



**Linneuniversitetet**  
Kalmar Växjö

Master Thesis Project

Computational Analyses of  
Scientific Publications Using Raw  
and Manually Curated Data with  
Applications to Text Visualization



*Author:* Imran Shokat

*Supervisors:* Prof. Dr. Andreas Kerren,  
Kostiantyn Kucher

*Examiner:* Welf Löwe

*Semester:* VT 2018

*Course Code:* 5DV50E

*Subject:* Computer Science

## Abstract

Text visualization is a field dedicated to the visual representation of textual data by using computer technology. A large number of visualization techniques are available, and now it is becoming harder for researchers and practitioners to choose an optimal technique for a particular task among the existing techniques. To overcome this problem, the ISOVIS Group developed an interactive survey browser for text visualization techniques. ISOVIS researchers gathered papers which describe text visualization techniques or tools and categorized them according to a taxonomy. Several categories were manually assigned to each visualization technique. In this thesis, we aim to analyze the dataset of this browser. We carried out several analyses to find temporal trends and correlations of the categories present in the browser dataset. In addition, a comparison of these categories with a computational approach has been made. Our results show that some categories became more popular than before whereas others have declined in popularity. The cases of positive and negative correlation between various categories have been found and analyzed. Comparison between manually labeled datasets and results of computational text analyses were presented to the experts with an opportunity to refine the dataset. Data which is analyzed in this thesis project is specific to text visualization field, however, methods that are used in the analyses can be generalized for applications to other datasets of scientific literature surveys or, more generally, other manually curated collections of textual documents.

**Keywords:** scientific literature analysis, meta-analysis, trends, correlation, NLP, text mining, topic modeling, LDA, HDP, text visualization

## **Preface**

I would like to thank my supervisors Professor Dr. Andreas Kerren and Kostiantyn Kucher, who gave me a chance to work on an interesting, live project and learn new skills. I am especially thankful to Kostiantyn Kucher who managed to set several meetings during his busy time, from start to the end of this thesis project. He has been guiding me and providing feedback on the thesis project until the end. It was not possible for me to finish my thesis project without the guidance of Kostiantyn Kucher. At the same time, I am also thankful to thesis course manager Dr. Narges Khakpour for her time, guidance and feedback—she has spent her time with her students to highlight the shortcomings in our theses and guided us on how to create a scientific Master's thesis. Also, I would like to thank my examiner Prof. Dr. Welf Löwe for the feedback that helped me to improve my thesis report. Finally, I am also thankful to my friends and family for supporting me during this time and motivating me to finish it.

# Contents

<b>1</b>	<b>Introduction</b>	<b>1</b>
1.1	Motivation . . . . .	2
1.2	Problem Statement . . . . .	3
1.3	Method . . . . .	4
1.4	Contributions . . . . .	4
1.5	Target Groups . . . . .	5
1.6	Report Structure . . . . .	5
<b>2</b>	<b>Background</b>	<b>6</b>
2.1	Visualization . . . . .	6
2.1.1	Information Visualization . . . . .	6
2.1.2	Text Visualization . . . . .	6
2.1.3	Text Visualization Browser (TextVis Browser) . . . . .	7
2.2	Data Mining and Text Mining Methods . . . . .	8
2.2.1	Bibliometrics . . . . .	9
2.2.2	Temporal Analysis . . . . .	9
2.2.3	Correlation Analysis . . . . .	9
2.2.4	Content Analysis . . . . .	9
2.2.5	Natural Language Processing (NLP) . . . . .	10
2.2.6	Unsupervised Machine Learning . . . . .	10
2.2.7	Topic Modeling . . . . .	10
2.2.8	Latent Dirichlet Allocation (LDA) . . . . .	11
2.2.9	Hierarchical Dirichlet Processes (HDP) . . . . .	12
2.3	Technical Terms . . . . .	12
2.3.1	Unigram, Bigram, and N-gram . . . . .	12
2.3.2	Stop Words . . . . .	12
2.3.3	Stemming and Lemmatization . . . . .	13
2.3.4	Jaccard Index . . . . .	13
<b>3</b>	<b>Method</b>	<b>14</b>
3.1	Scientific Approach . . . . .	14
3.2	Method Description . . . . .	14
3.3	Reliability and Validity . . . . .	15
3.4	Ethical Considerations . . . . .	16
<b>4</b>	<b>Data Analyses</b>	<b>17</b>
4.1	Data Collection and Preprocessing . . . . .	17
4.1.1	TextVis Browser Dataset . . . . .	17
4.1.2	Collection of Raw Texts of Scientific Publications . . . . .	18
4.2	Temporal Analysis of the Labeled Dataset . . . . .	18
4.2.1	Extraction and Visualization of Overall Temporal Distribution Based on Publication Year . . . . .	18
4.2.2	Extraction and Visualization of Temporal Distribution for Individual Data Categories . . . . .	19
4.3	Correlation Analysis of Data Categories . . . . .	23
4.3.1	Extraction of Data Series for Data Categories . . . . .	23
4.3.2	Computation and Visualization of Correlation Coefficients . . . . .	24
4.4	Computational Data Analysis of Raw Scientific Publication Texts . . . . .	26

4.4.1	Collection of Raw Textual Data . . . . .	26
4.4.2	Preprocessing of the Raw Textual Data . . . . .	28
4.4.3	Topic Modeling of the Textual Data . . . . .	30
4.4.4	Experimentation with Topic Modeling Algorithms and Parameters . . . . .	34
4.4.5	Computational Matching of Topic Modeling Results to the Labeled Dataset . . . . .	38
4.4.6	Additional Matching Method Bypassing the Topic Modeling Stage . . . . .	41
4.4.7	Evaluation of the Computational Matching Results . . . . .	42
<b>5</b>	<b>Discussion</b>	<b>48</b>
5.1	Original Dataset . . . . .	48
5.2	Data Collection and Preprocessing Results . . . . .	48
5.3	Temporal Analysis Results . . . . .	49
5.4	Correlation Analysis Results . . . . .	49
5.5	Publication Text Analysis Results . . . . .	50
<b>6</b>	<b>Conclusions and Future Work</b>	<b>52</b>
	<b>References</b>	<b>54</b>
<b>A</b>	<b>Appendix 1</b>	<b>A</b>
A.1	List of Categories Used in the TextVis Browser Dataset . . . . .	A
A.2	List of Key Terms Used for Category Matching . . . . .	A
A.3	Yearly Statistics for Category Labels . . . . .	D
A.4	Overall Category Statistics . . . . .	D

# 1 Introduction

The use of the internet, social media, wireless correspondence, digital libraries, and electronic documents are increasing each day. A large collection of scientific publications are available online, and the sizes of these collections are increasing day by day [1]. This increase is depending on dedicated researchers who are working and publishing new articles in their respective scientific research domain. The problem is arising when we want to get insights from these publications, In order to get the idea of a topic of interest in a particular field. That means we will have to read several publications, but it is a challenging task. However, it is important to go through as many as publications in order to get deep insights from these publications for the topic of interest. There is a need for a smarter way to go through most of the target publications to extract the needed information. This smart way could be through providing some analyses or surveys of target publications after extracting the information from the publications. This could help the researchers to get the proper insight topic of interest without reading all target papers. This type of analyses of publications can also provide researchers with additional insights using the state of the art method in the corresponding field, highlight the existing trends and gaps in the literature and point out the opportunities for future work, which are all important for researchers. The potential methods to read such data by machines or automatic way, in order to get insights in a short time, could be using machine learning and topic modeling approach.

Text visualization is one of the examples of the scientific discipline that is taken into account for providing the surveys, and the ISOVIS Group developed a TextVis Browser. They want to provide an overview of text visualization techniques by using TextVis Browser to the researcher. Visualization of data is now becoming more popular as compared to last decades [2]. Visualization helps us to provide the quick abstract idea of our data or information in the form of graphs, diagrams and charts. Several visualization techniques are introduced for text visualization, and we can choose a technique on the base of our data type. Text visualization as a subfield of information visualization is now becoming important due to the continued availability of a large amount of data online in the form of books, videos, and social networks, etc. Finding the right information from this huge amount of data is now becoming difficult, and harder for the researcher to find the related work for their required field. The ISOVIS Group tried to handle this type of problem for text visualization techniques, and they developed an interactive Text Visualization Browser [2], which will help the researcher to get exact information and provide the state of the art in this field. They collected abstract details of about 400 scientific text visualization techniques for TextVis Browser. Several categories are manually assigned in TextVis Browser to these techniques/publications. They assigned these categories on the base of publications content.

In this thesis, we will find temporal trends related to individual categories in survey data over time. We will examine how the categories which are assigned in TextVis Browser how become less or more popular over time. Also, we will check if there is any correlation between pairs of categories in survey data. We will see the correlation between categories. How pairs of categories were used together in these scientific publications. Finally, we will check if there is any correspondence between the category labels assigned to text visualization techniques manually and

prominent topics described in the corresponding text visualization publications by using machine learning, topic modeling approach.

## 1.1 Motivation

Firstly, we can give different motivations for visualization techniques. Hundreds of visualization techniques online available nowadays. If a person (student, researcher etc.) is new in visualization of data field and want to use visualization techniques, it is hard for him/her to choose a best possible visualization technique among hundreds which fits for his/her tasks. Lu and Liu [3] describe that we need to provide analysis and summarization of such online available large data.

A problem similar to the one discussed in the present thesis has also arisen in sentiment visualization as there are more than a hundred sentiment visualization techniques available. Kucher et al. [4] developed a web-based interactive survey Browser for sentiment visualization techniques. They manually labeled the categories for each technique. They collected 132 visualization techniques from peer-reviewed publications and categorized them in 35 categories. They discussed the state of the art in sentiment visualization and opportunities for future research in this survey. However, their dataset was quite small and was just relevant to sentiment visualization techniques. Federico et al. [5] conducted a survey on visual approaches of patents and scientific articles. They summarized the state of the art in the visualization of patents and scientific publication. Their survey is categorized into two aspects: data types (text, citations, authors, metadata) and analyses types. They computed the temporal analysis to find the patterns and relations between the entities for visualization of patents and scientific publications. Federico et al. [5] conducted this survey for information visualization, data visualization, and visual analytics but in this thesis project, our work is for text visualization approaches (tools). Suominen and Toivanen [6] conducted a comparison of unsupervised learning and human-assigned subject classification. They analyzed the scientific publications by using the topic modeling approach to classify scientific documents. They compared topic modeling results with manually labeled data of ISI-WoS and OECD classifications from 1995 to 2011 to validate the human-assigned topics with unsupervised learning (topic modeling). They illustrated the trends of topics how they grow, stable or decline year to year. Yau et al. [7] measured the evaluation of four (K-means, LDA, CTM, HDP) topic modeling algorithms for seven scientific publications. They compared the results of four topic modeling algorithms with the same dataset and concluded that the performance of HDP algorithm is better than others. According to Yau et al. [7], we can classify documents by using topic modeling with significant accuracy. Chen et al. [8] applied topic modeling to visualize collaboration over time from document metadata. They presented the temporal analyses over topics by implementing LDA topic modeling approach. Griffiths and Steyvers [9] applied a topic modeling algorithm (LDA) to find the relationships between different scientific disciplines. A purely unsupervised machine learning approach is used to discover the scientific topics and, find the trends and “hot topics”. They used the dataset of Proceedings of the National Academy of Sciences of the United States of America (PNAS) from 1991 to 2001 of scientific papers. They argue that Latent Dirichlet Allocation (LDA) is a statistical model that is suitable for any type of documents to discover the topics, meaningful trends and visualize their content. Natural language processing algorithms are used to extract the information from the

textual data. To provide the summary of textual data, first, we need to analyze the textual data and find the latent topics from the text. Havre et al. [10] introduced the ThemeRiver system, which is developed to visualize the thematic trends, patterns, or topics from the corpus. They visualize and discover the topics from the corpus, and uncover the hidden trends, themes or topics over time. They used a large collection of data and show that which themes most used over time and visualize it in the form of a river. In our implementation, we will show the trends in the form of histogram for a specific field of interest. The Blei [11] identifies that Latent Dirichlet Allocation (LDA) is a popular approach to discover latent topics from the text document. To discover topics from the textual data is called topic modeling. We can uncover hidden relationships by using topic modeling between the documents after getting the abstract topics from the document.

Topic modeling technique is now commonly used in scientific publications to find the trends or hidden relationships of different topics. The ISOVIS Group [2] conducted some analyses in the past to investigate the trends/popularity of different approaches. Their manually labeled data set is much larger now, so they are interested in conducting more analyses to get new insights about the state of the art in text visualization. Also, want to see how different approaches became more or less popular over the years and how they interacted with each other. The categories/labels were all assigned manually based on their expertise in this field of research, so now we are also interested to see if their manually labeled data corresponds to the topics/concepts/clusters extracted from the texts of the publications themselves.

The dataset that is used in this thesis is arguably the largest curated collection of text visualization techniques (used in peer-reviewed scientific publications) in the world. Its analysis is interesting and important for researchers in text visualization. On the base of our least knowledge, nothing like this exists for this particular application (survey analysis and evaluation of scientific publications about text visualization).

## 1.2 Problem Statement

The currently available manually labeled data about scientific publications for a particular subfield (text visualization) lacks summarization with regard to its temporal and topical contents. The researchers in this subfield require such summaries to gain insight into the current state of the scientific subfield and outline the future research work. The aim of this thesis project is, therefore, to solve this problem by applying several types of computational analyses for the manually labeled data and corresponding raw textual data. The research questions to be answered in this thesis project are presented in Table 1.1.

<b>RQ1</b>	What are the temporal trends related to individual categories in survey data over time?
<b>RQ2</b>	Is there any observable correlation between pairs of categories in survey data?
<b>RQ3</b>	Is there any correspondence between the category labels assigned to text visualization techniques manually and prominent concepts/topics described in the corresponding text visualization publications?

Table 1.1: Research Questions



The results of this project will provide new insights about the current state of scientific publications in text visualization field and will be useful for the scientific community. It will provide the summaries to researchers with regard to its temporal and topical contents and outline the future research work. Furthermore, some of the analyses will be generalizable to other datasets related to scientific publications.

### **1.3 Method**

The methods used in this thesis project to find the answers for research questions are briefly described below.

First of all, a literature review is carried out to see the latest approaches and how others solved the similar problems. We followed the well-known algorithms and methods to compute the results for research questions. In order to answer the research questions, we used the dataset of TextVis Browser provided by ISOVIS Group. Data analyses are computed for RQ1 and RQ2, and a software implementation is carried out to find the answer for research question RQ3.

The dataset is analyzed to find the answer for research question RQ1 and temporal analyses are computed for each category. Python programming language is used to extract and preprocess the data. The temporal trends are visualized using spreadsheet software (MS Excel). We found how each category became less or more popular during previous 27 years. The result of temporal analyses show that few categories became more popular in the previous 27 years.

Correlation analysis is computed in research question RQ2. Dataset is further analyzed to compute correlation analysis and a categories matrix is created to visualize the results. We are interested to see if some categories are used together or maybe they "compete" with each other. Pearson's coefficient is used to estimate the linear correlation between pair of categories. The results of correlation show that more categories have positive correlations between each other as compared to negative correlations.

A software implementation has been created to compute the answer for research question RQ3. First of all, we have collected the raw texts of scientific publications using online resources which are used in the TextVis Browser dataset. We preprocessed the raw text of these publications to remove the trivial data from the text. We used two well-known algorithms, HDP and LDA, for topic modeling. A third approach based on simple text matching is also used to test whether topic modeling is necessary for this analysis at all. The results of these analyses are presented to the ISOVIS Group members to investigate the discrepancies in manually assigned categories and also in the TextVis Browser dataset.

### **1.4 Contributions**

The results of this project provide a summary of text visualization techniques with regard to temporal and topical contents, which was previously unavailable for this dataset. The results of the comparison between the human-labeled dataset and topic modeling output can help the ISOVIS Group to validate manually labeled data, detect discrepancies and analyze them in more detail, and perhaps help them to refine the manual categorization that is used for the TextVis Browser's survey dataset. Overall, the results of this project (statistics, correlations, trends, text topics) can provide them with a better overview of the field in general, and also identify gaps and opportunities for future work in text visualization. Furthermore, the approach

applied in this project is generalizable to scientific publications and can be re-used in other fields and disciplines for textual data.

### **1.5 Target Groups**

The target groups are researcher, practitioners, students and people who are new in text visualization and want to know about text visualization in a short time. By viewing the results of this thesis project they can easily imagine which text visualization techniques are becoming less or more popular now and also they can find the temporal trends of categories individually. Also, they can see the correlation between pairs of categories, how pairs of categories are used together in Text Visualization Browser over time.

### **1.6 Report Structure**

The thesis work is divided into six chapters. Chapter 1 is already described, which is about the introduction of this thesis project, the background of the problem area, motivations for doing this project, problem statement of the thesis, thesis report, contributions, and target groups. Chapter 2 is about the background of the problem. In this chapter, we will discuss the existing work, what had been done before this thesis project for this problem. In Chapter 3, we will discuss which scientific approaches are used to get the thesis project results and also the methods are described in detail which was used in the implementation. Also, we will check the reliability and validity of the thesis project results and we will describe if there is any ethical consideration in this thesis project. The actual data analyses and their results are described in Chapter 4. The evaluation of the project results based on the further interpretation of the analytical results by domain experts is presented in Chapter 5. The conclusions of this thesis project and future work are discussed in Chapter 6.

## **2 Background**

In this chapter, we will present the background information about text visualization field in detail, where the data is used for this project originates from, and the information about various analytical methods and tools that were used throughout the rest of this report to discover trends and insights from the data.

### **2.1 Visualization**

Visualization in general sense is any technique or method that is used to communicate a message using diagrams, images, or animations. Nowadays interactive computer-aided visualization [12, 13] is becoming more popular in various applications and computer systems. We can explain several words, information, results, and ideas easily in one 2D or 3D representation by using visualization. Visualization is a new way of communication. The use of visualization is increasing in all kind of fields, like in medical, business, finance, or marketing. There are several fields and subfields of visualization as a scientific discipline, for example, scientific visualization, information visualization and text visualization [1, 2, 12, 13, 14].

#### **2.1.1 Information Visualization**

Information visualization is a process/study that is used to visualize (e.g., in the form of dashboards and scatterplots) the abstract data to provide insights. This data could be numerical and non-numerical. Information visualization [12] aims to amplify human cognition by using interactive, computer-aided visual representations of data. Typically the data is abstract (i.e., non-spatial) in this case, including multidimensional numerical datasets, networks, and relationships, or texts. Manovich [14] explains that information visualization is a process to work with data and uncover the hidden relations of data. Information visualization [13] is helpful to visualize the data in abstract information, such as visualize information on the World Wide Web, documents, and computer file systems. We can use different types of data in information visualization such as numerical, or non-numerical and text. The information visualization helps the end user to see a lot of information just in one graph, chart or diagram.

#### **2.1.2 Text Visualization**

Text visualization [1, 2] is a subtype of information visualization. Text visualization is a technique that is used to visualize trends and insights from textual data in the form of graphs, charts, timeline, etc. The digital data, like digital libraries, social networks, archived reports the amount of data is increasing. To read this large digital data is becoming impossible for a human. Now we need to find a way to save the reading time. Text visualization is a technique to read and analyze the textual data by using a computer technology and find the hidden statistical patterns from the data. We can visualize the results of these analyzed data through graphs, charts, maps, word clouds, timelines, etc. Text visualization shows the graphic view of the textual data and it helps us to find the hidden trends, and themes from the textual data. Wise et al. [15] explain that text visualization represents text information in the visual representation and it uncovers the hidden patterns and relationships in the document.

Furthermore, ISOVIS Group developed an interactive TextVis Browser for text visualization publications. We use the TextVis Browser’s dataset to further analyze it and get the state of the art of text visualization techniques.

### 2.1.3 Text Visualization Browser (TextVis Browser)

The ISOVIS Group [2] at Linnaeus University developed a web-based interactive visual survey Browser of text visualization techniques<sup>1</sup>, which can help researchers and practitioners to get an idea about text visualization field and find the related work on the base of various categories. TextVis Browser includes numerous manually curated entries corresponding to visualization techniques for raw textual data or text mining results. Each visualization technique is assigned to several labels corresponding to different categories such as “Data Source” or “Visual Representation”, as depicted in Figure 2.1 and also shown in Appendix A.1. The dataset of the Browser consists of 400 categorized visualization techniques as of August 21, 2018, and is expected to eventually grow in the future.

