NAVIGATING
THE “ACM” DIGITAL LIBRARY
WITH A
NEW VISUALIZATION INTERFACE

A THESIS SUBMITTED
TO KENT STATE UNIVERSITY IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF MASTERS OF COMPUTER SCIENCE

BY
JACKSON JACOB CHEENATH

UNDER SUPERVISION OF
Dr. YE ZHAO

AUGUST 2013
Thesis written by
Jackson Jacob Cheenath
M.S., Computer Science, Kent State University, USA, 2012
B.E., Mumbai University, India, 2007

Approved by

Dr. Ye Zhao , Chair, Masters Thesis Committee
Dr. Ruoming Jin , Members, Masters Thesis Committee
Dr. Austin Melton , Members, Masters Thesis Committee
Dr. Javed I. Khan , Chair, Department of Computer Science
Dr. Raymond A. Craig , Associate Dean, Graduate Affairs
# TABLE OF CONTENTS

LIST OF FIGURES .................................................................................................................. v

ACKNOWLEDGEMENTS .......................................................................................................... vii

CHAPTER 1 INTRODUCTION ................................................................................................. 1
  1.1 Problem Description and Motivation .............................................................................. 1
  1.2 Our Solution .................................................................................................................... 2

CHAPTER 2 DATA EXTRACTION AND COLLECTION .......................................................... 3
  2.1 How did we get the data? ............................................................................................... 3
  2.2 Data Storage ................................................................................................................... 3

CHAPTER 3 EXTRACTING MEANINGFUL DATA (EMD) ALGORITHM ......................... 5
  3.1 Current Problem ............................................................................................................ 5
  3.2 Our Solution .................................................................................................................. 6
  3.3 Extracting Meaningful Data (EMD) Algorithm ............................................................... 7
  3.4 Implementation ............................................................................................................ 12
  3.5 Algorithm Usage .......................................................................................................... 13

CHAPTER 4 VISUALIZING THE ACM DIGITAL LIBRARY .................................................. 15
  4.1 Literature Detail Summary ............................................................................................. 15
    4.1.1 Left Bookmark Section ............................................................................................ 15
    4.1.2 Middle Detail Section ............................................................................................. 16
    4.1.3 Right Search Result Section ................................................................................... 20
  4.2 Author Detail Summary .................................................................................................. 23
    4.2.1 Left & Right Sections ............................................................................................ 23
    4.2.2 Authors detail in the middle section ....................................................................... 24
  4.3 Conference ..................................................................................................................... 28
    4.3.1 Initial UI Design .................................................................................................... 28
4.3.2 4-way description ................................................................. 30
4.3.3 List Of paper ........................................................................ 31
4.3.4 Search Paper........................................................................ 32
4.3.5 Evolution path for a paper in a conference .............................. 33
4.4 Citeology .................................................................................. 36
  4.4.1 Initial UI Design ................................................................. 36
  4.4.2 Year Menu .......................................................................... 38
  4.4.3 Conference Menu ............................................................... 39
  4.4.4 Year/Conference Selection .................................................. 39
  4.4.5 Studying the evolution: ....................................................... 40
4.5 Search Detail Summary................................................................ 42
CHAPTER 5 RELATED WORK ........................................................... 45
  5.1 Digital Library User Interface .................................................. 45
  5.2 Phrase extraction .................................................................... 45
  5.3 Citeology ................................................................................ 46
CHAPTER 6 FUTURE WORK ............................................................. 47
CHAPTER 7 CONCLUSION ............................................................... 47
REFERENCES .............................................................................. 48
LIST OF FIGURES

Figure 1: MySQL Database Schema for storing the extracted data .................................................. 4
Figure 2: Example of single word extraction from the abstract of [4] ............................................. 5
Figure 3: Example of multi-word/phrase extraction from the abstract of [4] ................................. 6
Figure 4: Abstract from [3] showing the expected result of the algorithm ................................. 7
Figure 5: Output of the EMD algorithm ......................................................................................... 12
Figure 6: Schema for the WordNet SQL Server Database [11] ..................................................... 13
Figure 7: Bookmarks for different sections on the Literature Summary ........................................ 15
Figure 8: Abstract for the literature [2] ......................................................................................... 16
Figure 9: Ancestor & Descendent for the literature [2] ................................................................. 17
Figure 10: Conferences with immediate citations for literature [2] ............................................. 18
Figure 11: References & Citations for the literature [2] ............................................................... 19
Figure 12: Side-by-side Abstract comparison .................................................................................. 21
Figure 13: Search results when clicked on one of the highlighted words .................................... 21
Figure 14: Full View of the Literature Summary ......................................................................... 22
Figure 15: Author Detail Summary – Left & Right Column – Keywords & ACM Classification
Listing [6] with their count and percentages for the author ....................................................... 23
Figure 16: Conference Click & Hover ......................................................................................... 25
Figure 17: Authors details in the middle section ......................................................................... 26
Figure 18: Initial UI design of the conference summary tool for the SIGGRAPH conference from
1985 to 1991 ................................................................................................................................. 28
Figure 19: 4-way description for [5] ......................................................................................... 30
Figure 20: Filtering author, keyword or ACM classification [6]..................................................... 31
Figure 21: Conference Summary - List of Papers (a) Closed Accordion (b) Open Accordion.... 32
Figure 22: Initial screen before starting to search for evolution of a paper................................. 33
Figure 23: Step 2 Evolution – Introducing connectors to the leaf nodes................................. 34
Figure 24: Step 3 Evolution - Choosing another interesting leaf node.................................. 34
Figure 25: Final path, where the path cannot exceed anymore................................................. 35
Figure 26: Citeology for [2]...................................................................................................... 37
Figure 27: Citeology - Year Menu.............................................................................................. 38
Figure 28: Citeology - Conference Menu .................................................................................. 39
Figure 29: Selecting "OOPSLA" conference highlights the paper in that conference and a track is
drawn in the center of the screen. .............................................................................................. 39
Figure 30: Citeology for the paper [2]...................................................................................... 41
Figure 31: Showing search bar in the center of the screen......................................................... 42
Figure 32: Search Results ........................................................................................................... 43
Figure 33: Opening a literature in a modal pop to preserve the search results......................... 44
ACKNOWLEDGEMENTS

I would like to acknowledge the support of Dr. Ye Zhao for his guidance, support and encouragement. This thesis would not have been possible without his support and proper guidance. I would also like to express my thanks to my beloved family for their understanding and endless love, through the duration of my studies.

