

Big Data Analytics for Development: Events, Knowledge Graphs and Predictive Models

by

Sunandan Chakraborty

A dissertation submitted in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
Department of Computer Science
Courant Institute of Mathematical Sciences
New York University
September 2015

Lakshminarayanan Subramanian

© Sunandan Chakraborty
All Rights Reserved, 2015

Acknowledgments

It has indeed been a long and yet a deeply fulfilling journey. A journey that has been made special because of a number of wonderful and extraordinary people, without whose support I would not be where I am today. I would like to start with my advisors Prof. Lakshminarayanan Subramanian and Yaw Nyarko, for their guidance, and their contribution in helping me achieve substance from what started as a fledgling idea. I am also thankful to the Center for Technology and Economic Development (CTED), for the financial and other logistic support to make this otherwise difficult path smooth. I am deeply grateful to my dissertation committee members, Prof. David Sontag, