of the email address.
9961	frampton	75.7559348	No	“Fampton” is a part of the email address.
66950	savela	75.5916636	No	“Savela” is a part of the email address.
38326	cvtstu	75.5106009	No	“Cvtstu” is a part of the email address.
51497	kitchel	75.4753564	No	“Buhrow” is a part of the email address and the author’s lastname.

Table 12. Method2: LSI1 Analysis

6.2.2 LSI2

The algorithm was correct 7 out of 20 times, which is 35%. Most (12) of the errors are due to that the term is a part of an email address. If we removed email addresses from the documents (say by a regular expression), the algorithm would be correct 95% of the time ((12+7)/20 * 100% = 95%).

It took 1 hour and 14 minutes to finish the LSI2 computation. Let us examine the results.

```
select * from lsi2_UCI_KDD_SWL where length(term1)>4 and length(term2)>4 and
term1 != term2 and lsi2 > 84.5 order by lsi2 desc;
```

DOC	TERM1	TERM2	LSI2	SIGNIFICANT?	EXPLANATION
60563	luoma	binah	89.6383761	No	The phase is a part of the email address.
39632	gamma	correct	89.1723833	Yes	This document is about gamma correction.
59427	copper	bracelet	89.0597276	Yes	This document is about Copper Bracelet by the name of Sabona created by Dr. John Sorenson.
38653	mapsut	einstein	87.6227118	No	The phase is a part of path name and the email address.
38653	shmuel	einstein	87.6227118	No	The phase is the name of author and a part of the email address.
68085	riski	converg	87.3672945	No	The phase is a part of the email address.

58144	compart	syndrom	87.2173371	Yes	This document is about compartment syndrome - general information, references, etc. The phase is in the keyword search.
84068	jensen	peruvian	86.6405253	No	The phase is a part of the email address.
103434	battan	sequent	86.3348536	No	The phase is a part of the email address.
9975	instanc	handl	86.2749039	Yes	This document describes a module instance handle, HInstance.
38621	wisdom	attmail	86.0388875	No	The phase is a part of the email address.
51850	coral	bucknel	85.7520295	No	The phase is a part of the email address.
101610	steve	green	85.5840583	No	The phase is the author name and a part of the email address.
61015	stage	version	85.4190602	Yes	This document is about the Proton has been used in 2, 3, and 4 stage versions.
74727	small	claim	85.1371594	Yes	This document is about the small claims in the court.
77056	bitzm	columbia	85.0975749	No	The phase is a part of the email address.
20617	trade	unionist	84.9408964	Yes	This document is about the need help with "They came for the Jews" quote.
38935	gregori	winer	84.9408964	No	The phase is the author name.
9752	stephen	gibson	84.9408964	No	The phase is the author name and a part of the email address.
15276	deuelpm	craft	84.5603476	No	The phase is a part of the email address.

Table 13. Method2: LSI2 Analysis

6.2.3 LSI3

The algorithm was correct 14 out of 20 times, which is 70%. All 6 of the errors are due to that the term is a part of an email address or a path name in an email header. If we removed email addresses and header path names from the documents (say by a regular expression), the algorithm would be correct 100% of the time. The computation took about 2 hours. The following SQL query is used for selecting the results for analysis.

```
select * from lsi3_UCI_KDD_SWL where  length(term1)>4 and length(term2)>4 and
length(term3)>4 and term1 != term2 and term2 != term3 and term1 != term3 and
lsi3 > 90.1 order by lsi3 desc;
```

DOC	TERM1	TERM2	TERM3	LSI3	SIGNIFICANT?	EXPLANATION
60582	margin	drive	howev	95.182325	Yes	This document discusses DOS 6.0 and hard drive.
58100	immotil	cilia	syndrom	94.4361548	Yes	This document describes Immotile Cilia Syndrome.
59121	sbrun	oregon	uoregon	93.1021473	No	The phase is a part of the email address.
68277	server	window	hierarchi	93.1021473	Yes	This document is about XQueryTree, XGraberver, and robustness.
68277	custom	error	handler	93.1021473	Yes	This document discusses a BadWindow, an X protocol error.
104371	yanke	trade	kaminicki	91.4025184	Yes	This document mentions Yankees

						trade Kaminicki and Silvestri.
38342	decreas	speed	thank	91.4025184	Yes	This document discusses polygon orientation in DXF.
51303	electron	paper	trail	91.4025184	Yes	This document mentions “you leave an electronic paper trail on the net.”
51303	usual	theist	approach	91.4025184	Yes	This document discusses the usual theist approach.
53056	concert	ecsgat	tlcslip	91.4025184	No	The phase is a part of the path name in the email header.
67044	strip	chart	widget	91.4025184	Yes	This document is about an Athena strip chart widge. It includes it the summary.
75971	mildli	agress	justifi	91.4025184	Yes	This document mentions killing people. The is mildly agressive (justified, in your opinion).
59246	hidden	candida	infect	91.4025184	Yes	This document mentions hidden candida infections. This phase occurs 2 times in the document.
53056	tclark	tlcslip	uncec	91.4025184	No	The phase is a part of the email address.
53056	uvaarpa	concert	ecsgat	91.4025184	No	The phase is a part of the path name in the email header.
53056	ecsgat	tlcslip	uncec	91.4025184	No	The phase is a part of the path name in the email header.
52831	ubsil	msuvx	memst	91.4025184	No	The phase is a part of the email address.
51303	natur	argument	someon	91.4025184	Yes	This document mentions the "law of nature" argument someone posted recently.
179013	basic	pragmat	principl	90.4203319	Yes	This document discusses “a basic, pragmatic principle of day-to-day living". The phase occurs three times.
9975	modul	instanc	handl	90.3045273	Yes	This document is about module instance handle. The phase occurs four times.

Table 14. Method2: LSI3 Analysis

6.2.4 LSI4

The algorithm was correct 12 out of 20 times, which is 60%. Half of the errors are due to that the term is a part of an email address or a path name in an email header. If we removed email addresses and header path names from the documents (say by a regular expression), the algorithm would be correct 80% of the time. The computation took about 2 hours 21 minutes. The following SQL query is used for selecting the results for analysis.

```
select * from lsi4_UCI_KDD_SWL where  length(term1)>4 and length(term2)>4 and
length(term3)>4 and length(term4)>4 and term1 != term2 and term2 != term3 and
term1 != term3 and term3 != term4 and lsi4 > 97.8641 order by lsi4 desc;
```

DOC	TERM1	TERM2	TERM3	TERM4	LSI4	SIGNIFICANT	EXPLANATION
51942	discuss	alreadi	pleas	excus	108.105569	No	The phase appears in the P.S in the document.
61027	softwar	develop	group	survei	108.105569	Yes	This document discusses 90% of the software development groups surveyed were at level 1.
50527	travi	grundk	macgam	digest	101.643947	No	The phase combined author's name and source of information
51302	extraordinari	claim	requir	extraordinari	101.643947	Yes	"Extraordinary claims require extraordinary evidence." Included
53056	concert	ecsgat	tlcslip	uncec	101.643947	No	The phase is a part of the path name in the email header.
59242	discuss	prescript	strength	although	101.643947	Yes	This document mentions discussed prescription strength.
83917	earli	christian	perhap	second	101.643947	Yes	The subject is about Ancient references to Christianity.
59575	submarin	grant	aquariu	rosemount	101.643947	No	The phase is the concatenation of the email address and a quote.
53877	board	decoupl	capacitor	insid	101.643947	Yes	The subject is about decoupling caps – onboard.
53056	uvaarpa	concert	ecsgat	tlcslip	101.643947	No	The phase is a part of the path name in the email header.
58976	least	intrus	orthoscop	method	100.067394	Yes	"The hernia was repaired using the least intrusive (orthoscopic?) method" is used.

102588	discuss	basebal	salari	addit	97.8641404	No	This document is about brewers injuries related with baseball salaries.
102606	outstand	predict	record	overal	97.8641404	Yes	“Mike Francesa has an *outstanding* prediction record” discussed.
104282	gatewai	mavenri	altcit	eskimo	97.8641404	No	The phase is a part of the email address.
104373	great	acquisit	decent	offens	97.8641404	Yes	“Mark Whiten was a great acquisition... decent offense and great defense in right field” discussed.
104674	frank	thoma	david	paschich	97.8641404	No	The phase combines the names of two people.
15353	besid	effect	tempest	shield	97.8641404	Yes	The document discusses effective TEMPEST-shielding.
15353	equip	besid	effect	tempest	97.8641404	Yes	The document discusses effective TEMPEST-shielding.
176946	homosexu	child	molest	simpli	97.8641404	Yes	Homosexual = child molester.
176946	sexual	orient	mortal	netcom	97.8641404	Yes	The document discusses “sexual orientation”. The phase combines with a part of the email address.

Table 15. Method2: LSI4 Analysis

6.2.5 Dimensional Trends

In LSI1 and LSI2, the precision of the algorithm is impaired by the fact that the introduction of the denominator in the calculation of N(ti, dj) favors short documents. Nevertheless, an interesting observation is that as the dimension goes higher, this adverse short-document effect becomes less pronounced. In shorter documents, the algorithm is more likely to be misled by tokens in email addresses and email header paths, which do not usually contribute to the main content of the document.

The results presented here came from the analysis of only the long documents (those with more than 200 tokens).

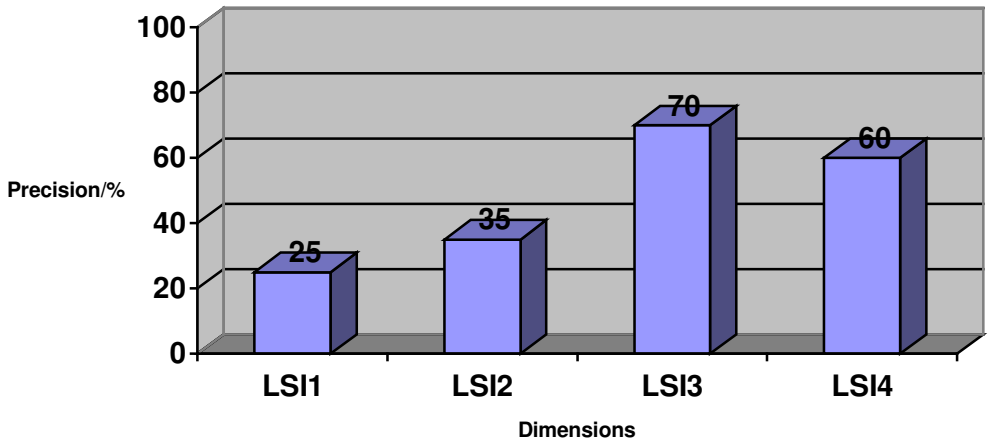

Figure 14. Dimensional Trends of Method 2

6.3 Method 3 - Use Document Frequency as a Threshold and N(ti, dj) as a Fraction

Method 3 differs from Method 2 only on how the input sizes of multi-dimensional LSI calculation.

6.3.1 LSI1

Since it is not possible to put a threshold to reduce the size of input to LSI1 calculation, Method 3 is the same as Method 2 for LSI1. The algorithm was correct five out of 20 times, which is 25%. It took 19 minutes to finish the computation of LSI1. The following SQL query is used for selecting the results for analysis.

```
select * from lsi1_TFDF_SWL where  length(term)>4 and lsi1 > 75.5 order by
lsi1 desc;
```

DOC	TERM	TF	DF	LSI1	SIGNIFICANT?	EXPLANATION
66435	xclrp	.000089421	8.9575774	83.5038806	No	“Xclrp” is a variable in the C code.
60354	satam	.000089421	8.44675178	78.7418875	No	“Satam” is a first name of the author and a part of email address.
38683	ilmenau	.000089421	8.38668754	78.1819596	No	“Ilmenau” is a part of the email address.
60654	uswnvg	.000089421	8.29617892	77.3382247	No	“Uswnvg” is a part of the email address.
68204	hardwarecolor	.000089421	8.26864076	77.0815098	Yes	This document is about creating your own ColorMap,
15927	anovak	.000089421	8.24673065	76.8772605	No	“anovak” is a part of the email address.
39620	dorsai	.000089421	8.24089973	76.8229038	Yes	This document describes “dorsai”, a community service
59059	spect	.000089421	8.21649076	76.5953597	Yes	This document discusses questions about SPECT imaging.

59427	bracelet	.000089421	8.18595861	76.3107344	Yes	This document is about Copper Bracelet.
53796	buhrow	.000089421	8.16726647	76.1364835	No	“Buhrow” is a part of the email address and the author’s last name.
38289	ederveen	.000089421	8.1662382	76.1268978	No	“Ederveen” is is an author’s last name and is a part of the email address.
60881	nsiad	.000089421	8.15804982	76.0505645	Yes	This document is about NASP: Key Issues facing the program GAO/T-NSIAD-92-26
67269	timessqr	.000089421	8.15347315	76.0079001	No	“Timessqr” is a part of the email address.
52210	callan	.000089421	8.14137013	75.8950739	No	“Callan” is an author’s last name and is a part of the email address.
51850	bucknel	.000089421	8.1398675	75.8810662	No	“Bucknel” is an organization and is a part of the email address.
66987	ledoux	.000089421	8.1398675	75.8810662	No	“Ledoux” is an author’s last name and is a part of the email address.
9961	frampton	.000089421	8.12644448	75.7559348	No	“Fampton” is a part of the email address.
66950	savela	.000089421	8.10882288	75.5916636	No	“Savela” is a part of the email address.
38326	cvtstu	.000089421	8.10012717	75.5106009	No	“Cvtstu” is a part of the email address.

Table 16. Method3: LSI1 Analysis

6.3.2 LSI2

The algorithm was correct seven out of 19 times, which is 37%. All of the errors are due to that the term is a part of an email address or a path name in an email header. If we removed email addresses and header path names from the documents (say by a regular expression), the algorithm would be correct 100% of the time. The computation took two hours and 12 minutes.

Method 3 uses document frequency (DF) as the threshold to pick up only the terms with high enough DF to feed into the computation of LSI2 to reduce its input size. The author chose 8.0 as the threshold. It took 2 hours and 12 minutes to complete the computation of LSI2. The following is the SQL statement that reduces the size of an inverted table by Method 3.

```
insert into reduced_inverted select distinct * from inverted_table where term
in (select term from lsi1 where df > min_df)
```

where *min_df* is 8.0.

The following SQL query is used for selecting the results for analysis.

```
select * from lsi2_TFDF_SWL where length(term1)>4 and length(term2)>4 and
term1 != term2 and lsi2 > 81.5 order by lsi2 desc;
```

DOC	TERM1	TERM2	DF	LSI2	SIGNIFICANT?	EXPLANATION
59427	copper	bracelet	9.22505677	85.9973634	Yes	This document is about Copper Bracelet by the name of Sabona created by Dr. John Sorenson.
60563	luoma	binah	9.1542877	85.3376435	No	The phase is a part of the email address.
39632	gamma	correct	9.14842945	85.283032	Yes	This document is about gamma correction.
58144	compart	syndrom	9.09519878	84.7868077	Yes	This document is about compartment syndrome.
38653	mapsut	einstein	9.04718956	84.3392586	No	The phase is a part of path name and the email address.
38653	shmuel	einstein	9.04718956	84.3392586	No	The phase is the name of author and a part of the email address.
68085	riski	converg	8.93987783	83.3388825	No	The phase is a part of the email address.
101610	steve	green	8.9156132	83.1126841	No	The phase is the author name and a part of the email address.
9975	instanc	handl	8.90200755	82.9858501	Yes	This document describes a module instance handle, HInstance.
53796	moria	nfbcal	8.87205523	82.7066301	No	The phase is the author name and a part of the email address
53665	black	demon	8.84936382	82.4950973	No	The phase is a part of the email address.
77056	bitzm	columbia	8.83662479	82.3763422	No	The phase is a part of the email address.
10099	bjorn	myrland	8.83451731	82.356696	No	The phase is the author name and a part of the email address.
9752	stephen	gibson	8.81162349	82.1432763	No	The phase is the author name and a part of the email address.
51732	meridian	demon	8.79327435	81.9722229	No	The phase is a part of the path name in the email header.
61015	stage	version	8.78723204	81.9158956	Yes	This document is about the Proton has been used in 2, 3, and 4 stage versions.
60866	space	clipper	8.78037095	81.8519354	Yes	The subject is about Space Clipper launch article.
74727	small	claim	8.77823647	81.8320376	Yes	This document is about the small claims in the court.
51607	adsdesign	analog	8.7440033	81.5129108	No	The phase is a part of the email address.

Table 17. Method3: LSI2 Analysis

6.3.3 LSI3

The algorithm was correct eight out of 18 times, which is 44%. The computation of LSI3 took about two hours and 22 minutes. The following SQL query is used for selecting the results for analysis.

```
select * from lsi3_TFDF_SWL where  length(term1)>4 and length(term2)>4 and
length(term3)>4 and term1 != term2 and term2 != term3 and term1 != term3 and
lsi3 > 81.3 order by lsi3 desc;
```

DOC	TERM1	TERM2	TERM3	DF	LSI3	SIGNIFICANT?	EXPLANATION
38497	nation	univers	canberra	9.43715048	87.9745328	No	The phase is a part of an organization
59023	diseas	exist	david	9.39941016	87.6227118	Yes	This document describes Candida Albicans disease. The phase combines it and author' s first name.
67107	displai	graphic	window	9.19873946	85.7520295	No	This document describes how to get the actual size of memory for running computer programming.The phases are the parameters of the function.
38279	engin	research	institut	9.1762666	85.5425342	Yes	This document is about the job. System Engineering Research Institute is looking for resumes.
15464	system	perform	group	8.99394505	83.8429052	No	The phase is about the organization.
59471	comput	scienc	nation	8.9575774	83.5038806	No	The phase is the department where author works at.
74784	histori	japanes	languag	8.9575774	83.5038806	Yes	This document is about the books with different subjects.
59471	scienc	nation	univers	8.9575774	83.5038806	No	The phase combines the department and University where author works at.
104405	color	stori	bradlei	8.88858453	82.8607187	No	This article is about Tribune baseball and New York Times.
53534	organ	harri	control	8.88858453	82.8607187	No	The phase is the organization, Harris Controls.
39009	sound	effect	music	8.88858453	82.8607187	Yes	The document is about giant software yard sales.

104925	georgetown	univers	washington	8.85579471	82.5550471	No	The phase is the name of an organization.
77277	kevin	cursor	demon	8.84509942	82.455344	No	The phase is a part of the email address.
105024	sport	basebal	organ	8.82404601	82.2590809	No	The phase is a part of the “Followup To:” in the email header.
59595	water	current	brian	8.76342139	81.6939291	Yes	This document mentions water current.
62386	space	organ	thoma	8.76342139	81.6939291	Yes	This document is about the Soyuz and Shuttle Comparisons.
75364	dream	about	islam	8.76342139	81.6939291	Yes	This document is about ISLAM borders.
62394	henri	spencer	would	8.76342139	81.6939291	Yes	This document is about who the Henri Spencer is.

Table 18. Method3: LSI3 Analysis

6.3.4 LSI4

The algorithm was correct six out of 20 times, which is 30%. Most (11) of the errors are due to that the phrases are the names of the organizations of the authors. The computation took two hours and 55 minutes to complete the computation of LSI4. The following SQL query is used for selecting the results for analysis.

```
select * from lsi4_TFDF_SWL where  length(term1)>4 and length(term2)>4 and
length(term3)>4 and length(term4)>4 and term1 != term2 and term2 != term3 and
term1 != term3 and term3 != term4 and lsi4 > 82.5 order by lsi4 desc;
```

DOC	TERM1	TERM2	TERM3	TERM4	DF	LSI4	SIG?	EXPLANATION
61154	henri	spencer	write	pluto	10.2103404	95.182325	Yes	This document is about space news with a discussion of Pluto dwarf planet.
38279	system	engin	research	hinstitut	9.76405327	91.0219696	Yes	This article is about Job opportunity with SERI (Systems Engineering Research Institute.)
104375	comput	scienc	engin	demer	9.65072458	89.9655026	No	The phase concatenates the department and user name in the email address.
84353	organ	montana	state	univers	9.65072458	89.9655026	No	The phase is the

								organization.
105564	organ	oregon	state	system	9.65072458	89.9655026	No	The phase is a part of the organization.
53598	receiv	system	organ	northeastern	9.39941016	87.6227118	Yes	This document discusses receiver system.
59471	comput	scienc	nation	univers	9.39941016	87.6227118	No	The phase is about computer science at National University, not about hives.
61216	distribut	comput	group	stanford	9.29404964	86.6405253	No	The phase is about the Distributed Computing Group at Stanford, not about computer cult.
102651	system	organ	indiana	univers	9.11172808	84.9408964	No	The phase is a part of the organization.
53553	system	organ	laurentian	univers	9.11172808	84.9408964	No	The phase is the organization.
54022	programm	organ	auspex	system	9.11172808	84.9408964	No	The phase is a part of the organization.
59013	organ	princeton	univers	distribut	9.11172808	84.9408964	No	The phase is a part of the organization.
178867	comput	scienc	engin	univers	9.03168538	84.1947262	No	The phase is a part of the organization.
9882	window	printer	driver	ashok	9.03168538	84.1947262	Yes	This document is about WinQVT/Net V3.4, which uses standard Windows printer drivers.
9882	standard	window	printer	driver	9.03168538	84.1947262	Yes	This document is about WinQVT/Net V3.4, which uses standard Windows printer drivers.
74805	system	organ	harvard	univers	9.03168538	84.1947262	No	The phase is a part of the organization.

39083	graphic	organ	templ	univers	8.9575774	83.5038806	No	The phase is a part of the organization.
75395	jewish	problem	serdar	argic	8.9575774	83.5038806	Yes	The document discusses Jewish problems.
60461	mcgill	univers	comput	scienc	9.65072458	83.2761301	No	The phase consists of university's name and CS department.
104925	organ	georgetown	univers	washington	8.88858453	82.8607187	No	The phase is an organization name.

Table 19. Method3: LSI4 Analysis

6.3.5 Dimensional Trends

Since it is not possible to put a threshold to reduce the size of input to LSI1 calculation, Method 3 is the same as Method 2 for LSI1. From LSI2 on, DF is used as the threshold instead of TFIDF for the purpose of reducing the size of the input to the algorithms. As shown in the chart, the precision does not seem to improve in the multidimensional cases.

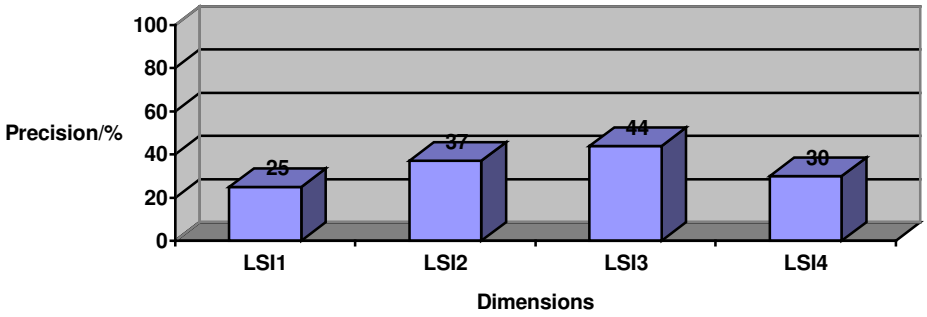

Figure 15. Dimensional Trends of Method 3

Among the many cases in which the algorithms made mistakes in the multidimensional cases, most multi-token phrases are pertinent to the authors, such as their name, email, and organization. These phrases are usually not significant in their documents in terms of content. DF is a good indicator of how prolific an individual or an organization is because the more documents that they appear, the higher the value of DF. However, TFIDF is a better indicator of the significance of a phrase in the documents where it occurs. This explains why TFIDF may be a better threshold than DF for reducing the size of the input in the multidimensional cases, as we compare the precision results of Method 3 with those of Methods 1 and 2.

6.4 Method 4 – Refined Method 2 by Removing Stop Words during Document Preprocessing

Method 4 differs from Method 2 in that it discards common function words (sometimes known as stop words) during preprocessing. The stop words are specified in a stop list, which is a text file downloaded from the WordNet Web site [W06]. The purpose of removing stop words is to reduce the size of the inverted table for LSI1 computation. Stop words, such as "a" and "the," often occur frequently in many

documents, but they usually do not contribute much to the content or the meaning. Therefore, it is often believed that removing stop words does not affect significantly the result of statistical content analysis while gaining some performance in the execution time. The author chose the stop analyzer of Lucene for removing stop words. In method 4, the document preprocessor takes 18 minutes to complete. There are 4175172 rows in the inverted table in the Oracle database.

6.4.1 LSI1

The algorithm was correct four out of 20 times, which is 20%. Most (15) of the errors are due to that the term is a part of an email address. If we removed email addresses from the documents (say by a regular expression), the algorithm would be correct 95% of the time. The computation of LSI1 took about 16 minutes. The following SQL query is used for selecting the results for analysis.

```
select * from lsi1_UCI_KDD_LDOC where length(term)>4 and lsi1 > 78.61 order
by lsi1 desc;
```

DOC	TERM	LSI1	SIGNIFICANT	EXPLANATION
66435	xclrp	85.1199559	No	“Xclrp” is a variable in the C code.
60354	satam	80.9712608	No	“Satam” is a first name of the author and a part of email address.
15927	anovak	80.8191165	No	“Anovak” is a part of the email address.
60654	uswnvg	80.8191165	No	“Uswnvg” is a part of the email address.
38683	ilmenau	80.4020113	No	“Ilmenau” is a part of the email address
59059	spect	79.8653086	Yes	This document discusses questions about SPECT imaging.
39620	dorsai	79.8298086	Yes	This document describes “dorsai”, which is a community-based service.
53880	traider	79.8045333	No	“Traider” appears seven times in this document. It is a part of the email address and the name of an organization.
38779	cogno	79.7441489	No	“Cogno” occurs seven times in this document. It is a part of the email address and the name of an organization.
53796	buhrow	79.6841504	No	“Buhrow” is a part of the email address and the author’s last name.
51850	bucknel	79.447918	No	“Bucknel” appears seven times in this document. It is a part of the email address and the name of an organization.
38308	talluri	79.2174834	No	“Taluri” occurs seven times in this document. It is a part of the email address and the name of an organization.
59427	bracelet	79.1043526	Yes	This document is about Copper Bracelet.
52210	callan	79.0918662	No	“Callan” occurs 9 times in this document. It is a part of the email address and the name of an organization.
38289	ederveen	78.9617523	No	“Ederveen” is is an author’s last name and is a part of the email address.
68204	hardwarecolor	78.8885621	Yes	This document is about creating your own ColorMap, i.e. Lookup Table in X11 R4.
52100	heurikon	78.791025	No	“Heurikon” appears 9 times in this document. It is a part of the email address and the name of an organization.
51989	pinghua	78.7187959	No	“Pinghua” occurs seven times in this document. It is a part of the email address and the name of an

				organization.
67269	timessqr	78.6649863	No	“Timessqr” is a part of the email address.
59059	eliez	78.6114834	No	“Elier” appears seven times in this document. It is a part of the email address and the name of an organization.