Jackson Cheenath

06/20/2013, Kent, Ohio
CHAPTER 1
INTRODUCTION

1.1 Problem Description and Motivation

Due to the increased adaptation of the Internet as a tool by users in recent years, there has been an exponential amount of growth in the amount of information stored electronically on the Internet. This information is stored in many different formats such as text, image, videos or sound for easy accessibility, recoverability and availability. There are many online repositories that attempt to collect and organize a large amount of information at one place. The ACM digital library stores a collection of publications from journals, conference proceedings and magazines. The purpose of this library is to provide an enormous collection of valuable information regarding a particular discipline. Since the number of publications stored in this library is so large, it is often necessary to search and filter through this collection of electronic documents to find relevant information about a particular topic. Each document in the library is stored with a collection of meta data attributes that make this searching and filtering possible but with the increasing amount of data and fixed screen space to explore that data, it has become more important to show concise meaningful information, and highlight the hidden data in such a way to make it easy to view and useful to the user at a glance.

The approach that many of these digital libraries has followed in the past is a traditional approach to navigate these collections via searching by the paper’s metadata attributes such as the paper’s author, title, year, conference, or abstract content. The user is then tasked with browsing through the paper for the citations and references to try and trace back the most relevant papers to their work. When researching a topic, it is always an added advantage to know the evolution of a paper or literature over the years, the path of research that an author took,
learning the particular kind of topics evolved in a specific conference or many other important inferences of the papers historical metadata patterns rather than just searching the library for papers with matching attributes.

1.2 Our Solution

To give the users the tools they need to visualize the data in this way, we propose a new visualization interface for navigating the ACM digital library. It supports effective literature exploration with a set of web-based functions including search, detail summary, conference summary, author summary, and citeology. These functions are designed and integrated with enhanced perceptual understanding and human-machine interaction, where colors, diagrams, layouts and other informative visualization factors are utilized to analyze the collective metadata from the ACM digital library.

We use a large-scale data set of the titles, authors, categories and abstracts based on the ACM digital library. A phrase extraction algorithm is designed to retrieve meaningful phrases from the data set. All the web based functions mentioned above uses this algorithm. These phrases, instead of single keywords, can represent the publications with improved semantics, which enhance the visualization output and user experience. We do this by sequentially scanning the paper’s abstract and using pre-defined dictionaries, punctuations to pull out groups of meaningful phrases and throwing out junk words such as verbs, possessive pronouns and other pre-defined stop words that would prevent us from returning meaningful data. The visual interface provides an advanced platform for researchers in their literature study. It can be further extended to the exploration of other libraries and databases.
CHAPTER 2
DATA EXTRACTION AND COLLECTION

2.1 How did we get the data?

Since the data from the ACM digital library was not readily available in a format that we could digitally analyze we had to write a scrapper to scrape the required data from the ACM digital library website. This scrapper was written in PHP to crawl through the proceedings list page and each of the literature detail pages to collect a sample of the necessary data for this thesis. The pieces of information we collected from each publication were credentials related to a literature paper from the site viz. the title, abstract, keywords, ACM’s computing classification system (2012) [6], year, references, citations and the conference where the paper was published. Since we scraped the data from the ACM digital library website we had to do an initial cleanup on the collected data before it could be used. This included filtering out special characters used to format the text for online viewing. After cleaning and gathering the data each conference was randomly assigned a color, but taking into consideration that no two conferences have the same color. We even collected the ACM’s Computing Classification System [6] into our database by parsing the XML provided by the ACM.

2.2 Data Storage

While the data was being extracted, the database used to store the data was MySQL, since PHP forms the best stack with MySQL. Our sample data set includes extracts from around 200K papers. This data includes approximately 215K authors, 550K paper-author relations, 160K keywords, 1000 conferences, 160K abstracts, 600K citations, and 600K references. The following diagram shows the schema for the database.
Figure 1: MySQL Database Schema for storing the extracted data
CHAPTER 3
EXTRACTING MEANINGFUL DATA (EMD) ALGORITHM

3.1 Current Problem

Consider a huge collection of electronic documents given to be analyzed and reviewed. There are many previous works that automatically tries to determine the hidden patterns or topics inside the documents to facilitate reviewing of a large collection of electronic documents. Most of these techniques are based on splitting the document by spaces and identifying the frequency of those words in the collection of documents. One of the difficulties in doing the latter is; there are many languages that can include multi-word groups / phrases that might give a different meaning to the context.

Figure 2: Example of single word extraction from the abstract of [4]

Consider the example in Figure 2, which is a single word extraction from the abstract of a 1978 paper [4]. As said above these words individually have their own meanings, but when grouped together as a multi-word or phrase has a different meaning as shown below in Figure 3.
3.2 Our Solution

We propose an algorithm which will extract multi-word / phrases from the text. Consider the example in Figure 4, an abstract from the paper [3]. The phrases like “diff utility”, “dataflow diagrams” and all others highlighted in black are the ones that give more meaning than just “diff”, “utility”, “dataflow” or “diagrams”. One thing to notice in this example is that most of the meaningful words are usually nouns and adjectives and the rest might either be an article, auxiliary verb, personal pronoun, action verb or so on. This algorithm does a sequential scanning from start to end and makes use of some pre-defined grammar, stop-word and discipline related single stop-word dictionary and extracts those particular nouns or adjectives and gets rid of the other non-meaningful terms. The tricky part here is to get rid of the non-regular verbs such as the words like “comparing”, “discovers”, “propose”, this is where we use a Lexical database like WordNet [13] to get rid of the non-regular verbs. In the next section we will describe in detail the EMD algorithm and how we go about extracting the meaningful data.
The diff utility is an important basic tool, providing a foundation for many of the fundamental practices of software development, such as source code management. While there are many file differencing tools for textual programming languages, including some that look at more than simple textual variations, there are few for visual programming languages. We present an algorithm for semantic comparison of programs in controlled visual dataflow languages; that is, languages in which dataflow diagrams are embedded in control structures. This algorithm performs depth-first search of call structures comparing embedded diagrams using subgraph isomorphism, to determine if two programs are semantically equivalent, and if they are not, discovers the differences. We use the visual language Prograph for illustration; however, the mechanism we propose could be applied to any controlled dataflow language, such as LabVIEW.

Figure 4: Abstract from [3] showing the expected result of the algorithm

3.3 Extracting Meaningful Data (EMD) Algorithm

The algorithm follows a series of steps as shown below to achieve the result

**Step 1: Initialize Dictionary**

The first thing we do is initialize dictionaries for stop words, English grammar and discipline related stop words.

- **Stop word list**: This list was taken from the MySQL full-text stop words [9], but deducted all those words that occur in the grammar list.
  
  Ex: “also”, “am”, “an”, “and” …

- **Grammar List**: This list was taken from various different sources on the Internet
- **English Articles:**
  - **Ex:** “a”, “an”, “the”

- **Auxiliary Verbs:**
  - **Ex:** “am”, “is”, “was”, “were”, “being”, “been”, “can”, “could” …

- **One word prepositions:**
  - **Ex:** “aboard”, “about”, “above”, “across”, “after”, “against” …

- **Complex prepositions:**
  - **Ex:** “according to”, “ahead of”, “along with”, “apart from” …

- **Adverb Suffixes:**
  - **Ex:** “entarily”, “entally”, “ionally”, “iately”, “ically”, “iously”

- **Verb Suffixes:**
  - **Ex:** “orize”, “iate”, “ivate”, “ize”, “own”, “ized”, “yze”

- **Personal Pronouns:**
  - **Ex:** “I”, “you”, “he”, “she”, “it”, “we”, “they”

- **Possessive Pronouns:**
  - **Ex:** “his”, “her”, “their”

- **Discipline related single stop words:** In case of ACM the common single word stop words will be like “algorithm”, “computer”, “experiment”, “study”, “research”, “result” …

**Step 2: Bracket Extraction**

There is a high probability that the content inside the bracket is an abbreviation, but this assumption is not always true. However, we can say that anything inside the brackets is probably something related to the context prior to it. So we will try to extract anything inside the brackets
rather than letting the content inside the brackets get mixed with the other content. After we extract the content inside the brackets we return the original abstract but this time without the brackets and the content inside the brackets.

**Ex:** In this paper we present a new global router appropriate for Multichip Module (MCM) and dense Printed Circuit Board (PCB) design.

### Step 3: Regex Cleaning

After the bracket contents and the brackets are extracted, we now use regular expressions to remove any characters other than the ones inside the bracket [a-z&;A-Z,#0-9'-]. We do preserve seven punctuation marks comma (,), full-stop (.), semi-colon (;), hyphen (-), single quotes (‘) and hash/pound (#). Since we are doing a sequential scanning of the statements, marks comma, full-stop and semi-colon plays a very important role in our algorithm recognizing the “stops” and “pauses” in a statement, so that we don’t have phrases that have one word from one statement and the other word from the other statement. We keep hyphen to preserve the words like “depth-first”, “3-D”, “2-D”.

**Ex:** …. design for CMOS VLSI chip, specifically to guard against electrostatic discharge (ESD) stress and latchup. We propose … computer software. You may publish on paper, in the usual sort of user's guide or procedure manual, or you may publish online.

### Step 4: Initial Split

The initial split of the statement is done based on the complex preposition list, to get rid of the less important words from the statements.

**Ex:** a wave profile, which changes according to wave steepness and water depth.
Step 5: Second Split

Now for each initial split the second split is done on the spaces between each word.

Ex: “a”, “wave”, “profile”, “which”, “changes”

Step 6: Loop through the bag of single words

Note that till this point we have not removed the stop words like “which”, “a” but, we have an array or a bag of single words in a sequence. We loop through this array and do the following steps for each value in the array.

Loop starts here

Step 6.1: Initialize variables

Initialize a Temporary String Collector (TSC) and a Final Phrase Collector (FPC).

Step 6.2: Excluding unwanted values

Whenever we come across a full stop, comma, semi-colon or a number, we stop and do step A.

Step 6.3: Personal Pronouns

Most of the time a word after a personal pronoun is a verb, thus we can exclude them, here we stop and do Step A

Ex: We propose an algorithm ... In this paper we present ...

Step 6.4: Regular Verbs

Some basic regular verb tricks like the “ing”, “ed”, “s” [14] [15] form check can be done to exclude the verbs, here we stop and do Step A

Ex: Comparing, utilizing, decided, the animated effect
Step 6.5: Irregular Verbs

This is the trickiest part, since it is difficult to know which word is a verb, since in English a single word can be used either as a noun, a verb or an adjective. This is where we plan to use WordNet [13] to help find out irregular verbs.

Ex: Arose, became, began, bid, brought, built

Step 6.6: Exclude Stop Words, One word prepositions and Auxiliary verbs

At any point of time, we find either a stop word, one word preposition or an auxiliary verb, we stop and do Step A

Step 7: If none of the above

Do Step B

Loop ends here

Step A: Check and Write

Check if we have anything in the TSC, if TSC is not empty then we check if TSC is a

- single word adverb or
- a discipline related single stop word or
- a single letter .

If TSC is any of these three TSC is not put into FPC, else put TSC into the FPC and re-initialize the TSC;

Step B: Append to TSC

Finally at the end of the loop, we get a bag of meaningful words for that particular document.

Figure 5 shows the output of the algorithm
Algorithm Output

The diff utility is an important basic tool, providing a foundation for many of the fundamental practices of software development, such as source code management. While there are many file differencing tools for textual programming languages, including some that look at more than simple textual variations, there are few for visual programming languages. We present an algorithm for semantic comparison of programs in controlled visual dataflow languages; that is, languages in which dataflow diagrams are embedded in control structures. This algorithm performs depth-first search of call structures comparing embedded diagrams using subgraph isomorphism, to determine if two programs are semantically equivalent, and if they are not, discovers the differences. We use the visual language Prograph for illustration; however, the mechanism we propose could be applied to any controlled dataflow language, such as LabVIEW.

Figure 5: Output of the EMD algorithm

3.4 Implementation

The code was written in C# as a console application, mainly to use the built in dictionary features that C# provides and secondly because we got the Wordnet [13] SQL Server ready database from another source [11]. Figure 6 shows the schema for the WordNet[13]. Note that, during all the above operations, care was taken to handle the UTF-8 characters. Following figure shows the schema for the WordNet [13] SQL Server database [11].
Figure 6: Schema for the WordNet SQL Server Database [11]

3.5 Algorithm Usage

This algorithm can be used in many scenarios. Two such scenarios where we could use this algorithm would be giving these phrases as input to the LDA and secondly as further used in our visualization part to highlight those important phrases in the abstract which gives a rough idea of what the paper is about. Most of the LDA implementation till date uses the basic method of splitting the statements on spaces, which finally just gives bag of single words. As explained in the earlier sections, single words sometimes have different meaning as compared to when joined
with another word, which would give more meaning than just the single words. We could then join those words inside those phrases with a separator and then pass it to the LDA in the following manner

**EX:**

*Doc1:* simple: textual: variations: visual: programming: languages: semantic: comparison:

CHAPTER 4
VISUALIZING THE ACM DIGITAL LIBRARY

4.1 Literature Detail Summary

While visualizing the details for a literature, every part in the detailed summary was carefully designed in a way that could help the reader of that literature understand what the paper is and how the paper has been helpful for other users and how it can be helpful to the reader.

There are three main sections

4.1.1 Left Bookmark Section

The left bookmark section has bookmark links to different sections in the detailed summary viz. abstract, ancestor and descendant details, references and citations. On clicking any of the links the page does a smooth animated scroll [8] to the target.

![Figure 7: Bookmarks for different sections on the Literature Summary](image)

Figure 7: Bookmarks for different sections on the Literature Summary
4.1.2 Middle Detail Section

4.1.2.1 Abstract of the literature

Here we make use of the EMD algorithm to highlight the important phrases inside the abstract. Figure 8 is a very good example where EMD algorithm is used. The phrases “ownership types”, “equivalence classes”, “data structure”, “object encapsulation”, “multithreaded programs” gives a very good idea of what the paper is about. Consider the same phrases split into individual words “data”, “structure”, “object”, “encapsulation”, “equivalence”, “classes”, each word has its own different meaning, but when together with another word can change the meaning of the context.

**Figure 8: Abstract for the literature [2]**
4.1.2.2 Ancestor & Descendant Details

Figure 9: Ancestor & Descendant for the literature [2]

The figure above gives some details with regards to the ancestors and descendants of the selected paper. The first row has four numbers in each column. The first number denotes the maximum number of citations that a paper has in that particular generation’s collection of papers. The number in dark red is the number of papers having a certain threshold value of citations in that particular generation. Currently we set the threshold value internally on the code side and the threshold value for minimum number of citations is 10. The number with the percent sign gives the percentage of papers having a threshold value of citations in that generation. The last number denotes the minimum number of citations that a paper has in that particular generation. Each column in the second row gives the range of years for the referenced/cited papers in that generation and the number in the parentheses gives the total number of papers in that generation.

The third row shows the distribution of ACM Computing Classification System [6] in each generation. This is sometimes good to know, which category of papers were used by the ancestors and after the selected paper was published what category of papers cited this paper. In
this way we come to know the potential of this paper in different categories. Currently we are showing only top ten categories in each generation.