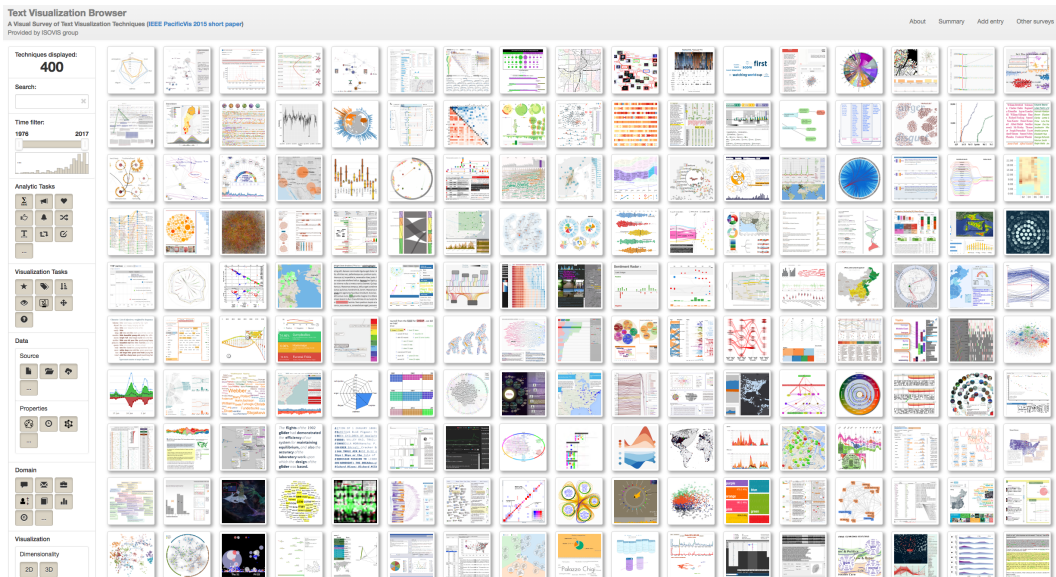


Figure 2.1: Web-based Text Visualization Browser by Kucher and Kerren [2] (last accessed on August 21, 2018)

In Figure 2.1, the right side of the figure shows the thumbnails of each scientific text visualization technique which were published during the previous 27 years. There are 400 text visualization publications, which are used in this TextVis Browser. Researchers and visualization community can view the further detail after clicking a thumbnail. After clicking a thumbnail, we can see the detail of publications about authors, publication year, publication URL to see the full publication in detail and manually assigned categories icons to that publication. The left-hand side of the above Figure 2.1 is the manually labeled categories, which are given by ISOVIS Group. By clicking any labeled category we can find the relevant publication on the right-hand side. For example, If a researcher or practitioner want to see the publications about sentiment analysis, he/she will just click on sentiment analysis category icon in the left-hand side then TextVis Browser will show the sentiment analysis related publications in the right-hand side. Similarly, all of these

<sup>1</sup><http://textvis.lnu.se>

41 categories help the researcher to find the relevant publications on the base of his interest.

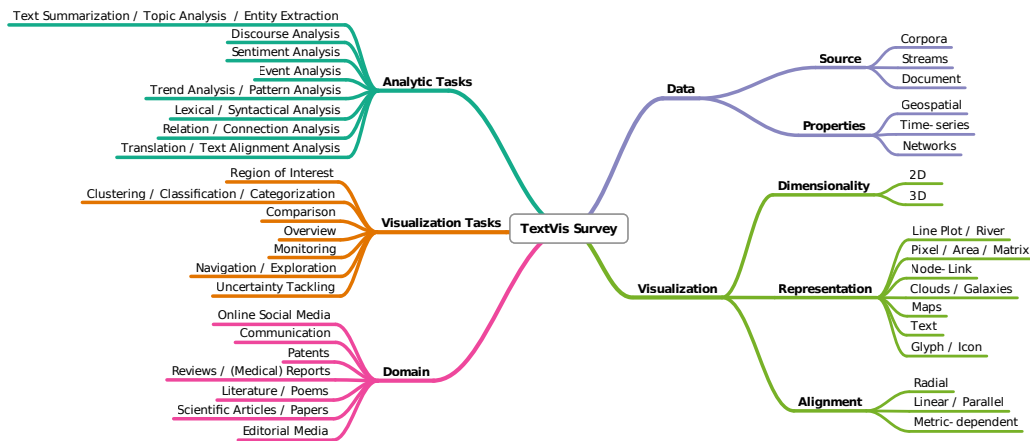


Figure 2.2: Taxonomy of text visualization techniques by Kucher and Kerren [2]

The above Figure 2.2 shows the taxonomy/categorization of text visualization techniques. These are 41 categories which are manually labeled by ISOVIS Group [2]. They manually assigned several labels to each visualization technique. They divided these categories into five main categories (analytic tasks, visualization tasks, domain, data, and visualization) and these main categories further have subcategories.

The ISOVIS Group conducted some analyses of this data previously, a portion of these analyses relied on metadata such as publication year or statistics of the assigned categories, and some would require careful analysis of the content of papers. These analyses were done manually and they followed the simple approach of doing it.

Besides the manual analysis of texts, there are other ways of identifying important concepts in textual data automatically, and one such method is topic modeling. This topic modeling method will extract the important topics from textual data after doing several analyses of the data. So, we will apply topic modeling to this visualization techniques dataset. This automatic technique for data analysis will help us to validate the results with the manual analyses. The description of topic modeling is discussed in Section 2.2.7.

## 2.2 Data Mining and Text Mining Methods

Data mining [16] is a discipline which is used to analyze the large dataset and find useful information from it. The use of data mining is increasing in different fields like science, medical, industry, marketing and finance. The text mining is also a type of data mining however it is specific only for textual data. The text mining is a set of technique, which is used for natural language processing text data to uncover the hidden structures, themes and patterns using the machine. Several methods or approaches are discussed in the literature for doing the data and text mining. There are several examples existing in literature with different kind of data. Bibliometrics is one example of approaches used for scientific literature analysis.

### **2.2.1 Bibliometrics**

Bibliometrics [17] is a measurement for text and information. Previously, the bibliometrics was used to discover the history of academic journal citations. According to Bellis and Nicola [18], the bibliometrics is used in the library and information science to provide quantitative analysis of academic literature. Also, bibliometrics is helpful for researchers to discover the hidden patterns and relationships from a large amount of historical text data. The good example of bibliometrics is Database Information Visualization and Analysis system called DIVA [19], which is used to visualize the document. This system explores the relationships of documents and provides a summary of each document.

### **2.2.2 Temporal Analysis**

Temporal analysis or temporal statistical analysis is used to find the behaviour of a variable in the dataset over time. By using temporal analysis, we can check how a variable becomes less or more popular over time. It can highlight the trends in given time. We can show these trends in the form of graphs or plots. Türkeş and Murat [20] computed the temporal analysis of annual rainfall variations in Turkey. They examined data ranging from 54 to 64 years, during the period 1930–1993 and analyzed how rainfall trend changes over time in Turkey.

### **2.2.3 Correlation Analysis**

Correlation analysis [21] is used to measure and interpret the strength of associations between several variables in a linear or nonlinear way. In its simplest form, correlation analysis is a method that is used to evaluate the linear relationship between two variables. Linear correlation between the variables could be positive or negative. We can check how two variables are used together within a certain period of time in the data. If they have a strong positive correlation with each other, it means they are used together in a similar way (e.g., both variables increase their values over time with a similar rate), and if they have a strong negative correlation, they are not used together in the data in a similar way (e.g., the values of one variable decrease as the values of another variable increase at the same time). The Pearson correlation coefficient [22] is commonly used to measure linear relationships between two variables. Linear correlation can be applied for our data analysis purposes to investigate whether certain pairs of categories co-occur in the dataset.

### **2.2.4 Content Analysis**

Content analysis is a technique [23], which is used to compress the large data into small content categories. According to Stemler and Bebell [24] content analysis is a method that is used to analyze the textual data and discover the patterns and trends from a document. The use of content analysis is increasing in different fields, for instance, it is used to examine the patterns in communication, find the answer of surveys and labelling the documents automatically by using machine learning approaches.

### **2.2.5 Natural Language Processing (NLP)**

Natural Language Processing [25] is a process which allows the system to understand the human language. By using NLP, we can translate the natural language spoken by humans into machine representation format. We can analyze a huge amount of information stored in free text files using NLP. NLP technology is commonly used in the healthcare industry in several organizations to improve patient engagement. There are several open source libraries available for NLP, however, the Natural Language Toolkit (NLTK) [26] is one of the most widespread libraries used for such purposes. NLTK is a suite of Python [27] modules (libraries), data types, corpus samples, and tutorials for NLP. NLTK is commonly used for research and teaching purposes in order to learn NLP.

### **2.2.6 Unsupervised Machine Learning**

Unsupervised machine learning is the subpart of machine learning to find the hidden patterns, relations between unlabeled data. Topic modeling is a good example of unsupervised machine learning (by using it, we can find hidden relations, trends, and patterns etc.). Gensim [28] is a Python module, which has a large range of machine learning algorithms for supervised and unsupervised problems.

### **2.2.7 Topic Modeling**

Kucher and Kerren [2] argued that due to high availability of large amount of data in different forms (e.g, images, videos, social networks, scientific articles, and books), it becomes more and more difficult to extract useful information from these large data collections. We need some tools and techniques to handle this problem and shows us exact information from this large amount of data. The topic modeling approach is well-known to handle this type of problems and can read a large amount of textual data in a short time. Topic modeling are algorithms (methods) which can organize and summarize the large collection of data. We can apply topic modeling algorithms to different types of data, for example, images, social networks, genetic data, textual data, streams, etc. Topic modeling algorithms discover the hidden topical patterns which are given in the collection, these algorithms don't require any annotations, the annotations will be given to the documents by algorithms while doing topic modeling according to the topics. It will use these annotations to organize, search and summarize the texts. We also don't need to apply the labelling of the documents, the labels will automatically become visible from the analysis of the texts. Topic modeling is also good for document clustering, organizing a large amount of textual data.

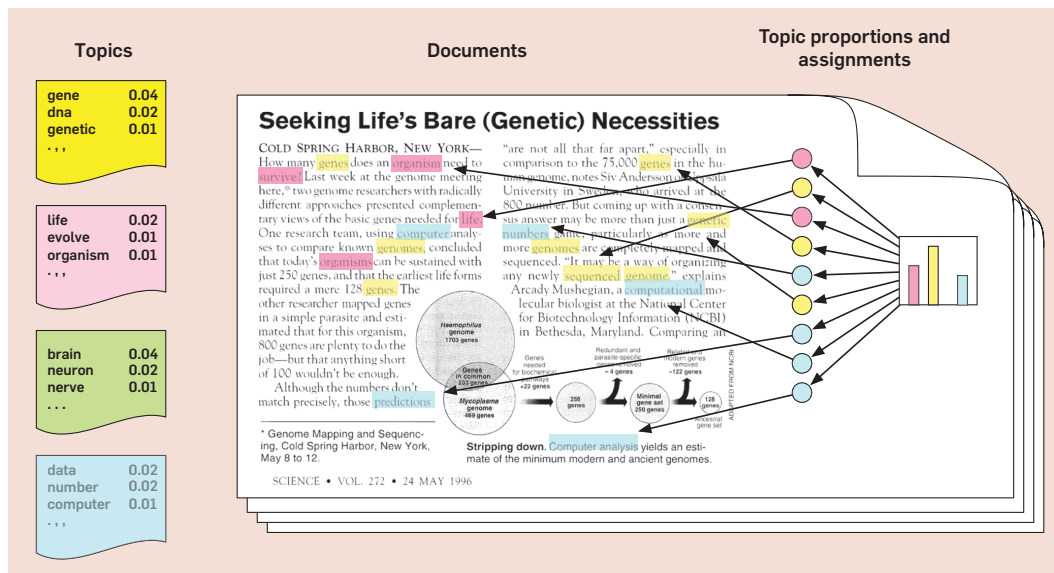


Figure 2.3: Example of Topic modeling by Blei [29]

In the above Figure 2.3, a simple topic modeling example is shown for textual data. By using topic modeling approach it is possible to automatically discover the topics from documents and find the hidden structure of the textual data. In the above Figure 2.3, the left side shows the mostly occurred topics which are discovered from the paper with their weights. The high weight of a topic shows that this topic is mostly used in the paper as compared to other. There are several approaches are commonly used for obtaining topics from a text, however, the most famous topic modeling approaches are the Latent Dirichlet Allocation (LDA) [11] and Hierarchical Dirichlet Process (HDP) [30]. These approaches are used to discover the topics that are present in the corpus. Several open source libraries exist for topic modeling with LDA and HDP to discover the topics from the text, such as MALLET, Scikit-learn, and Gensim. We have used Gensim [28] for topic modeling tasks in this thesis project. Gensim is a Python library developed by Radim Řehůřek and initially released in 2009. It is a powerful library that has the ability to handle large textual data. Gensim provides fast, memory efficient, and scalable implementations of both LDA and HDP algorithms.

### 2.2.8 Latent Dirichlet Allocation (LDA)

LDA [11] is a generative probabilistic model. It is used to discover the topics that are present in the corpus by constructing a statistical model of a document collection. It will get the frequently used words from the document. LDA model extracts a number of words and topics from a document by sorting them with a "weighting" variable. The weights are computed based on a number of times a word has occurred in the document. The words which are used more time in the document they have more weights. LDA show the topics with more weights in top topics. We can get the top most used keywords/topics from the document by using the LDA model. The algorithm will discover the topics from the whole corpus but it depends on the user how many topics he/she want to see from it. However, it will show the top topics/keywords in the top of the topics. The term "topic" represents term/row that contains a certain number of words/keywords.



### **2.2.9 Hierarchical Dirichlet Processes (HDP)**

HDP is an algorithm/model that is used for unsupervised analysis of grouped data. It shows the number of topics from the data. Teh et al. [30] present Hierarchical Dirichlet Processes (HDP) model for topic modeling. According to them, we can cluster problems by using HDP for different kinds of corpus. They explain that HDP is a Bayesian model. They compare the HDP with other models using three text corpora and found that HDP shows effective and superior performance results as compared to other models. HDP model takes random and unique topics from the corpus that are mostly occurred in the document. Similar to LDA, we can get the number of top words and topics with weight from the document using the HDP model.

## **2.3 Technical Terms**

In this section, we will explain the major technical terms or technologies which are used to analyze the data and solve the research problems. As mentioned above, in order to analyze the TextVis datasets, the Python [27] language was used for most tasks. We also used several standard Python modules as well as external libraries for our data analyses, such as Gensim [28] for topic modeling, NLTK [26] for reading the textual data, Pandas [31] to store the list of documents, NumPy [32] to calculate the numerical values, the JSON module to save data into JSON format, the glob module to read all files from a folder, etc.

### **2.3.1 Unigram, Bigram, and N-gram**

The n-gram model is used to get a sequence of n items from the text corpora, mostly it is used in Natural Language Processing. In n-gram, the size of n starts from 1,2,... and so on until n. For instance, if we give size 1 to n, it is called unigram and if we give size 2 to n then it is called bigram. For example, in text corpora, if we use the unigram model it will get the single words from the text. However, if we use the bigram model, it will get the words which are used together in the corpora. In our implementation, we will use bigram to get better topic modeling results. We used bigrams to get sequences of two adjacent elements from the document (e.g., "text visualization"), so by using bigrams we can see the dependency relationships and get more relevant words. However, if we use unigrams, it will only consider separate words. The trigram model can capture the usage of three word groups, but the probabilities of such groups' usage is small, while the performance of trigram calculation is much lower than for bigrams. That's why we have decided that bigrams is a good option as compared to unigrams and trigrams. The bigram model is used in preprocessing for topic modelling to automatically get common words from sentences of the given corpus. Gensim provides the functionality to capture the bigrams from textual data.

### **2.3.2 Stop Words**

Stop words [33] are the extremely common words that commonly appeared (e.g., is, am, are, was, were, has, have, etc.) in the text but they are trivial in a text. Stopwords usually include on conjunctions, prepositions, pronouns, etc. To gain better topic modeling results stop words should be removed from the text. This process will

save processing time for getting topics from the remaining text and also improve the topics accuracy. In our implementation, we will remove stop words from text visualization publications to gain better topic modeling results.

### **2.3.3 Stemming and Lemmatization**

Stemming and lemmatization [25] are preprocessing methods that aim to map the original forms of words in textual data to more general forms that might be more appropriate for automatic analyses. For instance, the words "hears", "heard", and "hearing" have meanings that could be considered identical in many cases, but simple string comparison operations would not identify them as such. Stemming [33] is a method which is used to remove the suffixes and identify the root/stem of a word. For instance, if we have similar words in our textual data, like the match, matches, matched, matching. However, after stemming it will merge every word into "match". Lemmatization [34] is a procedure to remove the inflectional ends from the document to produce the best results and returns the dictionary form of a word. It identified the lemma to the inflected word form in a document. By doing lemmatization to a document it will accept only 'noun', 'adjectives', 'verb' and 'adverb' from the document. Lemmatization can be applied, for instance, as a normalization method for document clustering [35], and similarly, this process can improve the topic modeling results.

### **2.3.4 Jaccard Index**

The Jaccard index [36] is a statistic to compare two different sets and check how these sets are similar. The result of the Jaccard index will return the ratio of the intersection of two sets over union of them. We will use the Jaccard index to match the manually labeled categories data with topic modeling results.

## 3 Method

In this chapter, we will describe the scientific approaches that are used to answer the research problems of this thesis project. Also, we will discuss in this chapter how we achieve reliability and explain how we validate the results of this thesis project.

### 3.1 Scientific Approach

In order to answer the research questions, we used several research methods. We reviewed the literature to analyze the existing solutions for the similar problem and gain an idea of how they solve similar problems. The dataset of TextVis Browser was provided in the form of a JSON file, and it contained both qualitative and quantitative data. We further analyze this dataset with both quantitative and qualitative methods and present the results in the form of listings, tables, and charts. The evaluation of the results is carried out through discussions with the domain experts in text visualization, thus following the qualitative approach.

### 3.2 Method Description

The particular source data which is used in this thesis project is based on the meta-data from a scientific publications survey on text visualization, including category labels from a set of 41 categories manually assigned to each survey entry/publication. We can imagine that reading 400 publications on this subject is an extremely time-consuming task for a researcher who wants to get acquainted with this research field. The applied results of this thesis project will, in fact, provide concise summarizations of the complete dataset. These summarizations (statistics, correlations, trends, and text topics) can provide us with a better overview of the field in general, and also identify gaps and opportunities for future work in text visualization.

In order to achieve the project results, the Python programming language is used to extract and preprocess the data, and the subsequent analysis is carried out with a spreadsheet software (MS Excel). Temporal analysis is used for the research question **RQ1** to learn how different categories were used over time, in other words, if some approaches became more or less popular. Correlation analysis is used for the research question **RQ2** to learn if some categories are used together or maybe they "compete" with each other. In order to answer the question **RQ3**, a software implementation has been created with Python and various text mining libraries were used to retrieve and process the text of scientific publications. Topic modeling with two different algorithms is then used to examine(analyze) from which structure we can gather raw textual data in an unsupervised fashion. The extracted results are then matched to manually labeled data with a proposed approach to check if there is a fit. This helps us to validate manually labeled data, detect discrepancies, analyze them in more detail, and finally, it can help the domain experts to refine manual categorization/labelling.

The evaluation/validation of this work includes experimental results for the topic modeling stage, a comparison to the existing meta-analyses of text visualization subfield, and a discussion with experts in text visualization. The feedback of data analysis and results is received in an iterative way during the stages of collecting the data and implementing the analyses. Also, when final results are presented to domain experts, then received feedback on its various aspects. So, it is considered similar to a semi-structured interview.

### 3.3 Reliability and Validity

Reliability and validity are two important factors of the project results, which were carefully considered when answering the research questions. The dataset of TextVis Browser was provided by the ISOVIS Group. This thesis project uses a specific version of the dataset copied on January 19, 2018, which contains 400 corresponding to scientific publications. The publications are already published and are available online. The content of these publications is not going to change in future in a critical way, and the metadata such as the publication year is included in the dataset file provided by the domain experts from the ISOVIS Group. The implementation of this thesis project is carried out with reliable methods as previously discussed in 2.2.7 section. The results of the research questions **RQ1** and **RQ2** would always be the same because the results are collected on the base of numerical values for a specific snapshot of the dataset. The analyses could (and probably will) be repeated for the updated dataset in the future in order to discover new insights. For research question **RQ3**, we used special data pre-processing techniques as a first step to get meaningful results. For instance, we used stopwords, tokenization, lemmatization, and bigram computation to remove the unnecessary and trivial data from the text. The well-known LDA and HDP algorithms were used to carry out topic modeling. While these algorithms are based on probabilistic methods and their output can vary between executions, we run these algorithms multiple times to gain reliable results. The variations in the topic modeling output would not affect the overall trends and insights discovered by the domain experts from these results. Additionally, we measured CPU execution time and memory usage while doing the topic modeling.

The validation of the project results is specially considered in this thesis project. The *Internal Validity* of the project is taken into account with regard to various analyses and computations. We run algorithms several times to validate if results are same in every time. We have provided different parameters to both algorithms for the same document to see the different results to assure which number of topics will provide good results. After deciding a number of parameters we have run the same document several times in different days to see the difference between the results but always it shows the same results for a document. The gained results of topic modeling were compared with manually labeled data and the resulting values have some relevancy. We can say that the results are promising as we had expected. The internal validity concerns are also addressed through the discussions with the domain experts (see below), who interpreted the results and found them to be reasonable.

The *External Validity* of the thesis project is a bit difficult to estimate since we could get different results using different topic modeling algorithms for this dataset. It also depends on the programming language, environment, CPU, memory, etc. If we used different algorithms, methods, and data, then results could be different. Therefore, we focus on a specific dataset to avoid overgeneralization in this thesis project. Nevertheless, the methods and algorithms which are used to compute the results could be applied to other types of textual data as part of future work.

The validation of the results of this thesis project is also carried out by the members of the ISOVIS Group who apply their domain expertise of text visualization field to make sense of the analytical methods and results for this particular dataset. It is possible that their involvement in the preparation and maintenance of the TextVis Browser dataset might provide certain biases for the interpretation of results. On the other hand, they are the main intended target group of this project who can make

use of its results, therefore, their opinions are critically important for validation and evaluation.

### **3.4 Ethical Considerations**

The ethical consideration has been followed during the completion of this thesis project. The dataset provided by the ISOVIS Group contains metadata on 400 text visualization techniques corresponding to scientific publications. These all papers, book chapters, and other documents are published and available for the scientific community to study and discuss. In this thesis project, the personal information of the authors of the papers is not used at any stage. Our focus was just on the content of the publications. The results of this project will only show the trends over time, correlation of categories, and correspondence between the topic modeling results and the manually labeled data.

## 4 Data Analyses

In this chapter, we will describe the data analyses of the project, including the steps, tools, techniques, and technologies we used to answer our research questions. We will describe the data collection process, temporal and correlation analyses of the existing TextVis Browser dataset, and computational analyses of scientific publication texts based on topic modeling.