Table 20. Method4: LSI1 Analysis

6.4.2 LSI2

The algorithm was correct 5 out of 20 times, which is 25%. All of the errors are due to that the term is a part of an email address or a path name in an email header. The author chose 10 as the threshold. It took one hour and 51 minutes to finish the computation of LSI2. The following SQL query is used for selecting the results for analysis.

```
select * from lsi2_UCI_KDD_LDOC where length(term1)>4 and length(term2)>4 and
term1 != term2 and lsi2 > 80.9 order by lsi2 desc;
```

DOC	TERM1	TERM2	LSI2	SIGNIFICANT?	EXPLANATION
68048	nordic	offshor	86.3737811	No	The phase is a name of the company and a part of the email address.
59427	copper	bracelet	84.8051865	Yes	This document is about Copper Bracelet by the name of Sabona created by Dr. John Sorenson.
10068	acadvm	uottawa	83.1536187	No	The phase is a part of path name and the email address.
84013	danwel	iastat	83.0531927	No	The phase is a part of path name and the email address.
58144	compart	syndrom	82.8555066	Yes	The document is about compartment syndrome - general information, references, etc.
9975	instanc	handl	82.7388624	Yes	This document describes a module instance handle, hInstance.
38653	mapsut	einstein	82.3973796	No	The phase is a part of path name and the email address.
38653	shmuel	einstein	82.3973796	No	The phase is the name of author and a part of the email address.
104697	tkevan	eplrx	82.1491782	No	The phase is a part of path name and the email address.
10838	georg	marengo	81.9605687	No	The phase is a part of path name.
60453	rebox	berlin	81.6200387	No	The phase is a part of path name and the organization.
68085	riski	converg	81.3662465	No	The phase is a part of the email address.
74727	small	claim	81.3662465	Yes	This document is about the small claims in the court.
58896	whole	blood	81.352346	Yes	The document is about Blood Glucose test strips.
101610	steve	green	81.3412405	No	The phase is the author name and a part of the email address.
51732	meridian	demon	81.3246067	No	The phase is a part of the email address.
53796	moria	nfbcal	81.1191338	No	The phase is a part of path name and the email address.

53665	black	demon	81.0462505	No	The phase is a part of the email address.
66962	pilgrim	umass	80.9979741	No	The phase is a part of the project name and email address.
10099	bjorn	myrland	80.918061	No	The phase is the name of author and a part of the email address.

Table 21. Method4: LSI2 Analysis

6.4.3 LSI3

The algorithm was correct 13 out of 20 times, which is 65%. All of the errors are due to that the term is a part of an email address or a path name in an email header. The computation of LSI3 took about two hours and 38 minutes. The following SQL query is used for selecting the results for analysis.

```
select * from lsi3_UCI_KDD_LDOC where  length(term1)>4 and length(term2)>4
and length(term3)>4 and term1 != term2 and term2 != term3 and term1 != term3
and lsi3 > 81.4 order by lsi3 desc;
```

DOC	TERM1	TERM2	TERM3	LSI3	SIGNIFICANT?	EXPLANATION
66451	implement	pointer	featur	85.1734588	Yes	This document discusses a pointer feature in Xlib.
52324	warren	laplac	biologi	83.7798334	No	The phase is part of the email address.
60878	convent	explos	proof	83.4615066	Yes	The document discusses ORION test film, which used conventional explosives as a proof-of-concept test, or another one?
9975	modul	instanc	handl	83.4615066	Yes	This document describes a module instance handle, HInstance.
60878	explos	proof	concept	83.4615066	Yes	The document discusses ORION test film, which used conventional explosives as a proof-of-concept test, or another one?
53194	gener	capabl	overpow	83.1536187	Yes	This document is about political atheists; all humans are generally capable of overpowering their instincts.
54067	close	caption	decod	82.8555066	Yes	This document is indeed about a telecaption decoder module.
54067	telecapt	decod	modul	82.8555066	Yes	This document is indeed about a telecaption decoder module.
59200	adren	gland	cortic	82.6618969	Yes	This document is about a rat cell line of adrenal gland / cortical cell type.
58100	immotil	cilia	syndrom	82.4721987	Yes	This document is indeed about immotile cilia syndrome. The phase occurs three times in the document.
51604	built	modem	bundl	82.2862571	Yes	This document is indeed about Apple machines, which have built-in modems and bundled software.
51604	modem	bundl	softwar	82.2862571	Yes	This document is indeed about Apple machines, which have built-in modems and bundled software.

51168	anthonyp	riscsm	scripp	82.2399358	No	The phase is a part of the email address. It occurs two times in the document.
82770	anthonyp	riscsm	scripp	82.2399358	No	The phase is a part of the email address. It occurs two times in the document.
61450	devdjn	space	alcbel	82.1039263	No	The phase is a part of the email address. It occurs three times in the document.
52820	gleasokr	rintintin	colorado	82.0140717	No	The phase is a part of the email address. It occurs two times in the document.
51295	measur	effect	realiti	81.9250687	Yes	This document is about God; "beyond measurement means it can have no measurable effect on reality".
67044	strip	chart	widget	81.5772608	Yes	This document is about how can the author forces an Athena strip chart to update.
67567	changj	qucdn	queensu	81.5772608	No	The phase is a part of the email address. It occurs three times in the document.
60453	sreck	rebox	berlin	81.4080717	No	The phase is a part of the email address. It occurs three times in the document.

Table 22. Method4: LSI3 Analysis

6.4.4 LSI4

The algorithm was correct 15 out of 20 times, which is 75%. Most (4) of the errors are due to that the term is a part of an email address or a path name in an email header. The computation of LSI4 took about three hours and 52 minutes. The following SQL query is used for selecting the results for analysis.

```
select * from lsi4_UCI_KDD_LDOC where  length(term1)>4 and length(term2)>4
and length(term3)>4 and length(term4)>4 and term1 != term2 and term2 != term3
and term1 != term3 and term3 != term4 and lsi4 > 84.8 order by lsi4 desc;
```

DOC	TERM1	TERM2	TERM3	TERM4	LSI4	SIGNIFICANT?	EXPLANATION
60878	convent	explos	proof	concept	93.7771915	Yes	The document discusses ORION test film, which used conventional explosives as a proof-of-concept test, or another one?
51604	built	modem	bundl	softwar	89.3639791	Yes	The document is indeed about Apple machines, which have built-in modems and bundled software
51168	anthoni	pelleti	anthonyp	riscsm	88.795183	No	The phase consists of author's first and last name and a part of the email address.
51168	pelleti	anthonyp	riscsm	scripp	88.795183	No	The phase consists of author's last name and a part of the email address.
82770	pelleti	anthonyp	riscsm	scripp	88.795183	No	The phase consists of author's last name and a part of the email address.

82770	anthoni	pelleti	anthonyp	riscsm	88.795183	No	The phase consists of author's first and last name and a part of the email address.
60835	eugen	mallov	gregori	matloff	88.2580269	Yes	The document is about an excellent reference on ORION system - the handbook published by Eugene Mallove and Gregory Matloff.
60835	handbook	eugen	mallov	gregori	88.2580269	Yes	The document is about an excellent reference on the ORION system - the handbook published by Eugene Mallove and Gregory Matloff.
84068	interest	spread	toler	pleas	88.2580269	Yes	The document is about experiences with Mormons; the author does this "in the interest of spreading tolerance, so please, no flames."
60835	starflight	handbook	eugen	mallov	88.2580269	Yes	The document discusses "The Starflight Handbook", by Eugene Mallove and Gregory Matloff.
60835	technic	reader	orion	system	88.2580269	Yes	The document is about an excellent reference for non-technical readers on the ORION system.
176933	theodor	kaldi	wrote	enter	87.268719	No	The phase consist of the author name ,Theodore Kaldis, and the first sentence that he wrote "When I entered 1st grade, ..."
9703	humbl	opinion	power	access	87.268719	Yes	The document is about Borland's Paradox Offer with author's opinion "in my humble opinion, more powerful than Access."
20559	disagr	christian	resurrect	christ	85.5567668	Yes	The document is about religion.
20559	therefor	immedi	useless	doesn	85.5567668	Yes	The document is about religion.
51204	prove	wrong	illiad	contain	85.5567668	Yes	The document is about a discussion of God; the Illiad is the word of God.
51539	centri	quadra	machin	mention	85.5567668	Yes	The document mentions the new centris and quadra machines, which had ROM accelerated video.

51204	matter	prove	wrong	illiad	85.5567668	Yes	The document is about a discussion of God; the Illiad is the word of God.
104371	trade	kaminicki	silvestri	seattl	84.8051865	Yes	The author of this document thinking of why don't the Yankees trade Kaminicki and Silvestri to Seattle for Ken Griffey Jr and Randy Johnson...
52291	directli	floppi	drive	haven	84.8051865	Yes	The document is about author's opnion on Centris 610; the power switch is directly under the floppy dirve.

Table 23. Method4: LSI4 Analysis

6.4.5 Dimensional Trends

The results are similar to those from Method 2. The removal of stop words does not significantly affect the precision of LSI.

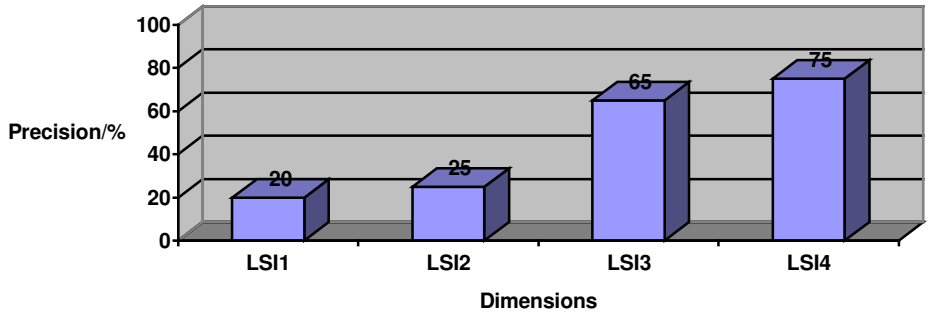

Figure 16. Dimensional Trends of Method 4

7. Conclusion

From the results of a straightforward computation of latent semantic indexing (in Method 1), the author discovered that the prospect of useful and meaningful extension of LSI to higher dimensions is promising. A challenge is posed by long documents whose content cannot be captured by the LSI algorithm. An important example is semi-structured documents.

To explore the possibility and practicality of normalizing the LSI computation against the length of documents, the author explored that idea of introducing the total number of tokens in a document as the denominator when calculating N(ti, dj) (Method 2). Although the results were disappointing for one and two dimensions, some prospect was shown in higher dimensions. Method 4 is a variant of Method 2 in which stop words were removed during the document preprocessing. The effect on precision is not significant.

The author also explored the method of using document frequency (DF) instead of TFIDF as a threshold to limit the size of the input to HD-LSI computation. The precision gets worse as the number of dimensions gets higher. This is probably because TFIDF is a better significance indicator than DF.

The invention of this project is to extend LSI to higher dimensions. The analysis of the research reveals the strengths and weakness of each approach to make the computation of HD-LSI tractable.

8. Appendices

8.1 Appendix A – Stop List

The following list of 199 stop words was downloaded from WordNet [W06].

```
a aboard about above across after against all along alongside although amid
amidst among amongst an and another anti any anybody anyone anything around
as astride at aught bar barring because before behind below beneath beside
besides between beyond both but by circa concerning considering despite down
during each either enough everybody everyone except excepting excluding few
fewer following for from he her hers herself him himself his hisself i idem
if ilk in including inside into it its itself like many me mine minus more
most myself naught near neither nobody none nor nothing notwithstanding of
off on oneself onto opposite or other otherwise our ourself ourselves outside
over own past pending per plus regarding round save self several she since so
some somebody someone something somewhat such suchlike sundry than that the
thee theirs them themselves there they thine this thou though through
throughout thyself till to tother toward towards twain under underneath
unless unlike until up upon us various versus via vis-a-vis we what whatall
whatever whatsoever when whereas wherewith wherewithal which whichever
whichsoever while who whoever whom whomever whomso whomsoever whose whosoever
with within without worth ye yet yon yonder you you-all yours yourself
yourselves
```

8.2 Appendix B – Database Samples

The purpose of this appendix is to allow any interested researcher to reuse the database for further research. In particular, the naming convention for table names is described.

