Fall 12-14-2011

Syntactic and Semantic Analysis and Visualization of Unstructured English Texts

Saurav Karmakar
Georgia State University

SYNTACTIC AND SEMANTIC ANALYSIS AND VISUALIZATION OF UNSTRUCTURED ENGLISH TEXTS

by

SAURAV KARMAKAR

Under the Direction of YING ZHU

ABSTRACT

People have complex thoughts, and they often express their thoughts with complex sentences using natural languages. This complexity may facilitate efficient communications among the audience with the same knowledge base. But on the other hand, for a different or new audience this composition becomes cumbersome to understand and analyze. Analysis of such compositions using syntactic or semantic measures is a challenging job and defines the base step for natural language processing.

In this dissertation I explore and propose a number of new techniques to analyze and visualize the syntactic and semantic patterns of unstructured English texts.

The syntactic analysis is done through a proposed visualization technique which categorizes and compares different English compositions based on their different reading complexity metrics. For the semantic analysis I use Latent Semantic Analysis (LSA) to analyze the hidden patterns in complex compositions. I have used this technique to analyze comments from a social visualization web site for detecting

the irrelevant ones (e.g., spam). The patterns of collaborations are also studied through statistical analysis.

Word sense disambiguation is used to figure out the correct sense of a word in a sentence or composition. Using textual similarity measure, based on the different word similarity measures and word sense disambiguation on collaborative text snippets from social collaborative environment, reveals a direction to untie the knots of complex hidden patterns of collaboration.

INDEX WORDS: Readability, Complexity depth of field, Grammatical structure, Visualization, Chernoff faces, Web mining, Web information retrieval, Online social visualization, Recommendation, Composition style, Matrix, Latent Semantic Analysis, Online collaborative web site, Social media, Co-occurrence frequency, Pattern searching, Statistical analysis, Categorical data, Semantic similarity, Word sense disambiguation, Natural text, Natural Language, Social network.

SYNTACTIC AND SEMANTIC ANALYSIS AND VISUALIZATION OF UNSTRUCTURED ENGLISH TEXTS

by

SAURAV KARMAKAR

A Dissertation Submitted in Partial Fulfillment of the Requirements for the Degree of

Doctor of Philosophy

in the College of Arts and Sciences

Georgia State University

2011

SYNTACTIC AND SEMANTIC ANALYSIS AND VISUALIZATION OF UNSTRUCTURED ENGLISH TEXTS

by

SAURAV KARMAKAR

Committee Chair: Ying Zhu

Committee: Rajshekhar Sunderraman

G. Scott Owen

Gengsheng Qin

Electronic Version Approved:

Office of Graduate Studies

College of Arts and Sciences

Georgia State University

December 2011

DEDICATION

To my parents

To my sister

To my Love

ACKNOWLEDGEMENTS

First of all, I would like to thank my advisor, Dr. Ying Zhu, for being my guide, master, and best friend. This dissertation and related research works would not have been possible without his help. His support, guidance, and encouragement create an excellent research environment for my research and his invaluable ideas always lead me to explore new directions of research territory. We have numerous times of long discussions on different research topics which are turning into valuable treasure to me. I have been very glad to work with Dr. Ying Zhu.

Special thanks go to Dr. Rajshekhar Sunderraman. He has helped me so many times in many different ways throughout my Ph.D. study. Not only in research, but teaching, learning he was the one who always guided me to where I am standing today. Also, many thanks to Dr. G. Scott Owen and Dr. Gengsheng Qin for his significant suggestions to improve the research work in this dissertation. Their guidance and comments worked as invaluable and beyond this dissertation.

Thanks to the continuous supports from the Computer Science Department and Brain and Behavior Program at Georgia State University. In addition I would like to thank Kireet Kokala for collaborating in the latent semantic analysis work and Jaya Kalidindi for the web data retrieval task. Also thanks to the Office of International Affairs, where I found the significance of such research while working with textual agreements of Georgia State with other Universities.

I would like to also thank Dr. Kimberly Nelson King for teaching me the big ideas and always being there as an advisor of my student organization life. Thanks goes to Shaochieh for his technical support and help which allowed me to conduct my experiments with most ease.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

1 INTRODUCTION

People have complex thoughts and they often express their thoughts with complex sentences using natural languages. This complexity may facilitate efficient communication among the audience with the same knowledge base. But on the other hand, for different or new audiences this mismatch of compositional complexity becomes cumbersome to understand and analyze. The natural languages generally consist of set of mathematically ambiguous grammatical rules, which adds the most complication and challenge for analysis and understanding. Analysis of such compositions using syntactic or semantic measure is a challenging job and defines the base step for natural language processing [1].

Analysis of natural languages or text mining [2-5] is a challenging domain where computer scientists, linguistics and statisticians work together. The overall approach for finding solution to this problem is to use different statistical techniques, artificial intelligence methodologies and modeling linguistic patterns. The goal of this field is to find a generalized pattern for natural human expressions.

Computer and internet has become a part of our daily life for every other need. A huge amount of information is available in the internet in the form of natural language based compositions. Analyzing natural languages would be the greatest help in this regard. Therefore this field carries a great amount of potential, but it has lot to achieve. For example if we need any information on a topic we throw some keywords from that topic in the search engines. As a result we expect to get the most appropriately related information on that topic. Today's search engines are very efficient but still they only understand everything in terms of keywords based match. What happens if we want to search a topic comprising of few lines of text? Unfortunately today's search engine fails on such a task. But in future we expect to work through machine's understanding of natural languages. In that case we expect that we would be able to communicate with the machine more easily and naturally. Retrieval of adequate and proper in-

formation is the first step for any knowledge based automation, control, communication. Therefore, retrieval of information and automation depend on the analysis of natural language a lot.

Due to the nature and diversity of natural languages, numerous divergent techniques are employed to figure out proper syntax and semantics of such language composition. Readability is one of such important syntactic aspect of a natural language; whereas the meaning of the components of a textual composition carries a primary role in semantics. Here I am introducing some visualization methods to syntactically analyze English textual compositions. Based on one of such measure I am trying to introduce a content oriented recommendation technique as well. To reveal the semantic meaning of a document I am executing some mathematical and statistical technique. In addition to this I am employing word meaning based measure to find textual similarity of compositions for small textual snippets. For this purpose I chose to retrieve some online text from online social network services and employ some descriptive and analytical statistics on the retrieved data.

1.1 Problem Statement

Analyzing natural language is an arduous job. The natural trend of such analysis begins with analyzing the structure of such compositions. The normal question pops out as: does the component of such structure provide any clue for analysis? Can those structures be presented in a more formidable way so that people can have better perception? Can there be some guidelines to categorize such structures? These aforementioned nominal questions come up from the syntactic structure of a natural language based text.

Now to understand the underlying meaning of such compositions, what kind of rules or techniques can we use? Can we find a way to compare meaning of such structure and if even exact comparison turns out too complicated, can we at least have a way to express the comparison in a more perceptive way? Can we employ some well learned techniques from other fields in this domain? All these concerns now relate to the semantic structure of the compositions.

1.2 Motivation

Most of the time our thoughts get expressed using the natural language based communication, which consists of complex sentences. This leads to creation of tons of compositions and documents, which are not very simple in nature always. Nowadays with the increase of quintessential popularity of web, almost all information is on the web. Still most of the aforementioned information is in the form of unstructured text composed from some natural human language.

Now the natural languages by their own nature have ambiguous grammars. Natural languages sometime contain components, which expresses different meaning at different times. Also sometime multiple components point toward same meaning. With such level of diverse complexities, the field itself brings serious challenges.

We have huge amount of important and interesting information present online today and most of them are in the form of unstructured texts. Analysis of such compositions using syntactic or semantic measure is a challenging job and defines the base step for natural language processing. If somehow we can create an automatic analytic measure for the documents, we could make all these information connected in appropriate order. If we can generalize the automatic measure and analytic understanding of such texts, it will solve enormous amount of unsolved problems. As if we feed that generalized understanding to the machine, it would understand human sense of expressions and perform exactly the way we expect. Starting from the information searching, to generalize ideas to recommend certain stuff, every daily need of ours could be done at their best potential if we can create an appropriate natural language analysis tool. The motif behind such work is practically limitless. Also the expected outcomes of such work are incredibly important as well. Therefore analyzing natural language carries a great challenge with outstanding interest.

1.3 Significance

Analysis of natural languages is the base step for natural language processing [1]. It is a very important and challenging field in today's aspect, as it deals with a lot of real world problems. This analysis directly impacts the understanding, organization and categorization of text documents or unstructured texts available on the web. Also this leads to locating areas of a composition which may need improvement. Following are the few of many topics where natural language analysis has significant impact.

- *Categorize Text*: Analyze and categorize the text as per its complexity level. It helps providing a grade level to the text which helps user to choose texts as per their level of comfort.
- *Information Retrieval (IR)*: As most of the information in web is in the form of unstructured text, information retrieval heavily depends on natural language analysis. It deals with searching and retrieving appropriate information.
- *Pattern Recognition*: Analyzing short text snippets in a social collaboration environment could reveal collaboration pattern. Using semantic understanding of texts spam/unwanted components could be found. This also can be used to measure web security in terms of analyzing its content.
- *Recommendation*: Content based recommendation works through analyzing the contents of recommendation left by a user. After analysis similar syntactic or semantic structural recommendations are suggested for the same category users.
- *Information Extraction (IE)*: Basically this works through the extraction of semantic information from text. This covers tasks such as named entity recognition, relationship extraction, coreference resolution, etc.
- *Named Entity Recognition (NER)*: This technique finds out which items in a text map to proper names, such as people or places, and also annotates the type definition of each such name (e.g. person, location, organization).

- *Coreference resolution* is basically determining which words in a given texts refer to the same objects or entities.
- *Relationship extraction* is given a chunk of text, identifying the relationships among named entities (i.e. who is the wife of whom) mentioned in the text.
- *Automatic summarization*: This deals with generating a summary from a given text. Generally it helps in information retrieval a lot.
- *Natural Language Generation*: This works through the help of automatic summarization and information retrieval. Given a chunk of text, an automatic summary could be made and that summary text could be used to retrieve meaningfully related information from the web to create new text composition.
- *Question Answering*: This works through determining answer of question composed in natural human languages. This is a part of automatic summarization for the case of decision type answer (e.g. yes/no) and for more descriptive answers it works through Natural language generation.
- *Natural Language Understanding*: Analyzing chunk of texts semantically and then form that into more formal representations such as first-order logic structures that are easier for computer programs to manipulate.
- *Word sense disambiguation*: The words used in a natural language based compositions, often carry multiple meanings; Natural language analysis could lead to the selection the most appropriate meaning of a word in its context. Generally to resolve this kind of problem, the words and its associated multiple senses are given as a repository or through a taxonomy and the job is to choose the exact word and its appropriate sense pair by looking at the compositional context

- *Clause level/Phrase level/Part-of-speech Tagging*: Given a sentence, its clause level/Phrase level/Part-of-speech distributions could be found. This helps in compositional structure recognition and natural language understanding.

1.4 Organization

The remaining dissertation is organized as follows: Section 2 provides the literature review in syntactic and semantic complexity measure of Natural languages, specifically English. It also covers the background work of the natural language composition analysis and some related visualization techniques and topics in reference to this dissertation. Section 3 demonstrates work related to the syntactic analysis. Section 4 describes work related to the analysis of social visualization sites. Section 5 provides the details of semantic analysis work. Finally, Section 6 provides the conclusion and related future works.

2 LITERATURE REVIEW

Having internet as the backbone of everyday need, we have outstanding number of machine readable documents available. It is estimated that 80% of information lives in the form of text [6, 7]. The usual approach of logic-based programming [8] paradigm has guided the initial direction of textual understanding from such information. The fuzzy and often ambiguous relations in natural language limit the effectiveness of this approach.

What is text mining? Data mining applied to textual data. Text mining aims at unveiling the hidden information in the textual compositions. To achieve this, it has to deal with large number of words and structures in natural languages along with the vagueness, uncertainty and fuzziness. Text is unstructured, amorphous, and difficult to deal with but also the most common vehicle for formal exchange of information. Therefore, the motivation for trying to extract information from it is compelling, even if success is only partial.

Back in 1958 H. P. Luhn noted "the resolving power of significant words" [9] in primary text in his influential paper on automatic abstracting of textual compositions. In 1961 Lauren B. Doyle mentioned "natural characterization and organization of information can come from analysis of frequencies and distributions of words in libraries"[10] to focus initial direction towards text mining and related methods.

By the time this field was consolidating in the domain of textual content analysis, Swanson [11] had already put the idea into practice by developing a system to discover meaningful new knowledge in the biomedical literature. Swanson and Smalheiser developed a software, now called ARROWSMITH [12, 13], helps by finding common keywords and phrases in "complementary and noninteractive" sets of articles or literatures and juxtaposing representative citations likely to reveal interesting co-occurrences. According them two literatures are complementary if together they can reveal useful information not apparent in the two sets considered separately – e.g., one may reveal a natural relationship between A

and B, and the other a relationship between B and C, so that together they suggest a relationship between A and C. The articles are defined noninteractive, if their articles do not cross-cite and are not co-cited elsewhere in the literature. Swanson's system remains far from fully automated; it is highly medical domain-specific. But as this works out least partially, Swanson has been recognized as an early pioneer by self-described text mining practitioners. Lindsay et al. [14] and Kostoff et al. [15, 16] have extended Swanson's approach afterwards without calling it text mining.

IBM created a product named "Intelligent Miner for Text" [17, 18] in 1998 for natural language composition analysis. It consists of set of tools which can be seen as information extractors that enrich documents with information about their contents in the form of structured metadata.

Text mining or the natural language text analysis is the basic building block for the natural language processing. The history of natural language processing dates back to 1950s while Turing Test [19] was developed by Alan Turing as a criterion of intelligence. But thereafter the progress was quite slow in the following years. In 1960s quite notably successful NLP systems developed of that time, called SHRDLU [20], but it was very restricted in its range with restricted vocabularies. Also in the 70's many programmers worked and published “conceptual ontologies” [21, 22], which structured real-world information into computer-understandable data. The actual revolution in NLP started in the late 80s with the introduction of machine learning [23] algorithms for language processing and the steady increase in computational power resulting from Moore's Law [24]. Since then tons of work has been done in this domain and this topic is still under focus and need much more development. Statistical natural-language processing is a prime focus in this field which comprises all quantitative approaches to automated language processing, including probabilistic modeling, information theory, and linear algebra [25].

Modern father of linguistics Noam Chomsky brought the avalanche of cognitive revolution by his study of language and its structures through another approach. He focused in multidimensional topics

with multiple levels of significant depth and from there the computational languages got their first clean grammatical form [26].

The Penn Tree Bank is a great initiative to construct humongous corpora of English texts to launch enhanced research of natural language processing, speech recognition, integrated spoken language systems as well as theoretical linguistics [27]. These kinds of treebanks are generally used in corpus linguistics (the study of language as expressed in samples or "real world" text) for studying syntactic phenomena. They are also employed in computational linguistics for training or testing parsers. Of course there exist multiple different focuses in different treebanks, but in general there are two main groups. One, which depends on the phrase structure or language constituent parts annotation like the Penn Treebank [27] and the other one focuses finding the relation between words and its dependents, not its order through dependency structure e.g. Prague Dependency Treebank[28].

To define the syntax and semantics of a natural language Barwise et al. [29] explored the limitations of first order logic quantifiers for the natural language and established an interesting and important relationship between the syntax, semantics and logic in a natural language. Since then there were manifold divergent approaches to crack the natural language syntax and semantics.

Following are few of many approaches to crack natural language syntax. The syntax of complex compound words and those involving derivational and inflectional affixation has been studied by Selkirk. In that study the focus revolve around a syntactic standpoint that encompasses both the structure of words and the system of rules for generating that structure [30]. Steedman [31] focused on an elaborated study of Natural grammar to find appropriate compatibility with language syntactic structures. I start my journey of syntactic analysis of natural language through the readability. The following section elaborates more on that.

2.1 Syntax Analysis through Readability

Readability of a document is an indicator of its understandability to particular groups of readers. National literacy surveys have shown that an average adult in the United States reads at the 8th grade level, and a college graduate at the 10th-grade level [32]. Jeanne Chall and Edgar Dale stated readability as "The sum total (including all the interactions) of all those elements within a given piece of printed material that affect the success that a group of readers have with it. The success is the degree to which they understand it, read it at an optimal speed, and find it interesting" [33]. On one hand, higher text readability improve readership, comprehension, memorization, reading speed, and reading persistence [34]. On the other hand, a mismatch of document's readability and reader's reading level can result in disinterest, misunderstanding, and even deception [35]. Long back in 1889 in Russia, revolutionary writer Nikolai A. Rubakin [49] concentrated on a study of over 10,000 texts written by soldiers, craftsmen, and farmers. There he found that the main obstacles to readability were firstly unfamiliar vocabulary and secondly the use of too many long sentences [48].

To quantify the readability, researchers have proposed many readability indexes that classify a document into a specific grade level [36]. A typical readability index is a single average number or classification for the entire document. These indexes are calculated from two categories of readability metrics: word complexity and sentence complexity. Although their simplicity can be beneficial in many cases -- such as quick classification -- the readability indexes are often too simple, formulaic, and abstract for in-depth analysis. For example, a multiple authored document may contain a section that is particularly difficult to read, but this inconsistency is often not reflected in the overall readability index. Two documents may have similar readability indexes but very different distribution of complex words and sentences. There are other perspectives of limitations to these formulas as well [38]. There are many readability indexes. The most well known ones include the Flesch-Kincaid Reading Ease [38], Flesch-Kincaid Grade Level, Gunning Fog Score [42], SMOG Index [37], Coleman Liau Index [43], Automated

Readability Index [37], Dale-Chall readability formula [44], etc. They have been used extensively to help evaluate and develop textbooks, medical literature, business publications, military documents, web site contents, etc. [36]. For example, both Microsoft Word and Google Docs can calculate the Flesch-Kincaid Reading Ease [38] score for a document. The state of Florida requires that life insurance policies have a Flesch-Kincaid Reading Ease [38] index of 45 or higher. Flesch-Kincaid Reading Ease formula is also a standard used by many U.S. government agencies for evaluating technical documents.

Although there are a variety of readability formulas, they are largely based on two metrics -- the complexity of sentences and the complexity of words. In those methods, the complexity of sentences is typically measured by the average words per sentence, while the complexity of words is measured in slightly different ways. For example, Flesch-Kincaid Reading Ease [38] test and Flesch-Kincaid Grade Level test use the average number of syllables per word, while the Coleman-Liau Index [43] and the Automated Readability Index [44] uses the average number of characters per word. Gunning Fog Score [42] and SMOG index [37] use the percentage of polysyllabels (complex words, or words with more than three syllables), while the Dale-Chall Readability Formula [45] uses the percentage of difficult words that are not on a 3,000 familiar word list.

In general, the different readability indexes correlates well at the document level, but not so at the paragraph level. There is a need for good tools that help writers or readers to analyze the complexity of writing at paragraph level. My work consists of visualizing the fundamental readability metrics -- word complexity and sentence complexity -- so that a writer or reader can visually analyze the complexity of each paragraph.

The proposed visualization techniques [50] supplement the conventional readability indexes. But more importantly, my visualization techniques address a major limitation of the readability indexes. While traditional readability indexes try to quantify text complexity, my method is an attempt to visualize text complexity that is hard to quantify.

There are some related works [46, 47] on text visualization, but my project differs from them in terms of visualization techniques, focus, and application areas. For example, Keim and Oelke developed a literature fingerprinting method [46] for visual literary analysis. In their method, a text document is divided into blocks, and various literary analysis variables for each block are color coded into a small square. The focus of their work is on author attribution and literature forensic analysis. On the other hand, my visualization technique attempts to preserve the form of the original text and make the syntactical complexity easily visible. In other words, Keim and Oelke's method provide a "zoomed-out" view of a document, while my method provides a "zoomed-in" view of a document that allows for more detailed analysis.

DocuBurst [47] provides a visual summary of the semantic content of text documents. Using a radial, spacing-filling layout of hyponymy, this visualization does not preserve the form of the original text. This technique is designed for literature forensic analysis, document categorization, and authorship attribution. Unlike DocuBurst [47], my method [50] here focuses on the syntactical complexity of the document (rather than the semantic content).

Also there are two main issues with the traditional readability indexes mentioned earlier. First, as seen from the above examples, a typical readability index is a single average number or classification for the entire document. It does not describe the readability variation at the paragraph level. In addition, the various readability indexes do not correlate well at the paragraph level. Therefore instead of relying on a specific readability index, it is better to calculate multiple readability indexes at the paragraph level and display them side by side. This is what motivates my study.

My work [51] is also inspired in part by the Web of Trust (WOT) [52], which is a community-based website reputation rating tool that uses a traffic-light style color coding to visualize a web site's reputation. My visualization technique employs similar color coding based ring shapes to mark the readability score category. Another inspiration is Herman Chernoff [53], who developed a novel method to

represent multivariate data using cartoon of a face whose features such as size and shape of the eye, curvature of the mouth would correspond to different variables. Since human cognition is very responsive to facial expression, using faces to visualize multivariate data is very efficient and can lead to quick identification of outliers.

Another motivation for my research [51] is that readability indexes are not integrated with the visual presentation of text. For example, Microsoft Word and Google Docs can calculate the Flesch-Kincaid Reading Ease [38] score for a document, but it only returns a single number for a document but my method provides visual parameters for each paragraphs of the text.

2.1.1 Readability Metrics Based Recommendation

Recommender systems attempt to help users by lowering the information overload and selecting a subset of items from a universal set. Examples of such systems include top-N lists [54], book [56] and movie [57] recommenders, and intelligent avatars [58].

The two main recommendation modeling approaches are content-based filtering [56] and collaborative filtering [55]. Collaborative filtering recommends items to users based on the user's previous choices. On the other hand, content-based filtering recommends items based on the information content present regarding the item. Balabanovic et al. [59] took a hybrid approach to design a content-based collaborative system Fab that incorporates multiple topics of interest in storage and then clusters the information to generate recommendations.

Green et al. [60] developed a content-based recommendation technique by collecting text descriptions and using this textual aura to compute the similarity between items. Semeraro et al. [61] infused knowledge into words by associating knowledge sources and reasoning model. Also there are some works on the semantic knowledge based on the contents for recommendation. For example Tsatsou et al. [62] developed a recommendation system that combines ontological knowledge with content extracted linguistic information, derived from pre-trained lexical graphs, in order to produce high quali-

ty, personalized recommendations. Yu et al. [63] proposed a technique for e-learners which takes knowledge about the learner (user context), knowledge about content, and knowledge about the domain being learned into consideration. Ontology is also utilized to model such knowledge for recommendation.

My approach [68] is different from these earlier approaches in that I focus on recommendation by compositional structure. I want to find documents that not only contain the user specified keywords, but also possess user preferred composition style. This idea is inspired in part by the shape based 3D model search methods proposed by Funkhouser et al. [64], where a new query interface is generated which integrates text, 2D sketches, 3D sketches, and 3D models for searching the 3D models. In this work they also have developed a new matching algorithm that uses spherical harmonics to compute discriminating similarity measures without requiring repair of model degeneracies or alignment of orientations for the shape-based queries.

A number of methods have been proposed to compare the structural similarity of text based documents. For example, resemblance and containment of the documents are considered by Broder et al. [65], Latent Semantic Analysis based indexing method is employed by Deerwester et al. [66], and Fourier transform based similarity measures between XML documents are introduced by Flesca et al. [67]. Here I take a different approach by examining the clause level syntactical structures of texts which is done employing the Stanford Natural Language parser [39, 40] which parse sentences and return its grammatical structure.

2.2 Semantic Analysis

This section is concerned with the semantics of natural languages. Semantics is defined as the study of meaning expressed by elements of a language or combinations thereof. As like syntactic, there are numerous different approaches in semantic exploration of natural language.

Liddy et al. provided a natural language processing system [69] through semantic vector representation of the text where the text is summarized through it's subject code look up and psycholinguistic

approach based word sense matching. Resnik presented a measure [70] of semantic similarity in an IS-A taxonomy based on the notion of natural language based shared information content. Lytinen presents an approach [71] to natural language processing where the syntactic and semantic processing take place at the same time. There are better working systems developed in this domain with restricted exploration. For example, a general biomedical domain-oriented NLP engine, MedScan [72] is developed that efficiently processes sentences from MEDLINE (National Library of Medicine's premier bibliographic database) abstracts and produces a set of regularized logical structures representing the meaning of each sentence. By combining a lexical taxonomy structure with corpus statistical information Jiang et. el. presents a new approach for measuring semantic similarity/distance between words [73] and concepts. A natural language interface system [74] is created based on trained statistical model by Miller et al. which goes through basically three stages of processing: parsing, semantic interpretation and discourse. Over the time the deployment of statistical techniques in the domain of semantic understanding of natural languages found heavily growing and influencing.

2.2.1 Latent Semantic Analysis

LSA is a method for extracting and representing the contextual meaning of words through statistical computations over a large text corpus [66, 75, 76, 79]. It has been applied to fields such as psychology, sociology, data mining, etc. [66, 75-83]. It was first applied to Information Retrieval (IR) and was known as Latent Semantic Indexing in the late 1980s. Later, it was used to deal with the synonym and polysemy problems in IR [66]. LSA starts by using an algebraic method called Singular Value Decomposition (SVD) to condense the large input data into smaller and manageable rectangular matrices of words, grouped by logical passages. Each cell of the matrix contains a transform of the frequency of the given word in the passage. Next, the matrix is decomposed so that each passage is represented as a vector whose value is the sum of all vectors representing its component words.