### 4.1 Data Collection and Preprocessing

The data is collected from different sources. The following are the main sources where data was collected.

#### 4.1.1 TextVis Browser Dataset

The main dataset was used in this project is the TextVis Browser dataset provided by the ISOVIS Group in JSON format. It contains the metadata about visualization techniques discussed in scientific publications. For this thesis project, we have used a copy of the dataset created at the beginning of the project on January 19, 2018. This version of the dataset contains 400 entries. For the glance of the dataset content, given below is a snapshot of a single entry from the dataset.

---

```
{
  "id": "Robertson1993",
  "title": "Document Lens",
  "year": 1993,
  "authors": "George G. Robertson and Jock D. Mackinlay",
  "reference": "George G. Robertson and Jock D. Mackinlay. <i>The Document
    Lens</i>. Proceedings of the Annual ACM symposium on User
    Interface Software and Technology (UIST), pp. 101-108, 1993.",
  "url": "http://dx.doi.org/10.1145/168642.168652",
  "categories": ["overview", "navigation", "3d", "document", "text"]
}
```

---

Listing 4.1: An example of one of the entries in the TextVis Browser dataset (in this case, corresponding to the paper by Robertson and Mackinlay [37])

The above Listing 4.1 shows the details of one publication. It contains the basic detail of one publication, such as id, title, year, authors, reference, URL and categories. The entire dataset has 400 values similar to Listing 4.1 but with different details for every publication. The unique id has been assigned to every value by the ISOVIS Group. The title of the value shows which type of text visualization technique is discussed in that publication. The year represent in which year was the paper published and the authors are the actual names of the authors that wrote a specific publication. The reference attribute contains an individual scientific reference for every paper. The URL contains the online available URL of the publication. The last field is the categories. These categories represent various aspects of text visualization techniques. They are manually assigned by the members of the ISOVIS Group based on the content of papers. The taxonomy of these 41 categories has been discussed above in Section 2.1.3. The additional details about the categories are also provided in a separate file, which is used by us for the task of the computational category matching discussed below in Section 4.4.5. The additional remarks on the original dataset from the domain experts are provided in Section 5.1.

### 4.1.2 Collection of Raw Texts of Scientific Publications

In order to answer the research question **RQ3**, we need the raw texts of these all 400 publications which are used in TextVis Browser. The collection of PDF/HTML is done from different sources. It was difficult to find the PDF/HTML of all papers online. The different source has been used to collect the raw texts PDF/HTML of these publications. For instance, OneSearch, Google Scholar, ACM Digital Library, and IEEE Xplore Digital Library are the main sources which were used to get the PDF/HTML of the publications. Almost entirely publications were not publicly available that is why, we gained the access using OneSearch<sup>2</sup> after logging in by university credentials. We downloaded the PDF files of most of the publications. However, a few papers were not available in the PDF format, that is why we tried to get their HTML representation and saved it to the PDF format for the sake of consistency. The further processing steps, which accounted for a major portion of working hours spent on implementation of this project, are described below in Section 4.4.1. The domain experts' notes about the data collection and preprocessing stages are discussed in Section 5.2.

## 4.2 Temporal Analysis of the Labeled Dataset

In order to answer the research question **RQ1**, temporal analyses are carried out on the dataset of TextVis Browser which was provided by the ISOVIS Group. Following are the steps we followed to analyze the dataset and answer the research question.

### 4.2.1 Extraction and Visualization of Overall Temporal Distribution Based on Publication Year

First, we extracted the categories, years and number of publications from the dataset. Data preprocessing and analysis are carried out with Pandas [31]<sup>3</sup> and NumPy [32]<sup>4</sup> libraries. These libraries have great functionalities for data analysis. The programming script is written in Python. We draw the matrix table using Python and saved these results into MS Excel file for further process to find an answer to the research question. In this analysis, we computed how many times each category is used in publications in a specific year. We found which category is more popular and in extension most commonly used in the previous 27 years in scientific text visualization publications. Several researchers are always working to publish new publications in their field of research. The researchers who work on text visualization are also publishing text visualization publications each year. We are interested to see the state of the art of text visualization. After further analyzing the dataset of TextVis Browser, we found how text visualization publications increase each year during the previous 27 years. The results of the analysis show that the publications of text visualizations techniques gradually increased in every year for the previous 27 years.

We can see in Figure 4.1 that text visualization becomes more popular with regard to publication year. The histogram in Figure 4.1 indicate that from 2006 to onward there is an upward trend of visualization in each year. For instance, 55 publications were published in 2014. This ratio increased in 2016 according to the

---

<sup>2</sup><https://lnu.se/en/library>

<sup>3</sup><https://pandas.pydata.org/>

<sup>4</sup><https://pypi.org/project/numpy/>

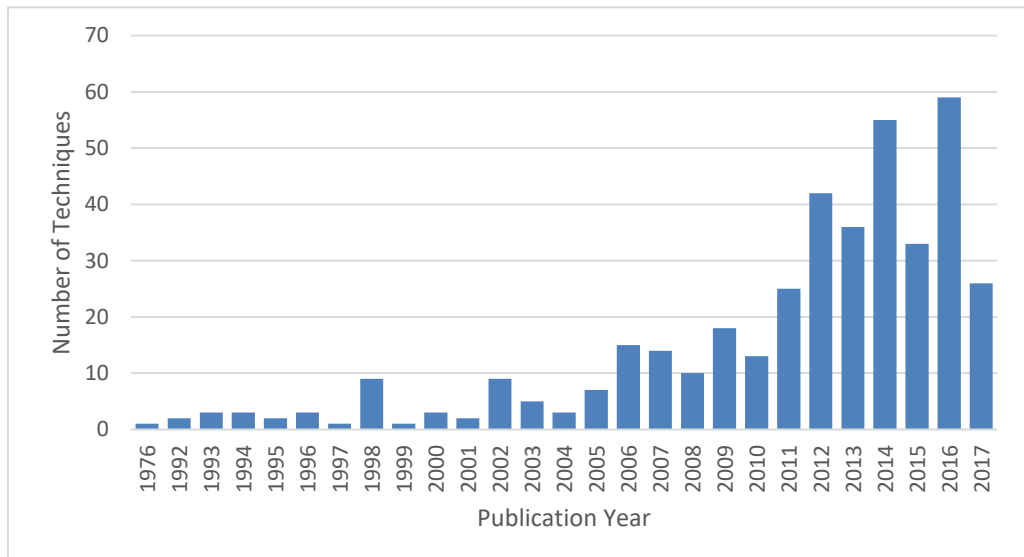


Figure 4.1: Histogram of the publications with regard to publication year

histogram (59 scientific text visualization publications). If we compare the number of publications of text visualization with previous years, we can see that the average number of publications from 1992 to 2010 is almost 15 but from 2011 to 2017 the average number of publications is approximately 30. We can clearly say that text visualization publications doubled in the previous 15 years, and visualization of text is becoming more popular.

#### 4.2.2 Extraction and Visualization of Temporal Distribution for Individual Data Categories

We are also interested to see the temporal trends related to individual categories in survey data over time. As we already mentioned the ISOVIS [2] Group they manually curated 41 categories in visual survey Browser. They manually assigned a few categories to each publication regarding their content. These categories also help the user to easily filter and access the required publications in Text visualization Browser. For instance, if we want to look at sentiment analysis related publications in TextVis Browser, we will just click on sentiment analysis category then it will show all publications related to sentiment analysis. In research question **RQ1**, we want to see what are the temporal trends related to individual categories in survey data over time, how these categories become less or more popular each year. In other words, we want to see that in text visualization publications, how the use of these manually curated categories becomes less or more popular in the previous 27 years. We further analyzed the TextVis dataset to find the temporal trends by using the Python language. The results of this analysis saved to MS Excel to create the histograms. We analyzed each category individually and created a histogram of each category to see the temporal trends regarding years.

The following three figures show the normalized charts of 41 categories individually to see the temporal trends in the previous 27 years. The further interpretation of these temporal analysis results by the domain experts is discussed in Section 5.3. We will see how these categories are used in publications each year. In order to calculate the normalized values for these bar charts, the count of entries/publications



using a particular category in a particular year is normalized against the total number of entries for the same year. For example, the category "Text Summarization" is used for 21 entries/publications for 2017, and the total count of entries for 2017 is 26, thus the normalized value for "Text Summarization" in 2017 is 21 divided by 26, or 0.808. In the figure, we have a scale of 0, 20, 40, 60, 80, and 100 and the normalized value 0.808 is actually approximately 81%.

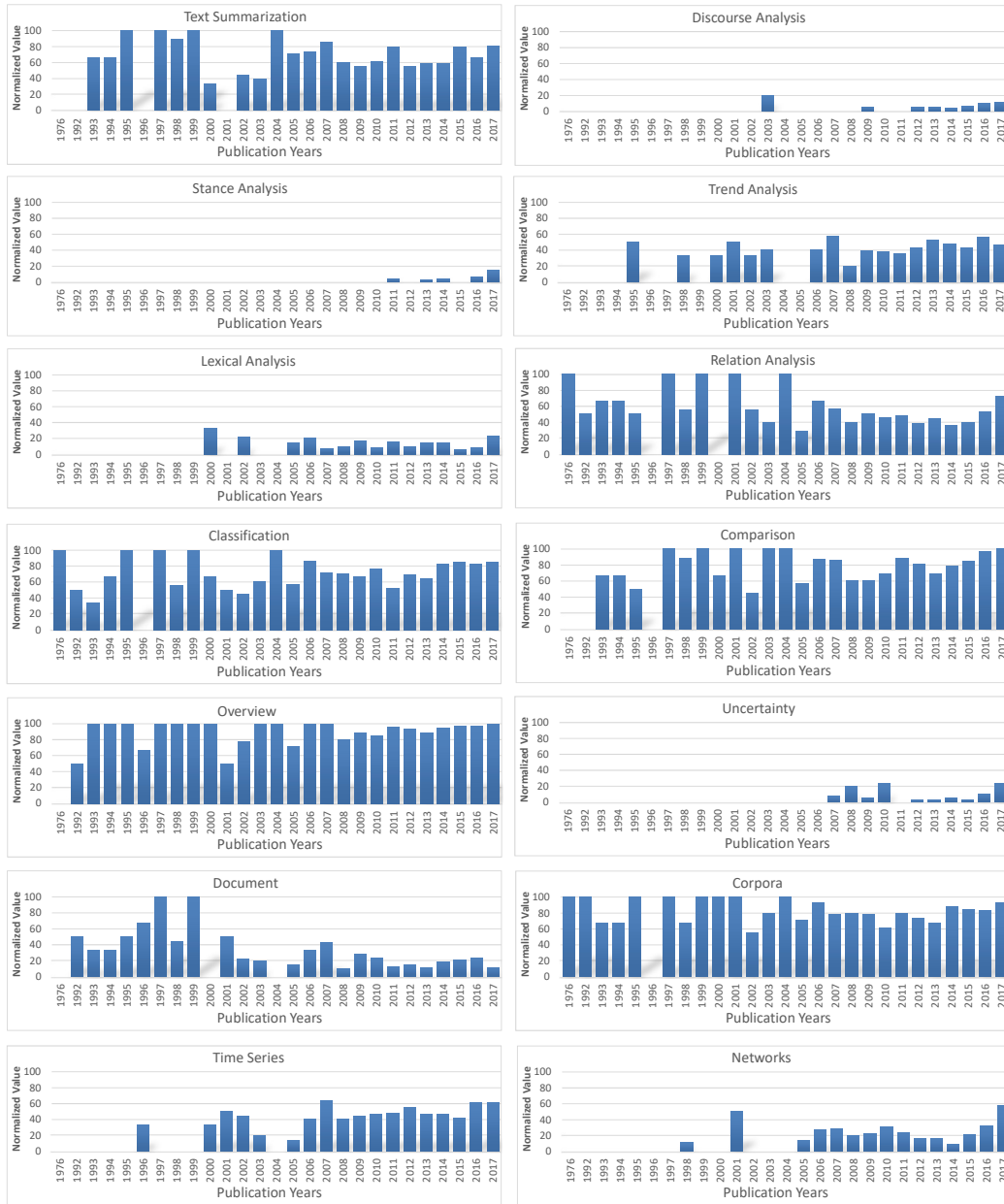


Figure 4.2: Temporal trends for individual categories

Figure 4.2 shows the temporal trends of 14 categories out of 41. Figure 4.2 demonstrate that the use of a few categories becomes less or more popular in text visualization publications in the previous 27 years. For instance, these categories 'Discourse Analysis', 'Stance Analysis', 'Lexical Analysis', 'Uncertainty', 'Networks', and 'Time Series' become more popular in previous years. Their use in text visualization publications increased. On the other hand, a few categories become less popular in the previous 27 years, e.g, 'Document' becomes less popular and

the use of 'Document' is decreasing each year in text visualization publications. An important notice is that in Document chart that after 2001 the use of the document is gradually decreasing. However, the use of a few categories remains probably the same in the previous 27 years, e.g, 'Overview' and 'Text Summarization'. The 'Overview' category is probably used in text visualization publications each year from 1992 to until 2017.

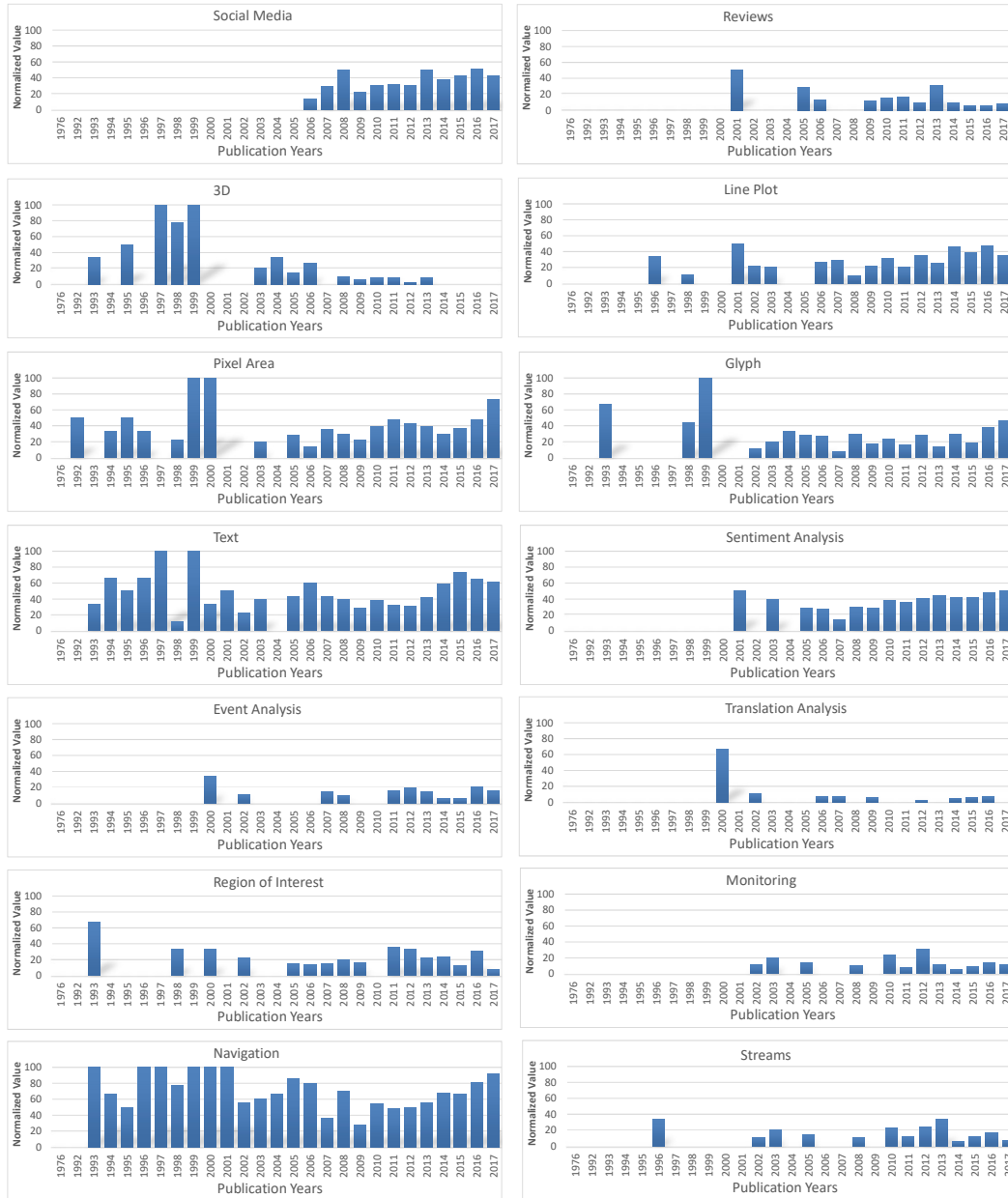


Figure 4.3: Temporal trends for individual categories (cont.)

Figure 4.3 shows the temporal trends of the other 14 categories out of 41. The histograms in Figure 4.3, clearly show that the use of a few categories in publications becomes less or more popular in the previous 27 years. According to histograms, the use of these categories in text visualization publications is becoming more popular, e.g, 'Social Media', 'Sentiment Analysis', 'Glyph', 'Pixel Area', and 'Line Plot'. The use of 'Social Media' in text visualization publications started in 2006 and gradually increasing each year until 2017. It means the social media be-

comes popular after 2006. On the other hand, the use of a few categories in text visualization publications becomes less popular, e.g. '3D', and 'Translation Analysis'. The use of '3D' category in publications from 1997 to 1999 was much but after that, it becomes very less popular and from 2014 to 2017 there is no use in publications exist. The number of categories was used often in text visualization publications in the previous 27 years, e.g. 'Streams', 'Region of Interest', 'Translation Analysis' and 'Reviews'. Although, the 'Navigation' category used almost every year from 1993 to 2017 in text visualization publication.



Figure 4.4: Temporal trends for individual categories (concl.)

Figure 4.4 shows the temporal trends of the other 13 categories out of 41. Similarly, in this Figure 4.4, few categories also become less or more popular in text visualization publication in the previous 27 years. For instance, the use of 'Geospatial' categories become more popular in text visualization publications after 2009 and it gradually increases until 2017. Although, few categories become less popu-

lar, e.g. 'Patents', 'Communication', 'Editorial Media', 'Papers', and 'Literature' in text visualization publications. Few categories are probably using in the publication from 1976, 1992 to 2017, e.g. 'Metric', 'Maps', and '2D'. The histogram of '2D' category shows that it was used probably each year in publications in the previous 27 years. The 'Radial' category used first time in 1993 but after that, in a few years it used some time in publications again but after 2008 it's becoming more popular until 2017 in publications.

After computing and discussing the temporal analysis results we observed that few categories become less or more popular. The reasons are different for each category why these categories become less or more popular. For instance, if we think about "Social Media" category it becomes more popular from 2006 the reason for it is that the use of social media starts from 2006 that is why publishers also tried to use the social media in their publications. On the other hand, if we think about "Document" category the use of document in publications is gradually decreasing. Because after digital libraries, and electronic document the use of document decreased in publications. In other categories they have also similar reasons, few categories was not exist in a few years ago but after digital era, the use of a few categories increased or decreased. The results of temporal analysis were also discussed with domain experts. The results fit their expectations.

### **4.3 Correlation Analysis of Data Categories**

In order to answer the research question **RQ2**, we further analyzed the TextVis Browser dataset. Taylor [38] describes that correlation analysis is a method which is often used to provide data summarization in medical and scientific research. The correlation analyses are useful to see the relationships between a pair of variables. We calculated the correlation matrix of our data. Below are the main steps which we followed.

#### **4.3.1 Extraction of Data Series for Data Categories**

To answer the research question **RQ2**, we have further analyzed the TextVis Browser dataset. Python language, Pandas, and Numpy are used for analysis of the data. We transformed the labeled dataset into a series of values for each category. After conducting several analyses of this dataset, we found the linear correlation coefficients for pairs of these categories in order to investigate their co-occurrence. We created a matrix table of all 400 publications and checked whether a category was used in that publication. Afterwards, the results of this analysis were saved to a MS Excel file to compute the correlation coefficients of these values.

	Paper id	comparison	overview	document	corpora	streams	patents	....	metric
1	Milgram1976	0	0	0	1	0	0	....	1
2	Eick1992	0	1	1	1	0	0	....	0
3	Lin1992	0	0	0	1	0	0	....	1
4	Spoerri1993	1	1	0	1	0	0	....	1
5	Olsen1993	1	1	0	1	0	0	....	1
6	Robertson1993	0	1	1	0	0	0	....	0
7	Hearst1994	1	1	0	1	0	0	....	1
8	Rennison1994	1	1	0	1	0	0	....	1
9	Salton1994	0	1	1	0	0	0	....	0
....	....	....	....	....	....	....	....	....	....
400	Huang2017	1	0	0	1	0	0	....	1

Figure 4.5: Part of the categories matrix used for correlation analysis

In Figure 4.5, we show the categories matrix which is created after analysing the TextVis Browser dataset using Python. The matrix is saved into MS Excel file for the computation of the correlation coefficients. In Figure 4.5, the 2nd column shows the entry IDs which are manually assigned to 400 entries corresponding to scientific publications. The other columns represent categories names. Figure 4.5 displays only a subset of the matrix for 10 entries out of 400 and 9 categories out of 41. The calculated values (0 or 1) indicate if a category was assigned to the paper in the source dataset.