4.1.2.3 Conferences with Immediate Citations

<table>
<thead>
<tr>
<th>Year</th>
<th>Conference 1</th>
<th>Conference 2</th>
<th>Conference 3</th>
<th>Conference 4</th>
<th>Conference 5</th>
<th>Conference 6</th>
<th>Conference 7</th>
<th>Conference 8</th>
<th>Conference 9</th>
<th>Conference 10</th>
</tr>
</thead>
<tbody>
<tr>
<td>2002</td>
<td>OOPSLA-2</td>
<td>PLDI-3</td>
<td>PLDI-2</td>
<td>OOPSLA-1</td>
<td>TLDI-1</td>
<td>POPL-1</td>
<td>ICSE-1</td>
<td>LTCS-1</td>
<td>ISSAT-1</td>
<td>PLDI-1</td>
</tr>
<tr>
<td>2003</td>
<td>PLDI-4</td>
<td>OOPSLA-1</td>
<td>POPL-1</td>
<td>ICSE-1</td>
<td>SAND-1</td>
<td>TLDI-1</td>
<td>ICSE-1</td>
<td>PACT-1</td>
<td>ICSE-1</td>
<td>TLDI-1</td>
</tr>
<tr>
<td>2004</td>
<td>OOPSLA-4</td>
<td>TLDI-1</td>
<td>PLDI-2</td>
<td>OOPSLA-2</td>
<td>PACT-1</td>
<td>POPL-1</td>
<td>VEE-1</td>
<td>ICSE-1</td>
<td>ICSE-1</td>
<td>TLDI-1</td>
</tr>
<tr>
<td>2005</td>
<td>PLDI-2</td>
<td>OOPSLA-1</td>
<td>POPL-1</td>
<td>ICSE-1</td>
<td>ICSE-1</td>
<td>OOPSLA-1</td>
<td>VEE-1</td>
<td>ICSE-1</td>
<td>ICSE-1</td>
<td>TLDI-1</td>
</tr>
<tr>
<td>2006</td>
<td>PLDI-3</td>
<td>POPL-1</td>
<td>ICSE-1</td>
<td>ICSE-1</td>
<td>ICSE-1</td>
<td>OOPSLA-1</td>
<td>ICSE-1</td>
<td>ICSE-1</td>
<td>ICSE-1</td>
<td>TLDI-1</td>
</tr>
<tr>
<td>2007</td>
<td>OOPSLA-3</td>
<td>POPL-1</td>
<td>ICSE-1</td>
<td>ICSE-1</td>
<td>ICSE-1</td>
<td>OOPSLA-1</td>
<td>ICSE-1</td>
<td>ICSE-1</td>
<td>ICSE-1</td>
<td>TLDI-1</td>
</tr>
<tr>
<td>2008</td>
<td>PLDI-3</td>
<td>POPL-1</td>
<td>ICSE-1</td>
<td>ICSE-1</td>
<td>ICSE-1</td>
<td>OOPSLA-1</td>
<td>ICSE-1</td>
<td>ICSE-1</td>
<td>ICSE-1</td>
<td>TLDI-1</td>
</tr>
<tr>
<td>2009</td>
<td>OOPSLA-2</td>
<td>POPL-1</td>
<td>ICSE-1</td>
<td>ICSE-1</td>
<td>ICSE-1</td>
<td>OOPSLA-1</td>
<td>ICSE-1</td>
<td>ICSE-1</td>
<td>ICSE-1</td>
<td>TLDI-1</td>
</tr>
<tr>
<td>2010</td>
<td>PLDI-2</td>
<td>POPL-1</td>
<td>ICSE-1</td>
<td>ICSE-1</td>
<td>ICSE-1</td>
<td>OOPSLA-1</td>
<td>ICSE-1</td>
<td>ICSE-1</td>
<td>ICSE-1</td>
<td>TLDI-1</td>
</tr>
<tr>
<td>2011</td>
<td>OOPSLA-2</td>
<td>POPL-1</td>
<td>ICSE-1</td>
<td>ICSE-1</td>
<td>ICSE-1</td>
<td>OOPSLA-1</td>
<td>ICSE-1</td>
<td>ICSE-1</td>
<td>ICSE-1</td>
<td>TLDI-1</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Year</th>
<th>Conference 1</th>
<th>Conference 2</th>
<th>Conference 3</th>
<th>Conference 4</th>
<th>Conference 5</th>
<th>Conference 6</th>
</tr>
</thead>
<tbody>
<tr>
<td>2012</td>
<td>PLDI-1</td>
<td>PACT-1</td>
<td>ICSE-1</td>
<td>ICSE-1</td>
<td>ICSE-1</td>
<td>ICSE-1</td>
</tr>
<tr>
<td></td>
<td>TLDI-1</td>
<td>PACT-1</td>
<td>ICSE-1</td>
<td>ICSE-1</td>
<td>ICSE-1</td>
<td>ICSE-1</td>
</tr>
<tr>
<td></td>
<td>POPL-1</td>
<td>TLDI-1</td>
<td>ICSE-1</td>
<td>ICSE-1</td>
<td>ICSE-1</td>
<td>ICSE-1</td>
</tr>
<tr>
<td></td>
<td>ISCA-1</td>
<td>TLDI-1</td>
<td>ICSE-1</td>
<td>ICSE-1</td>
<td>ICSE-1</td>
<td>ICSE-1</td>
</tr>
</tbody>
</table>

**Figure 10: Conferences with immediate citations for literature [2]**

- The above figure shows the distribution of conference per year just for the first year descendants or immediate citers.
- Some common attributes in each column are as listed below:
- The conferences are sorted in descending order.
- The color line under each conference represents that particular conference. The color lines help finding papers from the same conference in other years. For example, consider the TLDI conference in 2003, papers from the same conference can be found by comparing the colored lines in the other years. This comes in handy when there are many papers from different conferences citing this selected paper, as shown above.
• On selecting any of the conference, it lists the paper citing the selected paper in that year. And on clicking that paper, it goes to the offset of that paper in the references and citations section.

4.1.2.4 References & Citations

<table>
<thead>
<tr>
<th>Conference</th>
<th>Year</th>
<th>Title</th>
<th>Authors</th>
</tr>
</thead>
<tbody>
<tr>
<td>PLDI</td>
<td>2001</td>
<td>Enforcing high-level protocols in low-level software</td>
<td>Manuel Fahndrich, Robert DeLine</td>
</tr>
<tr>
<td>POPL</td>
<td>1998</td>
<td>Classes and mixins</td>
<td>Matthias Felleisen, Matthew Fiat, Shiriram Krishnamurthi</td>
</tr>
<tr>
<td>TLDI</td>
<td>2003</td>
<td>Type-safe multithreading in cyclone</td>
<td>Dan Grossman</td>
</tr>
</tbody>
</table>

**Figure 11: References & Citations for the literature [2]**

• As shown in the above figure, this section lists the references and citations for the selected paper. Some peculiar attributes for each row are as follows
  • Each row is a cited/referenced paper.
  • Each row gives the title, abstract, year the paper was published, the conference, list of authors, keywords, number of references and citations.
  • Each row has a colored header bar, which represents the conference color.
  • As mentioned above in the previous section, clicking the paper link in the list, when the conference is clicked, takes us to one of these rows.
- On clicking the title of the paper, we are redirected to the publication detail summary for that paper.
- Links are given to the author too, if anyone is interested in exploring the author’s research.
- Any highlighted keyword is clickable in the abstract, and when we click the keyword, it reaches the next section “Right Search Result Section”. Note that we don’t have abstracts for some papers, since the ACM digital library didn’t have it on the paper detail page.
- Since we are not sorting by the number of citations, we convert the range of values of citations to the range of 80-150% of the actual font, which gives us the font-size for the numbers.

4.1.3 Right Search Result Section

When the user clicks any of the highlighted keywords, a right pane fades in to show the list of papers that have the selected keyword in their abstract.

- Each row here in the list is similar to reference/cited list in the middle section, except for absence of author, keyword, abstract, and some design changes.
- “View following papers side-by-side” opens up a modal box [12] and lists the papers side by side with the abstract.
Figure 12: Side-by-side Abstract comparison

- The Hide button fades out the right pane for search results section.

Figure 13: Search results when clicked on one of the highlighted words
**Figure 14: Full View of the Literature Summary**
4.2 Author Detail Summary

Sections

Initially in an author detail summary there are three sections, the list of classification [6], the list of keywords and the middle section which includes the years when the author published, the conference where he published, the list of papers that he published and the co-authors who he published the papers with. We will go in detail in each section.

4.2.1 Left & Right Sections

Figure 15: Author Detail Summary – Left & Right Column – Keywords & ACM Classification Listing [6] with their count and percentages for the author
The left section gives the list of keywords attached to the papers published by the author, while the right section gives the list of the ACM’s classification [6] for the same. Both of these attributes contribute equally to explore the categories of disciplines that the author is involved in and the kind of work done in the field of discipline. The numbers on the sides of each keyword or classification displays the count of the papers related to that keyword or classification, while the percentage gives the percent of papers in that category amongst the total list of papers. The bar on the top is to quickly visualize these numbers rather than reading the numbers and the percentages. Clicking on either of them will show the papers related to them.

4.2.2 Authors detail in the middle section

4.2.2.1 Left Conference Section

The left conference section lists all the conferences in which the selected author has published.

- Here each rectangle in the left section is a conference.
- The color of the rectangle represents the conference and comes from the pre-assigned values.
- The height of the rectangle depends on the number of papers published by the author in that particular conference and the sum of the converted range of values from 1-10 for the number of citations.
- Hovering over the conference gives some detail about the author’s contribution (no. of papers and the total citation count) in that conference.
- On clicking the conference a path is drawn between the conference and the paper with varying path widths where the path widths are calculates using the range conversion formula and then the page scrolls down to the first paper published in that selected conference.
### Middle Paper Section

The middle paper section lists all the papers published by the author:

- Similar to previous interfaces, here each row is a paper.
- The top color bar represents the color of the conference.
- Figure below is a short detail of the paper showing the title, conference year, citation and the reference count.
When you click a paper connectors are drawn between the conference and the co-authors as shown below.

![Diagram showing paper connectors between conference and co-authors]

**Figure 17: Authors details in the middle section**
4.2.2.3 Right Co-Author Section

The right co-author section as the name suggests is the list of co-authors that worked with the selected author.

- The list is placed in descending order of the number of papers published by both the author and the co-author.
- The number besides each author is the number of papers published by the co-author along with the selected author.
  
  Ex: (17/37) – Andrew Jensen Ko has published only seventeen papers with this author from his total of 37 papers.

4.2.2.4 Years Section

This section gives the list of years when the author had published. Besides each year the number represents the number of papers published in that particular year. Since the papers are stacked in ascending order of the year, clicking on a year scrolls down to that particular section where all the papers published in that year exists. There is even a marker drawn on the left side of the papers which shows the grouping of all the papers published in that year.
4.3 Conference

Figure 18: Initial UI design of the conference summary tool for the SIGGRAPH conference from 1985 to 1991

The conference summary was designed to learn more about the evolution of topic during a period in that conference.

4.3.1 Initial UI Design

The above image gives an overview of the initial UI design of the conference summary visualization for the SIGGRAPH conference from 1985 to 1991. Key attributes involved in this visualization are:

- Each column here represents a year and under each year we list the papers published during that year. Each row in that column lists a max of three papers and advances along
the y-axis if there are more than three papers in a year. And as the number of years increase, we advance along the x-axis.

- We use circles here to represent a single paper.
- The size of the circle here represents the no. of citations received by that particular paper. So more the citations larger the size of the circle and less the citations smaller the size of the circle.
- Note that we have two numbers on either side of the circle, the left number indicates the number of references and the right number indicates the number of citations for that particular paper.
- Since this visualization is just for a single conference, it won’t make sense to use the color assigned to that conference as every paper will have the same color. The color of the circle here represents the range of citations received for that paper.

<table>
<thead>
<tr>
<th></th>
<th>&lt; 10</th>
<th>&gt; 10 &lt; 50</th>
<th>&gt; 50 &lt; 100</th>
<th>&gt; 100</th>
</tr>
</thead>
</table>

- We have a top menu which gives three possible options
  - List of paper, when clicked toggles a left menu, which lists the papers from the selected conference during the start and end year period.
  - Search Paper lets you search a paper inside the selected conference for that period.
  - You can change the conference, start year and end year for a different conference and a different period.
When we click any of the circles, a 4-way description opens up. Next section gives a brief description of a 4-way description.

Hovering over a paper shows a tool tip [10] with the year conference and paper title in it.

### 4.3.2 4-way description

![Figure 19: 4-way description for [5].](image)

The figure above itself is self-explanatory as to why we named it as a 4-way description. The top two boxes show the ACM’s “Computing classification system” [6] categories and the keyword for that paper. The bottom two boxes give the list of authors and the abstract with the highlighted phrases. Besides the author’s name the numbers represent (number of papers publish by this author in this conference / total number of papers published including other conferences). On clicking any of the list members in the 4-way description box, the tool highlights the papers that have the selected keyword, CCS, author or phrases related to them and opens up a menu on the right which gives the list of papers highlighted on the screen. Clicking on the paper title on the right automatically scrolls to that paper the 4-way description for the paper opens up. There is a
reset button on the left to clear out any of the selections, paths or any other modifications made, which helps the user to explore a different path.

Figure 20: Filtering author, keyword or ACM classification [6]

4.3.3 List Of paper

The list of paper has a simple accordion type design with years as the header and on clicking the header it shows the list of papers in that year for the selected conference. The sole purpose of this list is to facilitate browsing through the list of papers by year. The figure below gives an example of the List of papers
Clicking on any paper in the list does a smooth animated scroll [8] to the offset of that paper, where offset is the X and Y axis of the circle that represents that paper. Along with the previous step, the tool opens up a 4-way description for the selected paper as described in the previous section.

4.3.4 Search Paper

This feature lets a user search for papers within the selected conference and during that period range. The search result is shown in the right pane that fades in [8] when “Search paper” is clicked. Figure 12 is the exact representation of the search results in the right pane. Similar to earlier visualizations, a link is given at the top to hide this pane and another link is given to explore the papers in the search results in a side-by-side comparison view.
4.3.5 Evolution path for a paper in a conference

One of the important inferences that we plan to derive is the evolution of a paper in the conference. To facilitate this, we give users the option to click numbers on either side of the circle, where the number on the left represents the references and the one on the right represents the number of citations.

Consider Figure 18

Figure 22: Initial screen before starting to search for evolution of a paper

We are going to consider the paper marked inside the red box. When the citation count is clicked we search for the citations of that paper in that conference and draw connectors between them showing their relation. Following figure shows the result after clicking the citation count. A blue empty circle is drawn under each of the connected papers. This blue empty circle is a connector; it is called a connector because when you click that circle it cleans out any unwanted earlier connections and tries to retrieve the citations for the paper, it is attached to, and draws connectors between them. At any point of time if you click a citation count for a different paper, it cleans out all the paths to the earlier paper. Notice that the citation count is 15 for this paper but it throws out only 8 connectors, this is because this paper has been cited only 8 times in this conference and the rest 7 are in other conferences.
Consider we click the connector for the second box. It then shows the selected path on the right.

As shown in the next figure there will be times when the path will stop, because there are either no citations for the last paper or if there are citations, but those citations doesn’t come under the selected conference.
Figure 25: Final path, where the path cannot exceed anymore

The path drawn above was for the citations of each paper; similarly this can be done for the references of the same paper. When we do this the path and the paper path is re-made from the first paper in the path to the last paper. This can be considered as the citation path for the first paper in the path or may be an ancestor to the descendant path for the earlier paper. When in a path we can even view the 4-way description for each paper in the path. This implementation gives the evolution of a paper inside a single conference; the same evolution for multiple conferences can be studied with citeology which is described in the next section.
4.4 Citeology

The name comes from the paper [1]. As said in the paper “Citeology is an interactive visualization that looks at the relationships between research publications through their use of citations.” [1]. Following excerpt from the paper explains why the author named it as “Citeology: Visualizing Paper Genealogy”,

EXCERPT [1]

‘If one considers a research article in a genealogical sense, the papers, which an article referenced, could be considered the article’s “ancestors” or “parents” and the papers, which referenced the target article could be considered “descendants” or “children”. … Such information can be useful when tracing the history of a piece of work or trying to find related articles.’

The main goal as mentioned in the introduction is to learn the evolution of a paper or a literature over the years, or the path of research for an author, or learning the kind of topics evolved in a particular conference or any other important inferences. Though the citeology paper [1] covers some part of what we have to achieve, there are some missing features which we can add on, to achieve our goal.

4.4.1 Initial UI Design

The image below is the output of citeology for the paper [2]. We will dive into the design of each element in the figure.
Each square is considered as one paper.

The size of the square varies from $5 \ldots 20$. The lesser the size of the square, lesser the number of citations and greater the size of the square more is the number of citations. We convert the range of values of citations to the range of $5\ldots20$ which gives us the size of the square.

The color of the square determines the conference of that paper.

Each column here, as mentioned above ancestors / descendants. The paper with the column header as “Selected paper” is the paper for which we are having the citeology. To the left of the selected are five generations of ancestors and to the right are five generations of descendants.
• In each of the ancestor/descendant column, the papers are sorted by year in the ascending order and placed under the year when it was published. And each row in a column can have a maximum of 5 papers.

• Hovering over a paper shows a tool tip [10] with the year conference and paper title in it.

• The top menu gives other options
  o No of Generations – lets the user select the number of generations to view.
  o Toggle CRH – Toggles the citation range highlighter
  o Year - Lets the user highlight papers for a selected year
  o Conference - Lets the user highlight papers for a particular conference
  o Paper Path - Toggle the visibility of the paper path details;
  o Reset – Lets the user reset any selections made while exploring the citeology.

4.4.2 Year Menu

![Years.png](image-url)

Figure 27: Citeology - Year Menu

The “years” menu in the top menu of the citeology lists the years of all the papers listed in the citeology of the selected paper. Besides the year in the parentheses is the count of papers in that particular year.
4.4.3 Conference Menu

Similar to the “years” menu, the conference menu lists the conferences of all the papers listed in the citeology of the selected paper along with the count of papers in each conference within the parentheses.

4.4.4 Year/Conference Selection

![Figure 28: Citeology - Conference Menu](image)

![Figure 29: Selecting "OOPSLA" conference highlights the paper in that conference and a track is drawn in the center of the screen.](image)
On clicking either a year or a conference in the menu shown above, the paper belonging to that particular year or conference gets highlighted while the opacity of the other papers are reduced, so that the highlighted papers are seen easily in the crowd. Another important feature is the center track as shown above in the Figure. A track is drawn in red in between the ancestors and the descendants i.e. under the selected paper, when a year or conference is selected. The horizontal line coming out of the track shows the direction where the papers are highlighted. This comes in handy when the year/conference you select has only few papers among the huge set of papers and you want to find those papers in the crowd. The track starts from the selected paper box and ends at the y-coordinate of the last element. Note in the above figure that the horizontal line and the track between two horizontal lines along the y-axis changes from lower opacity to higher opacity, which indicates that lower the opacity of the line, less the number of papers in that row and higher the opacity, more the number of papers in that row.

4.4.5 Studying the evolution:

Selecting any paper creates a path between the selected paper and the original paper for which the citeology was generated. Of course, there will be multiple paths to the selected paper, but we are considering the shortest path to that paper or the path in which the selected paper has the closest generation. Selecting another paper will erase the earlier path and draw a path to the selected paper. Notice that a right pane opens up when selecting a paper, which shows the list of papers in the path, the rest of the details is similar to the description for Figure12. Here the path may contain papers not only from the same conference but also from different conferences.
Figure 30: Citeology for the paper [2]
4.5 Search Detail Summary

The search detail summary can be considered as a small accessibility and visually based interaction improvement to the existing or traditional way of rendering the search results. Whenever a user is inside the tool, and wants to search for something, they just have to start typing; automatically a search bar comes up in the center of the page with different options to select from. This comes in very handy because the user will never have to go and click inside the textbox to start writing. Secondly, you don’t see the text box on the screen, so this will also save some screen space that can be efficiently used. The following figure shows the search bar in the middle of the page.

![Figure 31: Showing search bar in the center of the screen](image)

Once the search query is submitted, the search results will come up as shown in the figure below.
Figure 32: Search Results

The search results have an improved user interface design, showing only concise valuable information. It also pulls in the keywords, ACM classifications [6], year, authors and the conference for all the literature/paper in the search results. When the attributes for the literature/paper are collected they then grouped and shown in the form of values and percentages. The top left and right columns show the keywords and the ACM classifications [6] associated with the papers in the search results. The green bar on the top gives a visual representation of the percentage of papers associated with that particular keyword or the classification [6]. The row
below shows the years, authors and the conference with their respective count of papers in the search results. The author label also gives the total number of papers published. All the values under the paper attribute columns are clickable and which when clicked filters the papers in the center column. Though while filtering, the process of filtration takes place on the client side which reduces bandwidth consumption on the server side since it does not make a server call. Care was taken so that whenever a paper title or the author name was clicked it doesn’t leave the search page. Thus, a modal popup is used to load the page inside the pop up, making a way to return back to the search results. The following Figure gives an example of this work.

![Modal popup example](image.png)

*Figure 33: Opening a literature in a modal pop to preserve the search results*
CHAPTER 5
RELATED WORK

5.1 Digital Library User Interface

Many online article repositories like the ACM Digital Library, the IEEE Digital Library and many more, uses the traditional way browsing through the papers and selecting the most relevant papers for their topic. Apart from that there are some information visualization approach that have been implemented such as the 3D Vase museum, ActiveGraph scatter plot of citation data, AquaBrowser, semantic Treemap of web links in [19], use of Concept Map, Tree Map and Self-Organizing Map in [20] and interactive animated 2-D scatter plot used in [21]. As seen in the previous chapters and as mentioned in the introduction we use shapes, sizes, colors, layout and other interaction factors to better visualize and analyze the collective metadata from the ACM digital library. We use HTML5 and SVG elements to build these visualization tools, PHP and APACHE to run the server and MySQL database to store the data.

5.2 Phrase extraction

There are many different parsing techniques that have been used to extract meaningful phrases such as the noun phrases from a given text corpus [16], [17], [18]. The algorithm we wrote is based on a combination of both linguistic rules such as the morphological and the syntactic description of the text and the usage of a lexical database such as WordNet [13]. The use of WordNet [13] gives us a better way to get rid of irregular verbs. We even make use of punctuations which indicates where stops and pauses occur, so as to separate the phrases accordingly. With this approach we have achieved a much similar but, improved results.
5.3 Citeology

There have been many previous works that have tried to learn the citation trends for a paper using different information visualization techniques. Three such previous works are the butterfly system [22], CiteSense [23] and Citeology [1]. Most of these implementations are java based applications. Thus we have tried to create a system which achieves a similar system using some of the HTML5 based elements with many other additional features that lacked in the previous systems. The earlier system [1] was built upon two conferences with few papers in them. Our system exceeds those numbers by a huge amount due to which many of the visualizations done in the earlier system could not be implemented in our system. Some of those would be to list the first 25 characters as the node for the paper, which wouldn’t have been possible because of the huge data that we have. Secondly just showing the papers name in the tool tip [10] would be less information too as that one paper would be in midst of papers from so many other conference. Thus we provide the paper title, year and the conference in the tool tip [10]. Since there are many conferences in our dataset, which was not the case in the earlier system, thus we differentiate conferences with color. Other features that are not present in the earlier system is choosing all papers belonging to a year or a conference, toggling the citation range highlighter which helps to visualize the 4 ranges of citation count, selecting a paper from the path to view the citeology for that other paper, variations in the rectangle sizes to show visualize the citation count, a track in the center to find those highlighted papers in such a huge collection.
CHAPTER 6
FUTURE WORK

The EMD algorithm has used just a small chunk of rules in the English grammar. Since English Grammar can be considered as a vast pool of rules for the language, the EMD algorithm can be considered to be at an infant stage, where there is a lot of scope to improve the algorithm. The 4-way description helps to see the details of the paper in such a large collection of data, but the problem with the 4-way description is that it hides the papers that come in its path, leaving no way than to close that dialog to select the papers behind it. One other major improvement over the current work can be to apply topic modeling to the output of the EMD algorithm that was applied to the whole document of the literature instead of just the abstract and trying to visualize those results.

CHAPTER 7
CONCLUSION

We present an algorithm, which extracts meaningful phrases from the abstract of a paper rather than just single word keywords. We then provide an approach to visually navigate the ACM digital library with some web based visual interactive tools. One of the benefits of our approach is that, since everything is web based, these visualization tools can be used on almost all modern browsers. The suite of tools along with the EMD algorithm can be used to analyze the collective metadata from the ACM digital library and provide the user with different insights to explore many other important inferences of a paper’s historical metadata patterns.
REFERENCES


Example References


Tool References


**Web References**


**Related Work References**