The words-to-words, passages-to-words, and passages-to-passages similarities are computed as cosines, dot products, etc. [75, 81]. LSA has been well correlated with human studies with regards to association or semantic similarity [76]. One of the benefits of LSA is that the similarity estimates are not mere frequency counts or correlations based on word usage. Instead, the results reflect the semantic meaning of the text under the mathematical analysis.

A number of LSA variants have emerged since then [78, 80, 83]. Wang et al. [78] proposed a M-LSA technique that was used in establishing multiple co-occurrence relationships between different types of objects. The problem was that multiple co-occurrence relations need to be represented by multiple co-occurrence matrices. The researchers constructed an undirected graph G(V,E) to show this relationship. Specifically, the goal was to find the latent semantic representations for each type of object. And, based on the co-occurrence data of G(V, E), they identified the most significant concepts based on the mutual reinforcement principle. Finally, each object is represented in a unified low-dimensional space. The results of their experimentation show that the M-LSA variant outperformed standard LSA results and was applicable to collaborative filtering, text clustering, and text categorization.

In another study, Pino and Eskenazi [83] demonstrated the use of LSA in word sense discrimination for words with related and unrelated meanings within a tutor application of English vocabulary learning for non-native speakers. An indexed database containing manually annotated documents was used. LSA performance for words with related meanings and for words with unrelated meanings was investigated. Lastly, they examined if reducing the document to a selected context of the target word improved performance. Their method overcame the sparseness of short contexts such as questions and resulted in an improvement over the exact match baseline.

Comparing with the above two methods, my method [84] is more general and closer to the traditional LSA method. The target of my study is Many Eyes [85], a social visualization web site and also an IBM research project which allows users to upload their data, construct data visualizations, share the

data visualization with others, and comment on data visualizations. I performed LSA by constructing a similar database as in Pino and Eskenazi [83]. However, my input data was an amalgamation of cross domain topics, which, in the beginning phases did not readily result in any trends, compared to the predetermined dataset in [83]. In addition, my method [84] differs from previous work [78, 83] in the absence of a rank-lowering algorithm because my dataset is relatively small and not noisy. I also introduced an optimization of the term frequency (based in part on Landauer [78]) over the traditional LSA. By incorporating an initial co-occurrence matrix and TF-IDF solution, my LSA method is leaner and more efficient for small scale data sets when compared with M-LSA [77].

2.2.2 Semantic Analysis in a Social Collaborative Environment

In recent years, online social data visualization has emerged as a new platform for users to construct, share, and comment on data visualizations online. This emerging technology can be seen as the extension of Cloud Computing and Web 2.0 technologies to the field of data visualization. A typical online data visualization tool allows users to upload their data to a server, construct data visualizations online, and publish or share the data visualizations. Users can view and manipulate the data visualizations online and write comments.

The online data visualization shares the same advantages of cloud computing. The users do not have to install any special software on their own computer. They create data visualizations in their web browsers. The data and data visualizations are stored at the hosting company's server, and are accessible online from anywhere. In addition, the data and data visualizations can be easily shared with other people, and people can leave comments on the visualizations in the same way as blogs.

The first notable online social data visualization tools are Swivel [86] and ManyEyes [85, 87]. Swivel is a commercial product, with a private collection (paid access) and public collection (free access) of data visualizations. On the other hand, Many Eyes [85] is an IBM research project and all the data visualizations are publicly accessible. Both Swivel and Many Eyes are online since 2007.

More recently, a number of new online visualization tools have emerged. For example, Tableau Software announced the Tableau Public [88] in April 2010 – a free version of the Tableau visualization tool that allows users to publish and share their visualizations online. Google announced its Public Data Explorer [89] in March 2010, which also supports online data visualization construction and sharing. Microsoft's Pivot [90], launched in November 2009, is another new addition to the online social visualization tools.

With these developments, online social data visualization has been quickly gaining attractions. Therefore it is important to understand its impact on how people construct, view, and discuss visualizations. This is the motivation for my research.

I report [91] my preliminary analysis of Many Eyes [85], based on over 7,000 data visualizations and over 30,000 user generated comments from 2007 to 2010. I choose Many Eyes for my study over other online visualization tools for several reasons. First, Many Eyes have a longer history than many newer tools, such as Tableau Public, Google Public Data Explorer, and Pivot. As a result Many Eyes has a larger collection of data visualizations. Second, all the data visualizations on Many Eyes are publicly accessible. Therefore we can see the complete picture of user generated data visualizations on that web site. Third, Many Eyes has a large number of users who add many new data visualizations every day.

I believe Many Eyes is the largest experiment on user generated data visualizations. An analysis of the patterns and trends of these data visualizations can give us unprecedented insights into how people construct visualizations, what types of visualization are most associated with different subject areas, what types of visualizations receive the most interest (i.e. comments), etc. Such insights can help improve the design of data visualizations and visualization tools. The work reported here [91] is the first attempt and first step toward that goal. I am working further on this topic with advanced statistical techniques to establish and find more profound pattern of collaboration.

Collaboration is in overall an intense form of interaction, facilitates effective communication as well as the sharing of competence and other resources [105]. Due to the complicated and fuzzy nature of human natural behavior and expression stays active in collaboration, it becomes tremendously critical to track down such in quantitative domain. Despite all these, science indicators has provided additional quantitative information of a more direct and objective nature of geographical patterns in lieu of cooperation among scientific institutions [106]. In literature many studies tried to come up with the comparative measures of collaboration in two fields (or subfields) or to show the trend towards multiple authorships in a discipline. In this scenario as primitive approaches the mean number of authors per paper has been indicated and termed as Collaborative Index (CI) [107] and the proportion of multiple-authored papers has been called Degree of Collaboration (DC) [108]. Ajiferuke et al. [109] has shown the inadequacy of the above and defined a normalized 0-1 scale Collaboration coefficient (CC) (0 corresponds to single authors) [109] which comprise of the merits of both CI and DC. With the advent of internet online collaboration came in picture and nowadays it's the biggest hype. Hathorn et al [110] discussed and pointed out factors of participation, interaction, and interdependence in respect of collaboration with elaboration. Holding the hand of online collaboration the craze of social networking followed. The first one in the field is Sixdegrees [111], came out in 1997 and since then quite many appeared and today it's the age of facebook [112], twitter [113] and linkedin [114] etc.

Social data visualization platforms have taken a recent trend of interest amongst users for constructing, sharing, and commenting on data visualizations online. This aspiring direction of technology could be considered as the data visualization oriented advancement of Cloud Computing and Web 2.0 technologies. The first few prominent online tools in this domain are Swivel [11] and Many Eyes [12] [13]. Swivel is a commercial product, geared for both private collection (paid access) and public collection (free access) of data visualizations. On the other hand, Many Eyes [12, 13] is an IBM research project on the social data visualization, allows all the visualizations publicly available. Both Swivel and

Many Eyes came to the online domain since 2007. More recently, numerous new online visualization tools have came up. For example, Tableau Software started the Tableau Public [14] in April 2010 – a free version of the Tableau visualization tool that allows users to publish and share their visualizations online. Google came up with its Public Data Explorer [15] in March 2010, which also supports online construction and sharing of data visualizations. Microsoft announced Pivot [16], in November 2009, is another new addition in the domain. Many Eyes already received attention on their user behavior analytics [91].

In this online social collaboration domain users express the collaboration through expressions composed in natural languages, so natural language processing based analysis becomes important. One of the key approaches in natural language processing and related areas is textual similarity measure. Vectorial model in information retrieval might be one of the earliest applications in this domain, where the document most relevant to an input query is determined by ranking documents in a collection in reversed order of their similarity to the given query [115]. The application of textual similarity measure has been used in many direction: relevance feedback and text classification [116], word sense disambiguation [99], and more recently for extractive summarization [117], and methods for automatic evaluation of machine translation [118] or text summarization [119]. The stereotypical approach in textual similarity has been the lexical matching, while improvements came up on top as stemming, stop-word removal, part-of-speech tagging, longest subsequence matching, as well as various weighting and normalization factors [120]: which didn't provide appropriate success. Another outstanding approach is LSA (Latent Semantic Analysis) [74] in this domain, aims to find similar terms in large text collections, and measure similarity between texts by including these additional related words. However due its computational implementation cost complexity and also the "black-box" effect that does not allow for any deep insights into why some terms are selected as similar during the singular value decomposition process it hasn't been used in large scale.

Here I employ the word-to-word similarity metrics and word specificity based the textual similarity measure [121] on the textual snippets collected from online social visualization site to analyze and visualize the pattern of collaboration.

3 VISUALIZING THE SYNTACTIC ASPECT OF UNSTRUCTURED ENGLISH TEXTS

The syntactic aspect of unstructured English text is explored in terms of readability of the text. As readability caries the notion of how easy a document, it carries a good point in the analysis. Also as the readability metrics are comprised of different syntactic elements of the language.

3.1 Analyzing and visualizing Text Readability

Readability of a document is an indicator of its understandability to particular groups of readers. National literacy surveys have shown that an average adult in the United States reads at the 8th grade level, and a college graduate at the 10th-grade level [32]. To quantify the readability, researchers have proposed many readability indexes that classify a document into a specific grade level. On one hand, higher text readability improve readership, comprehension, memorization, reading speed, and reading persistence. On the other hand, a mismatch of document's readability and reader's reading level can result in disinterest, misunderstanding, and even deception.

A typical readability index is a single average number or classification for the entire document. These indexes are calculated from two categories of readability metrics: word complexity and sentence complexity. Although their simplicity can be beneficial in many cases -- such as quick classification -- the readability indexes are often too simple, formulaic, and abstract for in-depth analysis. For example, a multiple authored document may contain a section that is particularly difficult to read, but this inconsistency is often not reflected in the overall readability index. Two documents may have similar readability indexes but very different distribution of complex words and sentences. To address this issue, I propose a method for visualizing readability metrics.

My visualization method [50] highlights complex words and visualizes sentence complexity. The complex words are visualized with color coding. The sentence complexity is visualized as a stacked bar chart. Each sentence is represented by a single bar that may be divided into several sub-sections. Each

sub-section represents a sub-clause in the sentence. The gaps between the sub-sections encode the syntactical complexity of the sentence. This visualization preserves the familiar text document format and allows users to quickly compare word complexity; sentence lengthens, and sentence syntactical complexity.

My visualization [50] enhances the traditional formulaic readability calculation by enabling users to perform more sophisticated and nuanced readability analysis based on visual patterns. With the readability visualization, both readers and writers can visually identify the paragraphs or sentences that are difficult to read, or compare the readability of multiple documents or multiple versions of the same document. For example, the visualization may help readers visually identify different writing styles within a multi-author document. A writer can use it to assess the readability at the paragraph level, checking how various readability metrics are distributed across the document so as to identify specific areas for revision.

3.1.1 Methods

A. Measuring Word Complexity

There are three ways to measure word complexity: number of characters per word, number of syllables per word, and vocabulary based method. I adopt all three in my visualization so that users can choose different options and compare the results.

It is straightforward to count the number of characters per word, but counting the number of syllables is a little more complicated [41]. The number of syllables is the number of vowels (a, e, i, o, u) heard in a word. I first count the number of vowels (a, e, i, o and u), and then subtract the number of silent vowels (e.g. the silent 'e' at the end of a word) and diphotongs (e.g. oi, oy, ou, ow, au, aw, oo, etc.).

Dale and Chall [45] developed a vocabulary based approach for measuring word complexity. They have constructed a 3,000 familiar word list, and any word not on this list is considered a difficult one.

However, the 3,000 word list only contains the base form of words, and the complete word list is actually much larger. For example, the words “easy”, “easier”, and “easiest” are all considered familiar words, but only 'easy' is on the 3,000 word list. I use Mathematica's WordData functions to find each word and its morphological derivatives. In the end, the complete familiar word list contains 23,574 words.

B. Visualizing Word Complexity

The word complexity is visualized with color coding. By default, simple or familiar words are displayed in darker color, while complex words are displayed in lighter color. However, the color mapping can be adjusted by users for different purposes. In my program, users can use a slide bar to adjust and even reverse the color mapping. For example, an ESL student who is learning English may want to reverse the color mapping so that the complex words are highlighted to help her expand her vocabulary. A writer may also want to highlight all the complex words for possible revision. Figure 1 and 2 illustrate this capability.

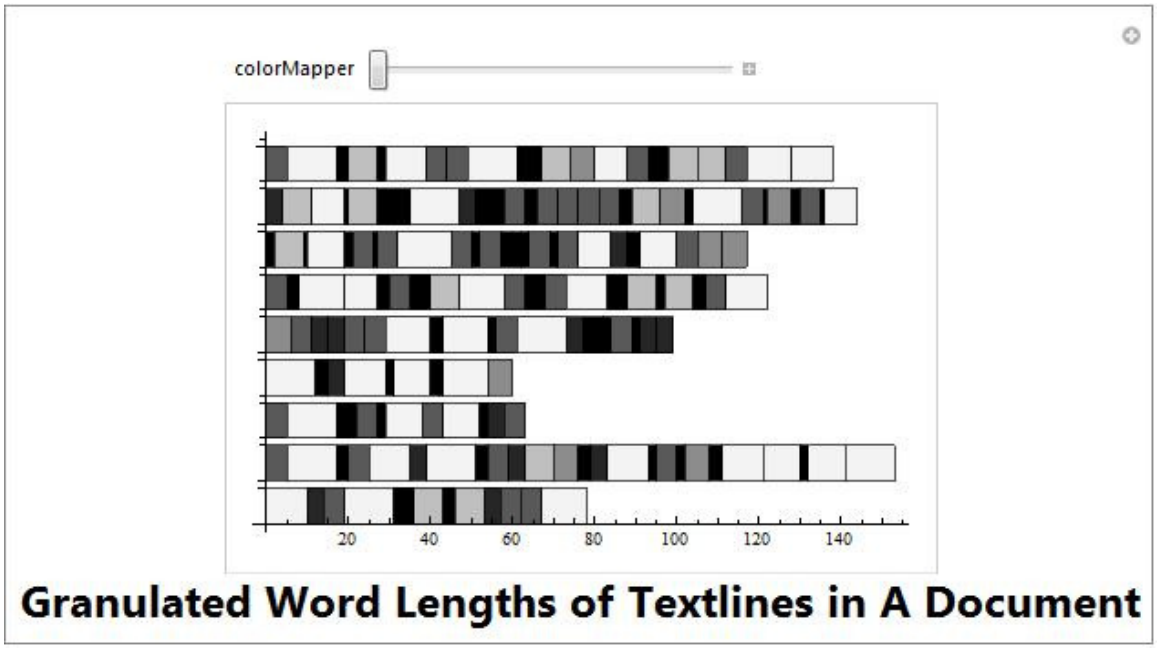


Figure 1. This is a visualization of word complexity. Each horizontal bar represents a sentence, and each section on the bar represents a word. Six different shades of gray color visualize word complexity in terms of word length. The shorter the word, the darker the color is.

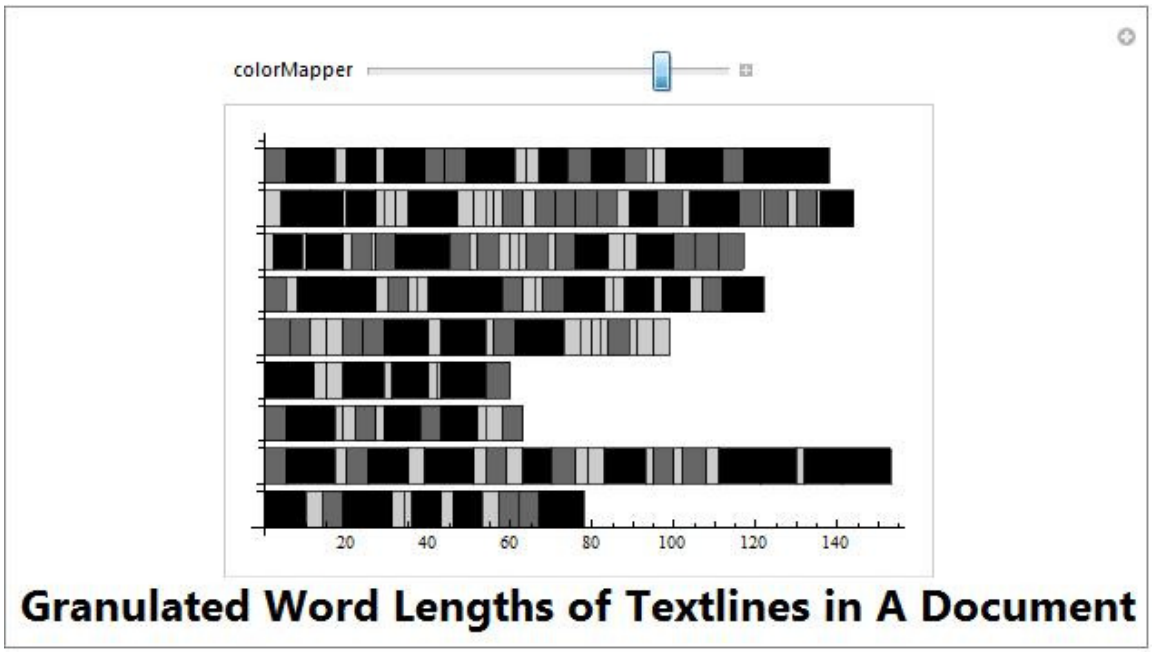


Figure 2. This is the same visualization as above, but the color mapping is reversed with the slider. Now the complex words are visualized in darker color.

C. Measuring Sentence Complexity

The most common metric for sentence complexity is the number of words per sentence. In addition to this, I also want to visualize the structural complexity of the sentences.

The syntactic structure of a sentence is generally divided into three levels: clauses, phrases, and words. Here I focus on the clause level. (In the future, extending this method to the phrase level would be interesting) A complex sentence typically contains two or more clauses – often connected by conjunctions -- and each clause may contain sub-clauses, and so on. My idea is to visualize the complexity of a sentence by visually marking the division of clauses in that sentence. But first, I need a tool that can automatically parse the sentences and return its grammatical structure. The tool I use is the Stanford Natural Language Parser (SNLP) [39, 40].

The Stanford Natural Language Parser [39, 40] is a statistical parser that can group words into clauses and phrases and classify words into different types, such as subject or object. For example, the result of parsing the following sentence is shown below.

"The people who live in that small yellow house with the picket fence across the street from me raise emus."

```
(ROOT
  (S
    (NP
      (NP (DT The) (NNS people))
      (SBAR
        (WHNP (WP who))
        (S
          (VP (VBP live)
            (PP (IN in)
              (NP
                (NP (DT that) (JJ small) (JJ yellow) (NN house))
                (PP (IN with)
                  (NP (DT the) (NN picket) (NN fence)))))
            (PP (IN across)
              (NP
                (NP (DT the) (NN street))
                (PP (IN from)
                  (NP (PRP me)))))))))
    (VP (VBP raise)
      (NP (NNS emus)))
    (..)))
```

Figure 3 shows a parse tree generated by SNLP.

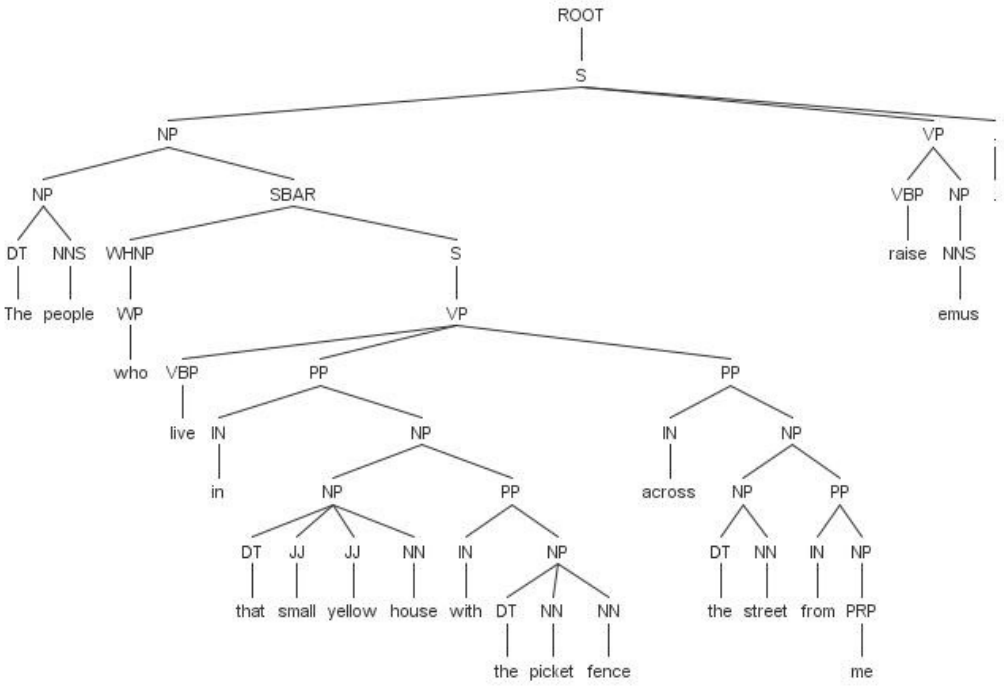


Figure 3. Parse tree generated by the SNLP parser.

My program reads in the SNLP output and then further parses it to identify the clause divisions. This information is then used to guide the visualization of sentence complexity.

D. Visualizing Sentence Complexity

The length of the sentence is visualized in the form of bar charts. Each sentence is represented by a horizontal bar that is scaled to occupy only one line. Therefore the readers can pre-attentively compare the sentence length. The original text can be displayed side by side with the visualization. When the user clicks on a bar, the corresponding word is highlighted in the original text (figure 9).

The visualization is based on parsing the output of SNLP, as described in the previous section. I visualize each sub-clause as a gray bar. The length of each gray bar is either character count or the word count of that clause. The gray bars are separated by white gaps. The length of a white gap depends on the depth of this clause in the SNLP parse tree. The shorter the white gap, the deeper the clause is placed in the parse tree. Therefore readers can quickly identify not only the number of clauses, but also the

depth of the clause division. The latter is also an indicator of the sentence complexity. In other words, the more divisions on a sentence bar, the more complex it is. The more variant the white gaps in a sentence bar, the more complex it is.

For example the sentence “While all restriction policies are based on the uniform guidelines, there may be minor variations in the details to account for local conditions” is visualized below.

Figure 4. This is a visualization of sentence structural complexity.

The gray sections represent sub-clauses, whose length is defined by the number of characters. The white gaps are indicators of the depth the sub-clause appears in the parse tree. The smaller the white gap, the lower the sub-clause is placed on the parse tree. This sentence is divided into three clause levels: the first sub-clause “While” appears at the second level of the tree, while the other two sub-clauses “all restriction policies are based on the uniform guidelines” and “there may be minor variations in the details to account for local conditions” appears at the third and fifth level of the parse tree, respectively. The above visualization clearly depicts the depth of division.

3.1.2 Implementation and Results

In this section I present examples of my readability visualization techniques (see figures 5 to 9). All of the visualizations were created in Mathematica 7.0, which provides a rich set of functions for visualization and text processing.

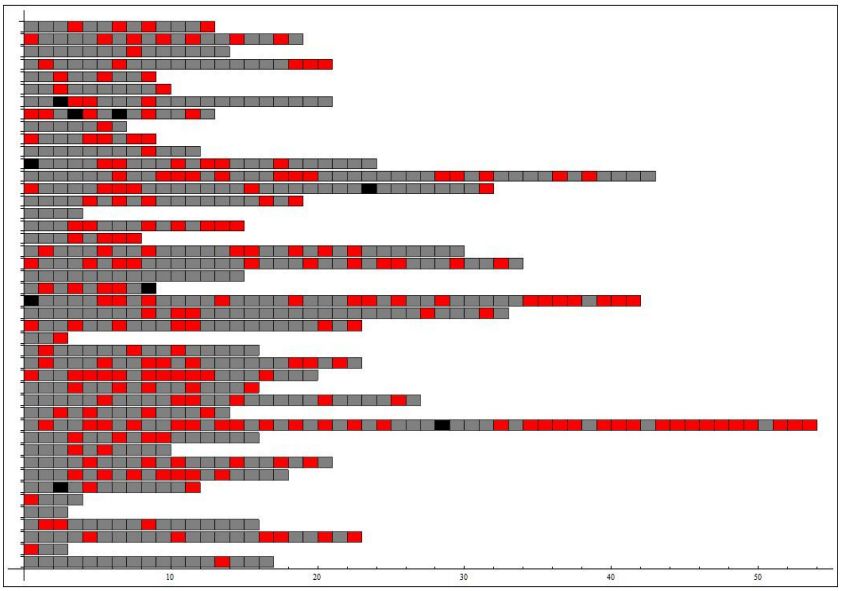


Figure 5. This is a visualization of the movie review of "Alice in Wonderland" by Roger Ebert (Chicago Sun-Times). Each bar represents a sentence. Each section represents a word. The gray sections represent words from the Dale-Chall list, red sections are words not from the list, and the black sections are non words (e.g. numbers).