8.2.1 Method 1

The suffix "_SWL_integer" indicates that the tables are used for Method 1. The "SW" indicates that stop words are included in the data. The "L" indicates that only long documents (those with more than 200 tokens) are used in the analysis.

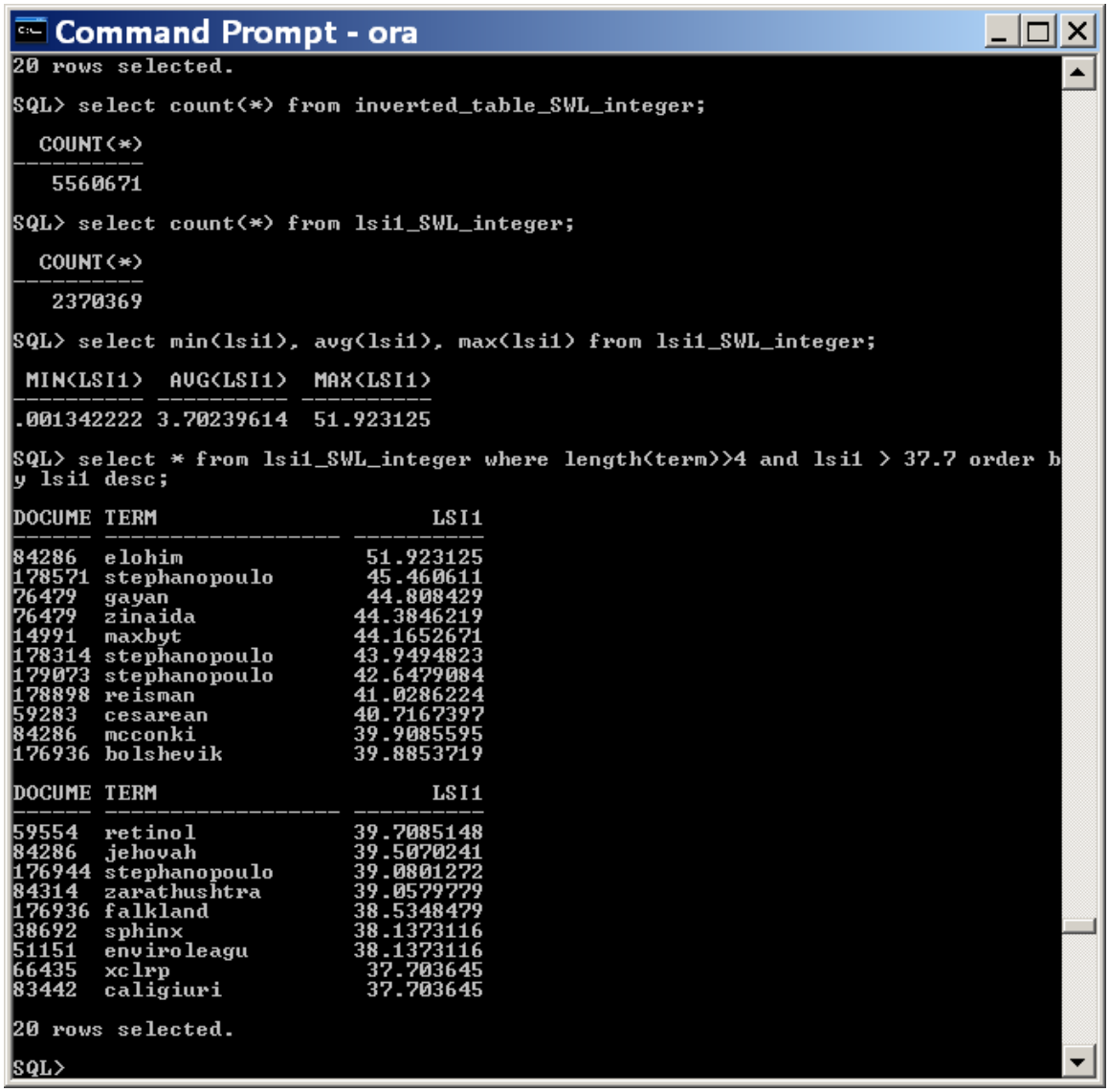

Figure 17. Method1: LSI1

The following screenshot shows the input and output tables for LSI2 computation by Method 1.

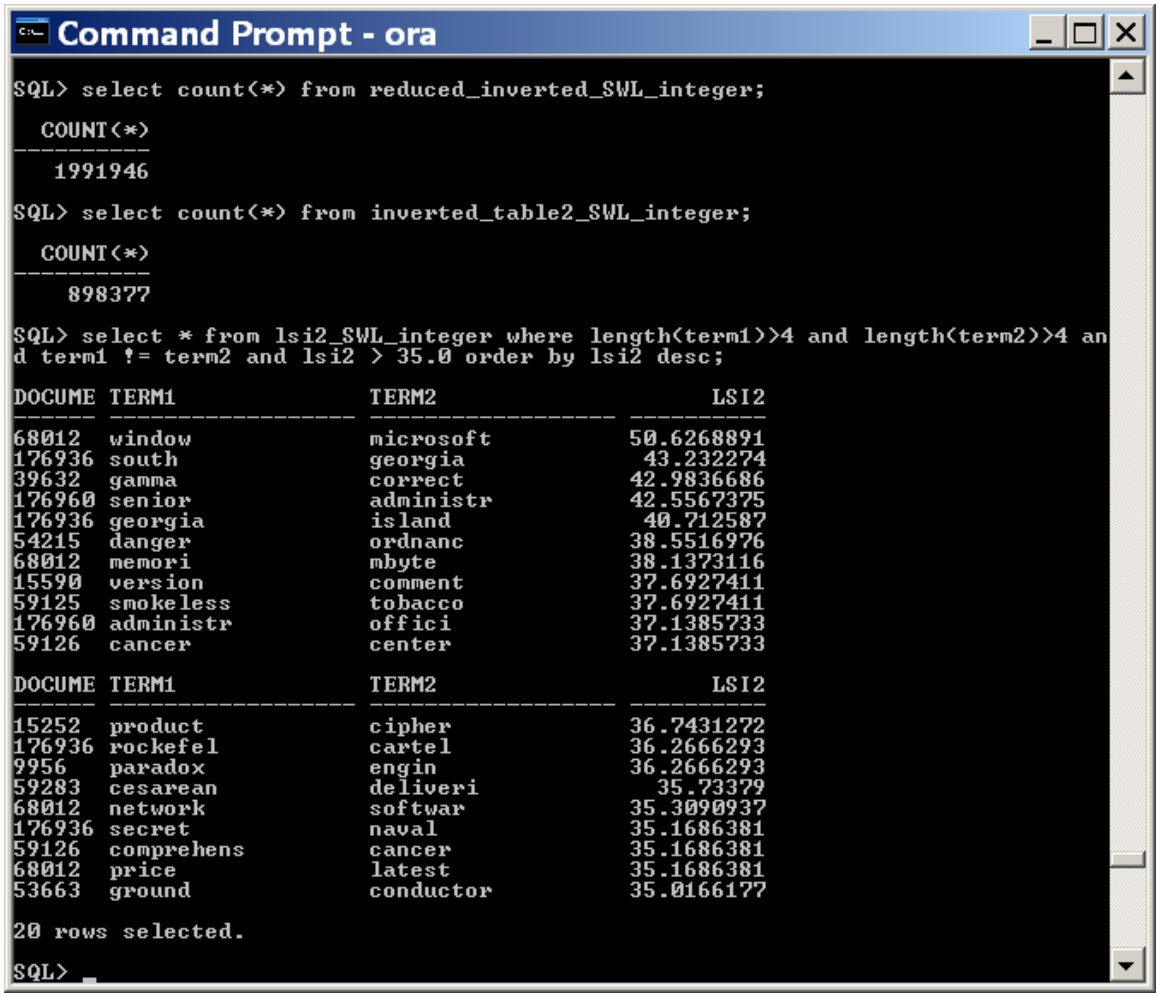

Figure 18. Method1: LSI2

The following screenshot shows the input and output tables for LSI3 computation by Method 1.

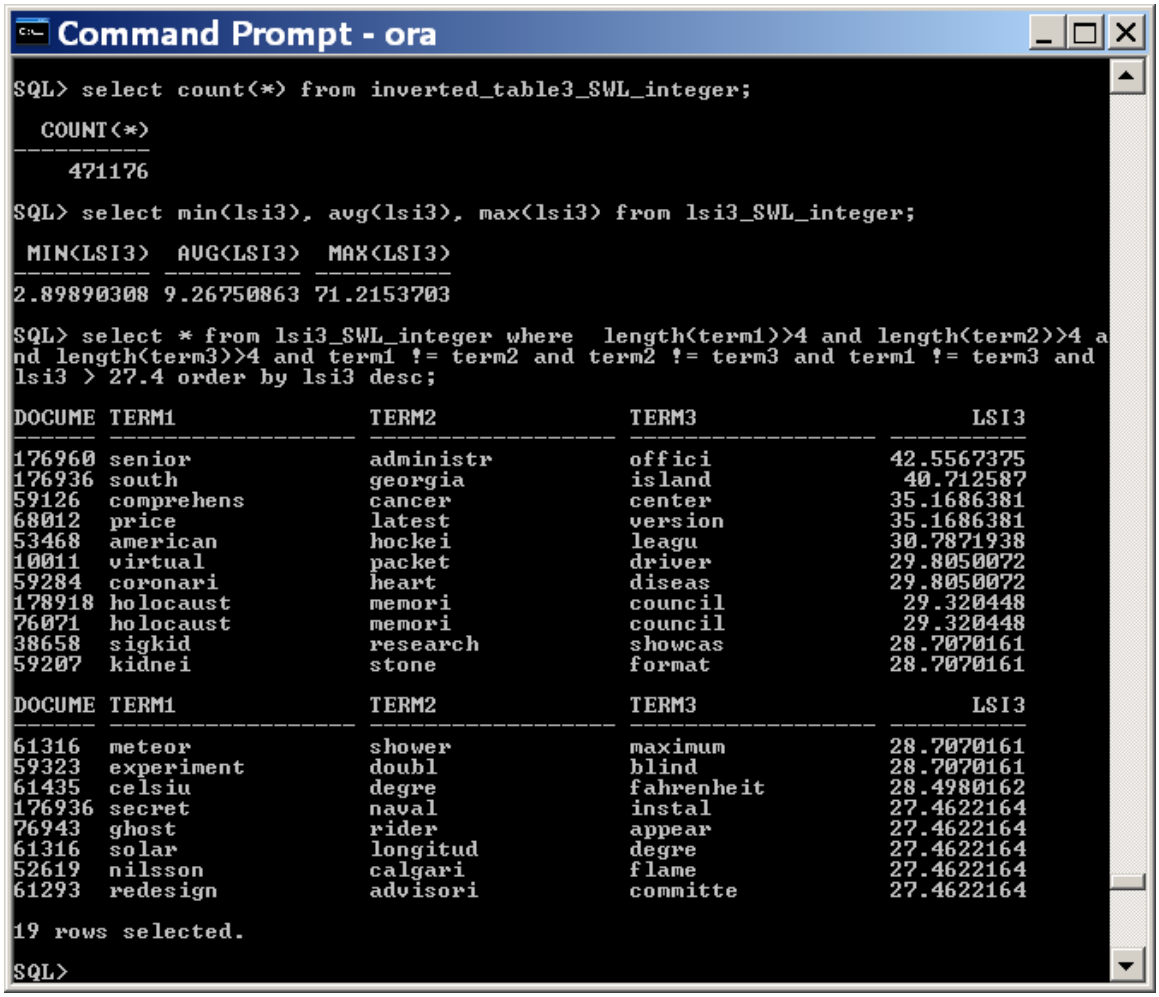

Figure 19. Method1: LSI3

The following screenshot shows the input and output tables for LSI4 computation by Method 1.