### 4.3.2 Computation and Visualization of Correlation Coefficients

In research question **RQ2**, we observed the correlation between the pair of categories in survey data. We will distinguish how these categories are correlated with each other. We have conducted the correlation analysis of these categories which are manually assigned to text visualization techniques in survey Browser. The linear correlation analysis is applied to find the correlation between pairs of categories. Figure 4.6 shows the Pearson’s  $r$  correlation coefficient values [38] between the categories. In Section 5.4, there is going to be further interpretation of the results by the domain experts are discussed. We have assigned the green and red colour to these pair of correlations to highlight the positive or negative correlation. The green colour shows the positive correlation and the red colour shows the negative correlation between the pair of categories. The dark red or green colour represents that the categories have strong negative or positive correlations with each other. In the literature, the correlation analysis is used to measure the strength of the relationship between two variables. Pagano and Robert [39] describes that correlation coefficient quantitatively represents the direction and degree of relationships (in our case, linear relationship). The authors [38, 39, 40] say that the range of correlation values is from -1 to +1. For the purposes of our analysis, medium and strong positive correlation cases mean that pairs of categories tend to co-occur. Negative correlation cases mean that pairs of categories are not used together, but rather used instead of each other.

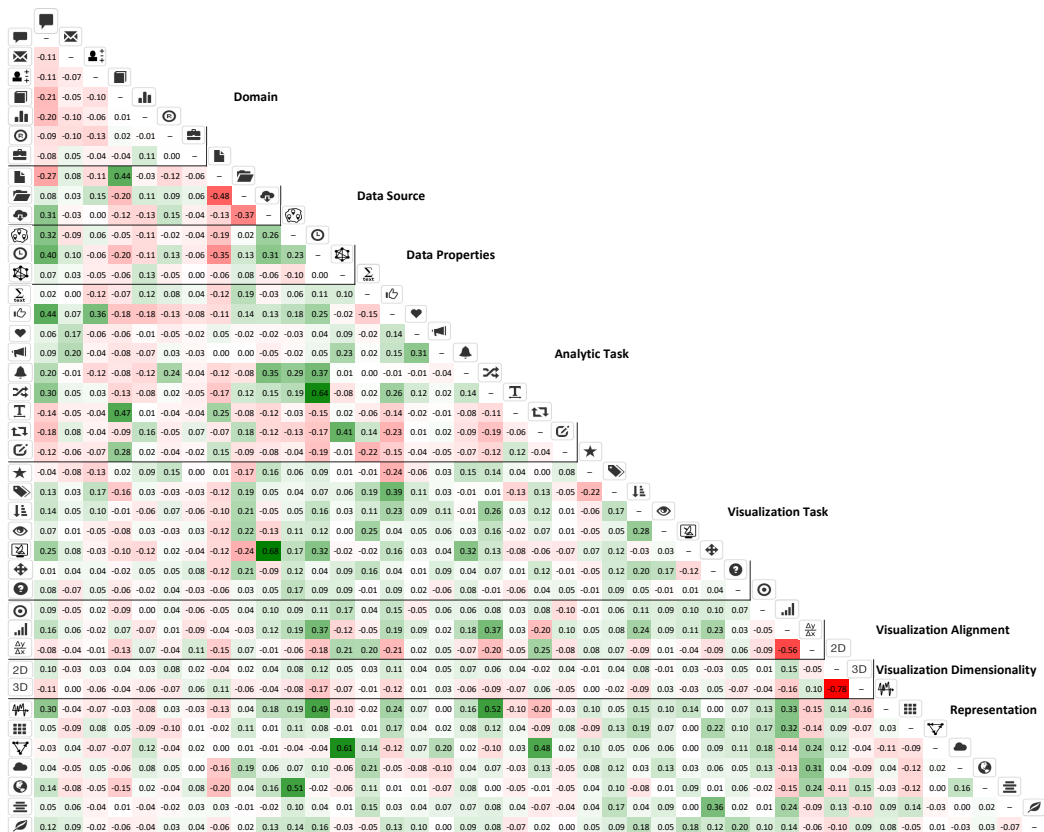











Figure 4.6: Correlation Matrix of categories

Figure 4.6 demonstrates the correlation between the pair of categories, how pairs of categories have correlations to each other. The name of categories is represented by icons to reduce the space. Unique icons are assigned to each category. The categories with icons are shown in Appendix A.1. These 41 categories are divided into 5 groups (Analytic Tasks, visualization Tasks, Domain, Data, and Visualization) in TextVis Browser. The Data and Visualization groups have further groups, the Data has two subgroups (source and properties) and visualization has 3 subgroups (Dimensionality, Representation and Alignment). There are total 8 groups of categories including subgroups. The dark black line in Figure 4.6 differentiates each group. Figure 4.6 shows the correlation of pairs of categories within their group and with other groups of categories.

The correlation matrix in Figure 4.6 shows that most categories have positive correlations but few have negative correlations. We will describe the few main strong positive and negative correlation between the pair of categories below.

First, we will discuss the negative correlations of few pairs of categories. There is a most strong negative correlation between 2D and 3D categories in Figure 4.6 and they have  $-0.78$  value. It means one category used at a time in a visualization technique. These both 2D and 3D categories belong to the same group. The reason is behind the negative correlation is each time we can use only one category for visualization and in each paper one category is used to visualize the output. The 2nd most negative correlations are between linear and metric. The correlation value between  (linear) and  $\frac{\Delta y}{\Delta x}$  (metric dependent) is  $-0.56$ , which shows that they have a negative correlation with each other. These both linear and metric dependent categories also belong to the same group of categories. If a person wants to visu-

alize the results there is a possibility he/she will visualize it in one category among linear and metric dependent. In Figure 4.6 there is the 3rd most negative correlation is between  (document) and  (corpora) categories, they have a negative correlation with -0.48 value. The negative correlation between document and corpora shows that each paper contains one type of data either document or corpora that is why they have a negative correlation. The correlation matrix in Figure 4.6 shows that the most strong negative correlations between the pairs of categories are in the same group. The categories that belong to different groups have less negative correlations.

Now, we will explain the few positive correlations between pairs of categories. The results of the correlation matrix in Figure 4.6 shows that more categories have positive correlations with each other. For instance, the correlation between  (monitoring) and  (streams) is 0.63, which have a highest strong positive correlation in Figure 4.6 between each other. The strong correlation value shows that these both categories used together in text visualization publications. It means if a document has data source streams there is a possibility to visualize it in monitoring. These both categories belong to different groups. The second highest positive correlation in Figure 4.6 is between  (trend analysis) and  (time series) is 0.62, the strong value of correlation shows that these both categories are also used together in text visualization publications. We observed that if a data has property time series it means this type of data belongs to trend analysis technique and we can visualize this data in the form of trend analysis that is why they used together. These both categories also belong to a different group of categories. The 3rd most positive correlation in Figure 4.6 is between  (node link) and  (networks), with the value of 0.61, they use together mostly in text visualization publications. Similar in these correlations if a data belongs to networks category then it is a possibility to represent it in the form of node links. These both categories also belong to a different group of categories.

The correlation matrix results show that the most categories in the same group of categories have negative correlations and the most categories are positively correlated each other with the different group of categories.

#### 4.4 Computational Data Analysis of Raw Scientific Publication Texts

The solution of research question **RQ3** was achieved through several analyses and computations. In this chapter, we will explain step by step the necessary steps which we took to find the solution of **RQ3**. In order to achieve this aim, we carried out the topic modeling of these text visualization publications. For topic modeling, we need raw textual data from these publications.

##### 4.4.1 Collection of Raw Textual Data

For topic modeling first, we need the plain text of text visualization publications which are used in TextVis Browser by ISOVIS Group. In Section 4.1.2 we have described how we carried out the initial collection steps and downloaded all of these 400 publications in the PDF format. After downloading, we used different approaches to convert PDF format publications to plain text without missing any

important information. The PDF-to-text conversion was done very carefully because if we lose some content or paragraph from a paper it could affect the topic modeling results. Therefore, we checked the plain text of each paper and matched its content with original papers PDF. The PDF to text converter<sup>5</sup> application used to create the text files of all these publication. We succeed for conversion of PDF to text in most publications. However, a few publications could not be converted to plain text (TXT format). Several publications were written more than twenty years ago and they were not saved in proper format in PDF file. Some publishers took the image of the paper and saved it to the PDF file. A couple of papers had a weird format. The PDF of these papers looked good but when we tried to convert them from PDF to plain text then, as a result, we got strange syntax that was missing some words or had words without space in between or even displayed strange symbols (see Figure 4.7 for an example).

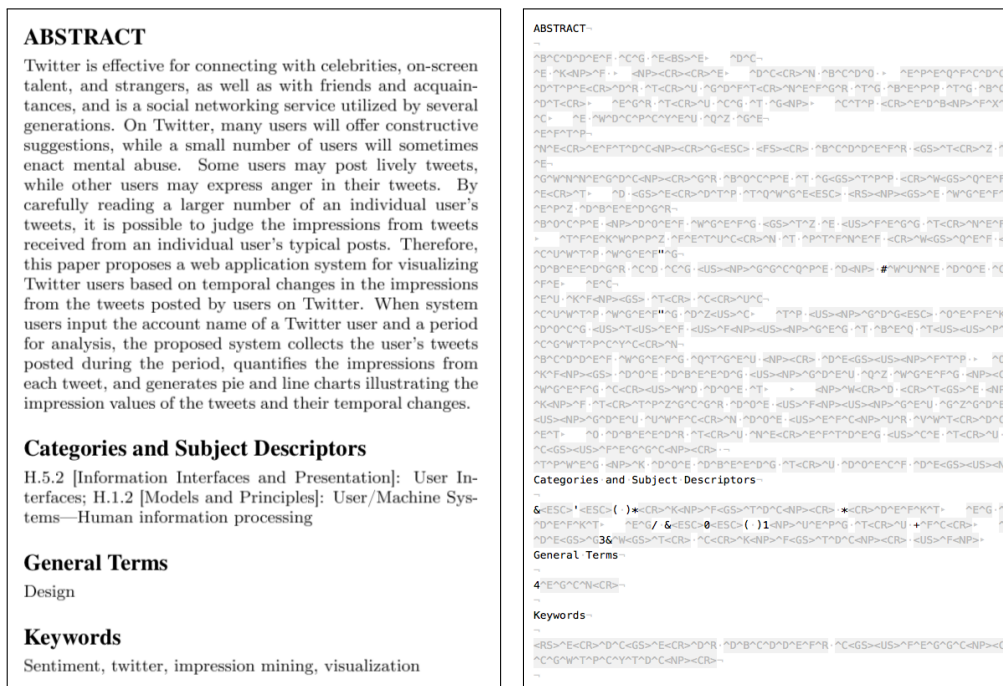


Figure 4.7: Example of a problematic PDF file and its PDF-to-text conversion results (in this case, for the paper by Kumamoto et al. [41])

Hence, we decided to use an Optical Character Recognition (OCR)<sup>6</sup> reader for these publications to convert their PDF to plain text. However, a couple of publications still had problems and we were unable to create their proper plain text. Then the snipping tool which is built in the Windows operating system was used to take the snapshot of the paper's text and save them to image format. Afterwards, we again used OCR reader and created the plain text files. At last, we finished the conversion process from PDF to the plain text of all of these 400 text visualization publications.

In the next step we discussed with domain experts and we decided that we will not use the acknowledgement and reference chapters during topic modeling. Because acknowledgement and reference chapters do not contain any serious informa-

<sup>5</sup><https://pdftotext.com/>

<sup>6</sup><https://www.onlineocr.net/>



tion in topic modeling. So we removed manually one by one the acknowledge and reference chapters from gained plain texts of all 400 publications.

The abstract of a paper describes the summary of that paper, therefore, after reading the abstract we can get a brief idea about the paper. We decided to perform topic modeling separately with abstracts and full texts. The motivation for this decision was caused by performance considerations and the consideration of abstracts as short representative summaries of the paper contents. Moreover, text mining of scientific literature with only abstracts rather than full texts has been discussed and used in the existing literature, e.g., in the works of Griffiths and Steyvers [9], Chen et al. [8], and Suominen and Toivanen [6].

We created two plain text files for each paper: one file for the abstract and another for the text body. We manually cut the abstract from all 400 plain text files and saved them in new plain text files which contain only abstracts. By this stage, we have created 800 plain text files (400 for abstracts and 400 for text body contents). Finally, we have organized this data into two folders with plain text files: one for abstracts of all papers and another for text body contents of all papers.

	PDF	Abstracts	Text Body	Full Text
Memory Occupied	860 MB	404 KB	13.5 MB	14 MB

Table 4.1: Memory occupied by all publications

Table 4.1 shows that PDF files of 400 publications took more memory space in the system as compared to plain texts. Because the PDF contains images, fonts, references, etc. The abstracts of the publications occupied just 404 KB of memory space in the system.

#### 4.4.2 Preprocessing of the Raw Textual Data

After collecting the PDF and converting them to plain text, the next step was preprocessing of these plain texts. Nayak et al. [42] state that preprocessing is an important step in Data Mining in general, and in Text Mining in particular. The preprocessing is used to extract the interesting knowledge from unstructured text data in Text Mining. Munková [43] explains that data preprocessing is the most time-consuming phase to discover the sequential patterns from textual data in Text Mining.

For topic modeling, we need only plain texts of papers. We will extract the important topics from these plain texts of papers. However, plain texts contain a lot of unnecessary data, for example, punctuations, URL, symbols, numbers, etc. This unnecessary data could affect the topic modeling results and could take a lot of extra time to process it. So, we need to remove this unnecessary data from all plain texts file before doing the topic modeling. Natural Language Processing libraries used to clean the unnecessary data from this textual data. Python programming language used to analyze and process the textual data. Several NLP and other libraries used to preprocess the textual data, for example, Gensim, NLTK, spacy for lemmatization, numpy, glob, pandas, etc., that are previously discussed in Chapter 2.

We have preprocessed the both (Abstracts and Text Body) files separately. However, we used the same processing techniques and algorithms in both files to clean the data. If we read all of these 800 plain text files one by one using our system and clean it, it could have taken too much time. So, we used glob Python library to read automatically all plain text files from a folder.

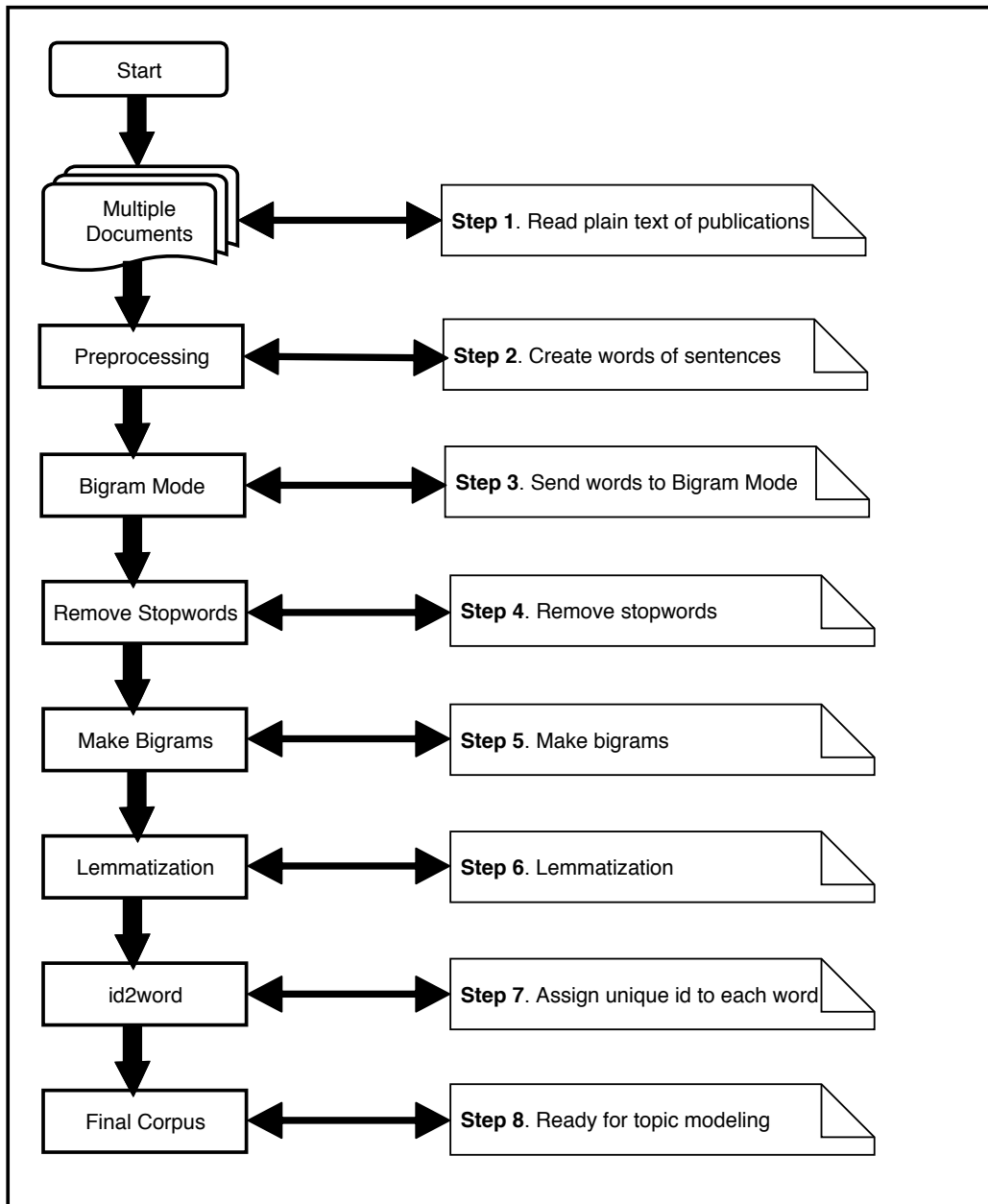


Figure 4.8: Text Mining Pre-Processing Techniques

Figure 4.8 displays the flow of preprocessing techniques, how we removed the unnecessary data from the textual data of publications. In Step 1, we read the texts of all files from the system by using Python and glob one by one and saved them in a list. After getting the list of texts of files in Step 2, we preprocess them by using Gensim and converted the sentences to words of all list of texts. Now we have a list of words, in Step 3 we define the bigram mode of words which used consecutively in a pair by using Gensim. In Step 4 we eliminated the stops words (the, is, an, am, are, etc.) from the text using NLTK because the stopwords are less important and they will not measure as keywords in text mining. In Step 5, we make the bigram of words which are used together consecutively in corpora by using NLTK. Stemming and lemmatization are done using NLP in Step 6 to remove the suffixes to reduce the number of words and takes the noun, adjectives, verb and adverb from the texts. Unique integer id is assigned to each word in Step 7, Gensim dictionary used for

mapping between words and their integer ids. After all these steps now the text is ready for topic modeling.

Table 4.2 provides an example of an abstract from publications before and after preprocessing. We can clearly see that after preprocessing the text becomes less. The trivial words are removed from the document, now we have meaningful words for topic modeling. The lemmatization process changes the words plural forms to singular, in this abstract example, few words are also lemmatized, for example, 'growing' to 'grow', 'researchers' to 'researcher', 'called' to 'call', 'based' to 'base' and so on. The lemmatization process is useful to reduce the inflectional forms and make a base form to different forms of words.

Abstract of a paper	Abstract after preprocessing
As a result of growing misuse of online anonymity, researchers have begun to create visualization tools to facilitate greater user accountability in online communities. In this study we created an authorship visualization called Writeprints that can help identify individuals based on their writing style ...	result, grow, misuse, online, anonymity, researcher, begin, create, visualization, tool, facilitate, great, user, accountability, online, community, study, create, authorship, visualization, call, writeprint, help, identify, individual, base, write, style ...

Table 4.2: Preprocessing example for part of an abstract (in this case, for the paper by Abbasi and Chen [44])

Preprocessing of the abstract and full text for each publication in the dataset was the last step before carrying out the topic modeling of the papers' contents. Preprocessing is additionally discussed with the domain experts in Section 5.2

#### 4.4.3 Topic Modeling of the Textual Data

After cleaning the text, now we start finding topics from the preprocessed text using Gensim. To gain the best results we did not rely just only on one topic modeling algorithm, after doing literature review we found that LDA and HDP are well-known and reliable algorithms. So, we decided that we will do topic modeling with two algorithms, Latent Dirichlet Allocation (LDA) and Hierarchical Dirichlet process (HDP). We divided the topic modeling into several steps. We carried out *local* topic modeling and *global* topic modeling of abstracts and full texts for these papers with LDA and HDP algorithms.

The local topic modeling means find topics from each paper individually from abstracts and full texts using LDA and HDP. The global topic modeling means find topics collectively from abstracts and full texts using LDA and HDP.

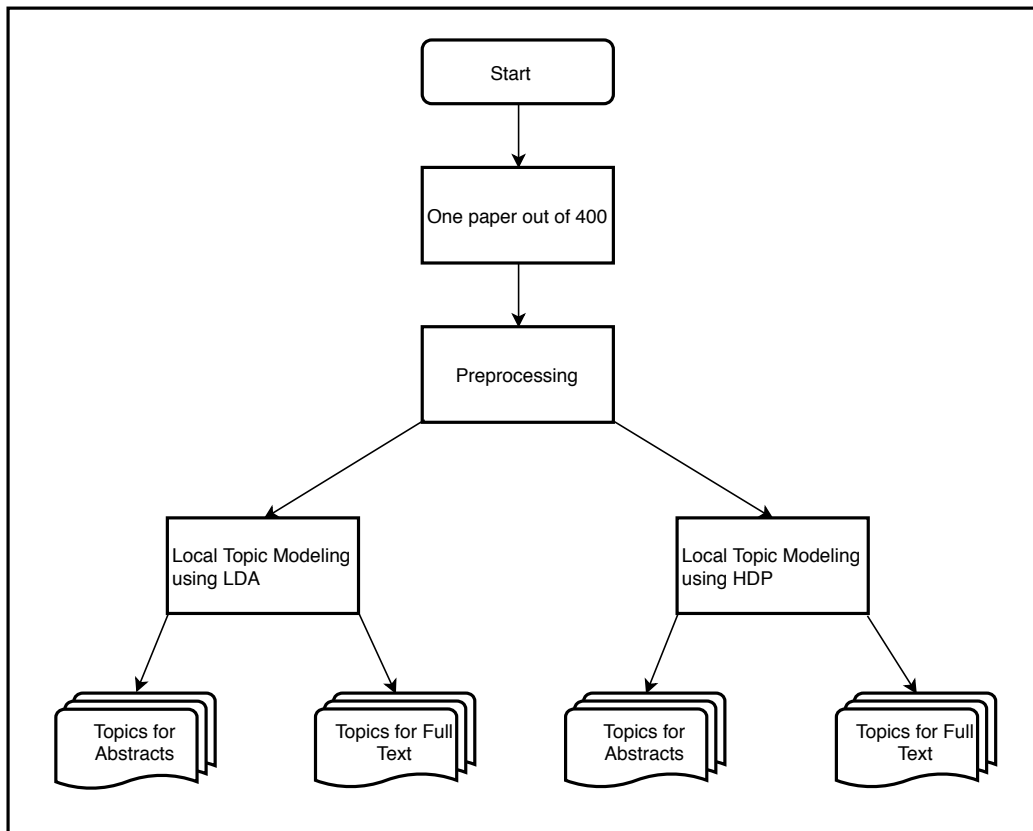


Figure 4.9: Flow of Local Topic modeling

The above Figure 4.9 shows the flow of local topic modeling, how we divide the topic modeling of papers into several steps. The preprocessing is the same for both algorithms (LDA and HDP).

After preprocessing, we first used LDA for finding topics from abstracts of each paper individually. The output of topics was saved into a new folder in plain text for each papers abstract individual. Same like abstracts, we also carried out topic modeling with full texts of all papers individually and saved the output into a new folder containing the topic result of each papers full text.

The second algorithm HDP is also used to find the topics from these papers. The preprocessing procedure was the same for HDP. The topics found from abstracts of each paper individually and saved the output of topics into a new folder in plain text for each paper abstract. Also, the topic modeling is applied using HDP with the full text of each paper individually. The output of topics saved into a new folder in the text file for full text also.

Now we have 4 folders: 1. topics from abstracts using LDA, 2. topics from full text using LDA, 3. topics from abstracts using HDP, 4. topics from full text using HDP, for local topic modeling results and each folder has 400 plain text files. In local topic modeling, the topics were found in each paper. Moreover, we are interested to see which topics are mostly used in all papers collectively.

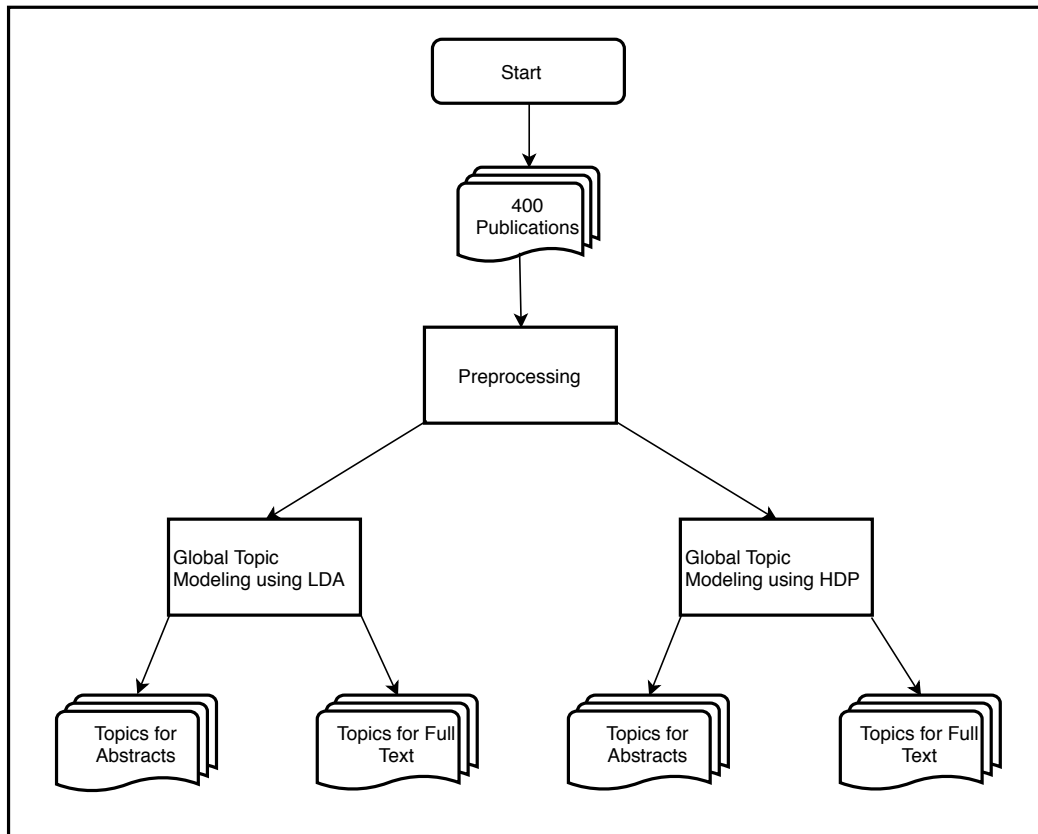


Figure 4.10: Flow of Global Topic modeling

In the global topic modeling, we found important topics from all papers collectively for abstracts and full texts using LDA and HDP. The preprocessing was same in global topic modeling but the difference here is, we considered all the papers as one corpus and start finding important topics from this corpus.

The flow of global topic modeling is shown in Figure 4.10. The preprocessing was same as local topic modeling in this step. The global topic modeling was carried out using LDA and HDP for abstracts and full text of all papers.

After preprocessing all paper's abstract, the LDA algorithm used to find the global topics from abstracts. To find the global topics from abstracts, we considered all paper's abstract as a single corpus and tried to find the top 30 most important topics from the corpus. Then we query the corpus of each paper from these topics and find the matching topics. The query process will check the relevant matched topics from global topics of each paper one by one. We matched how many, or which topics are in global topics belong to that paper. The matching topics for each paper saved into a new folder for abstracts. Similar procedure as applied for abstracts to find the global topics is also followed for the full text of papers using LDA to find the global topics. However, for finding the global topics for full text, we took top the 40 most important topics from the corpus.

The HDP algorithm also implemented to find global topics. Similar to LDA, first, we found global topics for abstracts and saved the output in the text file in our system. Thereafter, we also found global topics from full texts of papers and saved the results in our system.

The results of global topic modeling are saved into 4 folders: 1. global topics from abstracts using LDA, 2. global topics from full text using LDA, 3. global topics from abstracts using HDP, 4. global topics from full text using HDP.

While doing the topic modeling, at the same time, we also measured the execution time and memory consumption for both algorithms (LDA and HDP). We had a good specification system with 6th generation 2.40 GHz processor and 8 GB of Random Access Memory (RAM). The given below table shows the memory consumption and execution time of our system while doing the topic modeling using LDA and HDP. “TM” represents “Topic modeling” in the tables.

	LDA		HDP	
	Local TM	Global TM	Local TM	Global TM
Abstracts	302 seconds	12 seconds	188 seconds	24 seconds
Full Text	607 seconds	26 seconds	386 seconds	47 seconds

Table 4.3: Execution time while doing topic modeling using LDA and HDP

Table 4.3 shows the execution time while doing topic modeling using LDA and HDP. The measured time in Table 4.3 shows that both algorithms took different execution time while doing the same process. For instance, when we carried out local topic modeling of papers using LDA it took 607 seconds for full text and 302 seconds for abstracts but the HDP took just 386 seconds for full text and 188 seconds for abstracts. If we compare both algorithms execution time for local topic modeling of abstracts and full text, the HDP took less execution time of our system. However, while doing the global topic modeling using LDA it took 12 seconds for abstracts and 26 seconds for full text but the HDP took 24 seconds for abstracts and 47 seconds for full text. The comparison of both algorithms execution time shows that LDA took more execution time than HDP while doing local topic modeling and for global topic modeling HDP took more execution time than LDA.

	LDA		HDP	
	Local TM	Global TM	Local TM	Global TM
Abstracts	278 MB	249 MB	279 MB	294 MB
Full Text	290 MB	312 MB	285 MB	246 MB

Table 4.4: Memory consumption while doing topic modeling using LDA and HDP

Table 4.4 shows the memory consumption while doing topic modeling using LDA and HDP. We also measured the memory consumption of our system while doing local and global topic modeling of abstracts and full texts using LDA and HDP. According to Table 4.4, the memory consumption of our system for local topic modeling for LDA is 278 MB for abstracts and 290 MB for full texts, and HDP took 279 MB for abstracts and 285 MB for full texts. They took probably same memory space, just there is a very small difference between local topic modeling using both algorithms. Although, if we compare it with global topic modeling the LDA took 249 MB for abstracts and 312 MB for full text and the HDP took 294 MB for abstracts and 246 MB for full texts. The comparison shows that both algorithms occupied different memory space while doing global topic modeling for the same corpus. Also, after analyzing the comparison results we observed that HDP works better in large documents and it took less time and memory while doing global topic modeling.

#### 4.4.4 Experimentation with Topic Modeling Algorithms and Parameters

After doing the topic modeling with LDA and HDP, we have found that each algorithm has different results for the same paper. The topic modeling finds the topics with the weight of each topic. We took all topics with high weight from a corpus. However, in the given topic modeling results examples we ignored the weight and just took the words. First, we will see the difference between local topic modeling results for one paper out of 400 using LDA and HDP. Further interpretation of the topic modeling results for our scientific publications dataset by the domain experts is described in Section 5.5.

LDA allows us to get a certain number of topics from the corpus. We need to define a limit by using a parameter that restricts the number of topics that we want to extract from the corpus. We can return the number of top topics from the corpus using LDA and HDP while doing topic modeling. In our implementation, we extracted the following number of topics from the publications. We decided on the base of content of documents such as abstract has less data so we decided to get the just 5 topics and 10 words. We also decided after trying to get different number of topics. First, we tried extracting 10 topics and 10 words from abstracts their result shows same words in each topic. However, when we used these certain number of topics it shows less duplicate results for abstracts. Similar to the full-text and global topic modeling we chose more topics and words on the base of their larger content (see Table 4.5).

	LDA		HDP	
	Local TM	Global TM	Local TM	Global TM
Abstracts	5 topics, 10 words	30 topics, 10 words	5 topics, 10 words	30 topics, 10 words
Full Text	20 topics, 15 words	40 topics, 10 words	20 topics, 15 words	40 topics, 10 words

Table 4.5: The top number of topics we extracted using LDA and HDP during topic modeling

The abstract of a paper contains less data than a full text, that is why we extracted the less number of topics for it and limited it to just top 5 topics. The full text of a paper contains more data, therefore we extracted top 20 topics for each paper. We followed the same approach for both algorithms, LDA and HDP. In Table 4.6 we took just top 5 topics and 10 top terms for the abstract because the data of the abstract was quite less. Table 4.6 shows that the topic modeling results for LDA for 5 topics are the same in each term. There are duplicated results in LDA. However, if we see the topic modeling results of HDP for the same paper and for the top 5 topics and 10 terms HDP has good results with less duplication of topics than LDA. Also, HDP has probably different topics in each term.

In Table 4.6, we found topics just from the abstract. The abstract mostly contains a short summary of the paper. The whole paper contains more information as compared to abstract and in the papers body terms describes in details. Furthermore, we also carried out topic modeling for full text/full paper.

Latent Dirichlet Allocation (LDA)	Hierarchical Dirichlet Process (HDP)
<b>Topic 1:</b> order, misuse, result, researcher, provide, principal, performance, pattern, outperform, similar	<b>Topic 1:</b> visualization, base, identify, classification, instance, writeprint, create, arabic, authorship, online
<b>Topic 2:</b> order, misuse, result, researcher, provide, principal, performance, pattern, outperform, similar	<b>Topic 2:</b> find, biometric, well, begin, excellent, online, large, evaluate, authorship, call
<b>Topic 3:</b> order, misuse, result, researcher, provide, principal, performance, pattern, outperform, similar	<b>Topic 3:</b> forum, many, outperform, style, begin, accountability, feature, online, write, authorship
<b>Topic 4:</b> order, misuse, result, researcher, provide, principal, performance, pattern, outperform, similar	<b>Topic 4:</b> instance, misuse, create, believe, vector, technique, style, find, enforcement, automatically
<b>Topic 5:</b> order, misuse, result, researcher, provide, principal, performance, pattern, outperform, similar	<b>Topic 5:</b> deter, find, biometric, manner, excellent, algorithm, deception, provide, unique, online

Table 4.6: An example of *local* topic modeling results with LDA and HDP for one abstract (in this case, for the paper by Abbasi and Chen [44])

Latent Dirichlet Allocation (LDA)	Hierarchical Dirichlet Process (HDP)
<b>Topic 1:</b> message, online, author, authorship, use, feature, visualization, writeprint, identification, window, base, create, technique, algorithm, group	<b>Topic 1:</b> message, feature, authorship, online, use, visualization, writeprint, author, identification, base, technique, pattern, group, window, create
<b>Topic 2:</b> message, feature, visualization, online, use, authorship, writeprint, author, identification, technique, algorithm, window, base, write, pattern	<b>Topic 2:</b> associate, future, pattern-gram, online, enforcement, criminal, compensate, feed, letter, may, contribution, anonymity, use, effective, language
<b>Topic 3:</b> message, online, writeprint, feature, use, authorship, identification, author, base, visualization, technique, window, pattern, create, group	<b>Topic 3:</b> make, aqsa, sum, light, need, large, multiple, correct, couple, evaluate, vocabulary, secondary, self, much, speak
<b>Topic 4:</b> message, online, use, authorship, feature, visualization, writeprint, author, pattern, technique, base, create, identification, group, algorithm	<b>Topic 4:</b> technique, work, fit, pose, dire, rampant, parameter, kelly, limit, user, exception, however, variance, powerful, baseline
<b>Topic 5:</b> message, online, feature, authorship, visualization, author, writeprint, use, identification, group, write, algorithm, pattern, window, technique	<b>Topic 5:</b> karhunen, repeat, several, compare, range, attribute, efficient, reduction, fit, anonymity, could, frequency, identification, kelly, writing

Table 4.7: An example of *local* topic modeling results with LDA and HDP for one full text (in this case, for the paper by Abbasi and Chen [44])



Table 4.7 shows the results of one paper’s full text using LDA and HDP. The full text of a paper contains more data than abstract, so we took more top topics while doing topic modeling using LDA and HDP. We took top 15 topics and 20 terms for full text for both LDA and HDP. However, in Table 4.7 we show just top 5 topics and 15 terms. The preprocessing was the same for this paper but we have different topic modeling results using LDA and HDP. We have noticed that when we provided full text to the algorithms they provided more meaningful topics than abstract results with very fewer duplications. However, the LDA still has more duplicate topics in several terms as compared to HDP topics.

The topic modeling is carried out for each paper’s abstract and full text individually using LDA and HDP, and we found the top important topics from the papers. These topic modeling results just related to specific paper. Furthermore, we are interested to see which topic are more important in all 400 publications using the same algorithms (LDA and HDP). We refer to this approach for finding the collectively topics from all 400 publications as *global* topic modeling.

Latent Dirichlet Allocation (LDA)	Hierarchical Dirichlet Process (HDP)
<b>Topic 1:</b> text, visualization, analysis, time, system, present, topic, visual, document, sentiment	<b>Topic 1:</b> visualization, datum, analysis, new, time, base, system, event, information, paper
<b>Topic 2:</b> analysis, tool, datum, visualization, sentiment, model, present, user, information, analyst	<b>Topic 2:</b> text, document, use, user, topic, visualization, datum, sentiment, system, base
<b>Topic 3:</b> text, user, visualization, approach, use, analysis, visual, base, document, present	<b>Topic 3:</b> sentiment, user, base, analysis, propose, present, time, method, technique, review
<b>Topic 4:</b> visualization, text, system, document, information, user, use, datum, base, visual	<b>Topic 4:</b> text, visualization, information, document, user, paper, method, topic, visualize, visual
<b>Topic 5:</b> visual, analysis, sentiment, visualization, summary, present, text, base, time, user	<b>Topic 5:</b> information, use, show, document, text, visual, purpose, system, user, collection

Table 4.8: *Global* topic modeling results with LDA and HDP for abstracts of the complete dataset of 400 papers

The preprocessing was also same as local topic modeling for global topic modeling but we considered all 400 publications as a one textual data source of all papers abstracts and full texts separately.

In Table 4.8, we show the global topic modeling results of all 400 paper’s abstracts using LDA and HDP. We took top 30 topics and 10 words in both algorithms in our actual computation. However, in Table 4.8 we showed just 5 top topics with 10 terms/words. The results of global topic modeling show that both algorithms have different topic modeling results with the same corpus. In Table 4.8 the results of global topic modeling using LDA and HDP shows that few words are more important and they used mostly in all 400 publication’s abstracts, e.g, ’visualization’, ’text’, ’analysis’, ’document’, etc, both algorithms (LDA and HDP) extracted these topics during topic modeling.

The global topic modeling was also carried out for full texts of all 400 papers using LDA and HDP. The preprocessing was same as global topic modeling for

abstracts but the difference is here in this analysis, we took 400 paper’s complete texts as one corpus. In this analysis, we are interested to see which topics mostly used in all publications.

Latent Dirichlet Allocation (LDA)	Hierarchical Dirichlet Process (HDP)
<b>Topic 1:</b> visualization, datum, document, topic, system, figure, use, work, user, result	<b>Topic 1:</b> visualization, work, datum, analysis, user, text, figure, approach, system, use
<b>Topic 2:</b> figure, user, document, work, approach, visualization, datum, result, analysis, diagram	<b>Topic 2:</b> user, visualization, figure, sentiment, system, topic, work, use, information, study
<b>Topic 3:</b> work, document, figure, system, information, datum, use, visualization, user, first	<b>Topic 3:</b> text, figure, visualization, analysis, use, work, visual, study, paper, document
<b>Topic 4:</b> visualization, user, work, word, paper, figure, present, visual, show, system	<b>Topic 4:</b> user, topic, document, figure, visualization, sentiment, word, show, new, use
<b>Topic 5:</b> document, user, work, use, study, paper, show, system, base, approach	<b>Topic 5:</b> user, system, datum, figure, work, visualization, opinion, use, document, result

Table 4.9: *Global* topic modeling results with LDA and HDP for full texts of the complete dataset of 400 papers

Table 4.9 shows the global topic modeling results of full texts of 400 publications using LDA and HDP. For global topic modeling results we took top 40 topics and 10 words in our final analysis but in Table 4.9 we show just 5 topics with 10 words. Both algorithms extracted these topics ‘visualization’, ‘document’, ‘user’, ‘work’, ‘text’, etc from the corpus. Table 4.9 shows that the extracted topics mostly used in all 400 text visualization publications.

After doing the local and global topic modeling using LDA and HDP with abstracts and full texts of 400 publications, we have noticed that the extracted topics from both algorithms are not similar.

As the next step, after getting the global topics of 400 paper abstracts we further queried the all 400 papers one by one to get the relevant topics from the global topics.

Latent Dirichlet Allocation (LDA)	Hierarchical Dirichlet Process (HDP)
<b>Topic 1:</b> visualization, emotion, use, base, user, network, present, technique, study, content	<b>Topic 1:</b> topic, user, text, datum, use, analysis, system, sentiment, design, base
	<b>Topic 2:</b> categorize, ego, sentiment, shall, textdna, free, sub, lead, arise, default

Table 4.10: An example of topics for a single paper (in this case, for the paper by Abbasi and Chen [44]) based on global topic modeling results with LDA and HDP for abstracts

Table 4.10 shows the queried matched topics for the abstract of one paper out of 400 using LDA and HDP. These topics are queried from the top 40 global topics of abstract of 400 papers. The extracted results of both algorithms (LDA and HDP) in Table 4.10 shows that they have some similar topic terms, e.g, 'use', 'user', and 'base'. It is also evident that only several relevant topics were returned by the queries to the global topic models (just one for LDA and two for HDP, respectively) for this particular paper.

We followed a similar step with full text, after getting the global topic of 400 papers full text we further queried the all 400 papers one by one to get the relevant topics from the global topics.

Latent Dirichlet Allocation (LDA)	Hierarchical Dirichlet Process (HDP)
<b>Topic 1:</b> user, system, visualization, base, figure, work, text, result, use, present	<b>Topic 1:</b> work, visualization, user, figure, analysis, use, system, text, datum, present
<b>Topic 2:</b> user, visualization, work, figure, use, text, show, study, future, base	<b>Topic 2:</b> emergence, authorship, sentiment, boldface, trist, approach, test, base, several, work

Table 4.11: An example of topics for a single paper (in this case, for the paper by Abbasi and Chen [44]) based on global topic modeling results with LDA and HDP for full texts

Table 4.11 shows the queried matched topics for the full text of one paper out of 400 using LDA and HDP. We considered top 40 global topics and the query the papers full text and saved the matching topics. Both the LDA and HDP algorithms extracted more similar topics with terms such as 'work', 'user', 'system', 'visualization', 'figure', 'text', and 'present' as compared to the abstract-based results in Table 4.10 above. At the same time, both topic models returned only two relevant topics for this paper.

#### 4.4.5 Computational Matching of Topic Modeling Results to the Labeled Dataset

We compared the local and global topic modeling results with the manually labeled dataset. We extracted local and global topics from papers abstracts and full texts using LDA and HDP in the previous section. Now we will try to check if these topic modeling results have any correspondence with the manually labeled dataset of TextVis Browser. We will explain the gained matching results step by step. First, we will explain the matching results for LDA for abstract and full text, afterwards, the results for HDP, and finally, an additional baseline approach that does not rely on topic modeling at all.

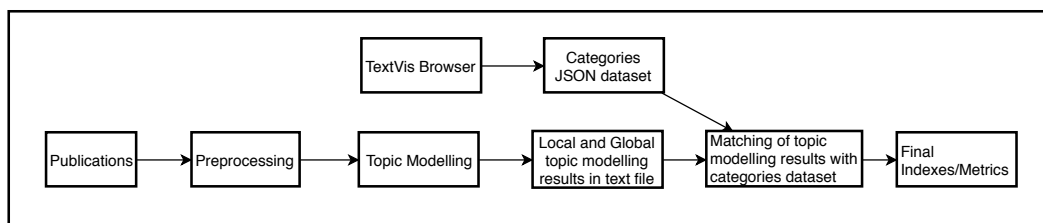


Figure 4.11: Flowchart for matching topic modeling results with categories dataset

Figure 4.11 shows the flowchart of the computational matching of local and global topic modeling results with the source data by using an additional dataset provided by the domain experts from the ISOVIS Group. The results of local and global topic modeling of LDA for abstracts of 400 papers were matched with manually labeled dataset using Python script. In this stage, we *merged* the local and global topic modeling results of abstracts using LDA for the purpose of matching. We considered them as one source and matched them with categories dataset described below. The matched results saved into one file. We checked that how many topics are similar in manually assigned categories of TextVis Browser. While matching the topic modeling results with labeled dataset we counted the number of matched values. We counted how many times a category matched with topic modeling results. The results of these matching are saved to JSON file.

In order to be able to match the information extracted from the texts of the publications, such as the topic modeling results in our case, to the category labels used in the source TextVis Browser dataset, we had to rely on additional information. This information was provided to us by the ISOVIS Group as an additional categories dataset that establishes the mapping between the abstract category labels (e.g., "sentiment-analysis" or "line-plot") and the corresponding lists/sets of key terms describing these categories (e.g., "emotion" or "polarity" for "sentiment-analysis"). We demonstrate several examples from this dataset below and the complete list of terms used for matching the categories is provided in Appendix A.2.

Listing 4.2 shows the first 3 categories out of 41 from the categories dataset. In Listing 4.2, the 'category' represents the name of the main category which is used in TextVis Browser. The 'terms' is the synonyms of the main category. We also tried to match the topic modeling results with 'terms'.

---

```

{
  "category": "text-summarization",
  "terms": ["summarization", "summary", "topic", "theme", "named", "entity", "entities"]
},
{
  "category": "discourse-analysis",
  "terms": ["discourse"]
},
{
  "category": "stance-analysis",
  "terms": ["stance", "intersubjectivity", "evaluative", "judgemental", "appraisal"]
}

```

---

Listing 4.2: Example of the mapping from the categories dataset

Listing 4.3 provides an example of the matching results for one paper based on LDA topic modeling for paper abstracts. While matching the local and global topic modeling results, the categories who do not match with topic modeling results are assigned 0. We can see the matched categories values like 'sentiment-analysis', 'trend-analysis', 'networks' are more than 0, it means these categories are matched with topic modeling results.

The local and global topic modeling results of full texts using LDA are also compared using Python script to see the matching values. We also *merged* here the local and global LDA topic modeling results for full texts. After that, we compared these results with categories dataset and stored the results into one output file. Listing 4.4 shows the matched results of local and global topic modeling using LDA for

---

```

{
  "id": "Abbasi2006",
  "matched_lda_abstracts": {"text-summarization": 0, "discourse-analysis":
    0, "stance-analysis": 0, "sentiment-analysis": 1, "event-analysis"
    : 0, "trend-analysis": 4, "lexical-analysis": 0, "relation-analysis"
    : 0, "translation-analysis": 0, "roi": 0, "classification": 0, "
    comparison": 0, "overview": 0, "monitoring": 0, "navigation": 0, "
    uncertainty": 0, "document": 0, "corpora": 0, "streams": 0, "
    geospatial": 0, "time-series": 0, "networks": 1, "social-media": 0,
    "communication": 0, "patents": 0, "reviews": 0, "literature": 0, "
    papers": 0, "editorial-media": 0, "2d": 0, "3d": 0, "line-plot": 0, "
    pixel-area": 0, "node-link": 0, "clouds": 0, "maps": 0, "text": 0, "
    glyph": 0, "radial": 0, "linear": 0, "metric": 0}
}

```

---

Listing 4.3: An example of category term matching results based on LDA for abstracts (in this case, for the paper by Abbasi and Chen [44])

---

```

{
  "id": "Abbasi2006",
  "matched_lda_fulltext": {"text-summarization": 0, "discourse-analysis":
    0, "stance-analysis": 0, "sentiment-analysis": 0, "event-analysis": 0,
    "trend-analysis": 19, "lexical-analysis": 0, "relation-analysis":
    0, "translation-analysis": 0, "roi": 0, "classification": 15, "
    comparison": 0, "overview": 0, "monitoring": 0, "navigation": 0, "
    uncertainty": 0, "document": 0, "corpora": 0, "streams": 0, "
    geospatial": 0, "time-series": 0, "networks": 2, "social-media": 0,
    "communication": 0, "patents": 0, "reviews": 0, "literature": 0, "
    papers": 0, "editorial-media": 0, "2d": 0, "3d": 0, "line-plot": 0, "
    pixel-area": 0, "node-link": 0, "clouds": 0, "maps": 0, "text": 2, "
    glyph": 0, "radial": 0, "linear": 0, "metric": 0}
}

```

---

Listing 4.4: An example of category term matching results based on LDA for full texts (in this case, for the paper by Abbasi and Chen [44])

the full text of 400 publications. When we matched topic modeling results of full text, we found more categories are matched with the manually labeled dataset.

After matching the local and global topic modeling results of LDA with the manually labeled dataset, we can say that the full texts have good matching results as compared to abstracts results.

Now we match the topic modeling results of HDP with the manually labeled dataset and see how many categories are matched with these results, similar to the LDA-based steps above. The local and global topic modeling results using HDP for the abstract are *merged* here for doing the matching with categories. The matching results saved into one output file.

Listing 4.5 shows the matching results of one paper's abstract out of 400 local and global topic modeling using HDP. If we compare these results with LDA's matched results, we can say that the results of HDP have more matching categories.

Now we will show the matching results of local and global topic modeling of full text with manually labeled dataset using HDP. The results of local and global topic modeling merged here and compared with categories dataset. The matched output stored into one file.

Listing 4.6 shows the results after matching local and global topic modeling results with the manually labeled dataset of full text using HDP. In the Listing 4.6 mostly values have more than 0 value it means several values of topic modeling results of HDP are matched with the manually labeled dataset in this paper.

If we compare the matching results of LDA and HDP we can say that HDP results have more matching values in the manually labeled dataset. HDP algorithm

---

```

{
  "id": "Abbasi2006",
  "matched_hdp_abstracts": {"text-summarization": 1, "discourse-analysis":
    : 0, "stance-analysis": 3, "sentiment-analysis": 4, "event-analysis"
    : 0, "trend-analysis": 0, "lexical-analysis": 0, "relation-analysis"
    : 0, "translation-analysis": 0, "roi": 0, "classification": 7, "
    comparison": 0, "overview": 0, "monitoring": 0, "navigation": 0, "
    uncertainty": 0, "document": 0, "corpora": 0, "streams": 0, "
    geospatial": 0, "time-series": 2, "networks": 0, "social-media": 1,
    "communication": 0, "patents": 0, "reviews": 0, "literature": 0, "
    papers": 0, "editorial-media": 0, "2d": 0, "3d": 0, "line-plot": 0, "
    pixel-area": 0, "node-link": 0, "clouds": 0, "maps": 0, "text": 1, "
    glyph": 0, "radial": 0, "linear": 0, "metric": 2}
}

```

---

Listing 4.5: An example of category term matching results based on HDP for abstracts (in this case, for the paper by Abbasi and Chen [44])

---

```

{
  "id": "Abbasi2006",
  "matched_hdp_fulltext": {"text-summarization": 3, "discourse-analysis":
    0, "stance-analysis": 1, "sentiment-analysis": 4, "event-analysis"
    : 0, "trend-analysis": 2, "lexical-analysis": 0,
    "relation-analysis": 2, "translation-analysis": 2, "roi": 3,
    "classification": 3, "comparison": 1, "overview": 1, "monitoring":
    0, "navigation": 1, "uncertainty": 1, "document": 1,
    "corpora": 1, "streams": 0, "geospatial": 0, "time-series": 2,
    "networks": 7, "social-media": 1, "communication": 2, "patents": 0
    , "reviews": 0, "literature": 2, "papers": 2,
    "editorial-media": 2, "2d": 0, "3d": 0, "line-plot": 0,
    "pixel-area": 1, "node-link": 2, "clouds": 0, "maps": 0,
    "text": 2, "glyph": 0, "radial": 0, "linear": 3, "metric": 0}
}

```

---

Listing 4.6: An example of category term matching results based on HDP for full texts (in this case, for the paper by Abbasi and Chen [44])

discovers and shows the less duplicate results from the corpus that is why this algorithm discovered good matching results as compared to LDA.

#### 4.4.6 Additional Matching Method Bypassing the Topic Modeling Stage

The above-mentioned results are based on local and global topic modeling using LDA and HDP, however, we also tried to match the paper's text without topic modeling using Python with the manually labeled dataset. The motivation for this step is that we were also interested to check if using topic modeling results for matching would be better than simply using the words/tokens from the source paper texts instead. This approach of simple text matching (without topic modeling) was carried out for abstracts and full texts of all 400 papers. After doing the preprocessing of publications and we directly compared the preprocessed whole text with categories dataset.

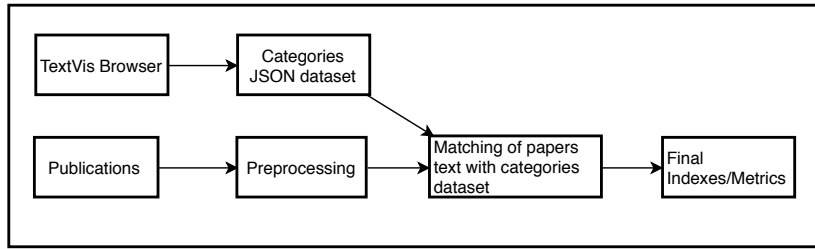


Figure 4.12: Flowchart for simple text matching (without topic modeling) results with categories dataset

In Figure 4.12 we showed the structure that we followed to match the simple text matching with categories dataset. The simple text matching is also carried out using Python language and the matched results saved to JSON file.

```

{
  "id": "Abbasi2006",
  "matched_abstracts": {"text-summarization": 0, "discourse-analysis": 0,
    "stance-analysis": 0, "sentiment-analysis": 0, "event-analysis": 0, "
    trend-analysis": 0, "lexical-analysis": 0, "relation-analysis": 0, "
    translation-analysis": 0, "roi": 0, "classification": 1,
    "comparison": 1, "overview": 0, "monitoring": 0, "navigation": 0, "
    uncertainty": 0, "document": 0, "corpora": 0, "streams": 0, "
    geospatial": 0, "time-series": 0, "networks": 0, "social-media": 0, "
    communication": 0, "patents": 0, "reviews": 0, "literature": 0, "
    papers": 0, "editorial-media": 0, "2d": 0, "3d": 0, "line-plot": 0, "
    pixel-area": 0, "node-link": 0, "clouds": 0, "maps": 0, "text": 0, "
    glyph": 0, "radial": 0, "linear": 0, "metric": 0}
}

```

Listing 4.7: An example of category term matching results based on simple matching approach (without topic modeling) for abstracts (in this case, for the paper by Abbasi and Chen [44])

Listing 4.7 shows the matching result of one paper out of 400 without topic modeling of abstracts. In Listing 4.7, we can see that only two categories 'classification' and 'comparison' is matched with the manually labeled dataset one time for this paper. However, if we see the matching results of local and global topic modeling in Listing 4.5 and 4.7 using LDA and HDP and compare it with these results, the topic modeling results have more matching values.

Now we will compare the full text of papers without topic modeling with the manually labeled dataset and see what are the matching results.

Listing 4.8 shows the matching results of one paper out of 400 without topic modeling of the full text. We can see in Listing 4.8 when we matched the full text of papers and we found a few matching values without topic modeling.

#### 4.4.7 Evaluation of the Computational Matching Results

In this subsection, we summarized and evaluated the results of matching between the data extracted from publication texts (with or without topic modeling) and the original labeled dataset that was presented in the previous subsections. We used different methods, approaches for computing the computational matching, as displayed above in Figures 4.11 and 4.12.

After computing the counts of corresponding terms for various categories in the matching stage, we are presented with a lot of detailed data, which, however,

```

{
  "id": "Abbasi2006",
  "matched_fulltext": {"text-summarization": 0, "discourse-analysis": 0, "
    stance-analysis": 0, "sentiment-analysis": 0, "event-analysis": 0, "
    trend-analysis": 0, "lexical-analysis": 0, "relation-analysis": 0, "
    translation-analysis": 0, "roi": 0, "classification": 9,
    "comparison": 3, "overview": 1, "monitoring": 0, "navigation": 0, "
    uncertainty": 0, "document": 2, "corpora": 1, "streams": 0, "
    geospatial": 0, "time-series": 0, "networks": 0, "social-media": 0,
    "communication": 5, "patents": 0, "reviews": 0, "literature": 0, "
    papers": 0, "editorial-media": 0, "2d": 0, "3d": 0, "line-plot": 0, "
    pixel-area": 0, "node-link": 0, "clouds": 0, "maps": 0, "text": 13,
    "glyph": 0, "radial": 0, "linear": 0, "metric": 0}
}

```

Listing 4.8: An example of category term matching results based on simple matching approach (without topic modeling) for full texts (in this case, for the paper by Abbasi and Chen [44])

does not provide us with a concise summary of how well the computational results fit/match the original dataset labels. Therefore, we have tried to summarize these matching results in a quantitative way in order to be able to evaluate the matching for any given publication entry (or the complete dataset on average) by looking at just one or several metric/index values. We did not rely on just one approach for computing such an index value, after reading the literature we followed two approaches to get the matching index of the computational results with the manually labeled dataset. We used *term match index* (similar to the *precision* metric) and *category match index* (corresponding to *Jaccard index*), which are both described below. The value range for both indexes is 0.0–1.0. The value of 1.0 means the ideal match between the computed categories and the categories present in the manually labeled dataset (i.e., the original TextVis Browser dataset).

To compute both indexes, we at first define set  $\mathbf{C}(entry)$  as the set of all categories pertaining to a single entry/paper in the source dataset (see Listing 4.1 for an example). We also define set  $\hat{\mathbf{C}}(entry)$  as the set of extracted categories for an entry with count values larger than 0:

$$\hat{\mathbf{C}}(entry) = \{category\} \mid count(entry, category) > 0 \quad (1)$$

For example, the set of extracted categories  $\hat{\mathbf{C}}(entry)$  in Listing 4.3 simply consists of three items: "sentiment-analysis", "trend-analysis", and "networks".

The term match index value  $TMI(entry)$  for a single entry/paper is then calculated according to the following formula:

$$TMI(entry) = \frac{\sum_{category \in \mathbf{C}(entry)} count(entry, category)}{\sum_{category \in \hat{\mathbf{C}}(entry)} count(entry, category)} \quad (2)$$

where  $count(entry, category)$  is the detected count of key terms corresponding to a specific category for a given dataset entry, e.g., in the example in Listing 4.3, the count for the entry "Abbasi2006" and the category "trend-analysis" is equal to 4.

To compute the term match index for an entry, we essentially calculate the sum of matched term counts for the categories present in the source data, then divide this value with the total sum of counts for that entry. These calculations were carried out for each of the 400 entries corresponding to the publications in the TextVis Browser dataset.



The term match index acts similar to the precision measure used in machine learning and NLP [25], which is focused on the *true positive* cases: it indicates how well are the categories from the source data (which could be considered the *gold standard* or the *ground truth*) represented in the computational results extracted from the publication texts. It also takes the absolute count values into account as weights, so if, for instance, the terms for categories present in the entry source data are matched 100 times in total, and the counts for other categories amount only to 10 in total, the term match index will still have a high value.

At the same time, it is not only *true positive* cases that are interesting to investigate. Therefore, we have decided to calculate an additional measure called the category match index, which is based on Jaccard index [36] for the sets of categories:

$$CMI(entry) = \frac{|\widehat{\mathbf{C}}(entry) \cap \mathbf{C}(entry)|}{|\widehat{\mathbf{C}}(entry) \cup \mathbf{C}(entry)|} \quad (3)$$

To compute the category match index, we essentially calculate the overlap between the set of categories from the source data and the set of categories detected in the textual data. If either many source data categories are missing in the extracted results (the *false negative* case), or there are many detected categories there which are not present in the source data (the *false positive* case), the category match index will have a rather low value.

---

```

{
  "id": "Abbasi2006",
  "source_categories": ["translation-analysis", "trend-analysis", "
    overview", "comparison", "classification", "corpora", "social-media",
    "2d", "clouds", "maps", "metric"],
  "matched_lda_abstract_categories": {"text-summarization": 0, "discourse-
    analysis": 0, "stance-analysis": 0, "sentiment-analysis": 1, "event-
    analysis": 0, "trend-analysis": 4, "lexical-analysis": 0, "
    relation-analysis": 0, "translation-analysis": 0, "roi": 0, "
    classification": 0, "comparison": 0, "overview": 0, "monitoring": 0,
    "navigation": 0, "uncertainty": 0, "document": 0, "corpora": 0, "
    streams": 0, "geospatial": 0, "time-series": 0, "networks": 1, "
    social-media": 0, "communication": 0, "patents": 0, "reviews": 0, "
    literature": 0, "papers": 0, "editorial-media": 0, "2d": 0, "3d": 0,
    "line-plot": 0, "pixel-area": 0, "node-link": 0, "clouds": 0, "maps":
    0, "text": 0, "glyph": 0, "radial": 0, "linear": 0, "metric": 0},
  "term_match_index": 0.67,
  "category_match_index": 0.08
}

```

---

Listing 4.9: An example of index values summarizing matching results based on LDA for abstracts (in this case, for the paper by Abbasi and Chen [44])

Listing 4.9 shows the matching results of term and category index for one paper’s abstract out of 400. The values calculated by LDA of abstracts after matching with the manually labeled dataset. In the Listing 4.9, the ‘source\_categories’ are the categories which are manually assigned by ISOVIS Group to this paper and the ‘matched\_lda\_abstract\_categories’ are matched categories results described in Section 4.4.5. The term and category match indexes are calculated in this step to check the matching ratio of matching results.

We also computed the computational matching with full texts matching results for LDA. Listing 4.10 shows the computational matching results of category and term index for one paper’s full text out of 400. We can see in the Listing 4.10 that the results of full text for category and term index are better as compare to abstract in Listing 4.9. In the abstract of this paper, the term match index is 0.67 but in the full

---

```

{
  "id": "Abbasi2006",
  "source_categories": ["translation-analysis", "trend-analysis", "
    overview", "comparison", "classification", "corpora", "social-media",
    "2d", "clouds", "maps", "metric"],
  "matched_lda_fulltext_categories": {"text-summarization": 0, "discourse-
    analysis": 0, "stance-analysis": 0, "sentiment-analysis": 0, "event-
    analysis": 0, "trend-analysis": 19, "lexical-analysis": 0, "relation-
    analysis": 0, "translation-analysis": 0, "roi": 0,
    "classification": 15, "comparison": 0, "overview": 0, "monitoring":
    0, "navigation": 0, "uncertainty": 0, "document": 0, "corpora": 0, "
    streams": 0, "geospatial": 0, "time-series": 0, "networks": 2, "
    social-media": 0, "communication": 0, "patents": 0, "reviews": 0, "
    literature": 0, "papers": 0, "editorial-media": 0, "2d": 0, "3d": 0,
    "line-plot": 0, "pixel-area": 0, "node-link": 0, "clouds": 0, "maps":
    0, "text": 2, "glyph": 0, "radial": 0, "linear": 0, "metric": 0},
  "term_match_index": 0.89,
  "category_match_index": 0.15
}

```

---

Listing 4.10: An example of index values summarizing matching results based on LDA for full texts (in this case, for the paper by Abbasi and Chen [44])

text, it has the value of 0.89 and category match index has 0.08 but in the full-text category, match index has 0.15. The results of the full text are more appropriate.

---

```

{
  "id": "Abbasi2006",
  "source_categories": ["translation-analysis", "trend-analysis", "
    overview", "comparison", "classification", "corpora", "social-media",
    "2d", "clouds", "maps", "metric"],
  "matched_hdp_abstract_categories": {"text-summarization": 1, "discourse-
    analysis": 0, "stance-analysis": 3, "sentiment-analysis": 4, "
    event-analysis": 0, "trend-analysis": 0, "lexical-analysis": 0, "
    relation-analysis": 0, "translation-analysis": 0, "roi": 0,
    "classification": 7, "comparison": 0, "overview": 0, "monitoring": 0
    , "navigation": 0, "uncertainty": 0, "document": 0, "corpora": 0, "
    streams": 0, "geospatial": 0, "time-series": 2, "networks": 0,
    "social-media": 1, "communication": 0, "patents": 0, "reviews": 0, "
    literature": 0, "papers": 0, "editorial-media": 0, "2d": 0, "3d": 0,
    "line-plot": 0, "pixel-area": 0, "node-link": 0, "clouds": 0, "maps":
    0, "text": 1, "glyph": 0, "radial": 0, "linear": 0, "metric": 2},
  "term_match_index": 0.48,
  "category_match_index": 0.19
}

```

---

Listing 4.11: An example of index values summarizing matching results based on HDP for abstracts (in this case, for the paper by Abbasi and Chen [44])

As the next step, we will see the computational matching results using HDP for abstracts and full-text. Listing 4.11 shows the computational matching results of category and term index for one paper out of 400 of its abstract using HDP. If we compare these results with LDA abstracts of term and category match index the results for the term of LDA are better but for the category, the results of HDP are better.

Listing 4.12 shows the computational matching results of category and term index for one paper out of 400 of its full text using HDP. If we compare these results with LDA full texts of term and category match index the results for the term of LDA are better but for the category, the results of HDP are better.

As discussed above in Section 4.4.6, besides using the topic modeling results to summarize the publication text contents, we also implemented an additional baseline method for simply using the preprocessed publication texts and matching the category key terms using the additional category-to-terms mapping (see List-

---

```

{
  "id": "Abbasi2006",
  "source_categories": ["translation-analysis", "trend-analysis",
    "overview", "comparison", "classification", "corpora",
    "social-media", "2d", "clouds", "maps", "metric"],
  "matched_hdp_fulltext_categories": {"text-summarization": 3, "discourse-
    analysis": 0, "stance-analysis": 1, "sentiment-analysis": 4, "
    event-analysis": 0, "trend-analysis": 2, "lexical-analysis": 0,
    "relation-analysis": 2, "translation-analysis": 2, "roi": 3,
    "classification": 3, "comparison": 1, "overview": 1, "monitoring":
    0, "navigation": 1, "uncertainty": 1, "document": 1,
    "corpora": 1, "streams": 0, "geospatial": 0, "time-series": 2,
    "networks": 7, "social-media": 1, "communication": 2, "patents": 0
    , "reviews": 0, "literature": 2, "papers": 2,
    "editorial-media": 2, "2d": 0, "3d": 0, "line-plot": 0,
    "pixel-area": 1, "node-link": 2, "clouds": 0, "maps": 0,
    "text": 2, "glyph": 0, "radial": 0, "linear": 3, "metric": 0},
  "term_match_index": 0.21,
  "category_match_index": 0.24
}

```

---

Listing 4.12: An example of index values summarizing matching results based on HDP for full texts (in this case, for the paper by Abbasi and Chen [44])

ing 4.2). As displayed in Figure 4.12, we have calculated the term and category match index values for the output of this additional method, too.

Listing 4.13 shows the computational matching results of category and term index for one paper out of 400 of its abstract for simple text matching.

---

```

{
  "id": "Abbasi2006",
  "source_categories": ["translation-analysis", "trend-analysis", "
    overview", "comparison", "classification", "corpora", "social-media",
    "2d", "clouds", "maps", "metric"],
  "matched_abstract_categories": {"text-summarization": 0, "discourse-
    analysis": 0, "stance-analysis": 0, "sentiment-analysis": 0, "event-
    analysis": 0, "trend-analysis": 0, "lexical-analysis": 0, "relation-
    analysis": 0, "translation-analysis": 0, "roi": 0,
    "classification": 1, "comparison": 1, "overview": 0, "monitoring":
    0, "navigation": 0, "uncertainty": 0, "document": 0, "corpora": 0, "
    streams": 0, "geospatial": 0, "time-series": 0, "networks": 0, "
    social-media": 0, "communication": 0, "patents": 0, "reviews": 0, "
    literature": 0, "papers": 0, "editorial-media": 0, "2d": 0, "3d": 0,
    "line-plot": 0, "pixel-area": 0, "node-link": 0, "clouds": 0, "maps":
    0, "text": 0, "glyph": 0, "radial": 0, "linear": 0, "metric": 0},
  "term_match_index": 1,
  "category_match_index": 0.18
}

```

---

Listing 4.13: An example of index values summarizing matching results based on simple matching approach (without topic modeling) for abstracts (in this case, for the paper by Abbasi and Chen [44])

Listing 4.14 shows the computational matching results of category and term index for one paper out of 400 of its full text for simple text matching. If we compare this matching result with LDA and HDP these results for category and term index have lower values.

After doing the computational matching for LDA, HDP and simple text matching approach for abstracts and full texts, we also calculated their average values to see which approach results are mostly matched with the labeled dataset.

```

{
  "id": "Abbasi2006",
  "source_categories": ["translation-analysis", "trend-analysis",
    "overview", "comparison", "classification", "corpora", "social-media",
    "2d", "clouds", "maps", "metric"],
  "matched_fulltext_categories": {"text-summarization": 0, "discourse-
    analysis": 0, "stance-analysis": 0, "sentiment-analysis": 0, "event-
    analysis": 0, "trend-analysis": 0, "lexical-analysis": 0, "relation-
    analysis": 0, "translation-analysis": 0, "roi": 0,
    "classification": 9, "comparison": 3, "overview": 1, "monitoring":
    0, "navigation": 0, "uncertainty": 0, "document": 2, "corpora": 1,
    "streams": 0, "geospatial": 0, "time-series": 0, "networks": 0, "
    social-media": 0, "communication": 5, "patents": 0, "reviews": 0, "
    literature": 0, "papers": 0, "editorial-media": 0, "2d": 0, "3d": 0,
    "line-plot": 0, "pixel-area": 0, "node-link": 0, "clouds": 0, "maps":
    0, "text": 13, "glyph": 0, "radial": 0, "linear": 0, "metric": 0},
  "term_match_index": 0.41,
  "category_match_index": 0.29
}

```

Listing 4.14: An example of index values summarizing matching results based on simple matching approach (without topic modeling) for full texts (in this case, for the paper by Abbasi and Chen [44])

	LDA		HDP		Simple Matching	
	Term	Category	Term	Category	Term	Category
Abstracts	0.52	0.18	<b>0.53</b>	<b>0.23</b>	0.43	0.06
Full Text	<b>0.60</b>	0.24	0.43	<b>0.31</b>	0.49	0.20

Table 4.12: Average results of computational matching for the complete dataset

Table 4.12 shows the average computational matching results for LDA, HDP and simple text matching for term and category/Jaccard index. If we compare the average computational matching results for abstract of LDA, HDP, simple text matching for term and category index, we can observe that the LDA and HDP have good average values as compared to simple text matching. The computational matching for full text for the term using LDA has good matching values as compared to HDP and simple text matching values. For the full text of category/Jaccard index, the HDP has good matching values as compared to LDA and simple text matching approach.

The additional interpretation of the matching results and further applications of this approach is discussed by the domain experts in Section 5.5.

## 5 Discussion

In this chapter, we describe the results of discussions with the members of the ISOVIS Group who provided their opinions and interpretation of the results and details of various stages of this project based on their expertise in text visualization field.<sup>7</sup>

### 5.1 Original Dataset

First of all, the experts made some comments about the origins and the status of the TextVis Browser dataset. The main JSON file with the metadata about various text visualization techniques corresponding to scientific publications was created, maintained, and edited since 2014 by two members of the ISOVIS Group. Some updates had to be done over time even for the basic fields of the metadata such as the publication year: on some occasions, the corresponding papers were initially published in online access, and then eventually published in specific journal issues after the turn of the year. The entries then had to be updated. By mid-2018, however, there are only a few entries in the dataset that correspond to the recent papers published in 2017, which might be affected by this issue in the future, thus, these changes should not affect the overall temporal trends discovered in this thesis project.

The process of assigning category labels to the entries also had its caveats. The experts carefully read the corresponding papers, then checked the list of categories one by one, and tried to make a judgement whether the paper contained sufficient evidence (or at least strong hints) of being related to each category. This process was subjective to a certain degree and had to rely on the domain knowledge. For instance, most text visualization techniques presented in the TextVis Browser dataset use only 2D for their visual representations, and this is usually obvious from the screenshots in the corresponding papers, but not mentioned explicitly in the text of the papers. Another example is the "overview" category that corresponds to the task of providing a visual overview of the complete dataset. Most of the researchers working in text visualization are aware of the famous InfoVis mantra "Overview first, zoom and filter, then details-on-demand" by Ben Shneiderman [45] and tend to follow it. Thus, providing an overview is a standard task that is often not mentioned explicitly in the text of the papers.

The variety of the categories in the dataset and the degree of uncertainty with regard to the corresponding evidence in some papers are the key factors why the dataset was eventually edited multiple times over time with regard to the category assignment to various entries/papers. The experts actually see the possibility of applying automatic matching of text mining results to their manually assigned category labels as an important contribution of this thesis project.

### 5.2 Data Collection and Preprocessing Results

The domain experts appreciated the efforts related to the data collection and preprocessing steps, noting that the process of converting the contents of PDF files with publications into plain text files is a very tedious and time-consuming process. The members of the ISOVIS Group have themselves recently described their experiences with a similar task for a different dataset [46], but it is worth noting that

---

<sup>7</sup>The text of this chapter was written together with the members of the ISOVIS Group who provided feedback for the thesis project.

the publication dataset mentioned in their paper consists of 221 PDF files, while the dataset used in this thesis project is almost twice as large (400 files). The experts have also noted that the task of converting PDF files of scientific publications into plain text is non-trivial and is still not supported well by available tools, despite existing efforts described by Constantin et al. [47] and Goodman et al. [48], among others. Therefore, the domain experts see the extracted dataset of plain texts for the scientific publications corresponding to the TextVis Browser dataset as an important asset and a contribution of this thesis project in itself.

### 5.3 Temporal Analysis Results

The domain experts have studied the results of several types of temporal analyses discussed in Section 4.2. They have noted that the results of the overall temporal analysis of the dataset based on publication year, as displayed in Figure 4.1, are consistent with the results and predictions are done by them in the original paper about the TextVis Browser dataset in 2015 [2]. The field of text visualization has continued to grow with a lot of new publications, especially in 2016, and some publications from the previous years were also added to the survey dataset over time. This general temporal trend is also consistent with the results of the SentimentVis Browser dataset analysis [4], which was focusing on a subset of the TextVis Browser dataset with a more specialized categorization in mind.

The results of the fine-grained temporal analysis presented in Figures 4.2, 4.3, and 4.4 are also consistent with the results for the SentimentVis Browser dataset with regard to the categories present in both categorizations/taxonomies, and they have confirmed the expectations of the domain experts. For instance, the support of the basic tasks of "Overview" and "Comparison" as well as the data source type of "Corpora" is consistent with regard to the complete dataset. The support for the data source type "Document" has decreased since the 2000s, and the interest for text visualization techniques using 3D visual representations in visualization community has almost disappeared since then, too. It is also possible to note that support for most data domains (such as literature and editorial media) has decreased since the invention of social media in the 2000s.

### 5.4 Correlation Analysis Results

The domain experts have agreed that the results of category correlation analysis presented in Section 4.3 are also consistent with the subset of the results for the SentimentVis Browser dataset. The interpretation of the correlation matrix displayed in Figure 4.6 depends on the definition of various levels of correlation strengths. While the guidelines existing in the literature vary in their suggestions for the weak, moderate, and strong correlation levels [38, 49, 50], the absolute value of 0.40 is agreed on by all of these sources as indicating at least a moderate level of correlation. Therefore, the experts focused on the cases with the positive correlation of  $\geq 0.40$  or the negative correlation of  $\leq -0.40$ .

First of all, the domain experts commented on the cases of moderate and strong negative correlation, which has occurred between pairs of categories from the same groups (or branches of the taxonomy). As noted above in Section 5.3, there is a relation between the decreasing support for the data source type "Document" (i.e., a single document/file used as data for visualization) and the increasing support for "Corpora" (i.e., a collection of documents used as data) in the dataset, and it is cor-

robored by the negative correlation value of  $-0.48$  between these two categories. 2D and 3D visual representations are clearly not often used together in the dataset, as the correlation of these categories is  $-0.78$ . The same applies for the "Radial" and "Metric" categories of visual alignment, which encode how the elements of visualizations are laid out and aligned. This pair of categories has the correlation value of  $-0.56$ : the authors of the corresponding text visualization techniques apparently prefer not to use both of them simultaneously.

With regard to the cases of positive correlation, the experts stated that interesting cases tend to involve correlations between categories from different groups. The task of "Lexical Analysis" (which is related to the investigation of lexical and syntactical properties using visualization) seems to be rather specific to the data domain of "Literature" with the correlation value of 0.47. The data in the "Literature" domain seems to mainly come in the "Document" form (correlation: 0.44). On the other hand, for the "Social Media" domain, the data is mainly characterized by the presence of the "Time Series" category (correlation: 0.40). The interesting cases related to positive correlations between analytical and visual/interaction tasks and specific data domains and properties include the pairs of the "Sentiment Analysis" task and the "Social Media" domain (0.44), the "Trend Analysis" task (i.e., investigation of trends and patterns over time) and the "Time Series" data property (0.64), the "Relation Analysis" task and the "Networks" data property (0.41), and finally, the "Monitoring" task and the "Streams" data property (0.68). With regard to particular visual representations, the "Line Plot" category (which also includes representations such as rivers/streamgraphs [51]) tend to be used together with the "Time Series" data property (0.49) and the "Trend Analysis" task (0.52), which is quite natural to the experts. The "Node-Link" visual representation (i.e., graphs are drawn with vertices/nodes and edges/links) is often used together with the "Networks" data property (0.61) and the "Relation Analysis" task (0.48), which is also confirming the experts' expectations. Finally, the "Geospatial" data property has a positive correlation value of 0.51 with the "Maps" visual representation, which can be interpreted as geo-tagged data being visualized using (geographical) maps.

In general, the domain experts have agreed that the correlation analysis results confirm their expectations of the usage of categories in the data. They also noted that further investigation of the cases with absolute correlation values  $\leq 0.40$  could be undertaken to gain additional insights.

## 5.5 Publication Text Analysis Results

The domain experts have studied the topic modeling results described in Section 4.4.4. They have noted that while the *local* topic modeling results could provide a lot of detailed information about any particular paper, the overall amount of data for all of the publications corresponding to 400 dataset entries is overwhelming for manual exploration. They have noted, though, that such a *local* approach could be useful for their workflow in the future to generate a quick summary for a new publication while studying and manually labelling it with category labels.

The experts have then studied the results of *global* topic modeling in order to get an overview of the complete dataset. They have stated that the results of both LDA and HDP algorithms for both abstract- and full text-based approaches are rather coherent topics related to research on information visualization. Some of the interesting topic terms present in the results in Table 4.8 and Table 4.9 include "sentiment",

"opinion", "topic", and "time". These topic terms are directly relevant to some of the most prominent categories in the TextVis Browser dataset, according to the experts. In general, the experts have noted that further experimentation with various stages of *global* topic modeling is possible in order to focus on more specific topics, perhaps, by excluding an extended set of domain-specific stop words such as "visualization" or "information", which are rather general for such a dataset.

As described above in Section 4.4.5, the domain experts from the ISOVIS Group were directly involved in the preparation of the additional dataset with the mapping between categories and corresponding key terms (see Listing 4.2 and Appendix A.2). According to them, some of the categories used in the TextVis Browser dataset was problematic to describe with a list of terms for the purposes of lexical matching, for instance, the category "Text" from the taxonomy/categorization is related to text being used as a visual representation, e.g., by applying various font attributes (size, weight, color) to encode sentiment, as discussed in the paper by Wecker et al. [52]. Using the actual word "text" as one of the key terms for the corresponding category was not practical, though, as it is commonly used in scientific publications, especially the ones dedicated to *text* visualization. The resulting categories-to-terms mapping and the lexical matching approach are, therefore, far from ideal and could be improved in the future, according to the experts.

The experts have been very interested in the results of matching between the categories in the source TextVis Browser dataset and the categories extracted using the computational approaches, including the average index values described in Section 4.4.7. They have noted that while the final results listed in Table 4.12 are far from the ideal value of 1.0, they generally correspond to the experts' expectations. As discussed above, some of the categories in the TextVis Browser dataset are rather ambiguous and their labelling depends on the experts' subjective opinion. Therefore, the experts from the ISOVIS Group had actually not expected the computational results extracted from texts to match the source dataset with high index values. They have also noted that one opportunity for future work is to compute index values for each category separately and use that to investigate whether some categories have particularly bad matching results over the complete dataset.

Rather than focusing on the average index values in the current project results, the domain experts have stated that values computed for each entry/publication provide them with an opportunity to find outliers in the existing TextVis Browser dataset, and to carefully check the contents of the publication against the manually labeled data. For instance, one of the entries checked by the domain experts is the entry corresponding to the paper by Havre et al. on ThemeRiver [10]. The experts have used the results of the matching based on HDP and full texts. They have noticed that multiple terms related to the "Text Summarization" category are detected, while this category is missing in the manually assigned set of categories in the TextVis Browser dataset. As another example, the experts have examined the entry corresponding to the paper by Lee et al. [53], which has the lowest index values for HDP / full-text results, namely, the term match index of 0.06 and category match index of 0.08. The experts have agreed that the source entry in the TextVis Browser dataset could be extended by some of the categories with matched terms, such as "Text Summarization", "Time Series", and "Clouds". These examples illustrate how the computational matching results achieved in this thesis project can be used by the domain experts to refine their manually curated dataset. The experts have noted that they find this contribution of the project important for their work.



## 6 Conclusions and Future Work

The goal we have defined in this thesis project was successfully achieved after doing several data analyses of the TextVis Browser dataset.

In order to answer the research question **RQ1**, we analyzed the TextVis Browser dataset. We discovered the current and historical state of the art of text visualization techniques, how text visualization becomes more popular in the previous 27 years. After doing the data analyses, we observed that after 2006 text visualization is becoming more popular and the publications for text visualization are gradually increasing. To show the trends in a visual form we draw the histograms for each category. We found the temporal trends of each category regarding year. The histograms of these 41 categories show that few categories became less popular in the previous 27 years. However few categories became more popular in the previous 27 years.

For the research question **RQ2**, we further analyzed the TextVis Browser dataset and created a categories matrix. In our implementation, we calculated a correlation matrix for the 41 categories to find out the correlation between a pair of categories which are used in TextVis Browser. The Pearson's coefficient is used to compute the correlation between the pair of categories. After computing the coefficients, we found that few categories have negative correlations between each other. However more categories have positive correlations between each other as compared to the negative correlations.

In order to address the final research question **RQ3**, the topic modeling is carried out using two algorithms (LDA and HDP) for the 400 publications which are used in TextVis Browser. For the topic modeling approach, we have collected the PDF files of 400 publications which are used in TextVis Browser and created the plain text files of each publication. The use of bigram in preprocessing helped us to get the better topic modeling results. We observe that the preprocessing was a good approach to remove the unimportant data from the text of the publications. The use of two algorithms LDA and HDP for topic modeling was good to get the different results for the same task in order to verify the accuracy of results. The results of topic modeling are compared with manually labeled dataset to find the correspondence between topic modeling results and the manually labeled dataset. After topic modeling, the 3rd approach is also applied for the matching of publications text directly with the manually labeled dataset. We named it simple text matching. The analyses with different methods, such as LDA, HDP and simple text matching helped us to see whether topic modeling is a good approach to discover the hidden semantic structures from the text. The results of these analyses were presented to the ISOVIS Group to look at the improvements in manually assigned categories and also in interactive TextVis Browser.

The results of this thesis project are useful for researchers to get an insight about text visualization techniques. They can immediately get an idea about text visualization when it became more popular from Figure 4.1 and also after viewing the temporal trends results they can imagine how some categories became less or more popular. They can easily get an idea about text visualization techniques and how it became more popular. The results of this thesis project are based on data provided by the ISOVIS Group and are relevant to text visualization. However, we can use the same approaches like text mining, topic modeling for other scientific fields

(e.g, bioinformatics, academic, scientific, etc.), which will help the researchers to provide the state of the art and summaries for their field of interest.

**Future Work** The possibility of evaluating the TextVis Browser and the taxonomy that the experts had manually constructed is one possible future work task. There are several other data analysis methods and algorithms that can be used for data analysis, text mining, and natural language processing. We can get different results for the same data after analysing it with different types of methods and algorithms. Rather than relying only on linear correlation analysis for pairs of data categories, it is possible to investigate the hidden relationships in the dataset further with more advanced data analysis methods. Other topic modeling and clustering algorithms also exist for text mining such as Latent Semantic Analysis (LSA), Correlated Topic Model (CTM), Hierarchical Latent Dirichlet Allocation (hLDA), K-means clustering, etc. The results of this thesis are based on topics and topic modeling algorithms but we could also check other properties such as topic reliability, perplexity, and coherence score, etc., in order to obtain even more reliable and optimized topics.

## References

- [1] N. Cao and W. Cui, *Introduction to Text Visualization*. Springer, 2016.
- [2] K. Kucher and A. Kerren, “Text visualization techniques: Taxonomy, visual survey, and community insights,” in *Proceedings of the IEEE Pacific Visualization Symposium*, ser. PacificVis ’15, 2015, pp. 117–121. [Online]. Available: <https://doi.org/10.1109/PACIFICVIS.2015.7156366>
- [3] H. Lu and G. Liu, “Text visualization and visual analytics based on multi-layer topic maps,” *Journal of Information and Computational Science*, vol. 8, no. 12, pp. 2459–2464, 2011.
- [4] K. Kucher, C. Paradis, and A. Kerren, “The state of the art in sentiment visualization,” *Computer Graphics Forum*, vol. 37, no. 1, pp. 71–96, Feb. 2018. [Online]. Available: <https://doi.org/10.1111/cgf.13217>
- [5] P. Federico, F. Heimerl, S. Koch, and S. Miksch, “A survey on visual approaches for analyzing scientific literature and patents,” *IEEE Transactions on Visualization and Computer Graphics*, vol. 23, pp. 2179–2198, 2017. [Online]. Available: <https://doi.org/10.1109/TVCG.2016.2610422>
- [6] A. Suominen and H. Toivanen, “Map of science with topic modeling: Comparison of unsupervised learning and human-assigned subject classification,” *Journal of the Association for Information Science and Technology*, vol. 67, no. 10, pp. 2464–2476, 2016. [Online]. Available: <https://doi.org/10.1002/asi.23596>
- [7] C. K. Yau, A. Porter, N. Newman, and A. Suominen, “Clustering scientific documents with topic modeling,” *Scientometrics*, vol. 100, pp. 767–786, 2014. [Online]. Available: <https://doi.org/10.1007/s11192-014-1321-8>
- [8] F. Chen, P. Chiu, and S. Lim, “Topic modeling of document metadata for visualizing collaborations over time,” in *Proceedings of the International Conference on Intelligent User Interfaces*, ser. IUI ’16, 2016, pp. 108–117. [Online]. Available: <http://doi.acm.org/10.1145/2856767.2856787>
- [9] T. L. Griffiths and M. Steyvers, “Finding scientific topics,” *Proceedings of the National academy of Sciences*, vol. 101, pp. 5228–5235, 2004. [Online]. Available: <https://doi.org/10.1073/pnas.0307752101>
- [10] S. Havre, B. Hetzler, and L. Nowell, “ThemeRiver: In search of trends, patterns, and relationships,” *IEEE Transactions on Visualization and Computer Graphics*, vol. 8, no. 1, pp. 9–20, 2002. [Online]. Available: <https://doi.org/10.1109/2945.981848>
- [11] D. M. Blei, A. Y. Ng, and M. I. Jordan, “Latent Dirichlet allocation,” *Journal of Machine Learning Research*, vol. 3, pp. 993–1022, Jan. 2003. [Online]. Available: <https://dl.acm.org/citation.cfm?id=944937>
- [12] S. K. Card, J. Mackinlay, and B. Shneiderman, *Readings in Information Visualization: Using Vision to Think*. Morgan Kaufmann, 1999. [Online]. Available: <https://dl.acm.org/citation.cfm?id=300679>

- [13] W. J. Schroeder, B. Lorensen, and K. Martin, *The Visualization Toolkit: An Object-Oriented Approach to 3D Graphics*. Kitware, 2004.
- [14] L. Manovich, “What is visualization?” *Poetess Archive Journal*, vol. 2, no. 1, 2010.
- [15] J. A. Wise, J. J. Thomas, K. Pennock, D. Lantrip, M. Pottier, A. Schur, and V. Crow, “Visualizing the non-visual: Spatial analysis and interaction with information from text documents,” in *Proceedings of the Information Visualization Symposium*, ser. INFOVIS '95, 1995, pp. 51–58. [Online]. Available: <https://doi.org/10.1109/INFVIS.1995.528686>
- [16] S. Tufféry, *Data Mining and Statistics for Decision Making*. Wiley Chichester, 2011.
- [17] M. J. Norton, *Introductory Concepts in Information Science*. Information Today, Inc., 2000.
- [18] N. De Bellis, *Bibliometrics and Citation Analysis: From the Science Citation Index to Cybermetrics*. Scarecrow Press, 2009.
- [19] S. Morris, C. DeYong, Z. Wu, S. Salman, and D. Yemenu, “DIVA: A visualization system for exploring document databases for technology forecasting,” *Computers & Industrial Engineering*, vol. 43, no. 4, pp. 841–862, 2002. [Online]. Available: [https://doi.org/10.1016/S0360-8352\(02\)00143-2](https://doi.org/10.1016/S0360-8352(02)00143-2)
- [20] M. Türkeş, “Spatial and temporal analysis of annual rainfall variations in Turkey,” *International Journal of Climatology*, vol. 16, no. 9, pp. 1057–1076, 1996. [Online]. Available: [https://doi.org/10.1002/\(SICI\)1097-0088\(199609\)16:9%3C1057::AID-JOC75%3E3.0.CO;2-D](https://doi.org/10.1002/(SICI)1097-0088(199609)16:9%3C1057::AID-JOC75%3E3.0.CO;2-D)
- [21] K. H. Zou, K. Tuncali, and S. G. Silverman, “Correlation and simple linear regression,” *Radiology*, vol. 227, no. 3, pp. 617–628, 2003.
- [22] R. Månsson, P. Tsapogas, M. Åkerlund, A. Lagergren, R. Gisler, and M. Sigvardsson, “Pearson correlation analysis of micro-array data allows for the identification of genetic targets for early b-cell factor,” *Journal of Biological Chemistry*, 2004.
- [23] S. Stemler, “An overview of content analysis,” *Practical Assessment, Research & Evaluation*, vol. 7, no. 17, pp. 137–146, 2001.
- [24] S. Stemler and D. Bebell, “An empirical approach to understanding and analyzing the mission statements of selected educational institutions,” in *Presentations of the Annual Meeting of the New England Educational Research Organization (NEERO)*, 1999.
- [25] C. D. Manning and H. Schütze, *Foundations of Statistical Natural Language Processing*. Cambridge, MA: MIT Press, 1999.
- [26] E. Loper and S. Bird, “NLTK: The natural language toolkit,” in *Proceedings of the ACL-02 Workshop on Effective Tools and Methodologies for*

*Teaching Natural Language Processing and Computational Linguistics — Volume 1*, ser. ETMTNLP '02, 2002, pp. 63–70. [Online]. Available: <https://doi.org/10.3115/1118108.1118117>

- [27] J. V. Guttag, *Introduction to Computation and Programming Using Python: With Application to Understanding Data*. MIT Press, 2016.
- [28] R. Řehůřek and P. Sojka, “Software framework for topic modelling with large corpora,” in *Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks*, 2010, pp. 45–50. [Online]. Available: <https://doi.org/10.13140/2.1.2393.1847>
- [29] D. M. Blei, “Probabilistic topic models,” *Communications of the ACM*, vol. 55, pp. 77–84, 2012. [Online]. Available: <http://doi.acm.org/10.1145/2133806.2133826>
- [30] Y. W. Teh, M. I. Jordan, M. J. Beal, and D. M. Blei, “Sharing clusters among related groups: Hierarchical Dirichlet processes,” in *Advances in Neural Information Processing Systems 17*, ser. NIPS 2004, L. K. Saul, Y. Weiss, and L. Bottou, Eds. MIT Press, 2005, pp. 1385–1392.
- [31] W. McKinney, “pandas: A foundational Python library for data analysis and statistics,” in *Proceedings of the Workshop on Python for High Performance and Scientific Computing*, ser. PyHPC '11, 2011.
- [32] T. E. Oliphant, *Guide to NumPy*, 2nd ed. Scotts Valley, CA: CreateSpace Independent Publishing Platform, 2015.
- [33] C. Ramasubramanian and R. Ramya, “Effective pre-processing activities in text mining using improved Porter’s stemming algorithm,” *International Journal of Advanced Research in Computer and Communication Engineering*, vol. 2, no. 12, pp. 4536–4538, 2013.
- [34] V. Balakrishnan and E. Lloyd-Yemoh, “Stemming and lemmatization: A comparison of retrieval performances,” *Lecture Notes on Software Engineering*, vol. 2, no. 3, pp. 262–267, 2014. [Online]. Available: <https://doi.org/10.7763/LNSE.2014.V2.134>
- [35] T. Korenius, J. Laurikkala, K. Järvelin, and M. Juhola, “Stemming and lemmatization in the clustering of Finnish text documents,” in *Proceedings of the ACM International Conference on Information and Knowledge Management*, ser. CIKM '04, 2004, pp. 625–633. [Online]. Available: <https://doi.org/10.1145/1031171.1031285>
- [36] P. Jaccard, “The distribution of the flora in the Alpine zone,” *New Phytologist*, vol. 11, no. 2, pp. 37–50, 1912. [Online]. Available: <https://doi.org/10.1111/j.1469-8137.1912.tb05611.x>
- [37] G. G. Robertson and J. D. Mackinlay, “The document lens,” in *Proceedings of the Annual ACM Symposium on User Interface Software and Technology*, ser. UIST '93, 1993, pp. 101–108. [Online]. Available: <http://doi.acm.org/10.1145/168642.168652>








- [38] R. Taylor, “Interpretation of the correlation coefficient: A basic review,” *Journal of Diagnostic Medical Sonography*, vol. 6, no. 1, pp. 35–39, 1990. [Online]. Available: <https://doi.org/10.1177%2F875647939000600106>
- [39] R. R. Pagano, *Understanding Statistics in the Behavioral Sciences*. Cengage Learning, 2012.
- [40] A. Asuero, A. Sayago, and A. Gonzalez, “The correlation coefficient: An overview,” *Critical Reviews in Analytical Chemistry*, vol. 36, no. 1, pp. 41–59, 2006. [Online]. Available: <https://doi.org/10.1080/10408340500526766>
- [41] T. Kumamoto, H. Wada, and T. Suzuki, “Visualizing temporal changes in impressions from tweets,” in *Proceedings of the International Conference on Information Integration and Web-based Applications & Services*, ser. iiWAS ’14, 2014, pp. 116–125. [Online]. Available: <http://doi.acm.org/10.1145/2684200.2684279>
- [42] A. S. Nayak, A. P. Kanive, N. Chandavekar, and B. R., “Survey on pre-processing techniques for text mining,” *International Journal of Engineering and Computer Science*, vol. 5, no. 6, pp. 16 875–16 879, 2016. [Online]. Available: <https://doi.org/10.1080/10.18535/ijecs/v5i6.25>
- [43] D. Munková, M. Munk, and M. Vozár, “Data pre-processing evaluation for text mining: Transaction/sequence model,” *Procedia Computer Science*, vol. 18, pp. 1198–1207, 2013. [Online]. Available: <https://doi.org/10.1016/j.procs.2013.05.286>
- [44] A. Abbasi and H. Chen, “Visualizing authorship for identification,” in *Intelligence and Security Informatics*, ser. LNCS, S. Mehrotra, D. D. Zeng, H. Chen, B. Thuraisingham, and F.-Y. Wang, Eds., vol. 3975. Berlin, Heidelberg: Springer Berlin Heidelberg, 2006, pp. 60–71. [Online]. Available: [https://doi.org/10.1007/11760146\\_6](https://doi.org/10.1007/11760146_6)
- [45] B. Shneiderman, “The eyes have it: A task by data type taxonomy for information visualizations,” in *Proceedings of the IEEE Symposium on Visual Languages*, ser. VL ’96, 1996, pp. 336–343. [Online]. Available: <https://doi.org/10.1109/VL.1996.545307>
- [46] K. Kucher, R. M. Martins, and A. Kerren, “Analysis of VINCI 2009–2017 proceedings,” in *Proceedings of the 11th International Symposium on Visual Information Communication and Interaction*, ser. VINCI ’18, 2018, pp. 97–101. [Online]. Available: <http://doi.acm.org/10.1145/3231622.3231641>
- [47] A. Constantin, S. Pettifer, and A. Voronkov, “PDFX: Fully-automated PDF-to-XML conversion of scientific literature,” in *Proceedings of the ACM Symposium on Document Engineering*, ser. DocEng ’13, 2013, pp. 177–180. [Online]. Available: <https://doi.org/10.1145/2494266.2494271>
- [48] M. W. Goodman, R. Georgi, and F. Xia, “PDF-to-Text reanalysis for linguistic data mining,” in *Proceedings of the Eleventh International Conference on Language Resources and Evaluation*, ser. LREC 2018, 2018.

- [49] L. H. Cohen, “Measurement of life events,” in *Life Events and Psychological Functioning: Theoretical and Methodological Issues*. SAGE Publications, 1988, pp. 11–30.
- [50] J. D. Evans, *Straightforward Statistics for the Behavioral Sciences*. Brooks/Cole Publishing, 1996.
- [51] L. Byron and M. Wattenberg, “Stacked graphs — geometry & aesthetics,” *IEEE Transactions on Visualization and Computer Graphics*, vol. 14, no. 6, pp. 1245–1252, Nov. 2008. [Online]. Available: <https://doi.org/10.1109/TVCG.2008.166>
- [52] A. J. Wecker, J. Lanir, O. Mokryn, E. Minkov, and T. Kuflik, “Semantize: Visualizing the sentiment of individual document,” in *Proceedings of the International Working Conference on Advanced Visual Interfaces*, ser. AVI '14, 2014, pp. 385–386. [Online]. Available: <http://doi.acm.org/10.1145/2598153.2600056>
- [53] B. Lee, N. H. Riche, A. K. Karlson, and S. Carpendale, “SparkClouds: Visualizing trends in tag clouds,” vol. 16, no. 6, Nov. 2010, pp. 1182–1189. [Online]. Available: <https://doi.org/10.1109/TVCG.2010.194>




# A Appendix 1

## A.1 List of Categories Used in the TextVis Browser Dataset




Given below is the list of categories used in TextVis Browser and correlation matrix, as well as the corresponding icons.

Data Domain	
	Online Social Media
	Communication
	Reviews / (Medical) Reports
	Literature/Poems
	Scientific Articles/Papers
	Editorial Media
	Patents










  

Data Source	
	Document
	Corpora
	Streams








  

Data Properties	
	Geospatial
	Time Series
	Networks








  

Analytic Tasks	
	Text Summarization / Topic Analysis
	Sentiment Analysis
	Stance Analysis
	Discourse Analysis
	Event Analysis
	Trend Analysis / Pattern Analysis
	Lexical / Syntactical Analysis
	Relation / Connection Analysis
	Translation / Text Alignment Analysis




  

Visualization Tasks	
	Region of Interest
	Clustering / Classification
	Comparison
	Overview
	Monitoring
	Navigation / Exploration
	Uncertainty Tackling

Visual Representation	
	Line Plot / River
	Pixel / Area / Matrix
	Node- Link
	Clouds / Galaxies
	Maps
	Text
	Glyph / Icon

Visual Alignment	
	Radial
	Linear / Parallel
	Metric- dependent

Visual Dimensionality	
2D	2D
3D	3D

Figure A.1: TextVis Browser categories with icons

## A.2 List of Key Terms Used for Category Matching

Given below is the complete list of category key terms used for matching the publication contents against the source dataset categories. The list was provided by the authors of TextVis Browser as domain experts in text visualization.

```
[
  {
    "category": "text-summarization",
    "terms": ["summarization", "summary", "topic", "theme", "named",
              "entity", "entities"]
  },
  {
    "category": "discourse-analysis",
    "terms": ["discourse"]
  },
  {
    "category": "stance-analysis",
    "terms": ["stance", "intersubjectivity", "evaluative", "judgemental",
              "appraisal"]
  },
  {
    "category": "sentiment-analysis",
```



```

    "terms": ["sentiment", "emotion", "opinion", "affect", "valence",
              "arousal", "subjectivity", "subjective", "polarity"]
  },
  {
    "category": "event-analysis",
    "terms": ["event"]
  },
  {
    "category": "trend-analysis",
    "terms": ["trend", "pattern", "forecast", "regression"]
  },
  {
    "category": "lexical-analysis",
    "terms": ["lexic", "syntactic", "syntax", "lemma"]
  },
  {
    "category": "relation-analysis",
    "terms": ["relation", "connection", "connectivity"]
  },
  {
    "category": "translation-analysis",
    "terms": ["translation", "alignment"]
  },
  {
    "category": "roi",
    "terms": ["interest", "region", "peak", "outlier", "anomaly",
              "anomalous"]
  },
  {
    "category": "classification",
    "terms": ["classification", "classify", "classified", "cluster",
              "categorization", "categorize", "categorise", "regression",
              "grouping"]
  },
  {
    "category": "comparison",
    "terms": ["comparison", "compare"]
  },
  {
    "category": "overview",
    "terms": ["overview", "summary"]
  },
  {
    "category": "monitoring",
    "terms": ["monitor"]
  },
  {
    "category": "navigation",
    "terms": ["navigation", "navigate", "exploration", "explore",
              "guidance", "guide", "provenance"]
  },
  {
    "category": "uncertainty",
    "terms": ["uncertain"]
  },
  {
    "category": "document",
    "terms": ["document", "individual"]
  },
  {
    "category": "corpora",
    "terms": ["corpora", "corpus", "collection"]
  },
  {
    "category": "streams",
    "terms": ["stream"]
  },
  {
    "category": "geospatial",
    "terms": ["spatial", "geo"]
  },
  {
    "category": "time-series",
    "terms": ["time", "temporal", "dynamic"]
  },
  {
    "category": "networks",
    "terms": ["network", "graph", "connection"]
  },
  },

```

```

{
  "category": "social-media",
  "terms": ["social", "media", "forum", "blog", "comment"]
},
{
  "category": "communication",
  "terms": ["communication", "email", "messenger"]
},
{
  "category": "patents",
  "terms": ["patent"]
},
{
  "category": "reviews",
  "terms": ["review", "customer", "report"]
},
{
  "category": "literature",
  "terms": ["literature", "poem", "story", "stories", "book", "fiction"]
},
{
  "category": "papers",
  "terms": ["article", "publication", "manuscript"]
},
{
  "category": "editorial-media",
  "terms": ["news", "wikipedia", "editorial"]
},
{
  "category": "2d",
  "terms": ["2d", "two-dimensional"]
},
{
  "category": "3d",
  "terms": ["3d", "three-dimensional"]
},
{
  "category": "line-plot",
  "terms": ["plot", "chart", "river", "streamgraph"]
},
{
  "category": "pixel-area",
  "terms": ["pixel", "area", "matrix", "bar", "pie", "donut", "treemap"]
},
{
  "category": "node-link",
  "terms": ["node", "vertex", "link", "edge"]
},
{
  "category": "clouds",
  "terms": ["cloud", "galaxy", "galaxies"]
},
{
  "category": "maps",
  "terms": ["map", "cartogram"]
},
{
  "category": "text",
  "terms": ["font", "label"]
},
{
  "category": "glyph",
  "terms": ["glyph", "icon"]
},
{
  "category": "radial",
  "terms": ["radial"]
},
{
  "category": "linear",
  "terms": ["linear"]
},
{
  "category": "metric",
  "terms": ["metric"]
}
]

```

---

Listing A.1: Complete list of category terms

### A.3 Yearly Statistics for Category Labels

The figure below shows how many category labels are assigned in the TextVis Browser dataset in total each year.

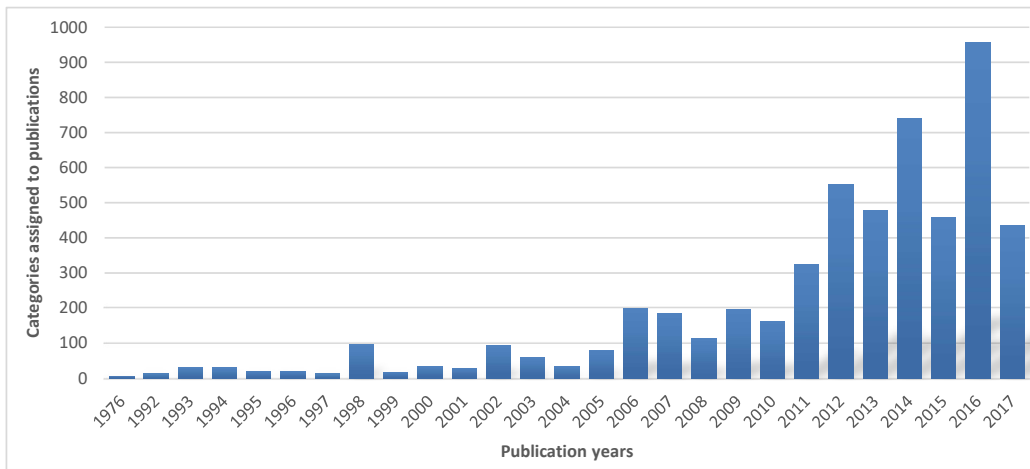


Figure A.2: Number of category labels assigned in the TextVis Browser dataset in total each year

### A.4 Overall Category Statistics

The figure below shows that how many times each category is assigned in the TextVis Browser dataset.

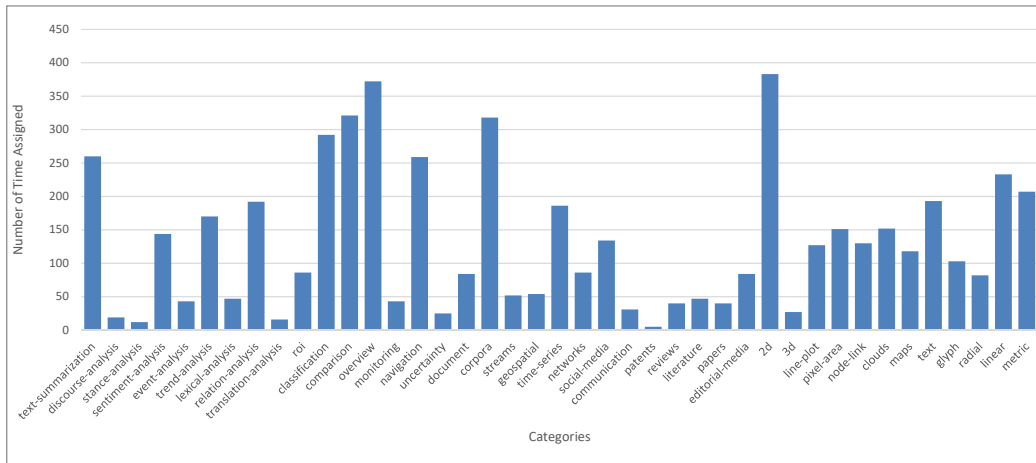


Figure A.3: Number of times each category is assigned in the TextVis Browser dataset

Figure A.3 shows how many times each category is assigned to publications in the TextVis Browser dataset. We can see that few categories are assigned several times as compared to others. The '2D' category is mostly assigned in the TextVis Browser dataset: it is assigned to 383 publications out of 400.