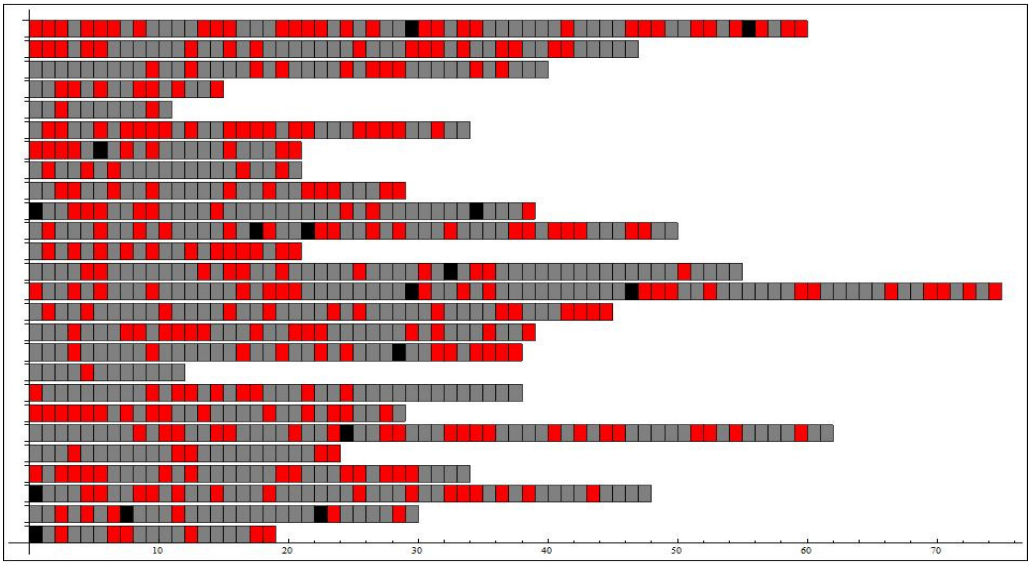


Figure 6. This is a visualization of the movie review of "Alice in Wonderland" by Dana Stevens (slate.com). Each bar represents a sentence. Each section represents a word. The gray sections represent words from the Dale-Chall list, red sections are words not from the list, and the black sections are non words (e.g. numbers). It is clear that Ebert's writing is more readable.

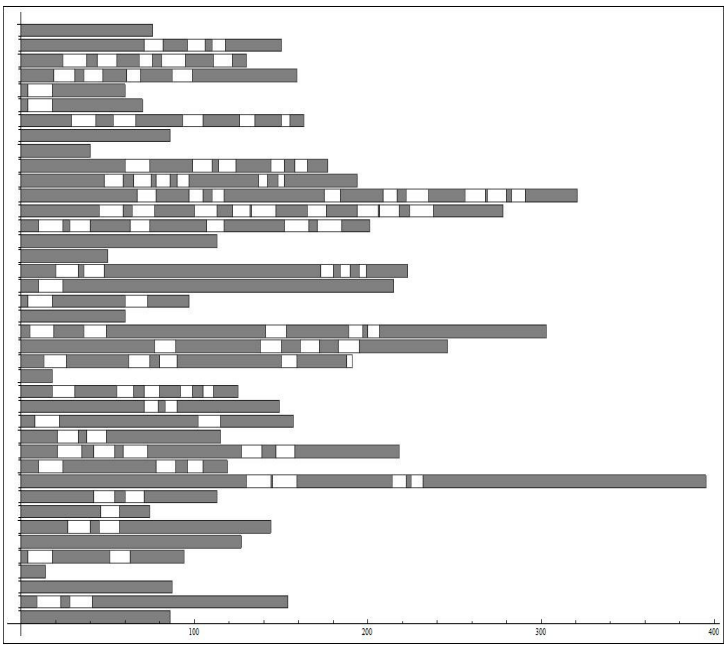


Figure 7. This is a visualization of the movie review of “Alice in Wonderland” by Roger Ebert. Each bar represents a sentence. Each gray section represents a sub-clause. The white gaps are indicators of the level of depth the sub-clauses appear in the parse tree. The higher the sub-clause is on the parse tree, the bigger the white gap.

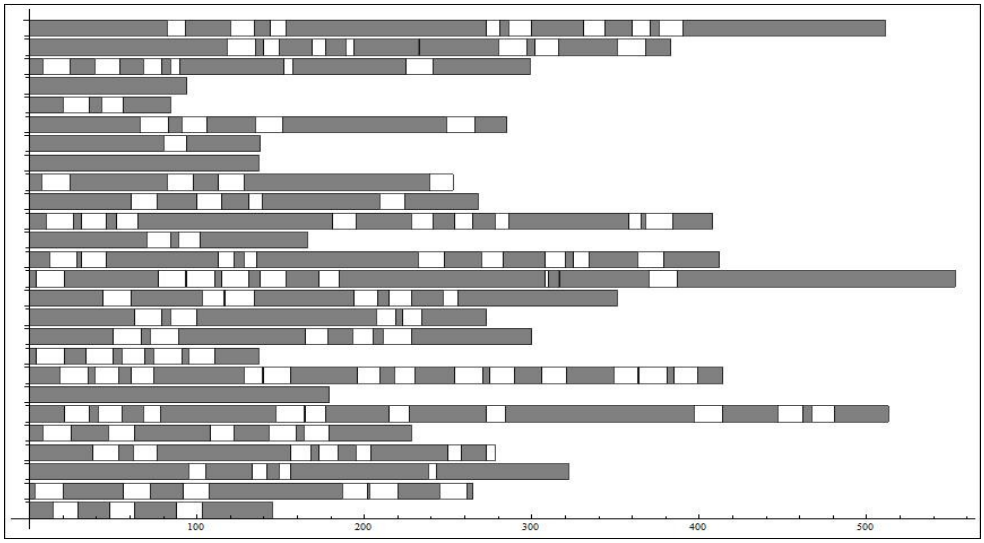


Figure 8. This is a visualization of the movie review of “Alice in Wonderland” by Dana Stevens (slate.com). Each bar represents a sentence. Each gray section represents a sub-clause. The white gaps are indicators of the level of depth the sub-clauses appear in the parse tree. The higher the sub-clause is on the parse tree, the bigger the white gap. It is clear that in general Stevens write more complex sentences than Ebert.

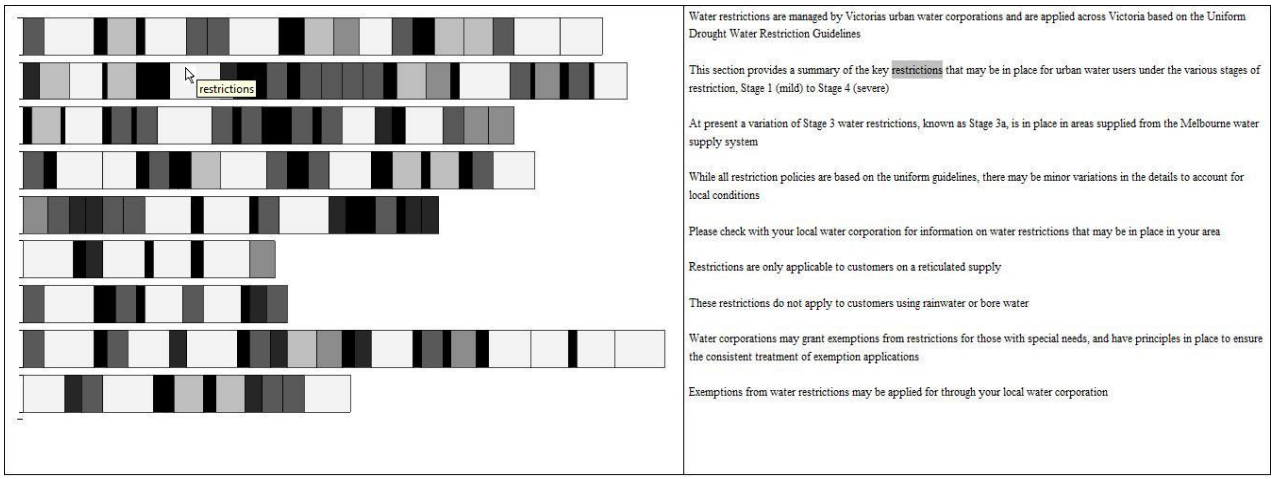


Figure 9. The visualization and the original text can be displayed side by side. When the user moves the mouse cursor over to a bar, the corresponding word is highlighted in the text. The word is also displayed in the tooltip.

3.1.3 Conclusion

My readability visualization is an attempt to address several issues. First, traditional readability indexes are too simplistic for in-depth and localized analysis. My visualization allows readers and writers to quickly identify the distribution of complex words and sentences across a document. Second, traditional readability indexes use only simple sentence length to measure sentence complexity. My visualization can help users quickly compare not only sentence lengths but also the syntactic structures of sentences. In this context the extension of the work to the phrase level structural complexity shows a promising one.

My case studies have demonstrated that the text readability can be effectively visualized. The different writing styles of different authors are clearly visible. The visualization is particularly useful for quick comparison, and can also serves as a map for writers to quickly locate areas to revise.

3.2 Recommendation by Composition Style

A Recommender systems attempt to reduce information overload by selecting a subset of items from a large data set based on user preferences. There are generally two types of recommendation systems: collaborative filtering and content-based approach. Collaborative filtering methods make recom-

mendations not based on content but rather on previous users' selections. Content-based approaches, on the other hand, explore the semantic aspect of the content using statistical and machine learning techniques.

However, there is one aspect that is often missing in the traditional recommendation system: composition style. Composition style is often an important factor in readers' selection of reading materials. For example, a reader may seek out articles written in similar style as his or her favorite writer. But neither the collaborative filtering nor the traditional content-based approaches address this issue. Here I propose a novel recommendation system based on the composition style. I use the Stanford Natural Language Parser (SNLP) [39, 40] to create clause level diagram of the syntactical structure of the document. Then my program searches other documents with similar syntactical structures. The syntactical structures are represented by matrices, and I compare the syntactical similarity based on the distance metric between two matrices. After the first round of automatic selections, users can then visually compare these documents to make a final selection. By incorporating human visual perception with computer based recommendation, users can find the document that fits their preferred composition style.

This approach is beneficial for recommending textual documents such as online product reviews, movie reviews, books, magazine articles, etc.

3.2.1 Methods

A. Converting sentence structure diagrams to matrices

With the help of SNLP I can visualize the compositional structure of a document by stacking horizontally each sentence's gray-white bar representation (see Figure 4). To compare the syntactic structural similarity between documents, I convert the sentence structural diagrams into matrices. As each gray section of the bar equals to the numbers of character that clause made of and each white section

length represent the depth level in the tree, I collect these numbers for each sentence as a row in the matrix and whole document constitutes the whole matrix this way.

B. Measuring the similarity between sentence structure matrices

I measure the similarity between two matrices by two metrics: distance and mismatched elements.

- *Distance:*

If the rank of two matrices A and B are the same, for example r, then the regular distance between them could be found as

$$Distance = Norm(A - B)$$

If the rank for matrices A and B are r_A and r_B respectively with the constraint $r_A > r_B$ then I try to find a matrix C of rank r_B that is the closest to matrix A through the best rank approximation method so that I can find the difference of the matrices. The regular distance between B and C are defined as,

$$Distance = Norm(B - C)$$

Here I chose Frobenius norm, which is defined for a matrix A of dimension m×n as,

$$\|A\|_F = \sqrt{\sum_{i=1}^{m}\sum_{j=1}^{n}(a_{ij})^2},$$

where a_{ij} is the *i* th row and *j* th column element of the matrix A.

If the two matrices in comparison are not of the same order, then I pad the smaller matrix with zeros to equate the dimension of the bigger one.

- *Mismatch Elements:*

If a small matrix is a subset of the big one and the big matrix consists of a few additional elements that are numerically large, then the similarity between the matrices in terms of Frobenius norms will be big. But this may not reflect the nature of their similarity. To address this issue I try to measure similarity in terms of the number of similar elements.

If C is the biggest common element between the case matrix A and control matrix B, and the dimension for *A*, *B*, *C* are d_A, d_B and d_C respectively, then

Control Mismatch = $1 - d_C / d_A$,

Case Mismatch = d_{B-C} / d_B

and considering the mismatch has similar effect on case and control I define,

$$TotalMismatch = 0.5 \times ControlMismatch + 0.5 \times CaseMismatch$$

- *Dissimilarity:*

Taking both distance and mismatched elements into account, I define the dissimilarity between two matrices as a linear combination of both factors:

$$Dissimilarity = \alpha * Distance + \beta * TotalMismatch$$

where α and β are two coefficients who's sum varies between 0 and 1.

Hence I can use the above dissimilarity measurement to compare the syntactical structure similarity of two text documents. In this regard the matrix is formed through the gray and white bar lengths of a sentence as a row of a matrix and the number different lengths acts as the different columns in that row of the matrix. The number of sentences in the text serves as the number of rows here as well.

3.2.2 Implementation and Results

In this section I present examples of my visualization and compare them. For this purpose I have chosen simonblog [92] for iphone related blog by a blogger as my control data and collected ipad related three more blogs by the same blogger as my case data. The visualizations are presented in figures 10 to 13.

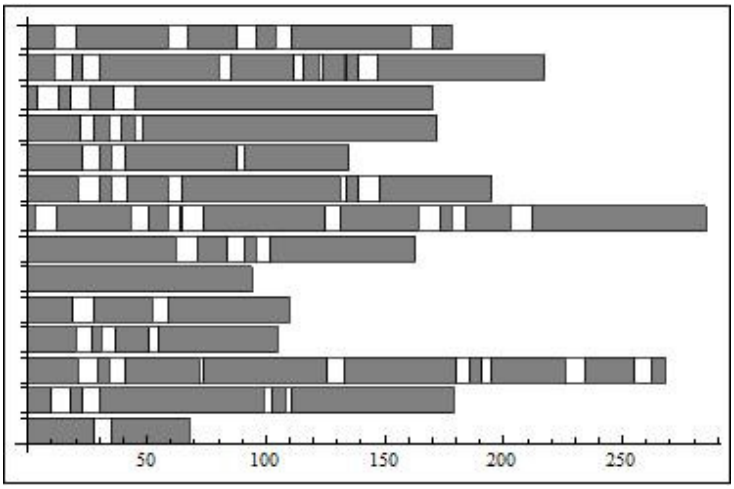


Figure 10. This picture shows the structural complexity of sentences taken from iphone related blog by a blogger. Each bar represents a sentence. Each gray section represents a sub-clause. The white gaps are indicators of the level of depth the sub-clauses appear in the parse tree. The higher the sub-clause is on the parse tree, the bigger the white gap.

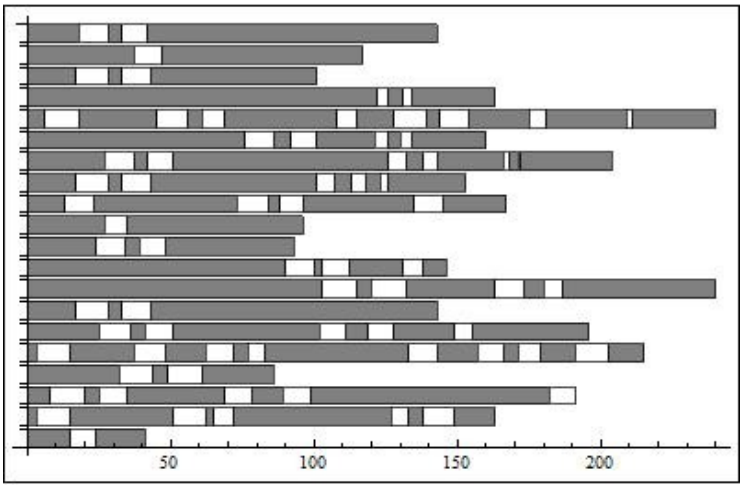


Figure 11. This picture shows the structural complexity of sentences taken from the first ipad related blog by the same blogger.

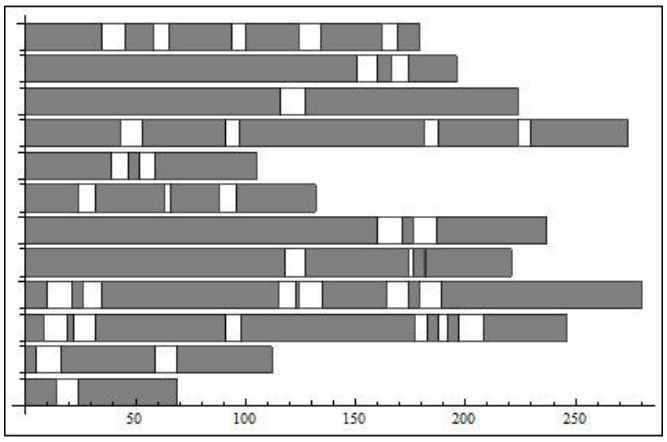


Figure 12. This picture shows the structural complexity of sentences taken from the second ipad related blog by the same blogger.

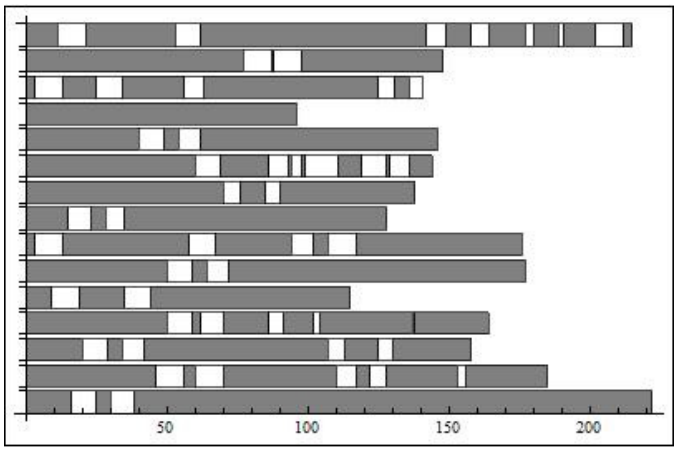


Figure 13. This picture shows the structural complexity of sentences taken from the third ipad related blog by the same blogger. The composition style is quite visually similar for figures.

I computed the aforementioned matrix distance for the figures 11, 12, 13 with respect to the figure 10 and they are respectively 0.786, 0.69 and 0.62 and the *TotalMismatch* for all the three figures came as 1. Interestingly as the distance here turns out to be less than one in for all the cases turns out there is some similarity. Unless two texts has exact similar subsections (i.e in case one text copied somepart of the other text) the TotalMismatch is generally 1. Now if one text has a multiplied volume subset of another text, TotalMismatch becomes less effective in finding mismatch and in that case more rigorous and computational expensive matrix matching techniques has to be employed. With the help of distance metric and visual perception I justify structurally similar texts.

3.2.3 Conclusion

I have described a recommendation method that adds composition style comparison on top of the traditional keyword based search. In addition to specifying a list of keywords, a user also provides a sample text document. My system will then search documents first based on keywords, but then rank the search results based on their syntactical structure similarity to the sample text. Users can then visually compare the structure of the documents and make a final selection.

As a result, my method helps users choose a document that is not only relevant in content but also conform to his/her preferred composition style. My preliminary results have shown that it has great potential for providing personalized search results, and thus improving user experience.

3.3 Visualizing Multiple Readability Indexes

A typical readability index is a single average number or classification for the entire document. Although readability indexes can help quickly classify a document, the traditional form of readability indexes are often too simple and abstract for in-depth analysis. For instance, two documents with very different distribution of complex words and sentences can have similar readability indexes. In case of a multiple authored document the overall readability index may hide a particularly difficult section written in more complex style compared to its overall readability index. Another issue is that although the different readability indexes correlate well at the document level, they often do not correlate well at the paragraph level.

To address this issue, I propose a method to visualize readability metrics for each paragraph of a document so that a writer or reader can visually understand and analyze the complexity of each paragraph.

My visualization method [51] employs two forms of visualization schemes. First, the overall readability index for each paragraph is visualized as a color coded ring. The color represents the readability score. This allows users to quickly understand the complexity of the document at the paragraph level. I also

created an explanatory visualization that shows all the different readability indexes for a paragraph. In this visualization different readability indexes are represented by color coded abbreviation such as “ARI” for Automatic Readability Index, “CL” for Coleman-Liau index, “FK” for Flesh Kinkaid readability ease, etc.

The second scheme in the aforementioned is to use Chernoff faces to encode multiple readability indexes for each paragraph. Each component of the face composition (such as size of the eye, curvature of the moth etc.) encodes a different readability index. The Chernoff face scheme uses the human cognitive system’s familiarity with facial expressions to help readers or writers to quickly identify outliers.

This visualization [51] improves the presentation of the readability indexes from its traditional numerical expressions to visual patterns that can be quickly recognized. With such readability visualization, both readers and writers can visually determine the difficult paragraphs, or tally the readability of multiple documents or multiple versions of the same document. Particularly, a writer can use it to evaluate the readability of the composed document at the paragraph level through inspecting how various readability metrics are distributed across the document and find specific areas for revision.

3.3.1 Methods

B. Calculating the readability indexes

I computed five readability indexes: Flesh Kinkaid reading ease, Gunning Fog score, Coleman-Liau index, SMOG index and Automatic Readability index. Here are their formulas [141].

Flesch-Kincaid Reading Ease index:

$$206.835 - 1.015 \times \left(\frac{total \quad words}{total \quad sentences} \right) - 84.6 \times \left(\frac{total \quad syllables}{total \quad words} \right)$$

Gunning Fog index:

$$0.4\times\left(\left(\frac{words}{sentence}\right)+100\times\left(\frac{complex\quad words}{words}\right)\right)$$

Coleman-Liau index:

$$5.89\times\left(\frac{characters}{words}\right)-29.5\times\left(\frac{sentences}{words}\right)-15.8$$

Simple Measure of Gobbledygood (SMOG) index:

$$1.043\times\sqrt{30\times\frac{number\quad of\quad polysyllables}{number\quad of\quad sentences}}+3.1291$$

All of the these indexes involve the computation of sentence length in terms of words, word length in terms of characters and syllables and paragraph length in terms of the number of sentences. Counting the number of characters per word is straightforward, but counting the number of syllables is a little more complicated [41]. The computation of syllable count is described in section 3.1.1 A.

C. Categorizing and color coding the indexes

I divide the whole range of possible readability index scores into five categories, from the highest to the lowest. Each category is assigned a color: red represents the worst readability index; green represents the best; orange, yellow, and cyan represent the indexes in between. After computing the readability index of a paragraph, the result is classified into one of the five categories. The overall readability index is calculated by averaging the different readability indexes for a paragraph. The overall readability index is then assigned the corresponding color code. A ring with the assigned color is then generated and displayed alongside the paragraph. An example is shown in Figure 14.

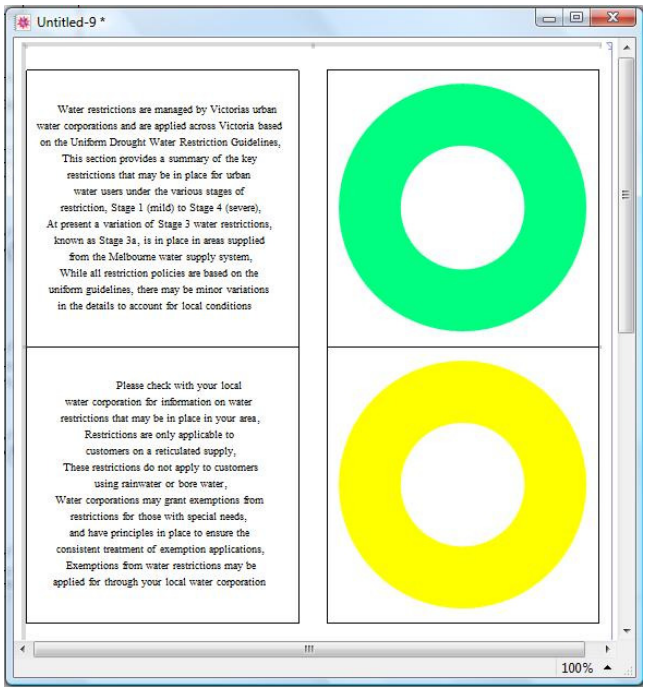


Figure 14. Overall readability index visualization as a color coded ring

Users can also choose to display the individual readability indexes as text abbreviations. The readability indexes are separated by '-'. An example is shown in Figure 15.

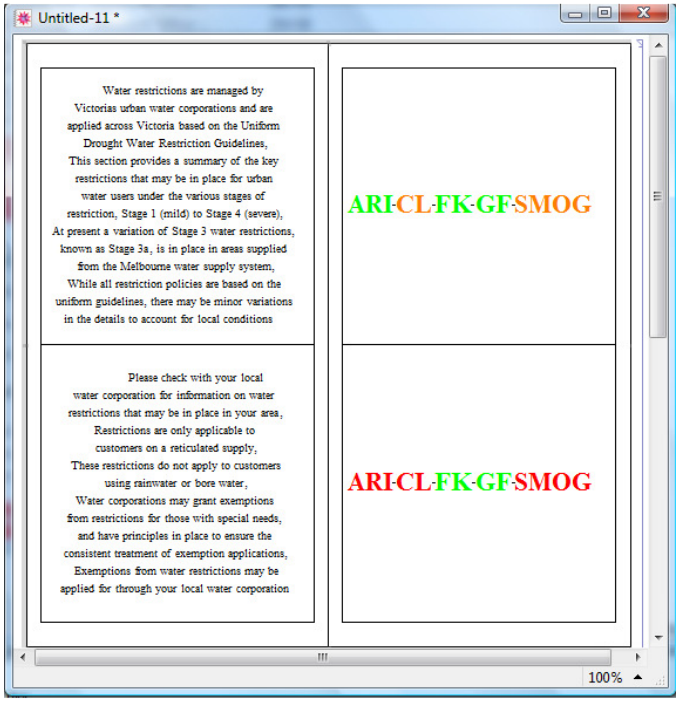


Figure 15. Visualization for different readability indexes

D. Visualizing readability indexes using Chernoff faces

Chernoff faces [53] are cartoon like characters that visualize multiple variables with different parts of the face. Here I have five different types of readability indexes mapped to five facial characters:

- Flesh Kinkaid is mapped to the shape of face (circular to oval);
- Gunning Fog is mapped to the size of the eyes;
- SMOG is mapped to the orientation of the eyes (acute angle to perpendicular with the nose);
- Coleman-Liau is mapped to the size of the mouth;
- ARI is mapped to the orientation of the mouth (smiling to sad).

Figure 16 and 17 show some examples.

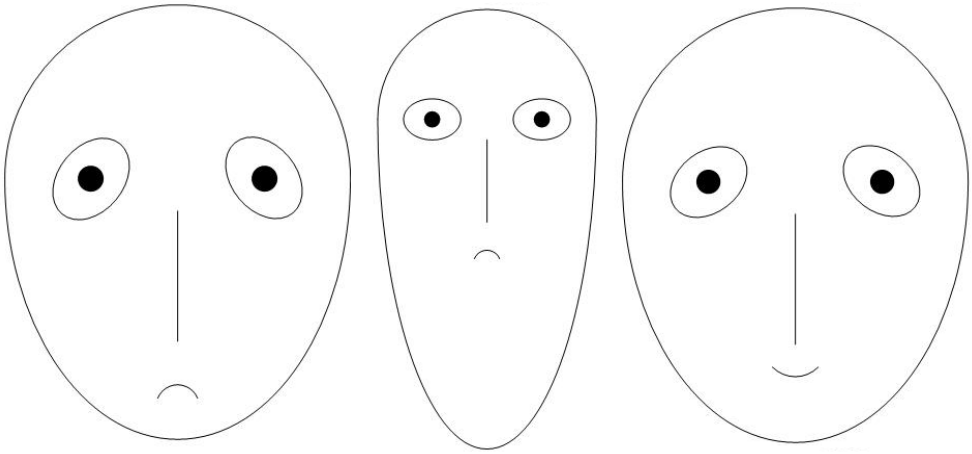


Figure 16. Example of Chernoff faces

Figure 17 shows an example of mapping different readability indexes to different facial characters.

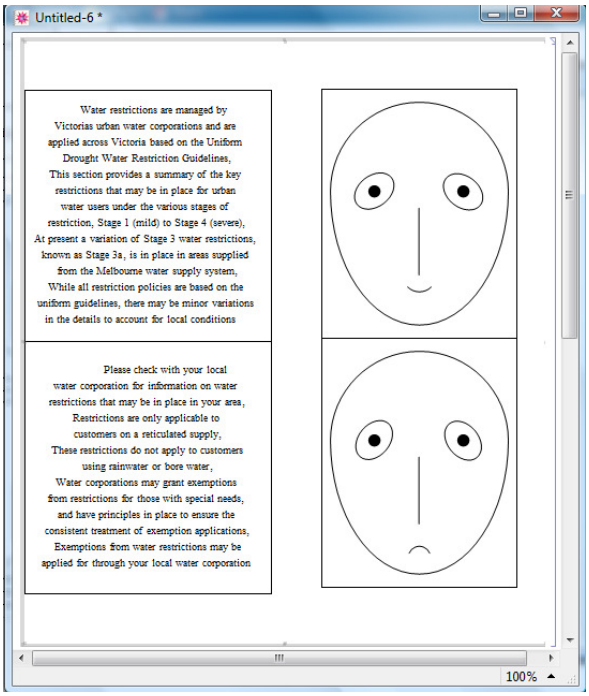


Figure 17. Visualizing five readability indexes with one Chernoff face per paragraph.

3.3.2 Implementation and Results

All of the visualizations were created in Mathematica 7.0, which provides a rich set of functions for visualization and text processing. As a case study, I present the visualization of two reviews of the movie "Alice in the Wonderland", one by Roger Ebert (Chicago Sun-Times) and the other by Dana Stevens (slate.com). The visualizations are presented in Fig. 18, 19 and 20.

Comparing Figure 18 and Figure 19 we can see that Roger Ebert's composition has more green indexes compared to Dana Stevens', suggesting that Roger Ebert's writing is more readable. In Figure 20, we can compare Roger Ebert's writing (left) with Dana Stevens' (right) by Chernoff faces. We can see that all the faces in Roger Ebert's section have smiley mouths, with some of them big smiles with more horizontal eyes. But the faces in Dana Stevens' section have one sad face with smaller smiles and more angular eyes. Therefore readers can quickly recognize that Stevens' composition is somewhat more difficult to read than Ebert's.

Figure 18. This is a visualization of the movie review of "Alice in Wonderland" by Roger Ebert (Chicago Sun-Times). The left section shows the overall readability index for each paragraph and the right section shows different readability indexes in their abbreviations.

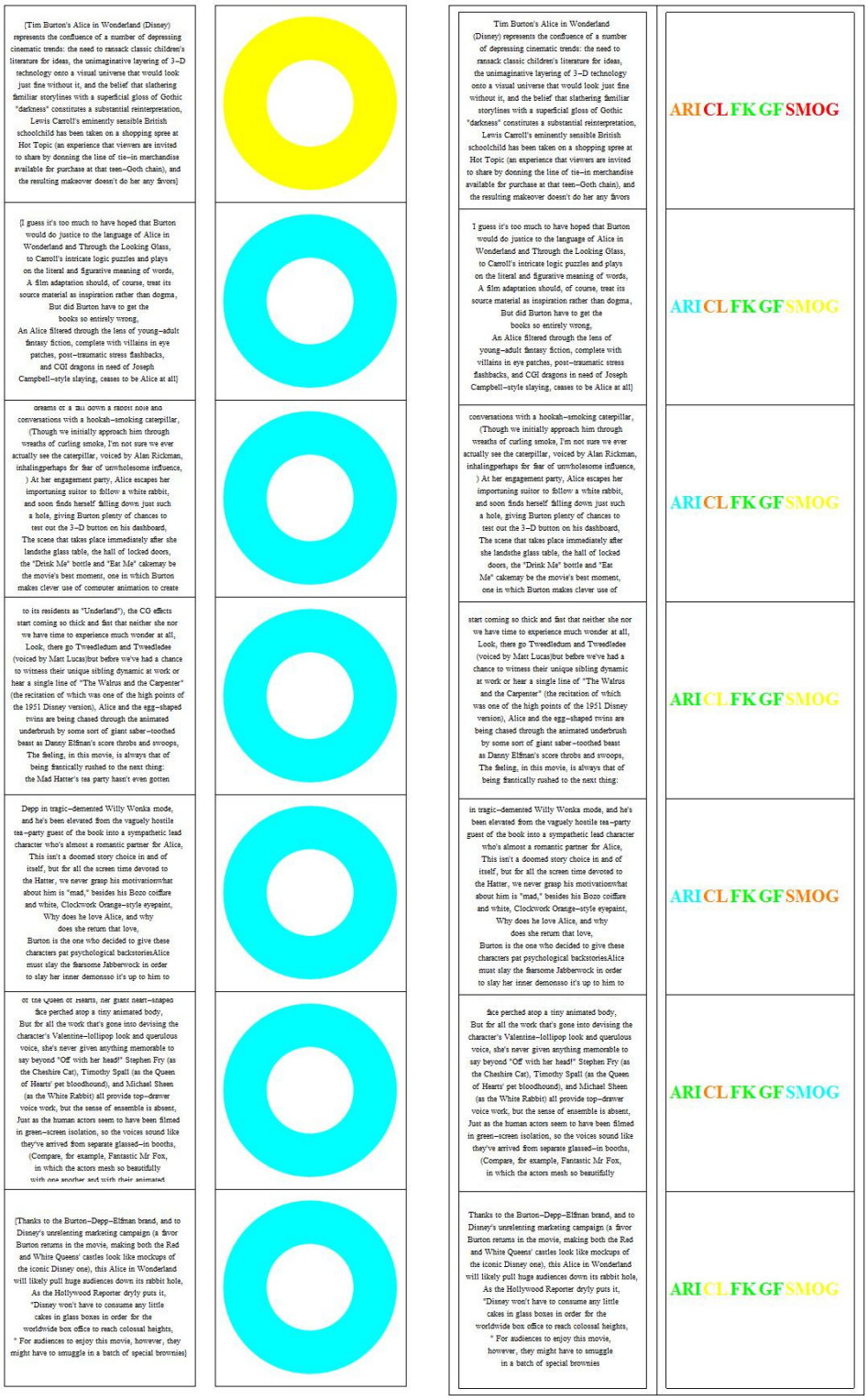


Figure 19. This is a visualization of the movie review of "Alice in Wonderland" by Dana Stevens (slate.com). The left section shows the overall readability index for each paragraph and the right section shows different readability indexes in their abbreviations.

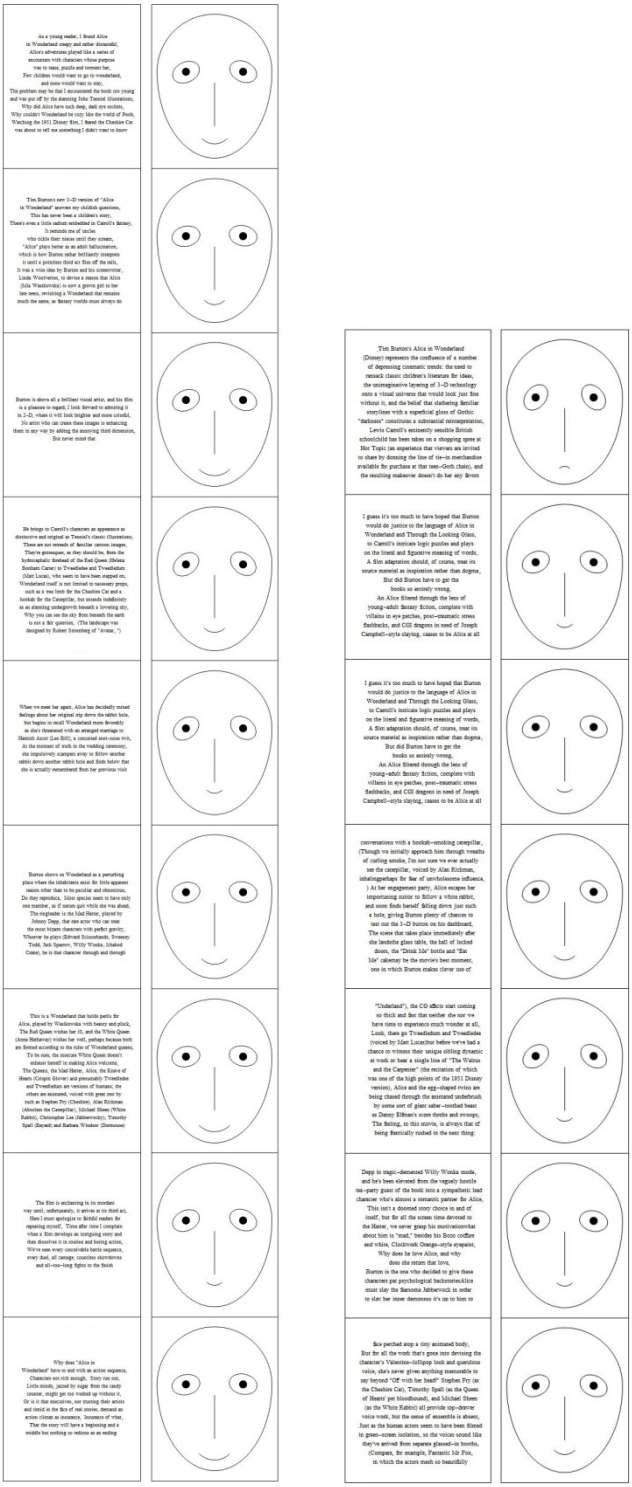


Figure 20. This is the Chernoff face visualization of the movie review of “Alice in Wonderland” by Roger Ebert (the left section) and Dana Stevens (the right section). Each Chernoff face encodes five readability indexes for that paragraph.

3.3.3 Conclusion

My readability index visualization addresses the issue that the traditional readability indexes are too abstract for in-depth analysis of the document. My visualization allows readers to quickly identify article sections with low readability. For writers, it can help quickly locate the sections that need revision. It is particularly helpful for analyzing and revising multi-authored documents.

4 ANALYSIS OF A SOCIAL DATA VISUALIZATION WEB SITE

In the past few years, online social data visualization has emerged as a new platform for users to construct, share, and comment on data visualizations online. The most well known online data visualization tools include Many Eyes, Swivel, and Tableau Public. In here, I report my analysis of Many Eyes – an IBM research project. By analyzing all the data visualizations constructed by users from 2007 to 2010, I provide insight into online user behavior as well as patterns and trends in social data visualization.

4.1 Descriptive Analysis

As I believe ManyEyes is the largest experiment on user generated data visualizations, I chose this site for analysis. In ManyEyes a user can create or search the data visualizations as well as comment on existing visualizations. This gives me enough motif to collect information from this social collaborative environment to explore the nature of collaboration. For every data visualization, I retrieved the following information

- Author's name
- Date posted
- Title of the visualization
- Type of the visualization
- Tag (if any)
- Rating (if any)
- All the comments

For each comment, I retrieved the following

- Information
- Author's name
- Data posted

- Comment

After the retrieval I concluded the report with the help of descriptive statistics.

- ***Descriptive Statistics***

Descriptive statistics are used to describe the basic features of the data in a study. This provides simple summaries about the sample and the measures. This forms the basis of virtually every quantitative analysis of data which could be enhanced with simple graphical analysis of the data as well.

In the single variable or Univariate analysis generally three major characteristics are explored:

- the distribution
- the central tendency
- the dispersion

The distribution is a summary of the frequency of individual values or ranges of values for a variable.

The central tendency of a distribution is an estimate of the "center" of a distribution of values. There are three major types of estimates of central tendency:

- Mean
- Median
- Mode

Dispersion refers to the spread of the values around the central tendency. There are two common measures of dispersion, the range and the standard deviation.

4.1.1 Implementation and Analysis

Collecting data from the Many Eyes web site is not trivial. Although a user can browse or search the data visualizations, the web site does not provide a downloadable database of all the visualizations. The web site does provide a RSS feed that broadcasts every new comment. However, at the time of my study, I could only retrieve comments made after March 2009. Therefore I used a Web data extraction tool to retrieve the data page by page.

Overall, I collected over 7,000 data visualizations and over 33,000 comments, made between January 2007 and June 2010, and saved them in a SQL database. My analysis was based on the queries made to this database. Note that I exclude comments automatically generated by the computer.

A. Activity Analysis

I) The number of data visualizations

Table 1. The number of data visualizations and comments by year

Year	Number of data visualizations	Number of comments	# comments / # visualization
2007	2172	13180	6.07
2008	1630	10619	6.51
2009	3260	10635	3.26
2010(up until June)	2433	7357	3.02
Total	9495	41791	

From Table 1, I see that the popularity of Many Eyes is getting stronger over the years. After a dip in 2008, the number of data visualizations bounced back strongly in 2009, and the number in 2010 is also strong. An interesting fact is that the average number of comments per visualization has been declining from 2007 to 2009, and then bounces back in 2010, meaning that people are commenting more on data visualizations in 2010. This suggests that the level of social interaction is getting higher this year. But it is unclear why there is a sudden increase of social interaction in 2010. It will be inter-

esting to monitor this number in the coming months and see which direction it is going.

Upon close inspection, I find that the comments are very unevenly distributed. The lively discussions were concentrated on a small number of data visualizations, while the vast majority of the visualizations attract little attention.

I also analyzed the number of data visualizations and the number of comments created per month but did not find clear patterns. There is often a big variation in the number of data visualizations or comments created for each month over different years. However, I did find some outliers in the number of comments made in certain months. For example, the number of comments made in January 2007 (2249 comments) and April 2010 (2308 comments) are much higher than the other months. April 2010 also has the highest monthly number of user generated data visualizations (499 data visualizations) in my study. The case of January 2007 is easy to explain because that was Many Eyes' first month of operation and there were a lot of curious new users. A possible explanation for the high number in April 2010 is that April 15 is the deadline for filing income taxes in the United States. There were probably a lot of data visualizations and comments about personal finance and the economy. Further investigation is needed to confirm this hypothesis.

II) Data visualization types

Table 2. Most created chart types by year

2007	**2008**	**2009**	**2010**
Bubble Chart	Bubble Chart	Word Cloud	Word Cloud
World Map	Bar Chart	World Map	Bubble Chart
Tag Cloud	Network Diagram	Word Tree	Word Tree
Bar Chart	Tag Cloud	Bubble Chart	Matrix Chart
Network Diagram	Word Cloud	Tag Cloud	World Map
Treemap	Matrix Chart	Bar Chart	Bar Chart
Line Graph	Word Tree	Matrix Chart	Treemap
Scatterplot	World Map	USA Map	Tag Cloud
US State Map	USA Map	Treemap	Network Diagram
Stack Graph for Categories	Line Graph	Network Diagram	USA Map

Table 3. Overall most created and commented chart types by year

10 Most created chart types	10 Most commented chart types
Bubble Chart	World Map
Word Cloud	Bubble Chart
World Map	Network Diagram
Bar Chart	US State Map
Tag Cloud	Stack Graph
Word Tree	Scatterplot
Network Diagram	Word Tree
Matrix Chart	Bar Chart
Treemap	Word Cloud
Line Graph	Tag Cloud

Table 2 and 3 shows the most created and commented visualization types. First of all, the popularity of Bubble Chart is somewhat surprising. It will be interesting to investigate the type of data the Bubble Chart were used for and how readers commented on them. It is not yet clear why Bubble Chart is so attractive to many users and whether it's an effective visualization type. Second, text visualizations such as Word Cloud, Tag Cloud, and Word Tree are quite popular, indicating that lots of users are using data visualization for text analysis. Third, the typical business and scientific data charts such as Line Graph and Scatterplot are ranked relatively low (e.g. lower than Treemap). This may indicate that the users of Many Eyes are more likely to use it for fun and curiosity than for business or scientific research purposes. Fourth, Pie Chart and Stack Graph are ranked low in popularity, perhaps indicating that after years of educational efforts by peoples like Tufte, the drawbacks of Pie Charts and Stack Graph are accepted by the general public.

B. User Analysis

I) Creators

Table 4. Registered users who created more than 20 visualizations (by year)

2007	2008	2009
Martin Wattenberg (273)	Martin Wattenberg (34)	Best_Baseball_Players_2009 (28)
Fernanda Viegas (122)	Belarius (26)	Irene Ros (26)
Fran Van Ham (95)	Iamcurious (20)	Jovirox (23)
Cgreen (78)		Fernanda B. Viegas (21)
Grjenkin (52)		
Belarius (41)		
Matt McKeon (37)		
Colm (35)		
Lee Byron (30)		
Jesse (26)		
0c73d86e-ad2f-11dd-84b8-000255111976 (25)		
Dcjohn (24)		
JasonW (21)		

Table 5. Creator analysis

	2007	2008	2009
Number of registered users who created at least one visualization	310	428	381
Number registered users who created 10 or more visualizations	22	14	12
Number of visualizations created by Anonymous users	627	692	2374

From Table 5, I can see that the vast majority of the visualizations are created by anonymous users. There is a big increase in the numbers of visualizations created by anonymous users in 2009, while the numbers of visualizations created by the registered users are relatively stable over the years. Since I know there is also big increase in the total number of visualizations created in 2009, it is reasonable to assume that there is a big increase in the number of anonymous users in 2009. Because the number of comments made in 2009 is similar to 2008, I speculate that these new anonymous users constructed many data visualizations but made few comments.

If I define an active contributor as a registered user who created more than 10 visualizations, then the percentage of active contributors was very low. Among the registered users who created at least one visualization, only 7% of them in 2007, 3% in 2008, and 3% in 2009 were active contributors. (Also note that these active contributors include some of the IBM research team members.) Only three registered users in 2008 made more than 20 visualizations and one of them was from the IBM research team. The situation in 2009 is similar: only 4 users created more than 20 visualizations and one of them was from the IBM research team. The year 2007 was different because it is the first year of Many Eyes' operation and the IBM research team (Wattenberg, Viegas, van Ham, McKeon, and perhaps Jesse) created large numbers of visualizations, probably for testing the system. (It seems that the IBM research team made far fewer data visualizations in 2008 and 2009.)

Overall, the vast majority of Many Eyes' users are casual users, making very small number of visualizations. This is not a surprise. However, I am surprised by how low the number of active contributors is.

There could be anonymous users who make many data visualizations. However, it is impossible for us to identify. But based on the registered users' behavior, it is unlikely that the anonymous users will be much different.

II) Commenter

Table 6. Commenter analysis

	2007	2008	2009
Number of registered users who made at least one comment	328	436	440
Number of registered users who made more than 10 comments	81	71	57
Comments posted by Anonymous users	4421	4840	5540

The situation with the commenters is similar to the situation with the contributors. From Table 6, I can see that the vast majority of the comments were made by anonymous users. Among registered users who made at least one comment, only 24% in 2007, 16% in 2008, and 12% in 2009 were active commenters (with over 10 comments). The percentage of active commenters though higher than that of the active creators, is decreasing over the years. In addition, the top commenters often include members from the IBM research team (e.g. Wattenberg and Viegas).

About 50% of the comments were made by anonymous users. Unlike the number of creators, I don't see a significant increase in the number of anonymous comments in 2009.

4.1.2 Conclusion

I have presented my preliminary analysis of online social visualization web site Many Eyes. Since a number of new online social visualization tools have been launched in the past year, the interest in this area is strong. Therefore the lessons I learned from Many Eyes web site will be valuable for the other online social visualization services.

My key findings are as follows.

- The number of data visualizations created on Many Eyes has been steadily increasing since 2008. User activity remains high. But the increase of data visualizations in 2009 is likely due to a large number of new anonymous users.
- Bubble Chart and World Map are the most created and commented visualization types.
- Text visualizations, such as Word Cloud, Tag Cloud and Word Tree, are among the most popular visualization types, indicating that the interest in text visualization is very high.
- After a steady decline over three years, the number of comments per visualization suddenly rises in 2010. It seems that new users who came to Many Eyes in 2010 are more interested in commenting on others' data visualizations than previous users. The social interactions on Many Eyes seem are getting stronger in 2010, which is an interesting trend.
- Overall the number of comments per visualization is still relatively low. A large number of comments are concentrated on a very small number of data visualizations. The vast majority of the data visualizations attract little attention.
- The number and percentage of active creators are very low. The number of active commenters is decreasing. About 50% of the comments were made by anonymous users.
- The vast majority of users seem to be casual users who create data visualizations for fun and curiosity. Very few people use this web site for serious data exploration or discussion.

Overall, online social visualization web sites such as Many Eyes need to find ways to attract more serious, active users and to promote more extensive social interactions among users. The success of a social media web site relies heavily on a relatively small group of very active contributors. The number of active contributors on Many Eyes, after excluding the IBM research team members, is surprisingly low and decreasing. The situation with the number of active comenters is similar. An extended analysis using Swivel, Tableau Public, and Google Public Data Explorer in the same context would be quite prom-

ising in future.

4.2 Exploring Relationship in Social Collaborative Website Retrieved Data

With potential amount of growth in today's world online social data visualization has been started gaining the hype for tomorrow. As a result people's interaction in such an environment has been analyzed at a very basic level [91] to understand how the users are constructing, viewing and discussing visualizations. Therefore the importance of such analysis continues in lieu of how different temporal factors affecting specific type of visualizations and their respective comments. Also Many Eyes attach some system generated tags to each and every visualization gets created on their platform; does this really relevant to the category of the charts or is it arbitrary? Therefore my motivation here to explore the answers for this questions using some analytical statistics.

The focused part of the data analysis covers categorical data collected from the Many Eyes site. In categorical data analysis two-way contingency tables formed by cross classifying categorical variables and are typically analyzed by calculating chi-square values testing the hypothesis of independence since Karl Pearson's phenomenal introduction in 1900 [94]. In the 1970s, a dramatic change was brought in the analysis of cross-classified data through the publication of a series of papers on loglinear models by L.A. Goodman [103][104].

4.2.1 Methods

C Data which are reflecting the classification of objects into different categories are referred as categorical data. A contingency table is a tabular representation of categorical data. It usually shows frequencies for particular combinations of values of discrete random variables represented in the table dimensions.

In contingency table the initial analysis is done through finding association between variables with the use of various kinds of Chi Square tests.

A. Pearson's Chi Square Test:

Pearson's chi-square test [94] (χ2) is one of a variety of chi-square hypothesis tests which employs the chi-square distribution to conclude the result of the test. It is used to test the association of row and column variables in a contingency table.

It tests a null hypothesis that the relative frequencies of occurrence of observed events follow a specified frequency distribution. For the test,

Null Hypothesis H_0 : There is no association between the row and column variables of the contingency table

Alternative Hypothesis H_1 : There is association between the variables.

The Chi-square statistic for the null hypothesis is calculated by finding the difference between each observed and theoretical frequency for each possible outcome (the total number of cells present in the table) in the contingency table, squaring them, dividing each by the theoretical frequency, and taking the sum of the results. The number of degrees of freedom is equal to the number of possible outcomes minus 1.

$$\chi^2_{n-1} = \sum_{i=1}^{n} \frac{(O_i - E_i)^2}{E_i} \qquad [137]$$

where O_i = an observed frequency;

E_i = an expected (theoretical) frequency, asserted by the null hypothesis;

n = the number of possible outcomes of each event.

This χ2 statistic value is compared with the χ2 distribution of n degrees of freedom.

Now if the probability of finding that value is found low in the distribution, then the null hypothesis is rejected with the conclusion that there is association between the row and column variables of the contingency table.

B. Cramer's V :

Cramer's [138] V is a method for calculating correlation in contingency tables, used mainly for the tables containing rows and columns. Once the Chi-square test establishes the association between the categorical variables of a contingency table, this technique is performed to figure out the strength of the association.

V is calculated by first calculating chi-square, then using the following calculation:

$$V = \sqrt{\frac{c^2}{n(k-1)}} \qquad [139]$$

where c^2 is chi-square and k is the number of rows or columns in the table.

The values of Cramer's V range from 0 and 1. If the value comes out close to 1, indicates the strong association between the variable under consideration from the contingency table, where as the value close to zero refers weakness.

C. Loglinear Modelling:

One of the specialized cases of generalized linear models is the loglinear model, which comes in action for Poisson distributed data [132].

Poisson distribution [132] represents counts or frequency of some event across time or over an area. According to statistical theory, the Poisson distribution is a discrete probability distribution that expresses the probability of a given number of events occurring in a fixed interval of time and/or space if these events occur with a known average rate and independently of the time since the last event. Loglinear modeling is an extension of the two-way contingency table analysis. There conditional relationship between two or more discrete, categorical variables is analyzed by taking the natural logarithm of the cell frequencies within a contingency table. This analysis doesn't distinguish variables as depen-

dent or independent variables; instead it treats all the variables as response variables and eventually results the analysis through directing the association between variables.

The following loglinear model [134], corresponds to a 2 x 2 contingency table with two categorical variables A and B each with two levels, is evaluated to find out the association between the categorical variables of the tbale.

$$Ln(F_{ij}) = \mu + \lambda_i^A + \lambda_j^B + \lambda_{ij}^{AB} \quad [134]$$

where, $Ln(F_{ij})$ = is the log of the expected cell frequency of the cases for cell ij in the contingency table.

μ = is the overall mean of the natural log of the expected frequencies of the table

λ = represeants the variable effects on the cell frequencies.

i and j = refer to the categories within the variables

Therefore: λ_i^A determines the main effect for variable A

λ_j^B determines the main effect for variable B

λ_{ij}^{AB} determines the interaction effect for variables A and B

The above loglinear model is the representation of the saturated one in this context, as it incorporates all possible one-way (variable A and B) and two-way effects (the interaction effect for variables A and B). When the model only incorporates some effects not all, considered as an unsaturated model. For example in this model if we assume that variable A has no effect on variable B, or vice versa then we can set the effect parameter λ_{ij}^{AB} to zero and the unsaturated model [134] turns out as :

$$Ln(F_{ij}) = \mu + \lambda_i^A + \lambda_j^B \quad [134]$$

As this particular model doesn't contain the interaction variable, it represents the independence model. This model eventually depicts the variables in the table are unassociated: therefore this serves as the alternative to the chi-square hypothesis test of independence.

The primary strategy in loglinear modeling involves fitting models to the observed count/frequency of the categorical variables from a contingency table. Basically the expected frequencies represent the model perfectly. Now depending on how good the model reflects the actual contingency table, the expected data reflects closeness to the observed data. Therefore after computing the expected frequencies through the modeling, multiple models are compared hierarchically for the best match. The model which reflects most parsimoniousness to the observed data is chosen. To figure out the appropriate best model, the goodness-of-fit statistics associated with models are compared.

The overall goodness-of-fit of a model is assessed by comparing the expected frequencies (F_{ij}) to the observed cell frequencies (f_{ij}) for each model. The Pearson Chi-square statistic or the likelihood ratio (L^2) can be used to test a models fit. The formula for the L^2 statistic is as follows [134]:

$$L^2 = 2\Sigma f_{ij} \ln(f_{ij}/F_{ij}) \quad [134]$$

L^2 follows a chi-square distribution with the degrees of freedom (df) equal to the number of cell counts minus the number of non redundant parameters (Therefore the saturated model always has degrees of freedom zero and the degrees of freedoms increase for the unsaturated cases). The larger the L^2 relative to the available degrees of freedom, the more the expected frequencies depart from the actual cell entries [134]. Therefore, the larger L^2 values indicate that the model does not fit the data well and thus, the model should be rejected.

4.2.2 Implementation and Analysis

Collecting data from the Many Eyes web site is not trivial. The web site does not provide a downloadable database of all the visualizations or comments in the site. But the web site provides a RSS feed that broadcasts every new comment. I used a Web data extraction tool to retrieve the data page by page. For each data visualization, I retrieved the following data for my current analysis

- Type of Visualization
- Timestamp Information of the visualization
- Tag for the visualization
- Comments on the visualization
- Timestamp Information of the visualization

On overall, the retrieval accumulated over 7,000 data visualizations and over 33,000 comments, made between January 2007 and June 2010. The retrieved data is saved in a MS SQL Server database. My analysis was based on the queries made to this database. Using the queries I excluded the system generated comments so that I can draw real collaboration from the users only.

In IBM's ManyEyes, the social collaborative environment, the data has been captured and visualized with time stamp information. This time stamp information includes year, month and day type of the week. I observe the creation or collaboration traffic of the different visualizations have quite a variation in their distribution. Inherently I try to find if there is any relationship between these different types of visualizations and time it has been created/collaborated.

A. Visualization Data

There are 31 different types of visualizations and out of which 12 are just specific different type of maps whose counts are very low so they are merged together into 'Other Map' type and eventually I

have 20 different types of visualizations. There were 7444 such visualizations out of which 6209s have a specific type.

I performed the contingency table analysis of the 20 types of visualization data with four different years (2007-2010) and twelve different months. The cells of each table contain the frequency of counts in that category. Running FREQ procedure with specific options I found the following for the above data.

Table 7. Chi-Square analysis of year wise creation of different visualization types

Statistic	**DF**	**Value**	**Probability**
Chi-Square	57	943.3191	<.0001
Cramer's V		0.2250	

Table 8. Chi-Square analysis of month wise creation of different visualization types

Statistic	**DF**	**Value**	**Probability**
Chi-Square	209	1176.5832	<.0001
Cramer's V		0.1313	

All the tables above have very hi chi-square value and the p values are really low; therefore it shows association of the visualization types with the temporal factors(years, and months) with respect to the comment counts. I can therefore say the comment counts of certain visualization types depend on the temporal factors. With the value of Cramer's V in the tables above I can't depict very strong association as it is closer to zero than one.

To test my analysis result I also executed both unsaturated and saturated loglinear models on the same data on SAS using CATMOD procedure and the unsaturated model fits, which again reflects the association as well.

B. Comment Data

As like the visualization data here are also 31 different types of visualizations and out of which 12 are just specific different type of maps whose counts are very low so they are merged together into 'Other Map' category and therefore I have 20 different types of visualizations.

There were 33258 comments retrieved. Similar to the visualization data analysis, I performed the contingency table analysis of the 20 types of visualization data with four different years (2007-2010), twelve different months and seven different day types of a week. The cells of each table contain the frequency of counts in that category. Running FREQ procedure with specific options I found the following for the above data.

To test my analysis result I also executed unsaturated loglinear models on the same data on SAS using CATMOD procedure and that guarantees the association as well.

Table 9. Chi-Square analysis of year wise comment counts on different visualization types

Statistic	**DF**	**Value**	**Probability**
Chi-Square	90	11139.7972	<.0001
Cramer's V		0.3341	

Table 10. Chi-Square analysis of month wise creation of different visualization types

Statistic	**DF**	**Value**	**Probability**
Chi-Square	209	18526.4748	<.0001
Cramer's V		0.2250	

Table 11. Chi-Square analysis of day type wise creation of different visualization types

Statistic	DF	Value	Probability
Chi-Square	114	4129.3310	<.0001
Cramer's V		0.1414	

All the tables above have very high chi-square value and the p-values are really low; therefore it shows association of the visualization types with the temporal factors(years, months and day types) with respect to the comment counts. I can therefore say the comment counts of certain visualization types depend on the temporal factors. With the value of Cramer's V in the tables above I can't depict very strong association as it more closer tro zero than one and definitely at the day type of the week level: I can say that the comment counts on the certain visualization doesn't depend that strongly on the different days of the week, so it's little arbitrary there.

To test my analysis result I also executed both unsaturated and saturated loglinear models on the same data on SAS using CATMOD procedure and the unsaturated model fits, which again reflects the association as well; following are the tables for the maximum likelihood analysis of variance for the log-linear models.

Table 12. Maximum Likelyhood Analysis of Variance for Unsaturated Loglinear Model of Year Effect

Source	DF	Chi-Sq	Pr > Chi-Sq
visType	19	12960.90	<.0001
year	3	2066.43	<.0001
Likelihood Ratio	54	8698.22	<.0001

Table 13. Maximum Likelyhood Analysis of Variance for Unsaturated Loglinear Model of Day Type Effect

Source	**DF**	**Chi-Sq**	**Pr > Chi-Sq**
visType	19	14129.54	<.0001
dayType	6	1414.75	<.0001
Likelihood Ratio	114	3907.82	<.0001

I saw there are total of 45 different comment counts starting from 1 to 45 and I performed contingency table analysis using FREQ procedure in SAS for these 45 different comment count categories and 4 different years, following is the outcome of that.

Table 14. Chi-Square analysis of year wise comment count categories

Statistic	**DF**	**Value**	**Probability**
Chi-Square	513	10710.4655	<.0001
Cramer's V		0.31	

The result here also shows association between these factors and the Cramer's V value shows little higher association direction than the earlier factors.

I also performed the loglinear modeling using SAS CATMOD procedure on the two categorical variables where I justify the association quite well too. Following is my findings from the modeling, here I show the result of the unsaturated model as it outperforms the saturated one:

Table 15. Maximum Likelyhood Analysis of Variance for Loglinear Model

Source	**DF**	**Chi-Sq**	**Pr > Chi-Sq**
countCategory	27	3531.58	<.0001
visType	19	1252.35	<.0001
Likelihood Ratio	143	538.45	<.0001

Here in the model it shows both ‘countCategory’ and ‘visType’ are significant as the probability is so low for the Wald test [133].

C. System Generated Tag Analysis

I also concentrated my focus onto the system generated tags on the visualizations. I wanted to explore if they have any relationship with the visualization types, i.e. I wanted to know if for certain tags correspond to certain visualization types. In this sector I had 20 visualization types but 2519 different system generated tags (which are possibly the collection of terms from the visualization title or so) and the contingency table has very low count distributed frequencies with total sample counts of 5156. I performed the contingency table analysis for this categorical variables using SAS FREQ procedure and later used the loglinear modeling (employed an unsaturated model, as it outperforms the saturated one) using the CATMOD procedure as well.

Table 16. Chi-Square analysis of System Generated Tags and Visualization Types

Statistic	**DF**	**Value**	**Probability**
Chi-Square	54607	105885	<.0001
Cramer's V		0.84151	

Table 17. Maximum Likelyhood Analysis of Variance for Loglinear Model

Source	**DF**	**Chi-Sq**	**Pr > Chi-Sq**
visTag	2E3	4159.19	<.0001
visType	26	175.34	<.0001
Likelihood Ratio	609	811.29	<.0001

The noticeable fact from the above is the Cramer's V value for the association. Here it is very close to 1, so reflects a strong association: so the system tags have a strong associative relationship with the visualization types.

4.2.3 Conclusion

I have presented some explorative analysis of the user interaction in online social visualization web site Many Eyes. As this online social data visualization service has grabbed quite an attention around us and multiple such tools has been launched lately, the interest of analysis in this field is potentially quite strong. The findings from the explorative data analysis from Many Eyes could be used for future guidelines of improvement for such online social visualization services.

My explorative analysis produced the following primary findings.

- The temporal factors such as the years and months have association with the type of visualizations created.
- The temporal factors such as the years, months and day type of the week have association with the type of visualizations in terms of comments.
- The temporal factors such as the years and have association to the different category of comments, considering each category representing certain number of comments. .
- The system generated tags have strong association with the visualization types.

The overall attention for Many Eyes is to provide the ease amongst users to interact, share and create data visualizations. The success of a social media web site relies heavily on a relatively small group of very active contributors. Also the temporal factors show a direction for the types of popular charts at different temporal points. So this could be taken into account and some improvement or focus could be employed on those to get more user attention. As the earlier analysis by Zhu et al. [91] provides some points to the most popular type of charts or mostly commented charts; those information

could be interested with this analytics and some more analytical aspects could be explored to pin down the user's attention and focus in such collaborative environment.

As a matter of fact some multifactor and constrained based multi dimensional contingency table analysis in this regard could result some more interesting pattern in user behavior in such a service platform.

5 ANALYZING THE SEMANTIC ASPECT OF UNSTRUCTURED ENGLISH TEXTS

The semantic aspect of a textual composition deals with the underlying meaning of the texts. As normal English texts are is one kind of a natural human language, it contains extreme ambiguities and fuzziness in its compositional meaning as usual. Nowadays people freely express themselves with such textual snippets in online social networks more often than anything else. Semantic exploration of such texts from online social network carries a tremendous promise in this regard.

5.1 Analyzing Social Collaborative Visualization using LSA

Enabled by Web 2.0 and Cloud Computing technologies, social media web sites have become an important part of our daily lives. These social media web sites succeed because they effectively facilitate communication and collaboration among large groups of people. For example, users can easily post comments and respond to each other's comments. However, the explosion of the social media web sites also leads to information overloading. To understand and analyze the huge amount of user generated contents, I need effective computing tools.

Here I present my study of a social visualization web site using Latent Semantic Analysis (LSA) [66, 75, 76, 79]. LSA is a statistical method for extracting and representing the contextual meaning of words. The main idea behind LSA is that the entire word collection of the text corpus provides mathematical clues that can help determine the similarities of the words' meanings [66, 75]. LSA has passage-based coherence and can handle noisy data.

The target of my study is Many Eyes [85], a social visualization web site and also an IBM research project. Many Eyes allows users to upload their data, construct data visualizations, share the data visualization with others, and comment on data visualizations. I chose Many Eyes because it is one of the first social visualization web sites and by far the most popular. Between 2007 and 2010, there are over 8,000 user created data visualizations and over 10,000 comments. Analyzing these visualizations

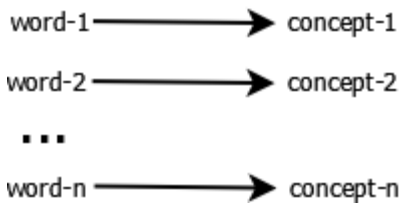
and comments can provide insights into the patterns and trends of user generated visualizations. These insights can lead to better visualization tools. The presented study is the first attempt to provide such an analysis.

My study shows that LSA is effective for analyzing and classifying user comments based on their semantic meanings. My method can help identify the most relevant comments and potential spam. The technique presented here is useful for developing effective search engines that automatically retrieve the most relevant user comments. It's also useful for developing spam filters that identify and block irrelevant comments.

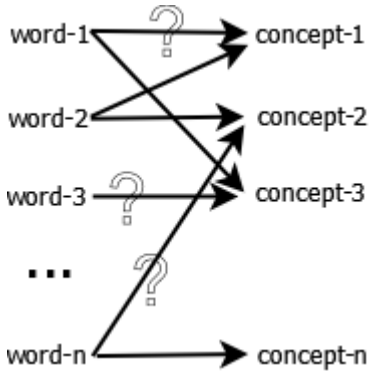
5.1.1 Methods

Latent Semantic Analysis

Latent Semantic Analysis (LSA) [66, 75, 76, 79], also known as Latent Semantic Indexing (LSI) literally means analyzing documents to find the underlying meaning or concepts of those documents. If each word only meant one concept, and each concept was only described by one word, then LSA would be easy since there is a simple mapping from words to concepts [131].



Unfortunately, this problem is difficult because English has different words that mean the same thing (synonyms), words with multiple meanings, and all sorts of ambiguities that obscure the concepts to the point where even people can have a hard time understanding [131].



For example, the word bank when used together with mortgage, loans, and rates probably means a financial institution. However, the word bank when used together with lures, casting, and fish probably means a stream or river bank.

Latent Semantic Analysis arose from the problem of how to find relevant documents from search words. The fundamental difficulty arises when I compare *words* to find relevant documents, because what I really want to do is compare the *meanings or concepts behind the words*. LSA attempts to solve this problem by mapping both words and documents into a "concept" space and doing the comparison in this space.

Since authors have a wide choice of words available when they write, the concepts can be obscured due to different word choices from different authors. This essentially random choice of words introduces noise into the word-concept relationship. Latent Semantic Analysis filters out some of this noise and also attempts to find the smallest set of concepts that spans all the documents.

In order to make this difficult problem solvable, LSA introduces some dramatic simplifications [131].

1. Documents are represented as "bags of words", where the order of the words in a document is not important, only how many times each word appears in a document.

2. Concepts are represented as patterns of words that usually appear together in documents. For example "leash", "treat", and "obey" might usually appear in documents about dog training.

3. Words are assumed to have only one meaning. This is clearly not the case (banks could be river banks or financial banks) but it makes the problem tractable.

LSA works through Singular Value Decomposition (SVD) on the word-document count matrix/ tf-idf(term frequency inverse document frequency) matrix, which is described below.

Singular Value Decomposition (SVD) :

Given a matrix M, it's SVD is found as

$M = U \times S \times V(T)$

Where U is the matrix of left singular vectors,

S is the diagonal matrix of singular values and

V(T) is the transpose of the matrix of right singular values

After the decomposition a reduced dimension of the matrix is considered to reconstruct the sense of the document by eliminating potential noise in text. The dimensionality reduction is then performed by discarding all but the top few hundred singular values.

To comprehend LSA, the following example is provided.

A. LSA Example [131]

As a small example, a search for books using the word "investing" is done at Amazon.com and the top 10 book titles that appeared there is taken. One of these titles was dropped because it had only one index word in common with the other titles. An index word is any word that:

- appears in 2 or more titles, and
- is not a very common word such as "and", "the", and so on (known as stop words). These words are not included because do not contribute much (if any) meaning.

In this example I have removed the following stop words: “and”, “edition”, “for”, “in”, “little”, “of”, “the”, “to”.

Here are the 9 remaining tiles. The index words (words that appear in 2 or more titles and are not stop words) are underlined.

1. The Neatest Little Guide to Stock Market Investing
2. Investing For Dummies, 4th Edition
3. The Little Book of Common Sense Investing: The Only Way to Guarantee Your Fair Share of Stock Market Returns
4. The Little Book of Value Investing
5. Value Investing: From Graham to Buffett and Beyond
6. Rich Dad's Guide to Investing: What the Rich Invest in, That the Poor and the Middle Class Do Not!
7. Investing in Real Estate, 5th Edition
8. Stock Investing For Dummies
9. Rich Dad's Advisors: The ABC's of Real Estate Investing: The Secrets of Finding Hidden Profits Most Investors Miss

Next the index word by title matrix is built. In the following matrix, I have left out the 0's to reduce clutter.

Index Words	Titles								
	T1	T2	T3	T4	T5	T6	T7	T8	T9
book			1	1					
dads						1			1
dummies		1						1	
estate							1		1
guide	1					1			
investing	1	1	1	1	1	1	1	1	1
market	1		1						
real							1		1
rich						2			1
stock	1		1					1	
value				1	1				

Figure 21. Word by title matrix for the aforementioned example [131]

Here is the complete 3 dimensional Singular Value Decomposition of my matrix. Each word has 3 numbers associated with it, one for each dimension. The first number tends to correspond to the number of times that word appears in all titles and is not as informative as the second and third dimensions, as I

discussed. Similarly, each title also has 3 numbers associated with it, one for each dimension. Once again, the first dimension is not very interesting because it tends to correspond to the number of words in the title.

book	0.15	-0.27	0.04
dads	0.24	0.38	-0.09
dummies	0.13	-0.17	0.07
estate	0.18	0.19	0.45
guide	0.22	0.09	-0.46
investing	0.74	-0.21	0.21
market	0.18	-0.30	-0.28
real	0.18	0.19	0.45
rich	0.36	0.59	-0.34
stock	0.25	-0.42	-0.28
value	0.12	-0.14	0.23

*

3.91	0	0
0	2.61	0
0	0	2.00

*

T1	T2	T3	T4	T5	T6	T7	T8	T9
0.35	0.22	0.34	0.26	0.22	0.49	0.28	0.29	0.44
-0.32	-0.15	-0.46	-0.24	-0.14	0.55	0.07	-0.31	0.44
-0.41	0.14	-0.16	0.25	0.22	-0.51	0.55	0.00	0.34

Figure 22. SVD decomposition of the word by title matrix of figure 21 [131]

Leaving out the first dimension, as discussed, the figure 23 is drawn using the second and third dimensions using a XY graph. We'll put the second dimension on the X axis and the third dimension on the Y axis and graph each word and title. It's interesting to compare the XY graph with the table I just created that clusters the documents.

In figure 23, words are represented by red squares and titles are represented by blue circles. For example the word "book" has dimension values (0.15, -0.27, 0.04). I ignore the first dimension value 0.15 and graph "book" to position (x = -0.27, y = 0.04) as can be seen in the graph. Titles are similarly graphed.

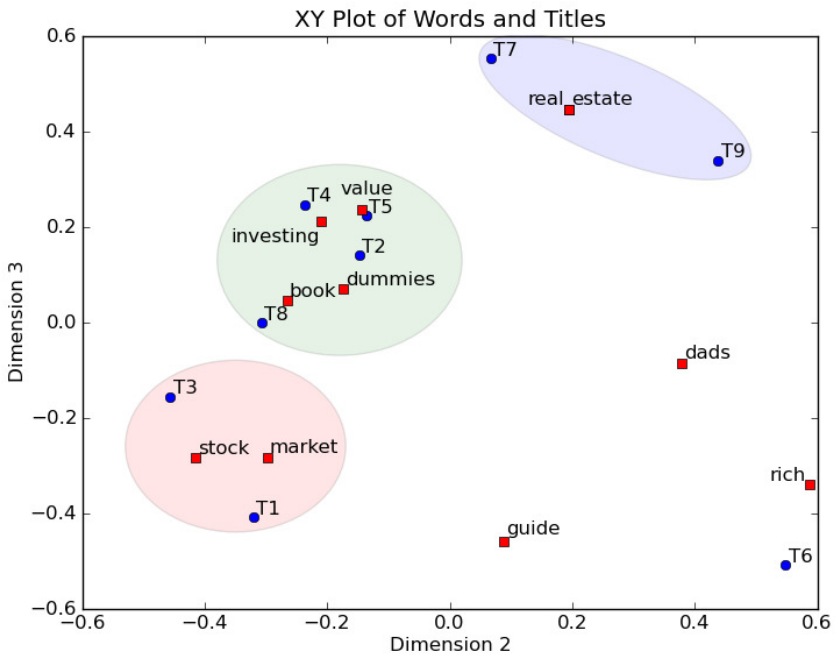


Figure 23. Word by title distribution from the aforementioned example [131]

B. Modify the counts by TFIDF

In sophisticated Latent Semantic Analysis systems, the raw matrix counts are usually modified so that rare words are weighted more heavily than common words. For example, a word that occurs in only 5% of the documents should probably be weighted more heavily than a word that occurs in 90% of the documents. The most popular weighting is TFIDF (Term Frequency - Inverse Document Frequency). Under this method, the count in each cell is replaced by the following formula [131].

$TFIDF_{i,j} = (N_{i,j} / N_{*,j}) * \log(D / D_i)$ where

- $N_{i,j}$ = the number of times word i appears in document j (the original cell count).
- $N_{*,j}$ = the number of total words in document j (just add the counts in column j).

- D = the number of documents (the number of columns).
- D_i = the number of documents in which word i appears (the number of non-zero columns in row i).

In this formula, words that concentrate in certain documents are emphasized (by the $N_{i,j}$ / $N_{*,j}$ ratio) and words that only appear in a few documents are also emphasized (by the log(D / D_i) term).

5.1.2 Implementation and Results

My implementation of LSA starts by creating a database of the words retrieved from the user comments on Many Eyes website. The text is formatted slightly to form lexemes and regular expressions are employed to determine the total number of objects in the title arrays for comparison. Pairings of the subject titles and comments are generated and the database is updated. The term frequency is calculated by obtaining the number of times a word occurs in a document (i.e. all the comments for a particular visualization). To distinguish words from other documents, I alternatively count the number of times each term occurs in each document and sum them. The second part of the term frequency is the inverse document frequency, which diminishes the weight of words occurring very frequently in the word bank. Together, the term frequency (TF) and inverse document frequency (IDF) help normalize the weight of infrequently occurring words. The results are stored in a matrix and visualized in a sparse plot with density markings indicating the rate of collaboration between authors on articles.

I visualize the result of my analysis using sparse graphs with an adapted visual marker technique [51]. Sparse graphs are effective for visualizing dense data and exploring visual trends in text data, such as blogging and online comments. With such visualization, I am able to identify comments that are most closely related to the subject of the visualization. Figure 24 shows the plot of comments on the data visualization: “US Government Expenses 1962-2004”. The original visualization subject line is the red line that is crowded with densely populated comments. In the figure, the semantic closeness of the comments to the subject is represented by the slopes of the line segments that connect the marker and the

origin. Further scaling of the plot can reveal the most closely related comments. In this case, C3 is the comment that is the most relevant to the subject of this visualization.

Figure 25 visualizes the semantic analysis of comments made to the visualization: “World Map of Social Networks in June 2009”. As seen in Table 18 and Figure 25, C9 is the most relevant comment to the subject line. The slopes of C6, C8, C12 are also close to the baseline and their similar word counts are also high. Of the total 11 words in the baseline subject title, the highest TF-IDF belongs to C9, which is the most closely related comment. On the other hand, comments C3 and C4 are very distant from the subject in terms of their semantic meanings.

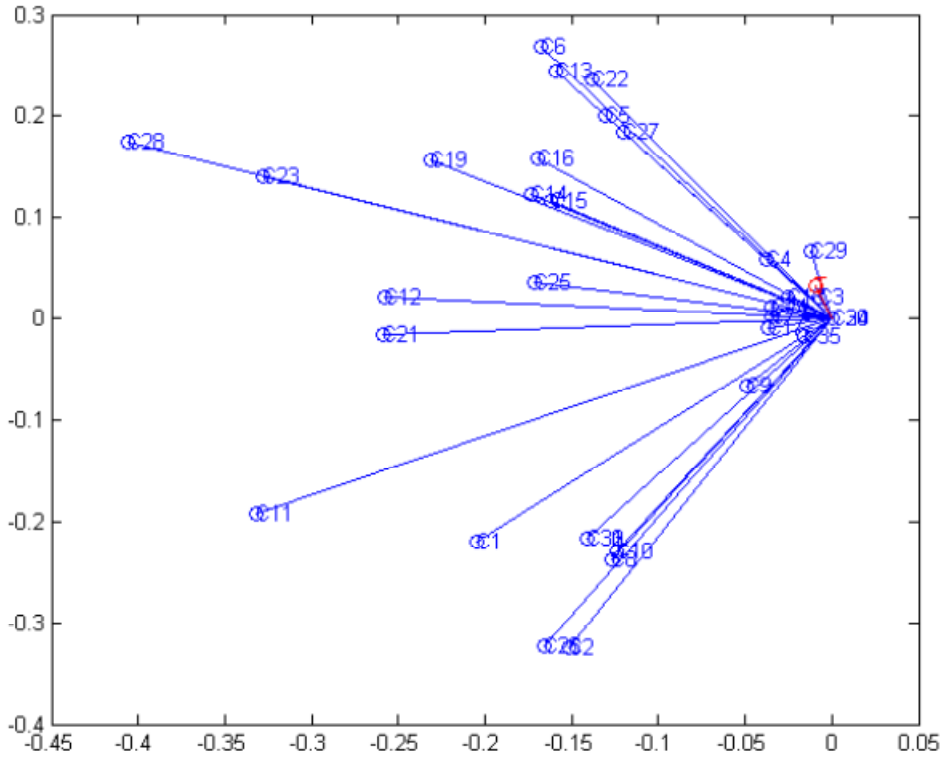


Figure 24. US Government Expenses 1962-2004. The red line is the subject line or baseline. C3 is the most relevant comment.

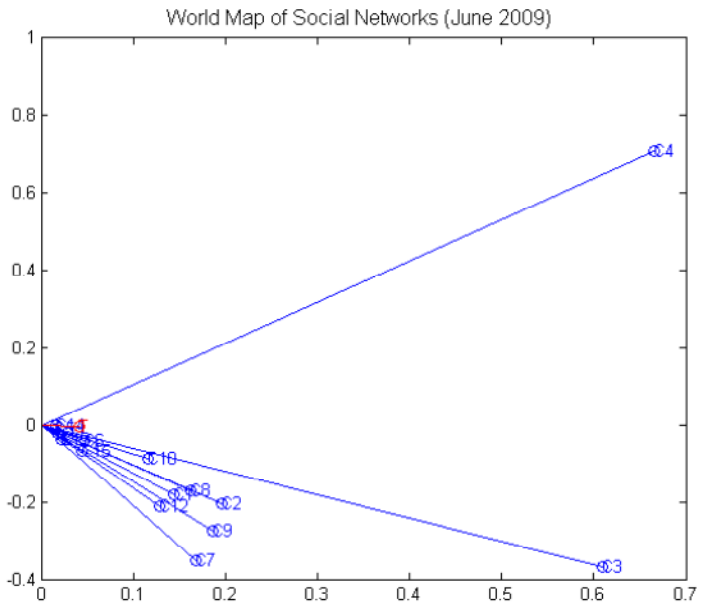


Figure 25. World Map of Social Networks in June 2009.

Figure 26 shows a stark comparison of the subject and comments. The visualization in question features comments that are unrelated to the baseline marker. Hence, the slopes of those lines do not match and I conclude that the disparate marker is spam, too little data, or outliers. Closer inspection of the data revealed that the comment is indeed Spam.

Table 18. World Map of Social Networks in June 2009. The total words in the article and the closely related comments are shown.

World Map of Social Networks (June 2009)

	Total Words	Words in Common	Percent in Common
C1	41	3	7
C2	25	0	0
C3	86	5	6
C4	107	8	7
C5	11	0	0
C6	21	2	10
C7	49	3	6
C8	39	3	8
C9	26	5	19
C10	30	1	3
C11	12	0	0
C12	23	2	9
C13	7	0	0
C14	11	0	0
C15	28	2	7
T	11	11	100

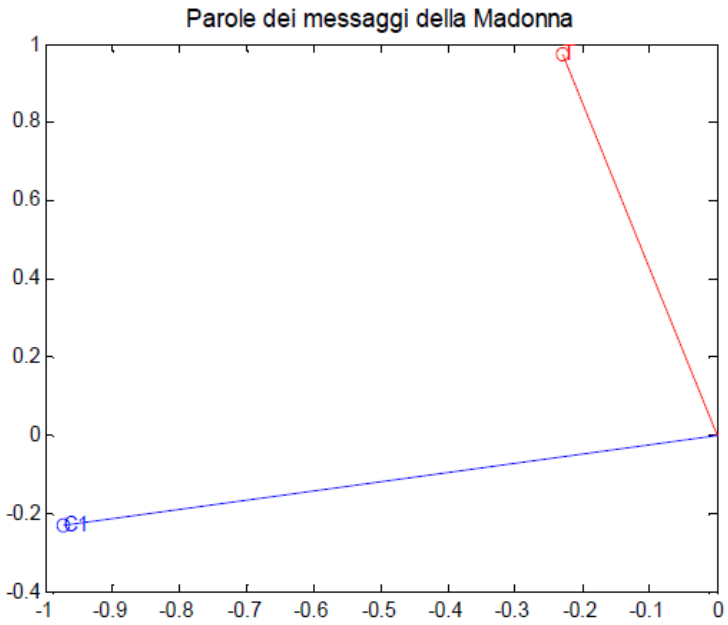


Figure 26. Parole dei messaggi della Madonna. Spam identification via disparate slopes and comment magnitude.

My analysis shows a strong correlation between words and phrases in user comments. Using the frequency of the comments and the number of words in common between the baselines and comments, I can establish basic trends in user comments. In general, data visualizations with many comments demonstrate strong collaboration between authors. However, some data visualizations have fewer but strongly correlated comments due to the quality and depth of the semantic meaning contained within such data.

Though I am able to capture the semantics in word pairings, my method has its limitations. For instance, polysemous words are not always properly accounted for due to the limitation of LSA. Another limitation is that the order of the words and sentence grammar is ignored when forming the word pairings. As a result, the subtle semantic meaning of some words may be lost in the analysis, particularly in dealing with long sentences.

5.1.3 Conclusion

I have discussed my method and results for analyzing user comments on a social media web site: Many Eyes. Based on the traditional LSA techniques, I optimized the term-frequency relations and developed a robust co-occurrence matrix and TF-IDF solution. I have used my method to analyze several thousand comments on Many Eyes. Based on the co-occurrence frequencies, I am able to identify the most relevant comments and potential spam. My method is useful for developing effective search engines that automatically retrieve the most relevant user comments. It's also useful for developing spam filters that identify and block irrelevant comments.

My experiments showed that LSA is effective for analyzing and classifying user comments based on their semantic meanings. The specific variant of LSA that I propose, designed for relatively small data size, is particularly useful for such application due to its efficiency. The proposed method can be readily applied to other social media web sites as well.

5.2 Mining Collaboration through Textual Semantic Interpretation

In recent days collaboration is the most and everywhere happening thing. Collaboration is a deep, collective, determination to reach an identical objective in recursive manner where two or more people or organizations work together to realize shared goals. Practically collaboration happens through sharing knowledge, learning and building consensus. Communication is a key aspect in collaboration. The greatest and broadest means of communication happens in today's world through internet. So while we communicate with a collaborative goal, how much do we achieve in that direction? Is there any direction so that we can achieve better? Do we at least understand the components which express non-collaboration? Even if we achieved some, can we do it better? These are pretty reasonable questions to answer as we are so much into this.

A typical social collaboration happens through communication in natural language expressions.

In the online media people come and collaborate on the topic of their interest generally through textual snippets written in natural languages as well. Therefore to disclose the mystery of collaboration natural language processing demands a darn importance. In this domain though syntactic and semantic understanding and analysis of the natural texts can lead the way to success but the later carry more weight as it provides the inner meaning of the whole composition.

In semantic textual analysis multiple directions has been approached based on training methods, mathematical techniques but direct component (terms/words) based analysis has still rare/limited success and attention. Therefore incorporating such approach based upon the textual elements of a composition i.e. the word or terms would be really interesting. Of course while taking such approach the limitation of the components as a part of the whole text has to be considered carefully through multiple angles; just for example word sense disambiguation, which tries to find the sense of the word used in a composition as words could have multifold meanings at different contexts.

To justify such textual similarity based measure upon collaboration analytics, a choice of appropriate collaboration platform is important. My work took the attention of online social data visualization hype as this allows users to upload to upload their data to a server, construct data visualizations, and publish or share the data visualizations with comments: which gives a perfect and complete object of analysis in this field. As the online data visualization tools utilizes the advantages of cloud computing, so that gives the user more comfort in using the application: helps to accumulate more user data, which is great for such analysis. In such analysis along with quantitative metrics as a figure of merit, creative visualizations always enhance the understanding and diagnosis.

5.2.1 Methods

A. Semantic Analysis

Given two textual comments on a visualization, I want to find how collaborative they are: i.e. given two textual compositions I want to find out a score that indicate their semantic level similarity, not usual syntactic level lexical matching or so. Ideally a comprehensive metric of text semantic similarity should be based upon the relation between the words in addition to the role played by the various entities involved in the interactions described by each of the two texts. Following this the semantic similarity of textual components are based upon the similarity of the component words in them. So the overall textual similarity is chosen to be based upon the word to word similarity and potential language models.

I) WordNet:

WordNet [122] is like a dictionary in that it stores words and their meanings, but differs from the traditional ones. Words in WordNet are arranged semantically instead of alphabetically for example. Synonymous words representing single distinct sense are gathered together there to form synonym sets, or synsets. For instance, the synset {base, alkali} represents “the sense of any of various water-soluble compounds capable of turning litmus blue and reacting with an acid to form a salt and water”.

Monosemous words or the words with one sense (e.g. ‘wristwatch’) appear in only one synset in the WordNet, but words with multiple senses (homonymous or polysemous words) are present in multiple synsets. For instance the word base occurs in two noun synsets,{ base, alkali} and{ basis, base, foundation, fundament, groundwork, cornerstone} , and the verb synset { establish, base, ground, found}. WordNet also stores information regarding the parts-of-speech of the words: nouns, verbs, adjectives and adverbs.

Along with single words WordNet synsets contains compound words consisting of two or more words but are treated like single words in concepts (e.g. “banking concern”, “depository financial Insti-

tution" etc.) [136]. Every synset in WordNet consist of a short entry, called definition or gloss, which explains the meaning of the concept represented by the synset (e.g. the synset { basis, base, foundation, fundament, groundwork, cornerstone} defines "lowest support of a structure").

WordNet also consists of an array of semantic and lexical relations between words and synsets. While semantic relations define a relationship between two synsets(e.g. the noun synset { robin, redbreast, robin redbreast} is related to the noun synset {bird} through the IS–A semantic relation since a robin is a kind of a bird), lexical relations describe a relationship between two specific words within two synsets (e.g. the 'antonymy' relation relates the words 'embarkation' and 'disembarkation' but not the rest of the words in their respective synsets which are {boarding, embarkation, embarkment} and {debarkation, disembarkation, disembarkment}). Following Figure 27 [135] shows an example of semantic relationship amongst synsets.

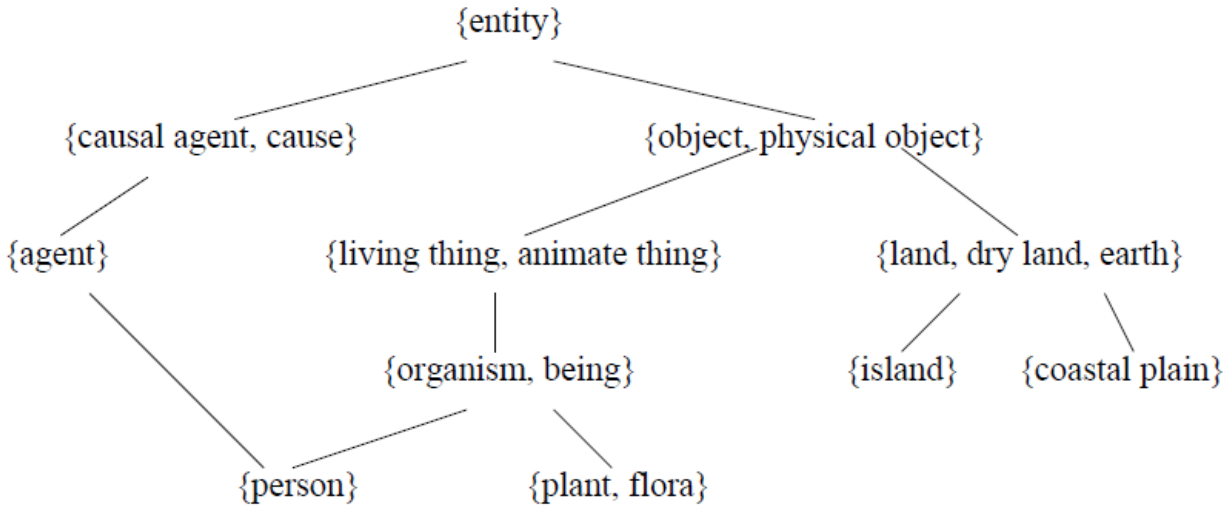


Figure 27. Sample WordNet Noun Taxonomy [135]

Now in a semantically related taxonomy such as in WordNet, a simple and practical approach to find similarity happens through presenting the taxonomy as an undirected graph and define the meas-

ure of similarity as the distance in terms of path length between the two synsets. The lesser the distance between two synsets, the more similar they are (e. g. in Figure 27 the synset {island} is closer/similar to {land, dryland, earth} than it is to {living thing, animate thing}). The similarity between synsets s1 and s2 is defined as $Sim_{(s1,s2)= 1/dist_{(s1,s2)}}$ [135], while $dist_{(s1,s2)}$ is the distance between them (either counted node wise or edge wise).

Therefore employing the node count, the distance between {person} and {object, physical object} is 4, so the similarity score is 1/4.

The depth of a synset, is a similar concept as distance, is simply the distance between that synset and the root of the taxonomy in which the synset is located. A shared parent of two synsets in the taxonomy is called a subsumer. The Least Common Subsumer (LCS) of two synsets is the subsumer that does not have any children that are also subsumers of the two synsets (e.g. in Figure 27 the subsumers of {living thing, animate thing} and {land, dry land, earth} are {object, physical object} and {entity} and the LCS of {object, physical object} since this is more specific than {entity} [135].

II) Word Level Semantic Similarity:

There exists numerous different word-to-word similarity metrics: ranging from distance based measures computed on semantic networks to metrics based on models of distributional similarity learned from large text collections. Amongst all these I concentrated upon six different metrics as they outperform in the domain of natural language processing (e.g. word sense disambiguation etc. Patwardhan et al. 2003 [140]) and relatively computationally efficient as well. In the following I describe them in brief. Now the metrics below described mostly based on concept to concept which could be easily mapped to word to word by following the concept behind the words. All the metrics mentioned are implemented using WordNet::Similarity package (Patwardhan et al., 2003 [140]).

i) Lesk:

The original Lesk algorithm [99] is used in disambiguating words in short phrases. Using Lesk algorithm a word's dictionary definition or gloss of each of its senses is compared to the glosses of every other word in the phrase and the largest match becomes the winner for the sense of the word.

ii) Leacock & Chodorow:

Here similarity [123] is defined as [121], $Sim_{lch} = -\log(\frac{dist_{(s1,s2)}}{2D})$,

where $dist_{(s1,s2)}$ is the length of the shortest path between two concepts/synsets s1 and s2 using node-counting, and *D* is the maximum depth of the taxonomy.

iii) Wu and Palmer:

This similarity metric [124] measures the depth of the two concepts/synsets in the WordNet taxonomy, and the depth of the least common subsumer (LCS) to find it's similarity score as [121]:

$$Sim_{wup} = \frac{2 * depth(LCS)}{depth(s_1) + depth(s_2)}$$

iv) Resnik:

This measure [70] returns the information content (IC) of the LCS of two concepts [121]:

$$Sim_{res} = IC(LCS)$$

where IC is defined as [121]:

$$IC = -logP(c)$$

Where and P(c) is the probability of encountering an instance of concept c in a large corpus.

v) Lin:

This measure [125] works through normalizing Resnik's [70] measure, and adds a normalization factor consisting of the information content of the two input concepts [121]:

$$Sim_{lin} = \frac{2 * IC(LCS)}{IC(s_1) + IC(s_2)}$$

vi) Jiang & Conrath :

This measure [126] finds information concept based semantic distance as [121],

$$SemDist_{jcn} = IC(s_1) + IC(s_2) - 2IC(LCS(s1, s2))$$

And the similarity measure as [121], $Sim_{jcn} = {}^{1}/_{SemDist_{jcn}}$

III) Language Models:

Other than the semantic similarity of the words, specificity of a word also accounts in the understanding of the natural language based text. Actually this emphasizes the higher weight of semantic matching of very specific words (e.g. 'hound' and 'whippet'), and lower importance to generic concept oriented word (e.g. 'go' and 'be'). Apparently the specificity of a word is addressed partially by the hierarchical taxonomy of semantic dictionaries, but the factor could be reinforced by incorporating a corpus based specificity measure, more specifically the distributional frequency of the word.

The frequency of words might not be a good measure of word importance but the distribution of words across an entire collection can be a good indicator of the specificity of the words. Words or terms occurring in few documents with higher frequency carry a distinguished factor in the understanding of the document compared to the terms appearing in every documents of the collection. Incorporation of the inverse document frequency introduced in [127], which is defined as the total number of documents in the corpus, divided by the total number of documents that include that word, is a good one for the enhancement of specificity of words in documents.

IV) Semantic Similarity of Text:

Incorporating the semantic similarity between words and words specificity Corley et al developed a semantic similarity measures between textual compositions. The measure works through pairing up those words that are found to be most similar to each other, and weighting their similarity with the

corresponding specificity score. The methods works by first creating sets of open-class words in the text, with a separate set created for nouns, verbs, adjectives, and adverbs. Using parts of speech tagger the text is basically tagged to different parts of speech and cardinal numbers are also tagged as well. Next the pairs of similar words across the sets corresponding to the same open-class in the two text segments are found. For noun and verb class the WordNet [122] based semantic similarity is measured in this technique and for other class the lexical matching is employed. Now I consider that if I cannot associate the other classes (e.g adjectives, adverbs) with the nouns and verb class properly then a lexical matching could result in the wrong direction (e.g. between “He is very good” and “He is very bad” if I cannot take account where ‘very’ is associating, then it could reflect higher score with lexical matching which is reverse of original semantic understanding); Therefore I only consider the noun and verb class pair matching here. For each noun (verb) in the set of nouns (verbs) belonging to one of the text segments, I choose the noun (verb) in the other text segment that has the highest semantic similarity (maxSim) is identified, using one of the six word similarity measures described earlier. If this similarity measure turns out a positive value, then the word is added to the set of similar words for the corresponding word class WSpos. In this technique the textual similarity of Ti with Tj is found using the scoring function [121],

$$Sim(T_i, T_j)_{T_i} = \frac{\sum_{pos}(\sum_{w_k \in \{WS_{pos}\}}(maxSim(w_k) * IDF_{w_k}))}{\sum_{w_k \in \{T_{i_{pos}}\}} IDF_{w_k}}$$

This score provides directional similarity for Ti (values ranging from 0 to 1)and similarly the directional similarity of Tj could be fund as well. The scores from both directions are combined into a bidirectional similarity using a simple average function [121] as:

$$Sim(T_i, T_j) = \frac{Sim(T_i, T_j)_{T_i} + Sim(T_i, T_j)_{T_j}}{2}$$

B. Visualization of Collaboration

The goal of the collaboration visualization is to visualize the strength and pattern of collaboration. Now I am trying to map the strength of collaboration in terms of semantic similarity of textual snippets in an online social collaborative visualization environment. Therefore a network graph containing all the users collaborating for one issue/topic would be interesting while the connection of the network strength depicts the similarity or the connection of the users in terms of collaboration. So basically if the network strength in terms of similarity measure is mapped to the edge attributes of the network would be interesting.

To employ such network graph I chose At&T research Lab's current popular tool in the market: Graphviz [128]. Graphviz is network visualization software for creating high-quality, readable node-link diagrams of large-scale data sets, brought in by the focus of John Ellson [129] of At&T recently. Graphviz works through a graph description language based on textual notation named the DOT [130] language and it has set of tools that can generate and/or process DOT files to produce visualizations.

5.2.2 Implementation and Results

A. Collection of Data

I chose IBM's collaborative data visualization tool Many Eyes [12][13] as the data repository. Users are allowed to upload data and then produce graphic representations on this website for others to view and collaborate through commenting upon the visualizations. Although a user can browse or search the data visualizations at Many Eyes [12][13], but collecting data from the site is a cumbersome process as it does not provide a downloadable database of all the visualizations. The web site does provide a RSS feed that broadcasts every new comment on the created visualizations. However, at the time of my experiment I collected data until March 2010. I used a Web data extraction tool to retrieve the

data page by page. Here for analysis I collected each visualization data along with the comments associated with them and the user information as well.

B. Cleaning and Preparation of Data

Using the web extraction tool the data is stored in a MS SQL server database. Multimodal queries are executed to find the user and their visualization and comments information. The web extraction tool collected the comments as string composed of the user and timestamp information. Therefore the string clean and up and processing queries are also executed to find the raw comments along with it's identifying user and timestamp information. Due to the availability of free form of collaborative writing quite some user data is found using spoken languages, symbols like smileys, use of non English platforms etc; so the final comment data has been manually cleaned up to be processed by the semantic analysis tool (described below) for the experiment.

C. Semantic Analysis

The comments data is semantically analyzed after being preprocessed. As the comments data is expressed in Natural language (English predominantly) it's semantic understanding and analysis is done using the word sense disambiguation. Using word sense disambiguation the identity and probable sense of the words present in a comment is found. All the word metrics are computed utilizing the WordNet repository. Later word sense based textual similarity between two textual comments of visualization is computed. Using this textual semantic similarity of multiple comments present in visualizations, the collaboration metric of IBM's ManyEyes has been explored. All the computations have been performed using Perl.

D. Data Preparation for Visualization

The similarity measures found were properly normalized first to aid the visualization come appropriate. Before the normalization the textual similarities calculated comes in the range of 0 to 1; the

ones closer to 1 signifies more similarity. In the normalization process these scores has been first mapped to a different range of 1-10 with one decimal approximation and then the scores has been reversed, i.e. higher the scores are now represent they are less similar (i.e. now they represent distance in terms of similarity). These distance measures would be now directly mapped to the edge length of the network graphs to be produced. As I used Graphviz to implement the commenting user's network using comment similarity, all the measures are transformed to the attributes of the network edges using the DOT language for processing. All these transformations are done using perl. These visualizations are very graphical in nature and provides a whole snapshot of the fully connected network in terms of the connection strengths as the length of the network edges.

Also the selection of such similarity measures has been taken to generate a quantitative visualization using MS-Excel.

I concentrated my analysis on the visualization where multiple users came and provided comments on a single visualization to understand collaboration. One of the shortcomings of such web service is if the user's decide to not register any identification and interact using anonymous or defaults user identity then it's little tough to understand the collaboration: this could be a good case for employing the full proof model to identify clusters of same identity users using similarity clusters though, once a full proof model is present.

Now in most of the visualization I found higher weight age of anonymous users, which represents the most users are casual visitors and they do not want to waste time by creating identity, instead they just want to provide their comment; therefore an identity creation driven directional goal should be implemented by such service to get more richer collaboration pictures. Now due to this I found some 19 commented visualizations have the more identified diverse users and they commented more than once in occasions; therefore I chose my attention on the analysis of such visualization profiles first. Following is

the user comment distribution of one such 19 commented visualization in IBM's Many Eyes, the one I chose to analyze next.

Table 19. Commenter Distribution in a 19 Commented Visualization in Many Eyes

User Identity	**Number of Times Commented**
Bernie_Hogan	2
Anonymous	3
Colm	4
Jbw	1
Sammy54	1
Martin_Wattenberg	1
Jacket	4
Fernanda_B_Viegas	1
Karim	1
Frank_van_Ham	1

Following are the six Graphviz visualizations (Fig. 28-33) generated of the 19 commenter's textual similarity distribution wise network graph. In these graphs the lower the length of the edges between the nodes i.e. the closer the nodes are in the network, they are more similar. These graphs show the whole network with the apparent idea of similarity difference; in case of closer similarity comparison these graphs are not that helpful though.

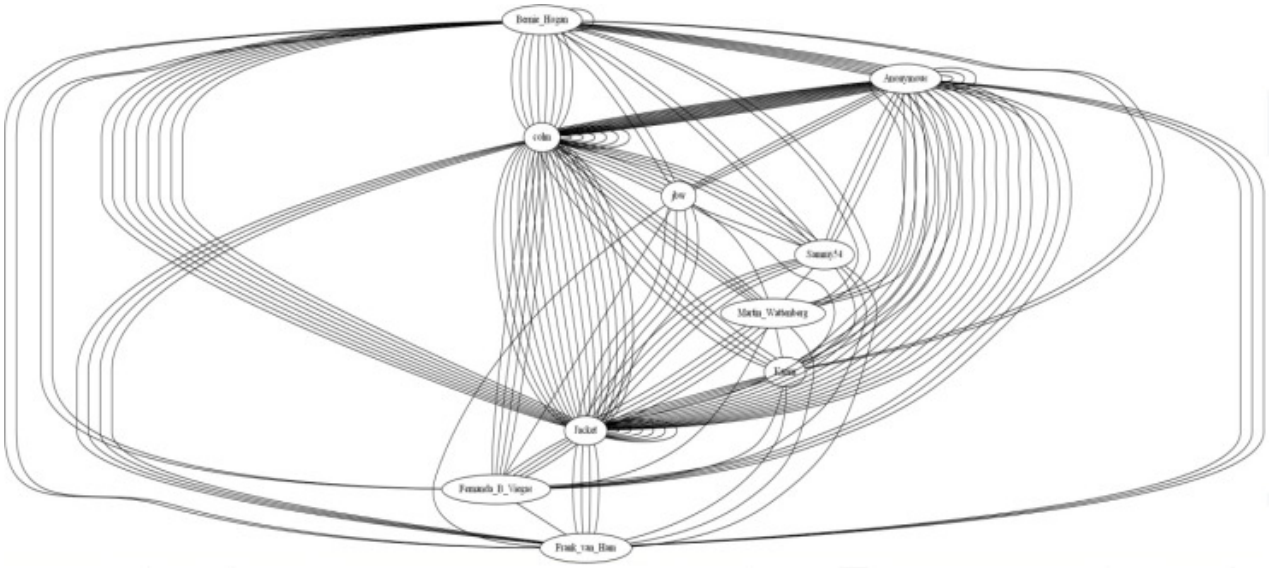


Figure 28. *Lesk* Measure of Word similarity wise textual similarity based collaboration pattern amongst the users off the 19 commented visualization

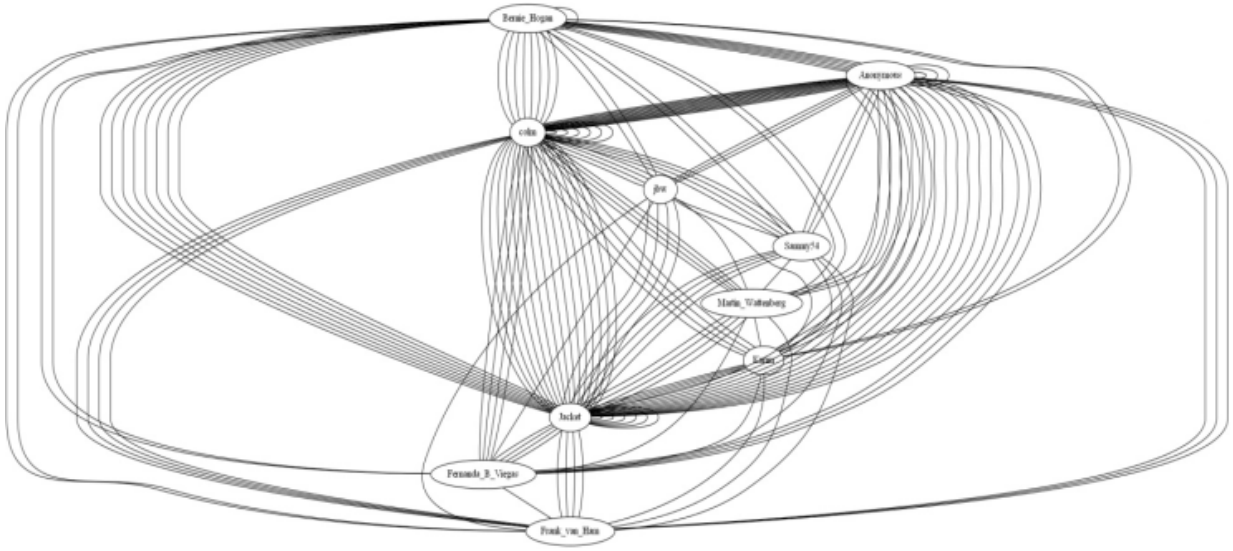


Figure 29. *Lacock-Chodorow* Measure of Word similarity wise textual similarity based collaboration pattern amongst the users off the 19 commented visualization

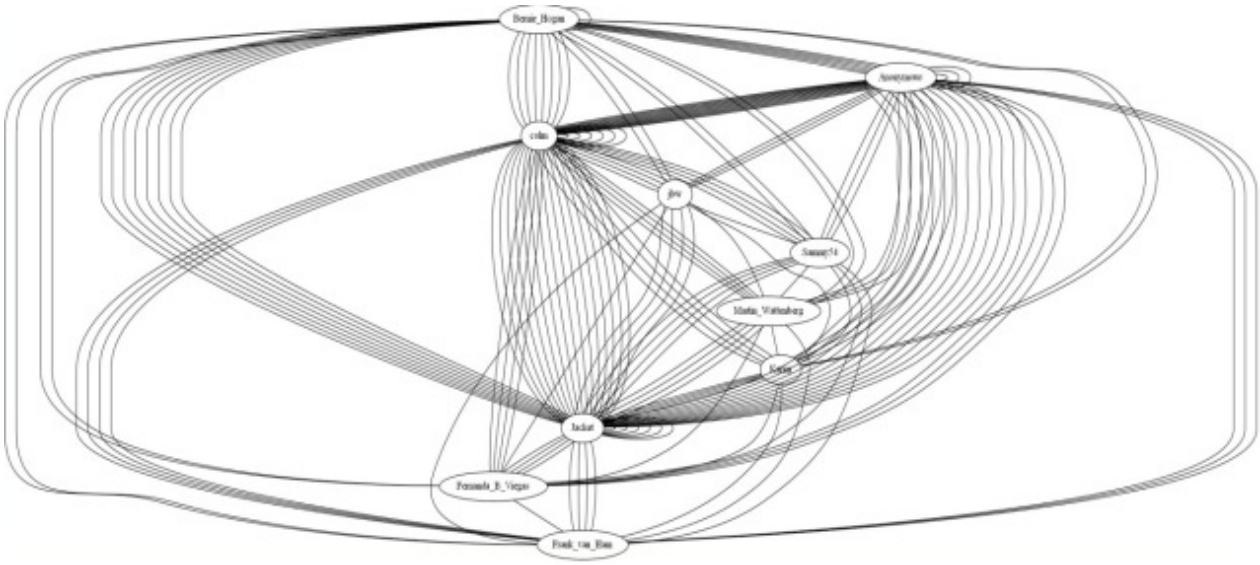


Figure 30. *Wu-Parmer* Measure of Word similarity wise textual similarity based collaboration pattern amongst the users off the 19 commented visualization

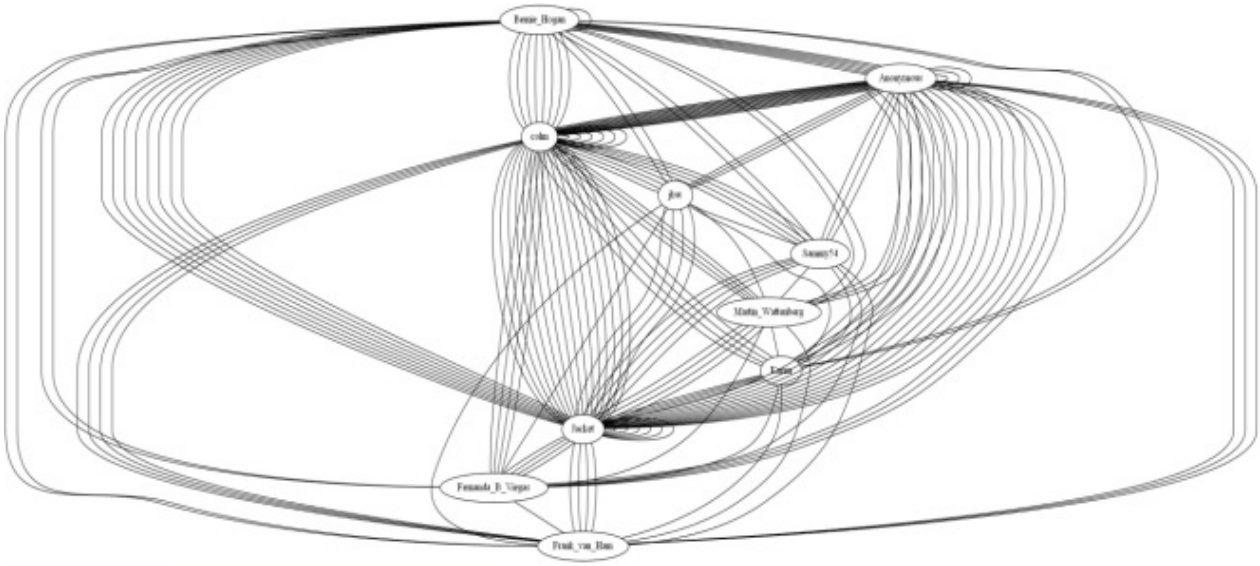


Figure 31. *Resnik* Measure of Word similarity wise textual similarity based collaboration pattern amongst the users off the 19 commented visualization

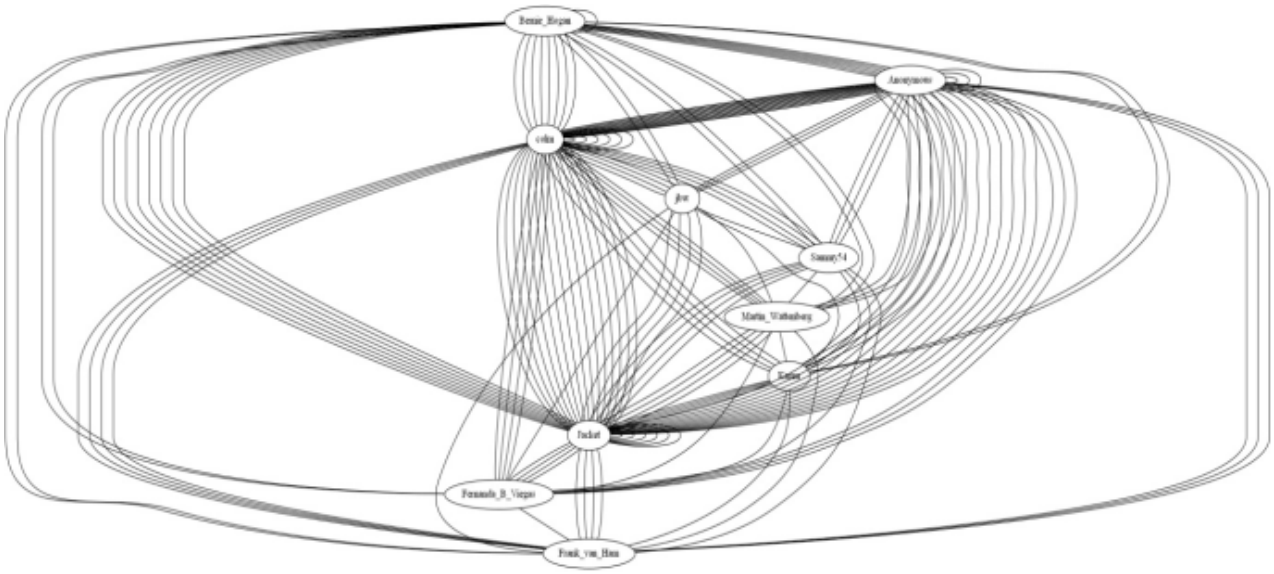


Figure 32. *Lin* Measure of Word similarity wise textual similarity based collaboration pattern amongst the users off the 19 commented visualization

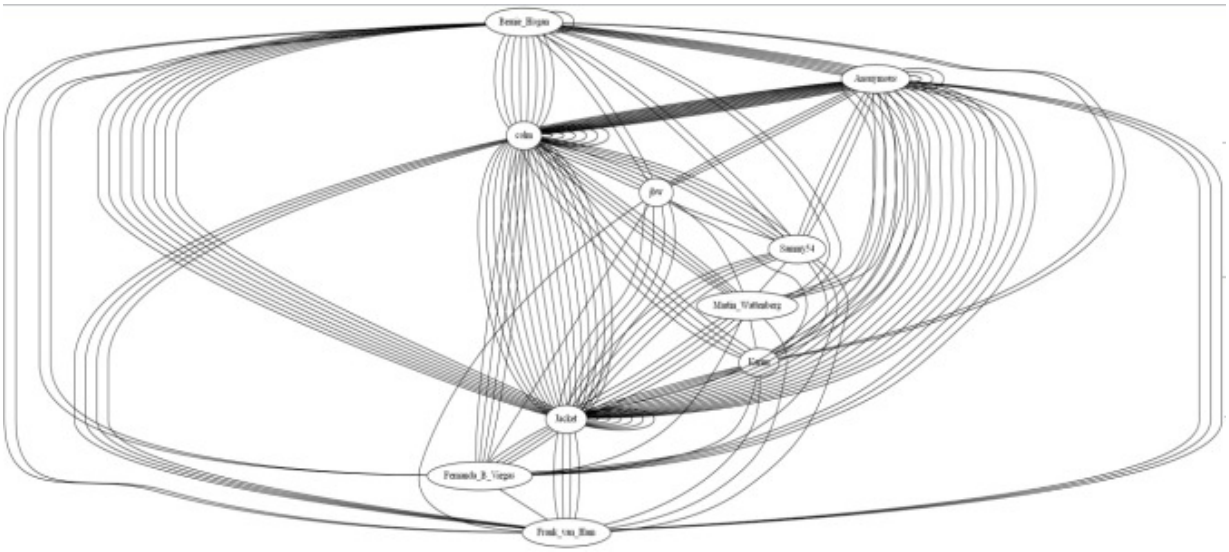


Figure 33. *Jiang-Conrath* Measure of Word similarity wise textual similarity based collaboration pattern amongst the users off the 19 commented visualization

As the comparative measure is not that apparent in the figures above, I picked up the top 20 similarity measures for each word based similarity techniques and displayed in the following cascaded bar graphs.

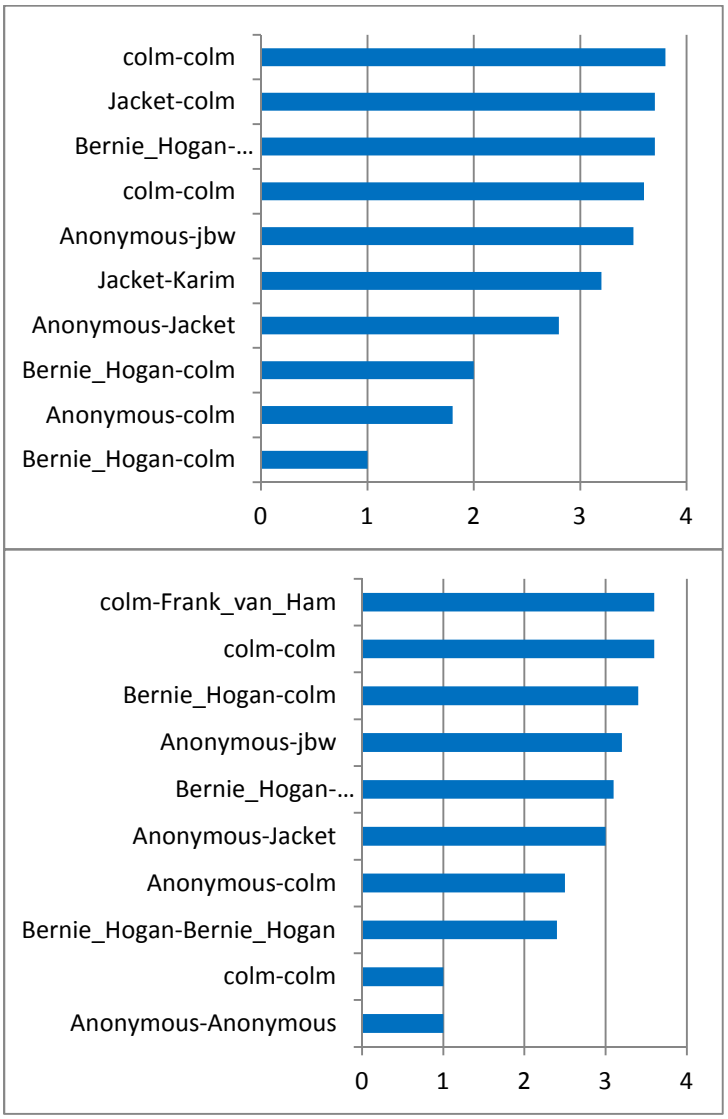


Figure 34. Lesk(top) and lch(bottom) word similarity measure based top 10 textual similarity measure with users in a 19 commented visualization

Figure 34 depicts the top 10 user similarity measure plotted as bar graph for the *Lesk* and *Leacock-Chodorow* measures. The other discussed method ones could be plotted and compared similarly.

From Figure 34 I see that two different measures produce little different similarity measure for same users, although some are same like the measures between ‘Anonymous’ and ‘Jacket’ or ‘Anonymous’ and ‘colm’. So I can certainly see this textual measure gives us certain positive direction towards the collaboration analysis using user comments’ textual similarity measure comparison.

5.2.3 Conclusion

Online social collaborative visualizations are in attention these days, so a proper analysis of the collaboration happening there carries a potential problem to enhance the service. In online social environment users collaborate through natural language based compositions, so textual similarity measure based comparison of textual snippets for the user collaboration analysis shows great potential and result. The use of different word similarity based measure, an enhanced word taxonomy, word sense disambiguators to measure textual similarity of compositions shows potential turn out.

In comparing such similarity measures using network graph shows the whole network in terms of its connectivity strengths. So if the collaboration in such an environment is loose, then a very sparse network graph would come up while a dense graph will reflect strong collaboration. Here I utilize such exiting measure and compare textual snippet based collaboration pattern visually, which turns out quite directive.

In the textual component similarity measure only nouns and verbs are accounted which basically holds the sense the backbone of an expression. Now the traditional way of the other connectors in a sentence like adjectives, adverbs, cardinals etc are lexically matched. But in traditional lexical matching a completely wrong result could be boosted as explained earlier. So instead if the sentential composition based structure [50] of the text is first considered and then the structural similarity is calculated through finding the apparent coordinate of its constituent words and their associated connectors: this kind of method would reflect much more appropriate result and that would be a very bright direction in the field of natural language processing.

Also if the data is collected from a social networking site where identity id mandatory such as facebook [8], then a lot more effective study could be done as unidentified users' information is mis-leading in this kind of analysis.

6 CONCLUSION AND FUTURE WORKS

6.1 Conclusion

In this dissertation, I tried to analyze the syntactic and semantic aspect of unstructured English texts through different techniques and visualizations.

I started from exploring the definition and understanding of syntactic understanding of natural English texts. As the existing methods of analysis on this perspective don't provide any clear and thorough idea, I introduced some visualizations for such analysis. The visualizations introduced seem quite effective in thorough explanation of the textual distribution in terms paragraph wise or every textual line based distribution. In this topic I initially focused on frequent/non frequent words, word length in terms of character lengths/syllable lengths. Later I defined the complexity of a textual composition through its clause level structural distribution. These visualizations immensely improve the syntactic understanding of text which actually impacts the readability. Based on such structural complexity distribution I introduced a content based recommendation technique as well, which focuses on providing better access to similar texts in terms of material's reading complexity.

Also I developed visual markers to express existing readability of a document at the paragraph level in terms of existing readability metrics. These visualizations give much more thorough understandings of a document compared to just the metrics.

While exploring the semantic domain, the initial challenge was to find appropriate textual data for analysis. As online social network provides the platform for people to come and communicate on some topic; I chose these kinds of networks as the source to capture texts which could have some sort of similarity in terms of the semantics. As I retrieved quite a huge amount of data for this purpose from an online social data visualization website (IBM's Many Eyes), my first focus was to understand and explore the data. In that sector I used descriptive and analytical statistics for such research and found interesting observations and results.

To experiment semantically first I employed latent semantic analysis technique on the retrieved data comments from Many Eyes. Based on the co-occurrence frequencies, I am able to identify the most relevant comments and potential spam. My method holds useful potential for developing effective search engines that automatically retrieve the most relevant user composed texts. It's also suitable for developing spam filters that identify and block irrelevant comments.

The final domain of work in this dissertation is textual snippet comparison based on the textual similarity measure. Word sense disambiguation methods based word distance measures in existing strong semantic word taxonomy was the play ground for this work. To figure out the pattern of textual similarity in an online collaborative environment, I calculated such measures and employed network graph to display such collaboration network's similarity and distance metrics. This visualization gives an overall idea of which users are in same cluster, collaborating about similar topic through their textual comment snippets. Also to compare thoroughly such similarity measure I used primitive bar graph, which could be used efficiently as the detail on demand feature of the just earlier mentioned visualization.

Therefore eventually quite some multi directional approach has been explored in here to approach panoramic disclosure of textual syntax and semantics mystery and in turn I received some satisfactory results and some potential prominent directions for the future.

6.2 Future Works

Although several directions in the syntactic and semantic analysis and visualization of natural text based compositions are explored in my research, but there are interesting close context ideas came up in my focus. In the syntactic analysis domain, an extension of the clausal structure of a sentence in a document seems very promising and interesting. If each sentence could be shaped as a tree consisting of clauses as branches, then the whole document could be a forest of such trees. In such development

then multi-document could be compared by comparing some forest measure techniques. Some close related directions in the semantic orientation, but not limited to, are listed below.

6.2.1 Employing LSA on Google Chart API

On user generated request, the Google Chart API [93] dynamically generates different type of charts. The Google Chart API [93] returns a chart image in response to a URL GET or POST request. It actually creates a PNG image of a chart from data and formatting parameters in an HTTP request. These charts could be embedded in a web page of user's choice, or downloaded as image for local and offline use. Actually using these API many different kind of charts could be created; currently line, bar, pie, and radar charts, as well as Venn diagrams, scatter plots, sparklines, maps, graphviz charts, google-o-meters, and QR codes are supported. All the information regarding the chart e.g. chart data, size, colors, and labels etc. are supplied as a part of the url in a specified format.

An exciting idea is to create a search tool for google chart with enhanced features. As a google chart could be created by producing a url following the chart grammars, parsing such url the parameters specific to a chart could be found. Now such parameters could be used to search for the similar charts. But enhancing such search through LSA [66] technique on the textual part of the charts like chart topics, chart legends would produce more appropriate and meaningful indexing on the chart search. This would result more appropriate and advanced chart search which will incorporate natural language understanding as a chart parameter to find out more pertinent results.

6.2.2 Studying LSA, p-LSA, LDA and Defining a Hybrid Approach

LSA fails to handle the polysemy (same words used with multiple meanings at different contexts) and concludes the outcome with noisy effects. Each occurrence of a word is treated in LSA as having the ditto meaning due to the word being represented as a single point in space. For example if the

word “bank” occur in a document as “river bank” and the “the federal bank”, LSA would take the meaning as same.

Some advanced variant of LSA such as pLSA [101] incorporates probabilistic approach and also shows profound promise in analysis of text. Unlike regular Latent Semantic Analysis which arises from linear algebra and executes a Singular Value Decomposition of co-occurrence tables, the pLSA [101] is based on a mixture decomposition derived from a latent class model. It is accepted as promising unsupervised learning method with a wide range of applications in text learning and information retrieval. There are other advancements on LSA as well. For example LDA [102] is a generative probabilistic model of a corpus. The basic idea behind LDA [102] is that the documents are represented as random mixtures over latent topics, where a topic is characterized by a distribution over words. LDA [102] is a generative model where sets of observations are explained by unobserved groups that explain why some parts of the data are similar. For example, if observations are words collected into documents, it assumes that each document is a mixture of a small number of topics and that each word's creation is attributable to one of the document's topics. LDA [102] is actually similar to probabilistic latent semantic analysis (pLSA) [101], except that in LDA [102] the topic distribution is assumed to follow a Dirichlet prior distribution. An explorative study on some of the related topic based compositions in English would be potentially interesting, which can lead to the derivation of a hybrid approach for the semantic disclosure of natural language based texts.

REFERENCES

[1] C. Manning. And H. Sch¨utze, Foundations of Statistical Natural Language Processing. The MIT Press, Cambridge, MA, US, 1999.

[2] A. H. Tan, Text mining: The state of the art and the challenges, In Proc of the Pacific Asia Conf on Knowledge Discovery and Data Mining, PAKDD'99 workshop on Knowledge Discovery from Advanced Databases, pp. 65–70, 1999.