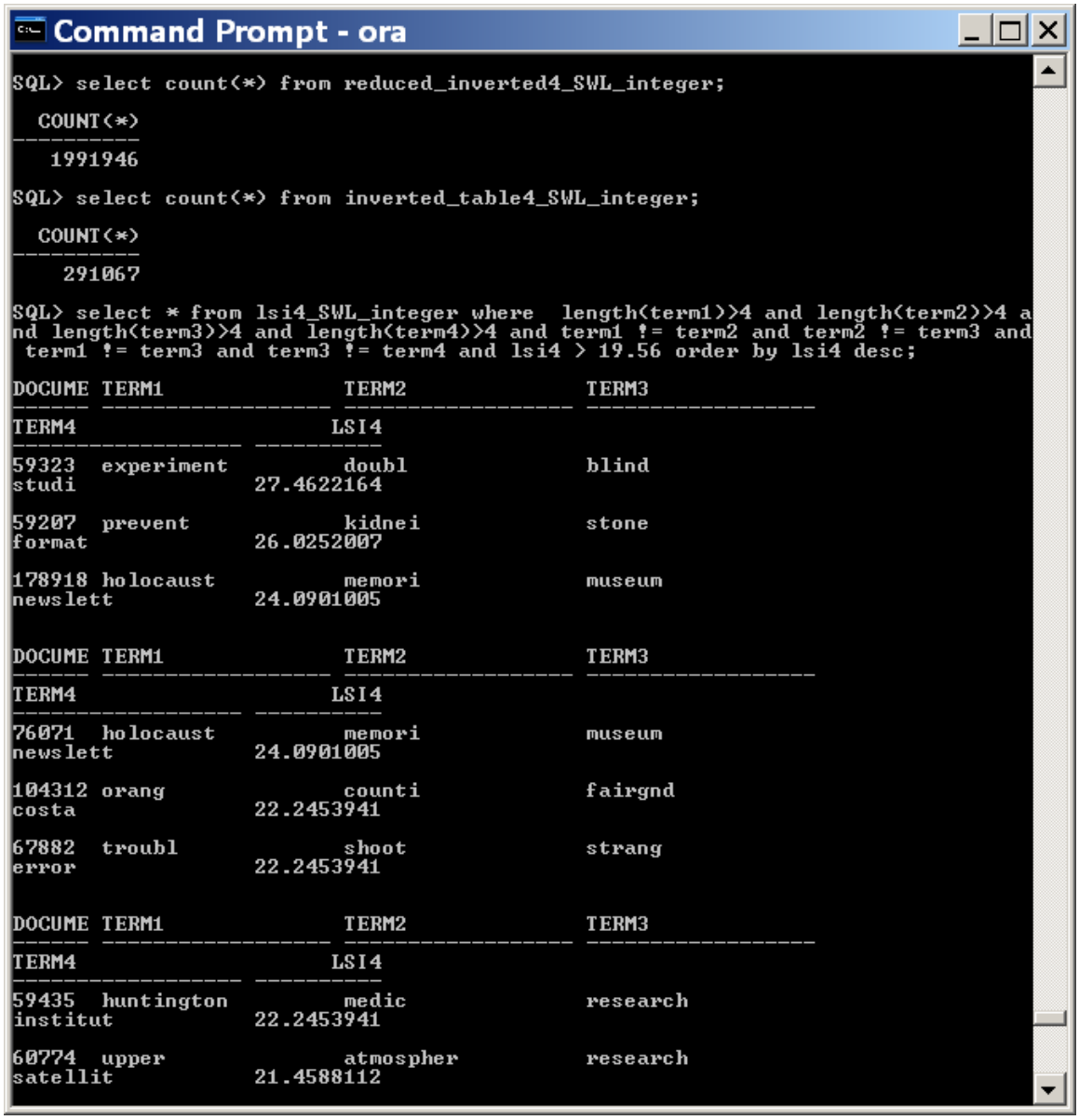

Figure 20. Method1: LSI4

8.2.2 Method 2

The suffix "_UCI_KDD_SWL" indicates that the tables are used for Method 2. The "SW" indicates that stop words are included in the data. The "L" indicates that only long documents (those with more than 200 tokens) are used in the analysis.

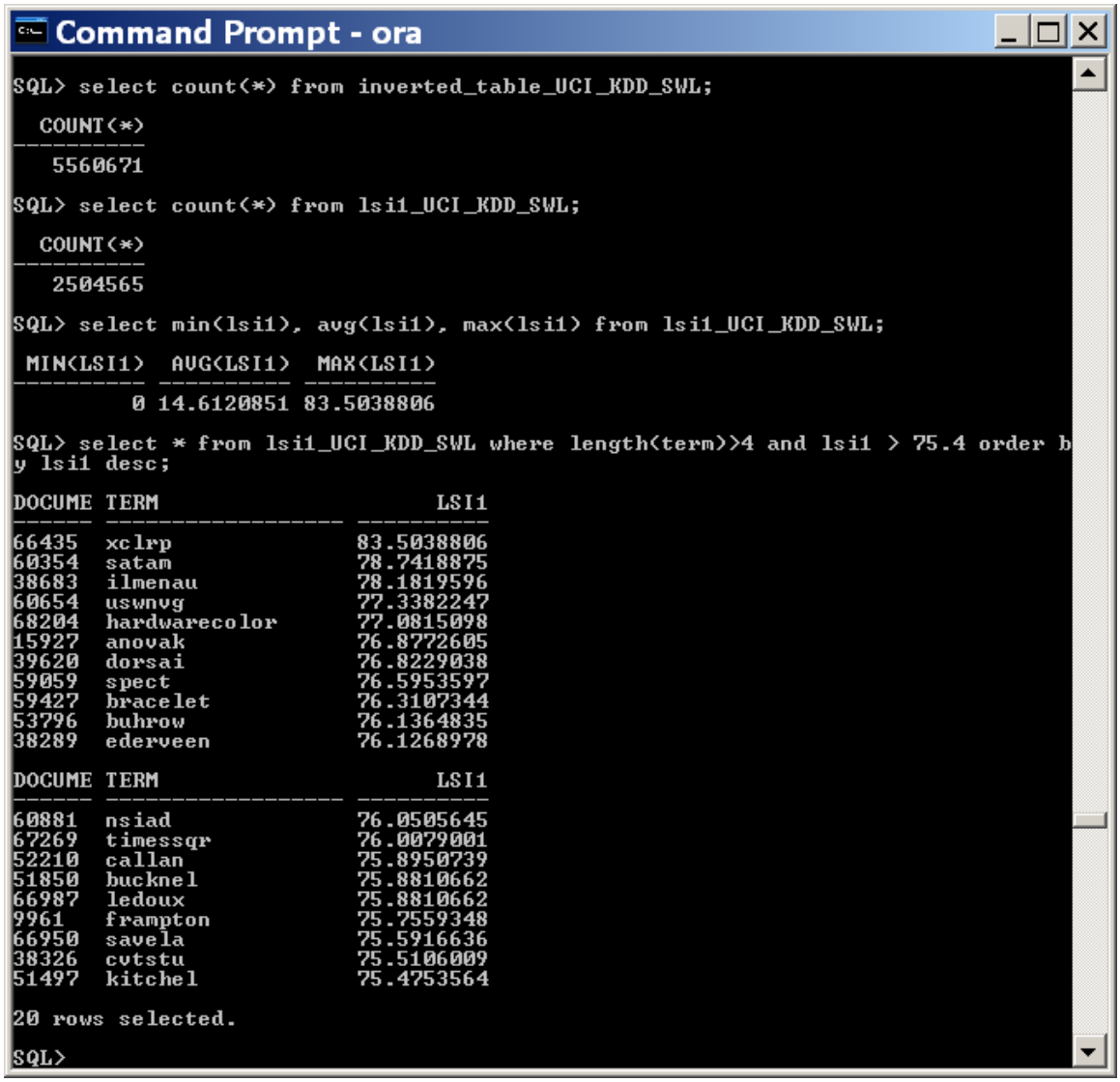

Figure 21. Method2: LSI1

The following screenshot shows the input and output tables for LSI2 computation by Method 2.

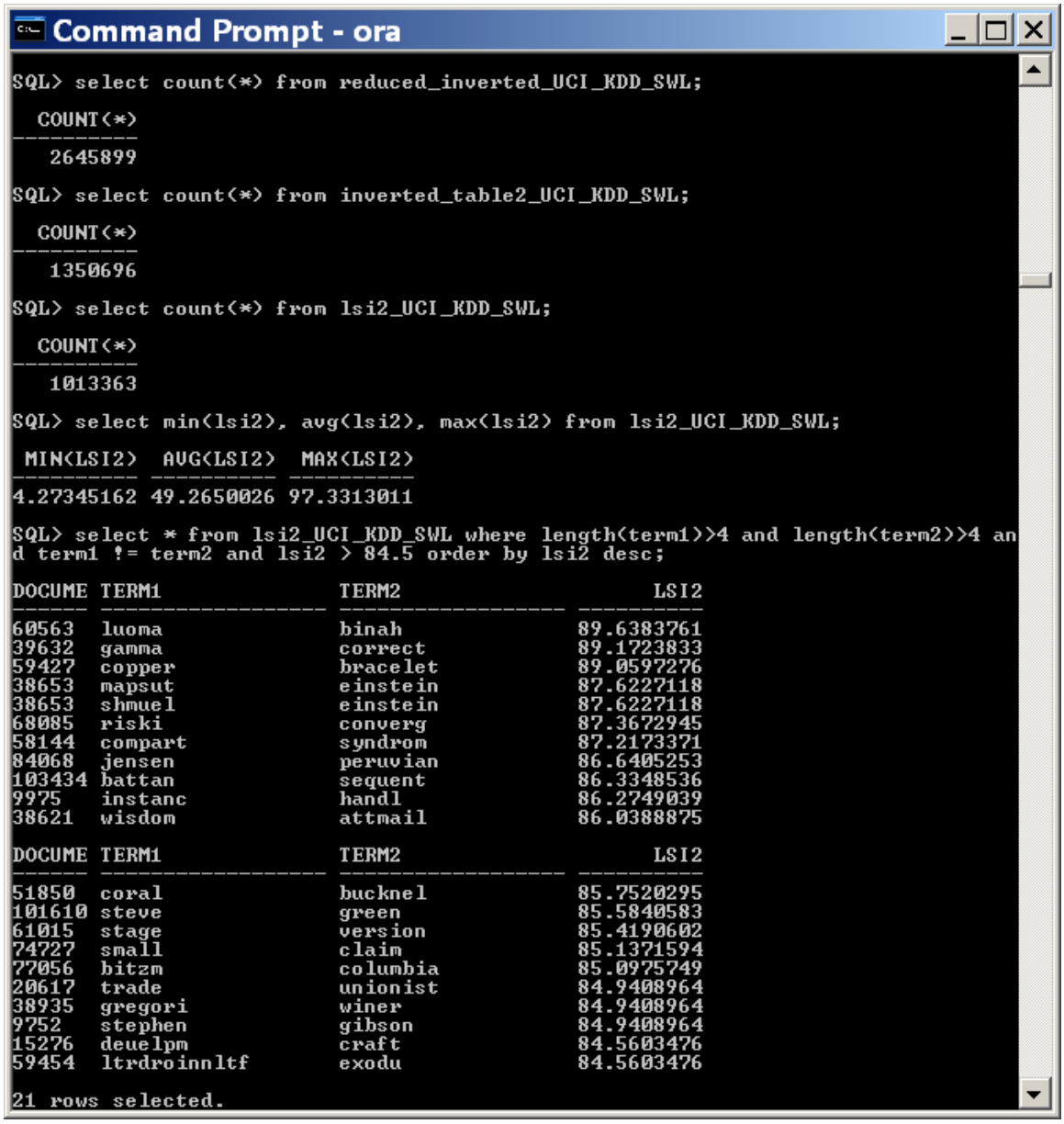

Figure 22. Method2: LSI2

The following screenshot shows the input and output tables for LSI3 computation by Method 2.

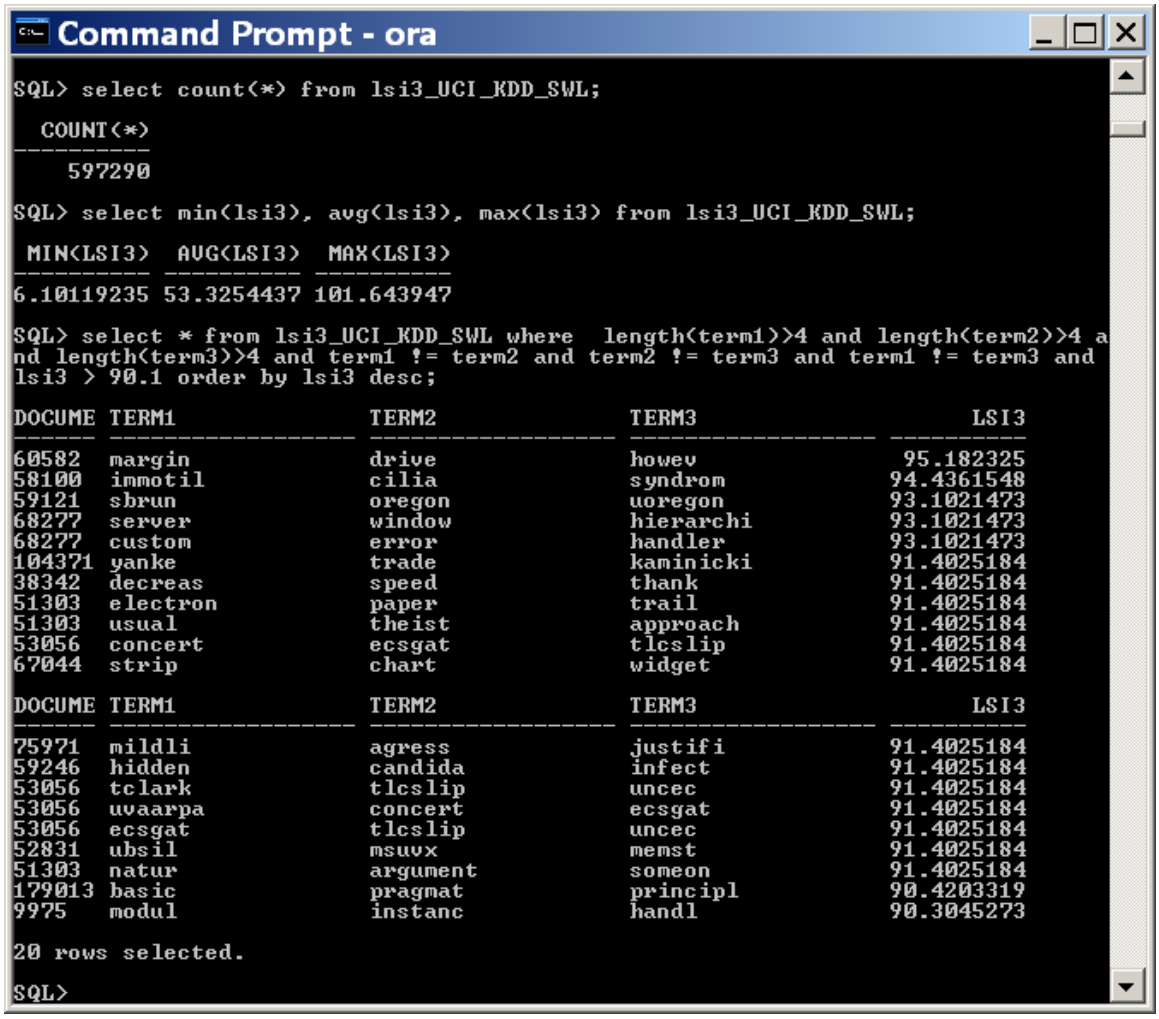

Figure 23. Method2: LSI3

The following screenshot shows the input and output tables for LSI4 computation by Method 2.

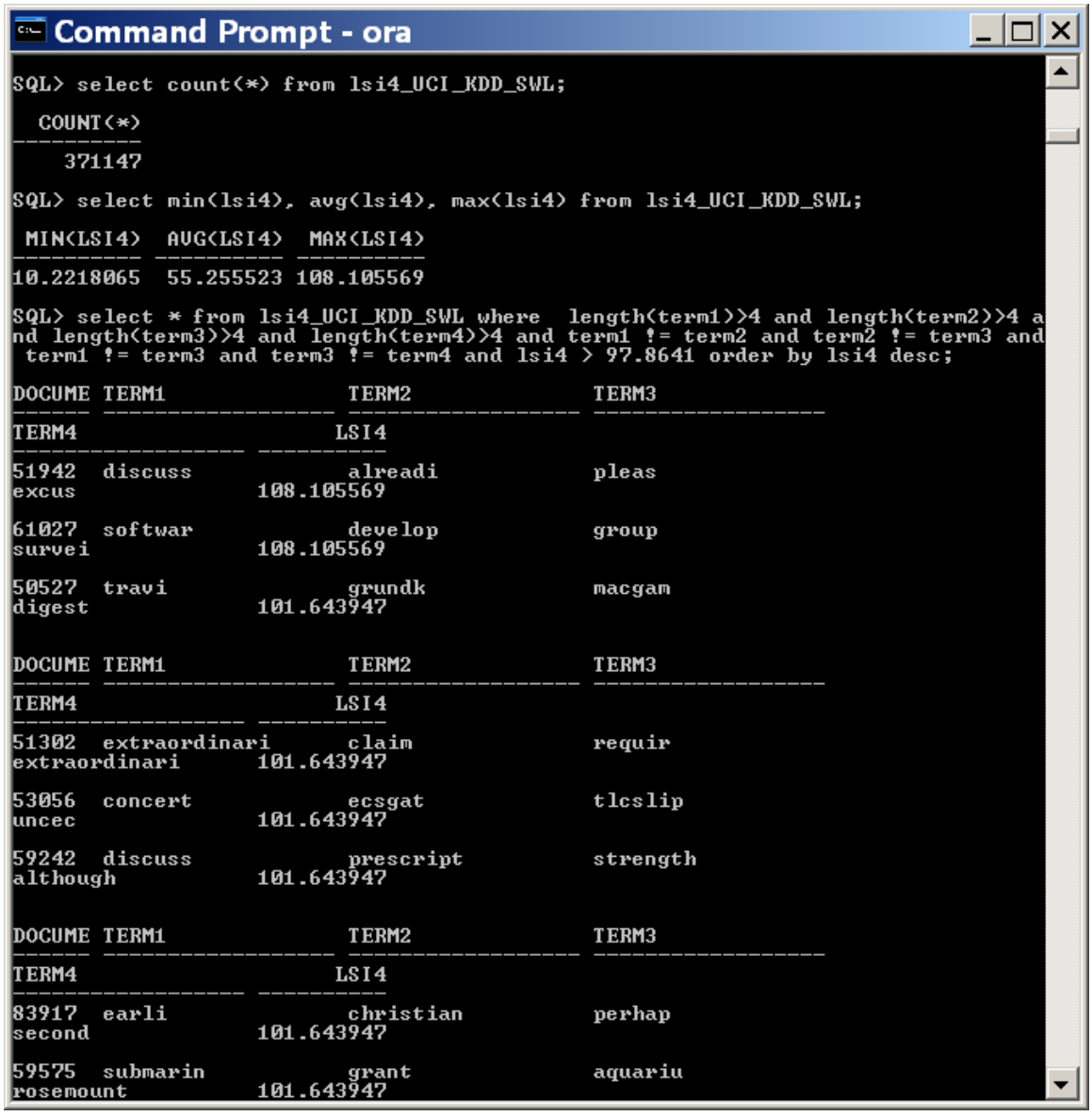

Figure 24. Method2: LSI4

8.2.3 Method 3

The suffix "_TFDF_SWL" indicates that the tables are used for Method 3. The "TF" stands for term frequency, and the "DF" stands for document frequency. The "SW" indicates that stop words are included in the data. The "L" indicates that only long documents (those with more than 200 tokens) are used in the analysis.

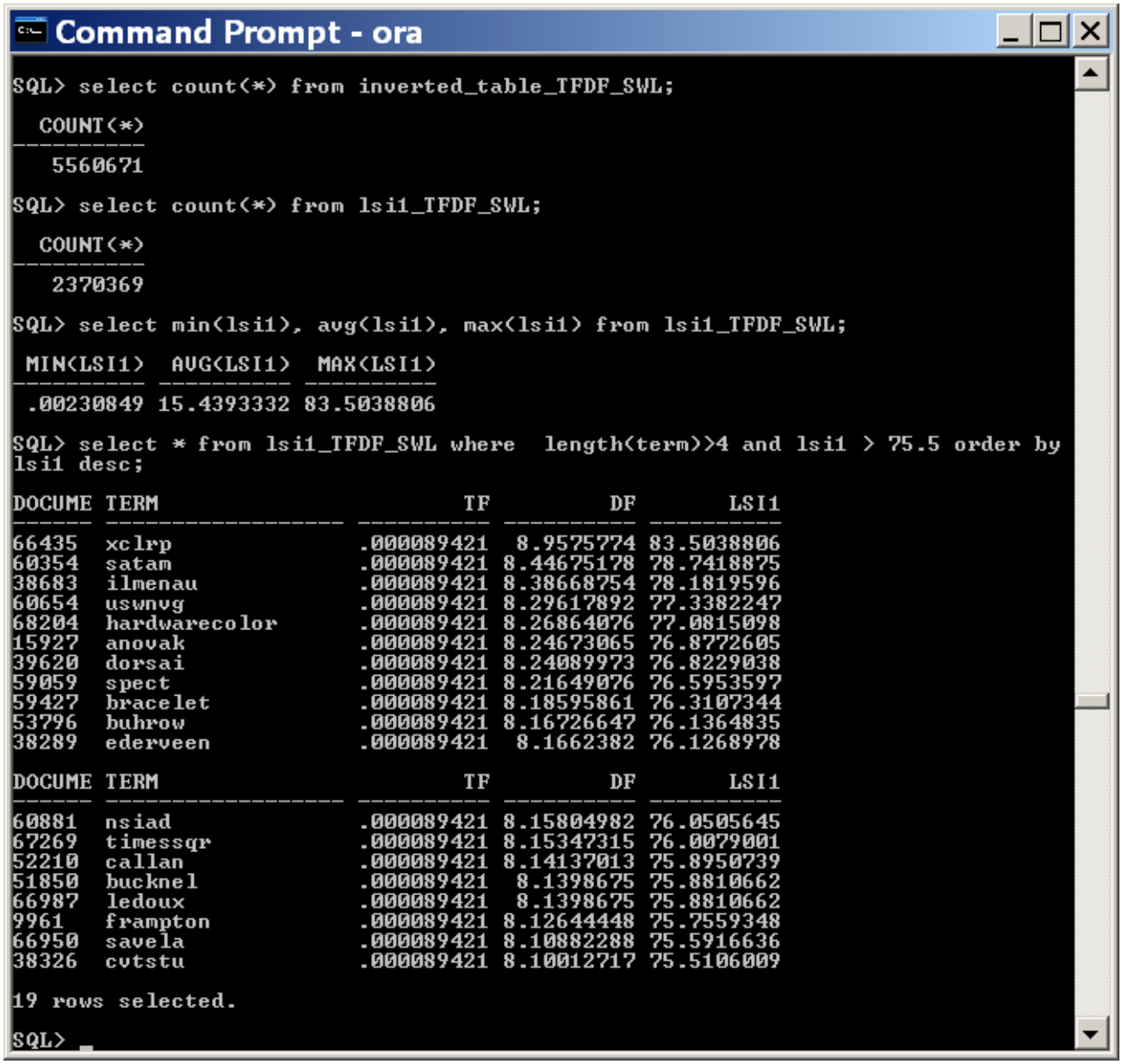

Figure 25. Method3: LSI1

The following screenshot shows the input and output tables for LSI2 computation by Method 3.

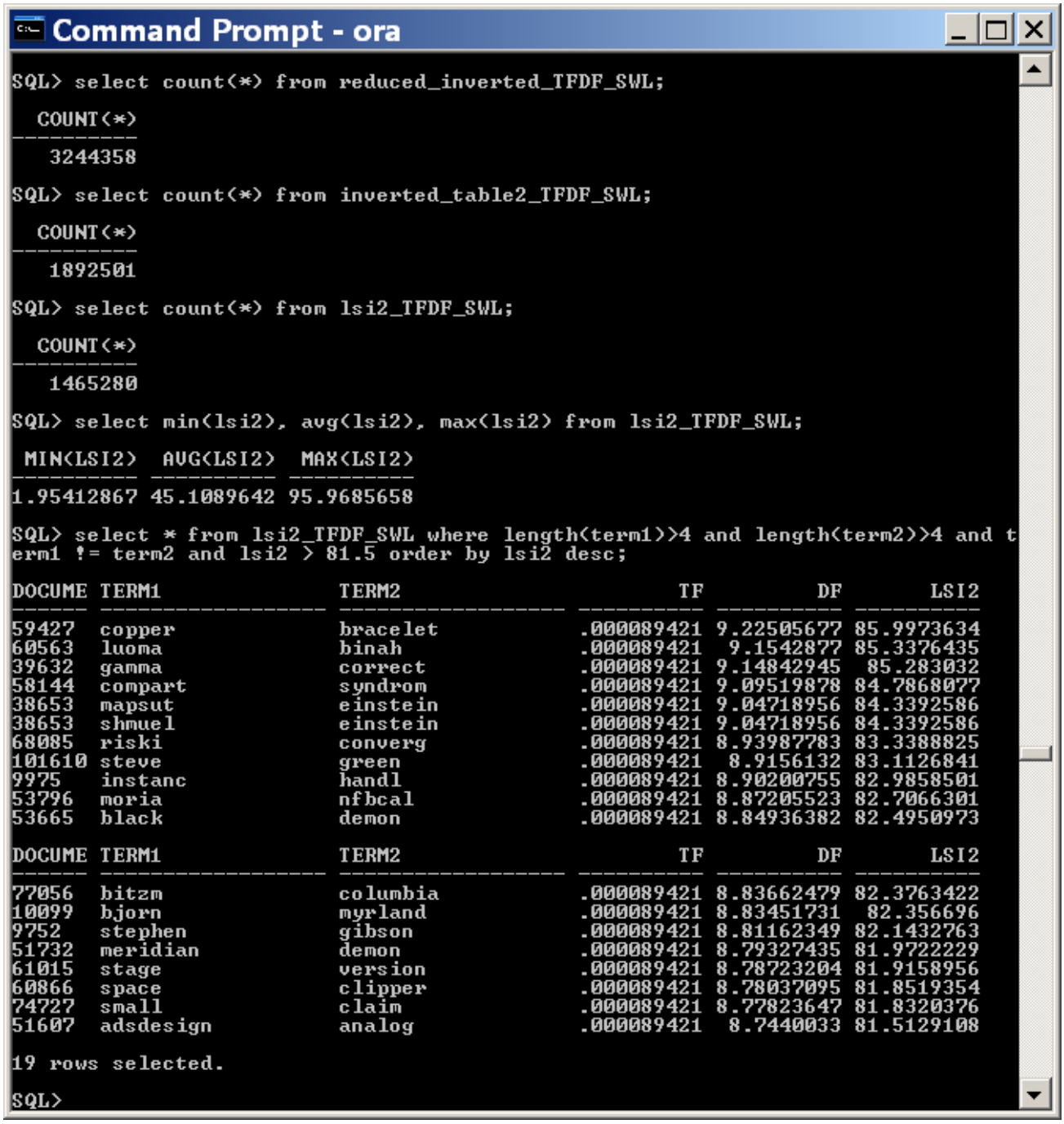

Figure 26. Method3: LSI2

The following screenshot shows the input and output tables for LSI3 computation by Method 3.

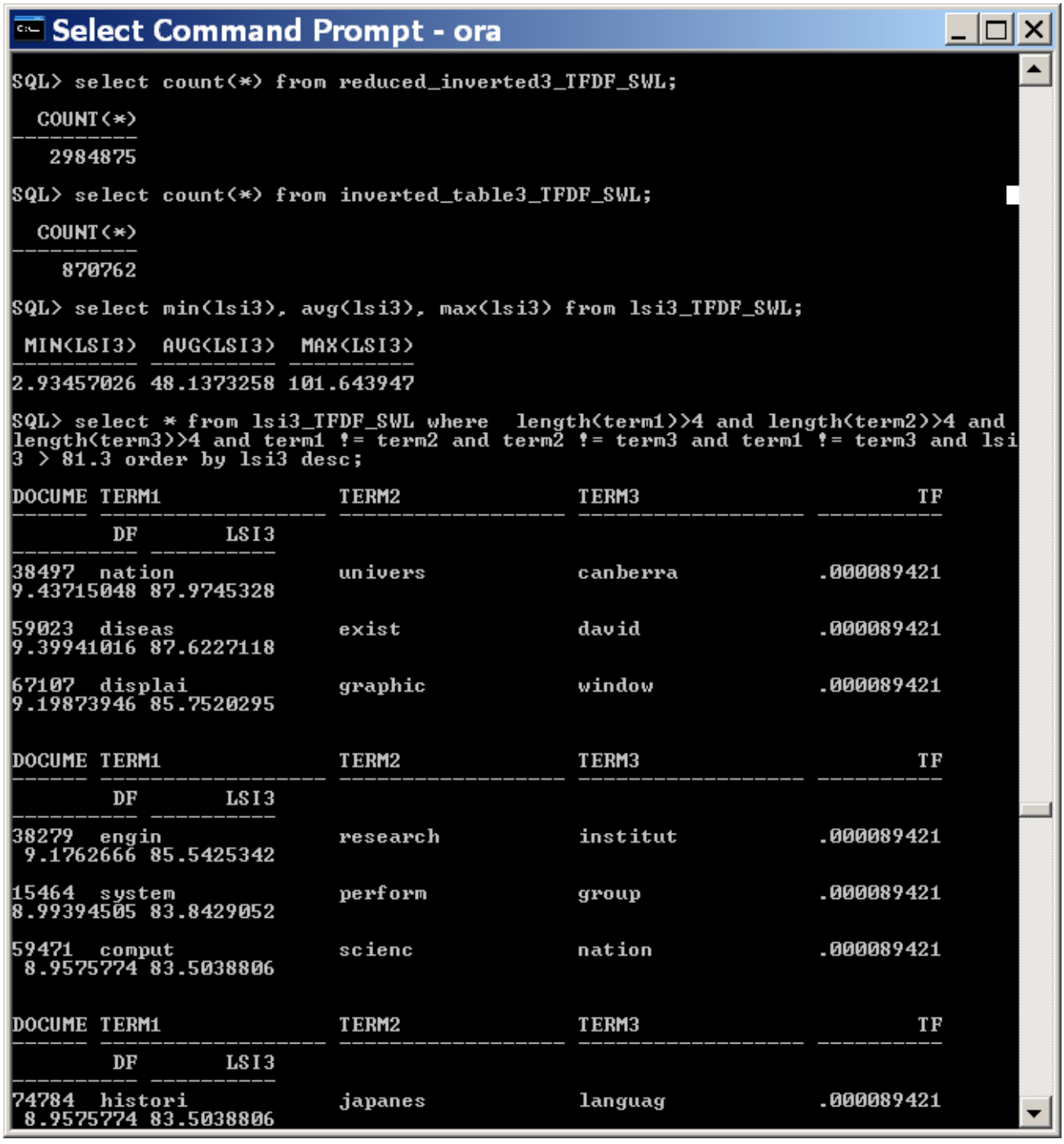

Figure 27. Method3: LSI3

The following screenshot shows the input and output tables for LSI4 computation by Method 3.

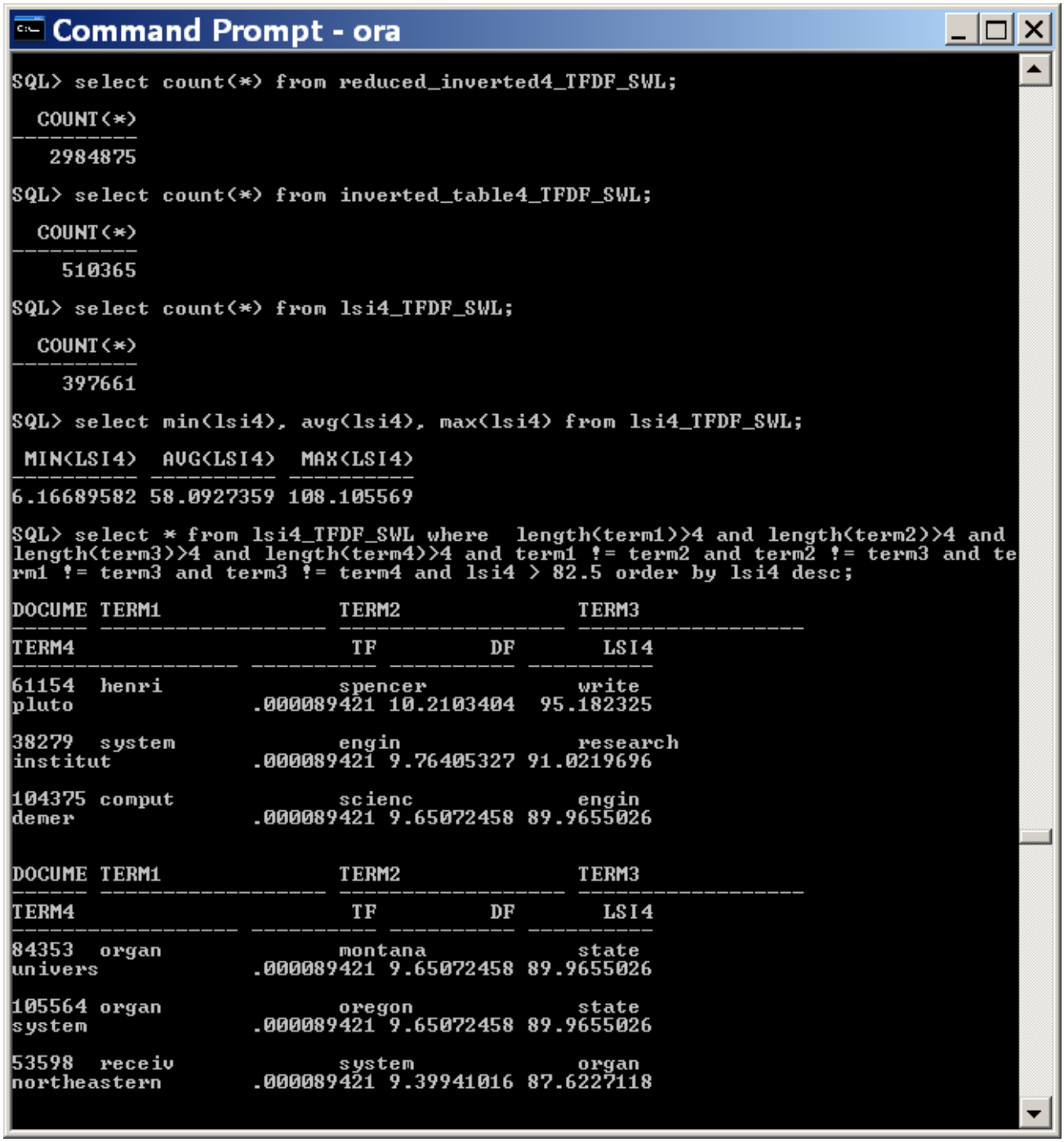

Figure 28. Method3: LSI4

8.2.4 Method 4

The suffix "_UCI_KDD_LDOC" indicates that the tables are used for Method 2. The "LDOC" indicates that only long documents (those with more than 200 tokens) are used in the analysis.

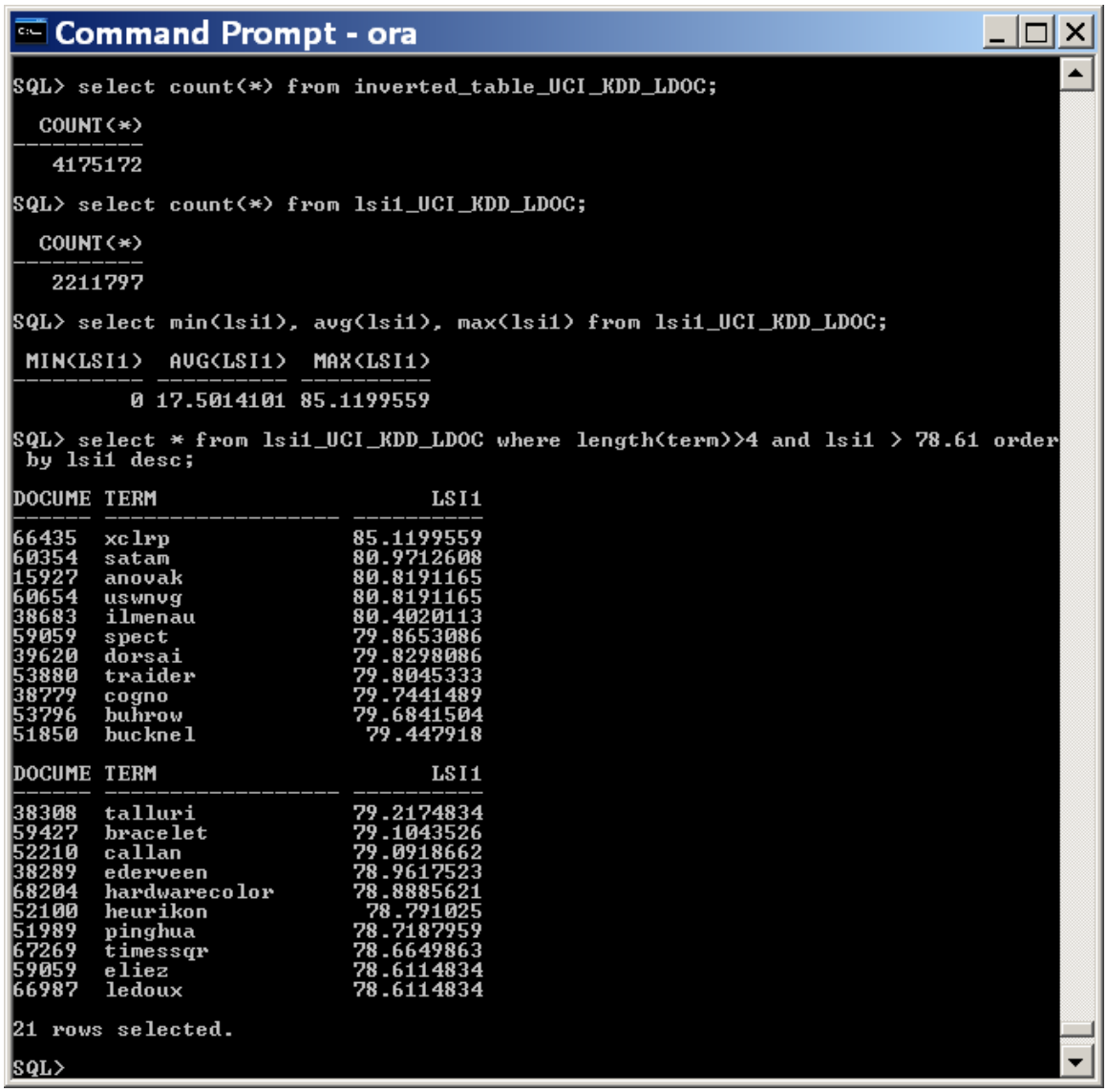

Figure 29. Method4: LSI1

The following screenshot shows the input and output tables for LSI2 computation by Method 4.

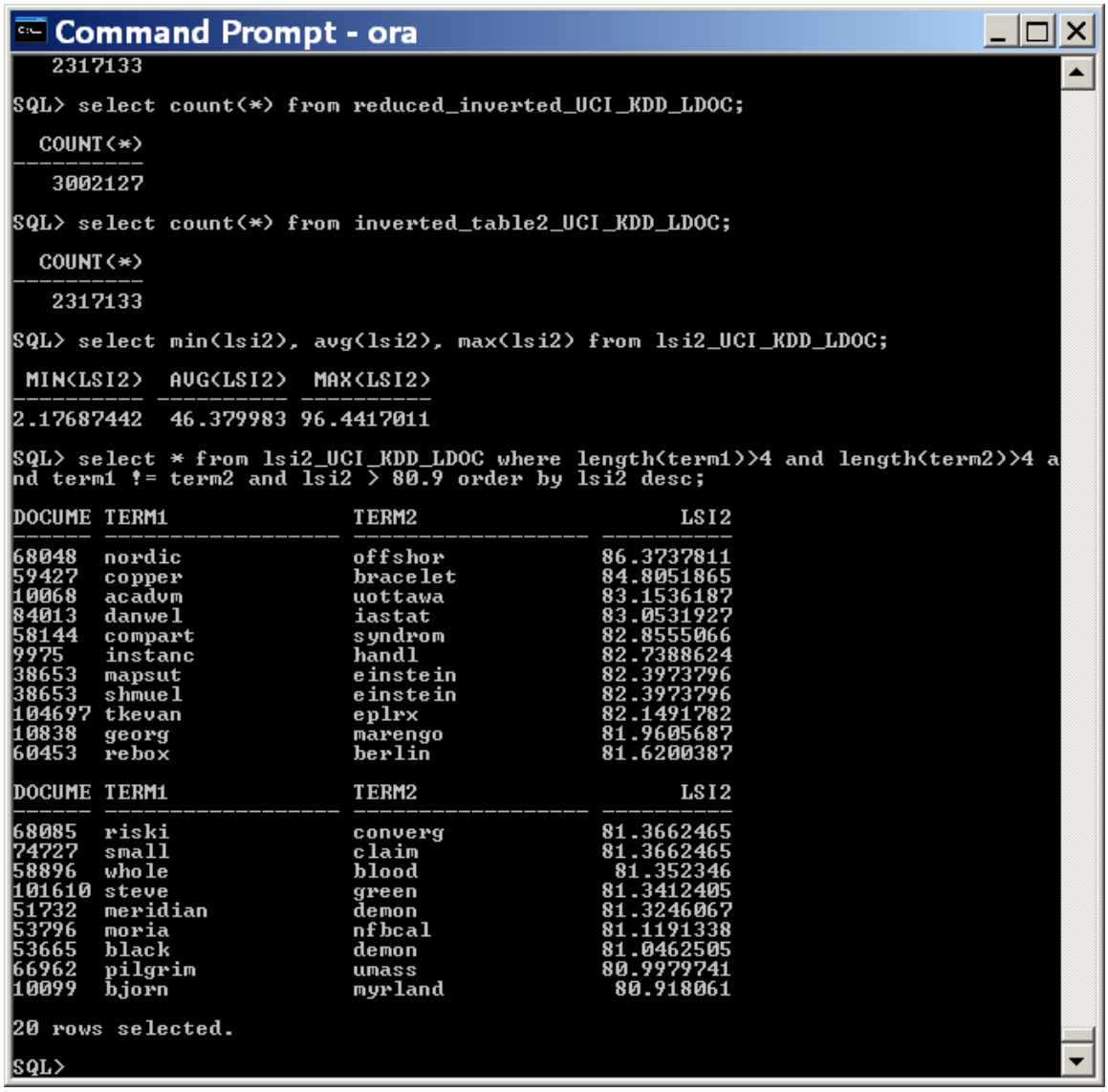

Figure 30. Method4: LSI2

The following screenshot shows the input and output tables for LSI3 computation by Method 4.

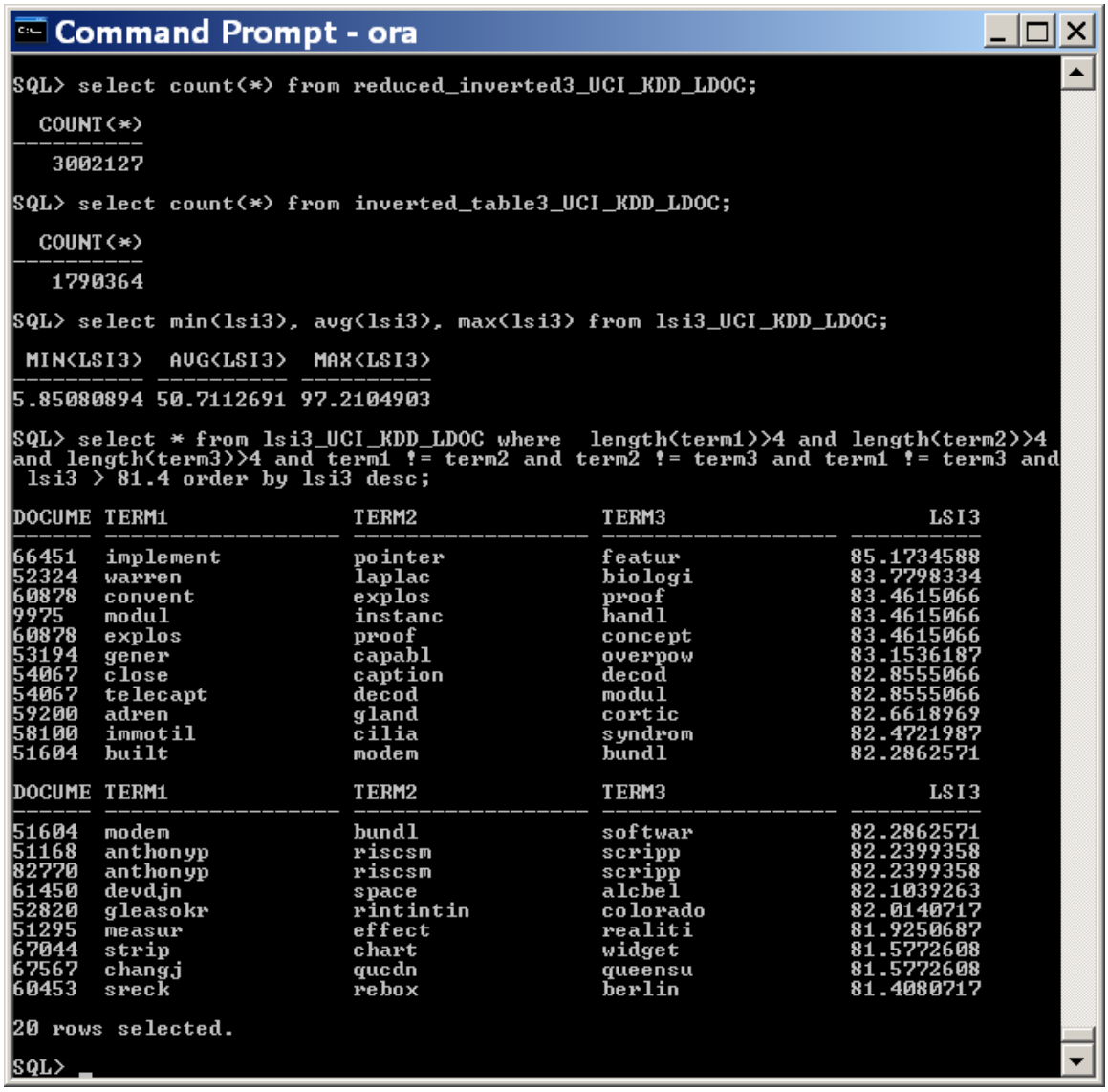

Figure 31. Method4: LSI3

The following screenshot shows the input and output tables for LSI4 computation by Method 4.

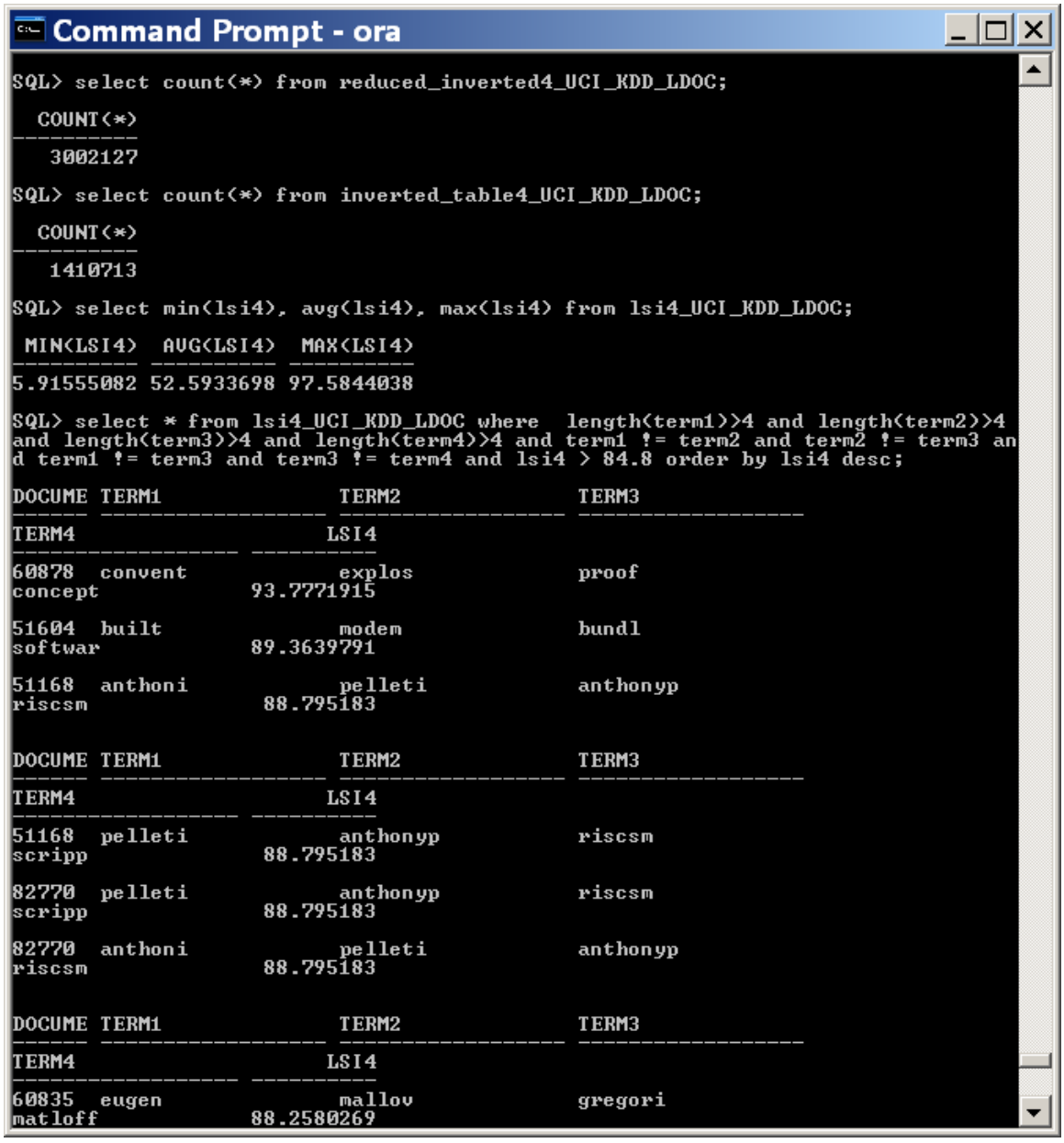

Figure 32. Method 4: LSI4

8.3 Appendix C – References

[A06] Apache (2006). Apache Lucene. [WWW Document] http://lucene.apache.org/java/docs/index.html (visited 2006, November 24).

[C97] Cheng, I. and Wilensky, R. (1997). An Experiment in Enhancing Information Access by Natural Language. Technical Report. UMI Order Number: CSD-97-963. University of California at Berkeley. [WWW Document] http://portal.acm.org/citation.cfm?id=893951&coll=GUIDE&dl=GUIDE&CFID=58602024&CFTOKEN=69210605 (visited 2005, December 2).

[D90] Deerwester, S., Dumais S., Furnas, S., Landauer, T., and Harshman R. (1990). Indexing by Latent Semantic Analysis. Journal of the Society for Information Science, 41(6), 391-407. [WWW Document] http://lsi.research.telcordia.com/lsi/papers/JASIS90.pdf (visited 2005, August 20).

[H99] Hofmann T. (1999). Probabilistic Latent Semantic Analysis. Proc. Uncertainty in Artificial Intelligence. [WWW Document] http://www.cs.brown.edu/people/th/papers/Hofmann-UAI99.pdf (visited 2005, August 19).

[J01] JUnit (2001). JUnit.orig. [WWW Document] http://www.junit.org/index.htm (visited 2005, December 13).

[L98] Landauer T., Foltz P., and Laham, D. (1998). Introduction to Latent Semantic Analysis Discourse Processes, 25, 259-284. [WWW Document]http://lsa.colorado.edu/papers/dp1.LSAintro.pdf (visited 2005, August 20).

[U06] University of California, Irvine (2006). Knowledge Discovery in Databases Archive. [WWW Document] http://kdd.ics.uci.edu/ (visited 2006, November 25).

[W05a] Wikipedia, the free encyclopedia. Latent Semantic Analysis. [WWW Document] http://en.wikipedia.org/wiki/Latent_Semantic_Indexing (visited 2005, August 16).

[W05b] Wikipedia, the free encyclopedia. TF-IDF (Term Frequency Inverse Document Frequency). [WWW Document] http://en.wikipedia.org/wiki/Tf-idf (visited 2005, August 16).

[W06] WordNet (2006). WordNet Similarity 1.02 Stoplist. [WWW Document] http://search.cpan.org/src/TPEDERSE/WordNet-Similarity-1.02/samples/stoplist.txt (visited 2005, August 16).