[3] M. W. Berry, Survey of Text Mining: Clustering, Classification, and Retrieval, Springer, 2003.

[4] M. W. Berry, Survey of Text Mining II: Clustering, Classification, and Retrieval, Springer, 2007.

[5] A. Hotho, A. N¨urnberger, and G. Paaß, A brief survey of text mining, GLDV-Journal for Computational Linguistics and Language Technology, Vol. 20(1), pp. 19–62, 2005.

[6] Clarabridge Article: Unstructured Data and the 80 Percent Rule. http://www.clarabridge.com/default.aspx?tabid=137&ModuleID=635&ArticleID=551, (retrieved in December 2010).

[7] Text mining summit conference brochure. http://www.textminingnews.com/, 2005.

[8] C. J. Van Rijsbergen, A non-classical logic for information retrieval, The Computer Journal, Vol. 29, Issue 6, pp. 481–485, 1986.

[9] H. P. Luhn, The automatic creation of literature abstracts, IBM Journal of Research and Development, Vol. 2, pp.159–165, 1959.

[10] L. B. Doyle, Semantic Road Maps for Literature Searchers, The Journal of the Association of Computing Machinery(ACM), Vol. 8(4), pp. 553-578, 1961.

[11] D. R. Swanson and N. R. Smalheiser, An interactive system for finding complementary literatures: a stimulus to scientific discovery, Artificial Intelligence, Vol. 91, pp. 183-203, 1997.

[12] N. R. Smalheiser and Don R. Swanson, Using arrowsmith: A computer-assisted approach to formulating and assessing scientific hypotheses, Computer Methods and Programs in Biomedicine, Vol. 57(3), pp. 149-153, 1998.

[13] D. R. SWANSON and N. R. SMALHEISER, Implicit text linkages between Medline records: using Arrowsmith as an aid to scientific discovery, Library Trends, Vol. 48(Summer), pp. 48-–59, 1999.

[14] R. K. Lindsay and M. D. Gordon, Literature-based discovery by lexical statistics, Journal of the American Society for Information Science,Vol. 50, pp. 574-587, 1999.

[15] R. N. Kostoff and R. A. DeMarco, Information extraction from scientific literature with text mining, Analytical Chemistry, 2001.

[16] R. N. Kostoff, J. A. Del Rio, J. A. Humenik, E. O. Garcia, and A. M. Ramirez, Citation mining: Integrating text mining and biometrics for research user profiling, Journal of the American Society for Information Science, Vol. 52, pp. 1148-1156, 2001.

[17] D. Tkach, Text mining technology: Turning information into knowledge. IBM White paper, 1997.

[18] J. Dorre, P. Gerstl and R. Seiffert, Text mining: Finding nuggets in mountains of textual data. In the Proc. of the fifth ACM SIGKDD international conference on knowledge discovery and data mining, San Diego, California, pp. 398–401, 1999.

[19] A. M. Turing, Computing machinery and intelligence, Mind, Vol. 59, pp. 433–460, 1950.

[20] T. Winograd, Procedures as a representation for data in a computer program for understanding natural language, Ph.D. dissertation, MIT, MA, 1971.

[21] R. C. Schank, N. M. Goldman, C. J. Rieger, and C. Riesbeck, MARGIE: Memory Analysis Response Generation, and Inference on English, In the Proc. of the International Joint Conference on Artificial Intelligence, pp. 255-261, 1973.

[22] R. E. Cullingford, Script application: computer understanding of newspaper stories, Ph.D. thesis, Yale University, New Haven, CT, 1977.

[23] R. S. Michalski, J. G. Carbonell, and T. M. Mitchell, editors. Machine Learning: An Artificial Intelligence Approach, Vol. I, Morgan Kaufmann, Los Altos, California, 1983.

[24] G. E. Moore, Cramming more components onto integrated circuits (Reprinted from Electronics, pg 114-117, April 19, 1965), In the Proc. of the IEEE, Vol. 86, pp. 82-85, Jan 1998.

[25] C. D. Manning, H. Schütze, Foundations of Statistical Natural Language Processing, MIT Press, MA, 1999.

[26] N. Chomsky, Syntactic structures, The Hague: Mouton, 1957.

[27] M. Marcus, B. Santorini and M. Marcinkiewicz, Building a large annotated corpus of English: The PennTreebank. Computational Linguistics, Vol. 19(2), pp.313–330, 1993.

[28] J. Hajic, Building a syntactically annotated corpus: The Prague Dependency Treebank, In E. Hajicova (Ed.), Issues of valency and meaning. Studies in honour of Jarmila Panevova, Czech Republic: Charles University Press, Prague, Czechoslovakia.

[29] J. Barwiseand R. Cooper,Generalized Quantifiers and Natural Language, Linguistics and Philosophy, Vol. 4, pp. 159–219, 1981.

[30] E. Selkirk, The Syntax of words, MIT press, Cambridge, MA, 1982.

[31] M. Steedman, The syntactic process. MIT Press, Cambridge, MA, 2000.

[32] W. H. Dubay, Smart language: Readers, Readability, and the Grading of Text, BookSurge Publishing, 2007.

[33] E. Dale and J. S. Chall. The concept of readability, Elementary English, pp. 26:23, 1949.

[34] E. B. Fry, Readability, Reading Hall of Fame Book, Newark, DE: International Reading Association, 2006.

[35] G. R. Klare, The measurement of readability, Ames: Iowa State University Press, 1963.

[36] W. H. Dubay, The Principles of Readability, White Paper, Impact Information, 2004.

[37] G. H. McLaughlin. SMOG Grading —– a New Readability Formula. Journal of Reading, Vol. 34(2), pp. 639-646, 1969.

[38] J. P. Kincaid, R. P. Fishburne, R. L. Rogers and B. S. Chissom. Derivation of new readability formulas (Automated Readability Index, Fog Count and Flesch Reading Ease Formula) for Navy enlisted personnel. Research Branch Report 8-75, Millington, TN: Naval Technical Training, U. S. Naval Air Station, Memphis, TN, 1975.

[39] D. Klein and C. D. Manning. Accurate Unlexicalized Parsing. In the Proc. of the 41st Meeting of the Association for Computational Linguistics, pp. 423-430, 2003.

[40] D. Klein and C. D. Manning. Fast Exact Inference with a Factored Model for Natural Language Parsing. In Advances in Neural Information Processing Systems 15, Cambridge, MA: MIT Press, pp. 3-10, 2002.

[41] D. Doyle, Syllable and Accent Rules. http://english.glendale.cc.ca.us/phonics.rules.html (retrieved in March 2010)

[42] R. Gunning. The technique of clear writing. MGraw-Hill, New York, 1952.

[43] M. Coleman and T. L. Liau. A computer readability formula designed for machine scoring. Journal of Applied Psychology, Vol. 60, pp. 283-284, 1975.

[44] E. A. Smith, and R. J. Senter, Automated readability index, AMRL-TR-66-22. Wright-Patterson AFB, OH: Aerospace Medical Division, 1967.

[45] J. S. Chall and E. Dale. Readability Revisited: the new Dale-Chall readability formula. Brookline Books, Cambridge, MA, 1995.

[46] D. A. Keim and D. Oelke, Literature fingerprinting: a new method for visual literary analysis, In the Proc. of IEEE Symposium on Visual Analytics Science and Technology, 2007, pp. 115-122.

[47] C. Collins, S. Carpendale, and G. Penn, DocuBurst: visualizing document content using language structure, Computer Graphics Forum, Vol. 28(3), pp. 1039-1046, 2009.

[48] I. Lorge, Word lists as background for communication, Teachers College Record, Vol. 45, pp. 543-552, 1944.

[49] A. Kent, H. Lancour, J. E. Daily, Encyclopedia of library and information science, Mercel Dekker Inc., NY, NY, Vol. 26, pp178-179, 1979.

[50] S. Karmakar, Y. Zhu, Visualizing Text Readability. In the Proc. of International Conference on Data Mining and Intelligent Information Technology Applications(ICMiA 2010), Seoul, Korea, 2010.

[51] S. Karmakar, Y. Zhu, Visualizing Multiple Text Readability Indexes. In the Proc. of International Conference on Education and Management Technology (ICEMT 2010), Cairo, Egypt, 2010.

[52] Web of Trust. WOT Services Ltd. http://www.mywot.com/ (retrieved in June 2010).

[53] H. Chernoff, The Use of Faces to Represent Points in k-Dimensional Space Graphically, Journal of American Statistical Association, Vol. 68, pp. 361-368, June 1973.

[54] M. Deshpande and G. Karypis, Item-Based Top-N Recommendation Algorithms, ACM Trans. Information Systems. Vol. 22(1), pp. 143-177, 2004.

[55] D. Goldberg, D. Nichol, B. M. Oki and D. Terry, Using Collaborative Filtering to Weave an Information Tapestry, Communications of the ACM. Vol. 35(12), pp. 61–70, 1992.

[56] R. Mooney and L. Roy, Content-Based Book Recommending Using Learning for Text Categorization, In the Proc. of the Fifth ACM Conference on Digital Libraries, San Antonio, TX: ACM Press. pp. 195–204, 2000.

[57] J. Alspector, A. Kolez and N. Karunanithi, Comparing Feature-Based and Clique-Based User Models for Movie Selection. In the Proc. of the Third ACM Conference on Digital Libraries, Pittsburgh, PA: ACM Press, 1998.

[58] E. Andr´e and T. Rist, From Adaptive Hypertext to Personalized Web Companions, Communications of the ACM, Vol. 45(5), pp. 43–46, 2002.

[59] M. Balabanovic and Y. Shoham, Fab: content-based collaborative recommendation, Communications of the ACM. Vol. 40(3), pp. 66-72, 1997.

[60] S. Green, P. Lamere, and J. Alexander, Generating Transparent, Steerable Recommendations from Textual Descriptions of Items, In the Proc. of the third ACM conference on Recommender systems, NY. pp. 281-284, 2009.

[61] G. Semeraro, P. Lops, P. Basil and M. D. Gemmis, Knowledge Infusion into Content-based Recommender Systems, In the Proc. of the third ACM conference on Recommender systems,

NY. pp. 301-304, 2009.

[62] D. Tsatsou, F. Menemenis, I. Kompatsiaris and P. Davis, A Semantic Framework for Personalized Ad Recommendation based on Advanced Textual Analysis, In the Proc. of the third ACM conference on Recommender systems, NY. pp. 217-220, 2009.

[63] Z. Yu, Y. Nakamura, S. Jang, S. Kajita, and K. Mase, Ontology-Based Semantic Recommendation for Context-Aware E-Learning, Lecture Notes in Computer Science, Vol. 4611, pp. 898-907, 2007.

[64] T. Funkhouser, P. Min, M. Kazhdan, J. Chen, A. Halderman, D. Dobkin, and D. Jacobs, A search engine for 3D models, Transactions on Graphics, Vol. 22(1), pp. 83–105, 2003.

[65] A. Z. Broder, On the Resemblance and Containment of Documents, In the Proc. of of Compression and Complexity of SEQUENCES, 1997.

[66] S. Deerwester, S. Dumais, T. Landauer, G. Furnas, and R. Harshman, Indexing by latent semantic analysis, Journal of the American Society for Information Science, Vol. 41, pp.391– 407, 1990.

[67] S. Flesca, G. Manco, E. Masciari, L. Pontieri, and A. Pugliese, Detecting structural similarities between XML documents, Fifth International Workshop on the Web and Databases, 2002.

[68] S. Karmakar, Y. Zhu, Recommendation by Composition Style. In the Proc. of International Conference on Intelligent Systems Design and Applications (ISDA 2010). Cairo, Egypt, 2010.

[69] E. D. Liddy, W. Paik, E. S. Yu, Natural language processing system for semantic vector representation which accounts for lexical ambiguity, United States Patent #5,873,056, Feb 16, 1999.

[70] P. Resnik, Semantic Similarity in a Taxonomy: An Information-Based Measure and its Application to Problems of Ambiguity in Natural Language, Journal of Artificial Intelligence Research, Vol. 11 pp. 95-130, 1998.

[71] S. L. Lytinen, Dynamically Combining Syntax and Semantics in natural language processing. InProc. of AAAI, pp. 574-578, 1986.

[72] S. Novichkova, S. Egorov, and N. Daraselia, MedScan, a natural language processing engine for MEDLINE abstracts, Bioinformatics, Vol. 19(13), pp. 1699–1706, 2003.

[73] S. Miller, D. Stallard, R. Bobrow, and R. Schwartz, A fully statistical approach to natural language interfaces, In the Proc. of the 34th Annual Meeting of the ACL, Santa Cruz, California, pp. 55–61, 1996.

[74] T. K. Landauer, P. Foltz and D. Laham, Introduction to Latent Semantic Analysis, Discourse Processes, Vol. 25, 1998, pp. 259-284.

[75] T. K. Landauer and S. Dooley, Latent semantic analysis: theory, method and application, In Proc. of the Conference on Computer Support for Collaborative Learning: Foundations for a CSCL Community, pp. 742- 743, 2002.

[76] T. K. Landauer, D. Laham, and M. Derr, From paragraph to graph: Latent semantic analysis for information visualization, In Proc. of National Academy of Science (PNAS), Vol. 101, Suppl. 1, pp. 5214-5219, 2004.

[77] X. J. Wang, J. T. Sun, et al. Latent semantic analysis for multiple-type interrelated data objects, In Proc. of the 29th annual international ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 236 – 243, 2006.

[78] T. K. Landauer, Handbook of Latent Semantic Analysis, Psychology Press, 2007.

[79] W. Zhu and C. Chen, Storylines: Visual exploration and analysis in latent semantic analysis, Computers & Graphics, Vol. 31, pp. 338-349, 2007.

[80] G. Gorrell, Latent Semantic Analysis: How does it work, and what is it good for?, LSA Tutorial (http://www.dcs.shef.ac.uk/~genevieve/lsa_tutorial.htm), (retrieved in August 2010).

[81] J. A. Larusson, and R. Alterman, Visualizing student activity in a wiki-mediated co-blogging exercise, In Proc. of the 27th Conference on Human Factors in Computing Systems, pp. 4093-4098, 2009.

[82] J. Pino, and M. Eskenazi, An application of latent semantic analysis to word sense discrimination for words with related and unrelated meanings, In Proc. of the Fourth Workshop on Innovative Use of NLP for Building Educational Applications, pp. 43-46, 2009.

[83] F. B. Viegas, M. Wattenberg, F. van Ham, J. Kriss, M. McKeon, Many Eyes: a site for visualization at Internet scale, IEEE Transactions on Visualization and Computer Graphics, Vol. 13(6), pp. 1121-1128, 2007.

[84] Y. Zhu, S. Karmakar, K. Kokala and R. K. N. Ayudhaya, Analyzing Comments on Social Media Web Sites with Latent Semantic Analysis, In the Proc. of The Fifth International Conference on Knowledge, Information and Creativity Support Systems (KICSS 2010) Chang Mai, Thailand, 2010.

[85] http://manyeyes.alphaworks.ibm.com (retrieved in July 2010).

[86] http://www.swivel.com (retrieved in July 2010).

[87] F. B. Viegas, M. Wattenberg, F. van Ham, J. Kriss, and M. McKeon, ManyEyes: a Site for Visualization at Internet Scale, IEEE Transactions on Visualization and Computer Graphics, Vol. 13, No. 6, pp. 1121-1128.

[88] http://tableausoftware.com/public (retrieved in July 2010).

[89] http://www.google.com/publicdata (retrieved in July 2010).

[90] http://www.getpivot.com (retrieved in July 2010).

[91] Y. Zhu, S. Karmakar, Analysis of the online social data visualization service: Many Eyes, In the Proc. of International Conference on Intelligent Systems Design and Applications (ISDA 2010). Cairo, Egypt, 2010.

[92] Simon Blog, http://www.simonblog.com/ (retrieved in 2010).

[93] Google Chart API, (http://code.google.com/apis/chart), Retrieved December 2010.

[94] K. Pearson, On the criterion that a given system of deviations from the probable in the case of a correlated system of variables is such that it can be reasonably supposed to have arisen from random sampling, Philosophical Magazine, Series vol. 550(302), pp. 157-175.

[95] G. A. Miller, R. T. Beckwith, C. D. Fellbaum, D. Gross, K. Miller, WordNet: An On-line Lexical Database, International Journal of Lexicography, Vol. 3(4), pp. 235-244, 1990.

[96] M. M.Masterman, The thesaurus in syntax and semantics, Mechanical Translation. Vol. 4(1-2), pp. 35-44, 1957.

[97] S. Madhu and D. Lytle, A figure of merit technique for the resolution of non-grammatical ambiguity, Mechanical Translation, Vol. 8(2), pp. 9–13, 1965.

[98] Y. Wilks, A preferential pattern-seeking semantics for natural language inference. Artificial Intelligence, Vol. 6, pp. 53-74, 1975.

[99] M. Lesk, Automatic sense disambiguation: How to tell a pine cone from an ice cream cone. In the Proc. of the 1986 Special Interest Group in Documentation of Association for Computing Machinery, New York, pp. 24-26, 1986.

[100] R. Mihalcea, C. Corley, and C. Strapparava. Corpus-based and knowledge-based measures of text semantic similarity. In the Proc. of Association for the Advancement of Artificial Intelligence (AAAI), 2006.

[101] T. Hofmann, Probabilistic Latent Semantic Analysis, In the Proc. of 15th Conference on Uncertainty in Artificial Intelligence, pp. 289-296, 1999.

[102] D. Blei, A. Ng, and M. Jordan. Latent Dirichlet allocation. In Advances in Neural Information Processing Systems, Vol. 14, 2002.

[103] L. A. Goodman, The multivariate analysis of qualitative data: Interactions among multiple classifications, *Journal of the American Statistical Association*, vol. 65, issue 329, pp. 226-256, 1970.

[104] L. A. Goodman, A general model for the analysis of surveys, American Journal of Sociology, vol. 77, issue 6, pp. 1035-1086, 1972.

[105] G. Melin, O. Persson, Studying research collaboration using co- authorships, Scientometrics, vol. 36, pp. 363-377, 1996.

[106] B. M. Gupta, S. Kumar, C.R. Karisiddappa, Collaboration profile of theoretical population genetics speciality, Scientometrics, vol. 39, pp. 293-314, 1997.

[107] S. M. Lawani, Quality, Collaboration and Citations in cancer reseach: A 268 bibliometric study, Ph.D. Dissertation, Florida State University, p395, 1980.

[108] K. Subramanyam, Bibliometric studeis of research collaboration: A review, Journal of Information Science, vol. 6, pp. 33-38, 1983.

[109] I. Ajiferuke, Q. Burrel, J. Tague, Collaborative coe_cient: A single measure of the degree of collaboration in research, Scientometrics Vol. 14 , pp. 421-433, 1988.

[110] L. G. Hathorn, A. L. Ingram, Online Collaboration: Making It Work, Educational Technology, vol. 42 , n1, pp. 33-40 , 2002.

[111] http://www.sixdegrees.org/ (retrieved in September 2011)

[112] http://www.facebook.com/ (retrieved in September 2011)

[113] http://www.twitter.com/ (retrieved in September 2011)

[114] http://www.linkedin.com/ (retrieved in September 2011)

[115] G. Salton and M.E. Lesk, Computer evaluation of indexing and text processing, Prentice Hall, Ing. Englewood Cliffs, New Jersey, pp. 143–180, 1971.

[116] J. Rocchio, Relevance feedback in information retrieval. Prentice Hall, Ing. Englewood Cliffs, New Jersey, pp. 313-323, 1971.

[117] G. Salton, A. Singhal, M. Mitra, and C. Buckley. Automatic text structuring and summarization, Information Processing and Management, vol. 33, issue 2, 1997.

[118] Papineni, S. Roukos, T. Ward, and W. Zhu, Bleu: a method for automatic evaluation of machine translation, Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics (ACL 2002), Philadelphia, PA, 2002.

[119] C. Lin , E. Hovy, Automatic evaluation of summaries using N-gram co-occurrence statistics, Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology, pp.71-78, 2003, Edmonton, Canada.

[120] G. Salton, and A. Bukley. Term weighting approaches in automatic text retrieval, Readings in Information Retrieval, Morgan Kaufmann Publishers, San Francisco, CA, 1997.

[121] C. Corley and R. Mihalcea, Measuring the semantic similarity of texts, Proceedings of the ACL Workshop on Empirical Modeling of Semantic Equivalence and Entailment, pp. 13-18, Ann Arbor, Michigan, 2005.

[122] C. Fellbaum, WordNet: An electronic lexical database. MIT Press, 1998.

[123] C. Leacock and M. Chodorow, Combining local context and WordNet similarity for word sense identification, In Christiane Fellbaum, editor, WordNet: An Electronic Lexical Database, chapter 11, pp. 265–283. MIT Press, 1998.

[124] Z. Wu and M. Palmer, Verb semantics and lexical selection, In Proceedings of the 32nd Annual Meeting of the Association for Computational Linguistics, Las Cruces, NewMexico, 1994.

[125] D. Lin, An information-theoretic definition of similarity. In Proceedings of the Fifteenth International Conference on Machine Learning (ICML-98), pp. 296–304, Madison, Wisconsin, 1998.

[126] J. Jiang and D. Conrath, Semantic similarity based on corpus statistics and lexical taxonomy, Proceedings of International Conference Research on Computational Linguistics (ROCLING X), Taiwan, 1997.

[127] K. Sparck-Jones, A statistical interpretation of term specificity and its applicatoin in retrieval, Journal of Documentation, vol. 28, issue 1, pp.11–21, 1972.

[128] J Ellson, S. North “Graphviz.” http://www.graphviz.org, 2004. (Retrieved in September 2011).

[129] J. Ellson, E. R. Gansner, L. Koutsofios, S. North, and G. Woodhull, Graphviz - open source graph drawing tools. In Graph Drawing, pp. 483–485, 2001.

[130] E. Koutsofios and S. North, Drawing graphs with dot, Technical report (available from the authors), AT&T Bell Laboratories, Murray Hill NJ, 1992.

[131] http://www.puffinwarellc.com/index.php/news-and-articles/articles/33.html?start=1 (Retrieved in September 2011).

[132] S.D. Poisson, Recherches sur la Probabilit´e des Jugements en Mati`ere Criminelle et en Mati`ere Civile, Pr´ec´ed´ees des R`egles G´en´erales du Calcul des Probabilit´es, Bachelier, Paris, 1837.

[133] A. Wald, Sequential tests of statistical hypotheses, Annals of Mathematical Statistics, vol. 16, issue 2, pp. 117–186, 1945.

[134] A. Jeansonne, Loglinar models, http://userwww.sfsu.edu/~efc/classes/biol710/loglinear/Log%20Linear%20Models.htm, 2002 (Retrieved September, 2011).

[135] J. Michelizzi, Semantic relatedness applied to all words sense disambiguation, Master's thesis, University of Minnesota, Duluth, July, 2005.

[136] S. Banerjee, Adapting the Lesk Algorithm for Word Sense Disambiguation to WordNet, Master's Thesis, University of Minnesota, 2002.

[137] http://en.wikipedia.org/wiki/Pearson%27s_chi-squared_test (Retrieved October 2011).

[138] Harald Cramér, Mathematical Methods of Statistics, Princeton: Princeton University Press, pp. 282,1946, ISBN 0691080046.

[139] http://en.wikipedia.org/wiki/Cram%C3%A9r%27s_V (Retrieved October 2011).

[140] S. Patwardhan, S. Banerjee and T. Pedersen, Using measures of semantic relatedness for word sense disambiguation, In Proceedings of the Fourth International Conference on Intel ligent Text Processing and Computational Linguistics, pp. 241–257, Mexico City, February 2003.

[141] http://www.wikipedia.org/ (Retrieved October 2011).

APPENDICES

Appendix A : Publications Related to this Research

Conference Proceedings

- Saurav Karmakar, Ying Zhu, "Visualizing Text Readability", In the Proceedings of IEEE International Conference on Data Mining and Intelligent Information Technology Applications(ICMiA 2010), November 30 - December 2, 2010, Seoul, Korea.
- Saurav Karmakar, Ying Zhu, "Recommendation by Composition Style", In the Proceedings of The Tenth IEEE International Conference on Intelligent Systems Design and Applications (ISDA 2010), November 29 - December 1, 2010, Cairo, Egypt.
- Ying Zhu, Saurav Karmakar, "Analysis of the online social data visualization service: Many Eyes", In the Proceedings of The Tenth IEEE International Conference on Intelligent Systems Design and Applications(ISDA 2010), November 29 - December 1, 2010, Cairo, Egypt.
- Ying Zhu, Saurav Karmakar, Kireet Kokala, Rawiroj Kasemsri Na Ayudhaya, "Analyzing Comments on Social Media Web Sites with Latent Semantic Analysis", In the Proceedings of The Fifth International Conference on Knowledge, Information and Creativity Support Systems (KICSS 2010), LNCS/LNAI, Springer, November 25 - 27, 2010, Chang Mai, Thailand.
- Saurav Karmakar, Ying Zhu, "Visualizing Multiple Text Readability Indexes", In the Proceedings of IEEE International Conference on Education and Management Technology(ICEMT 2010), November 2-4, 2010, Cairo, Egypt.
- Saurav Karmakar, Ying Zhu, "Mining Collaboration through Textual Semantic Interpretation", (Submitted)
- Saurav Karmakar, Ying Zhu, "Nature of Relationships in Social Collaborative Webiste : MayEyes", (Submitted)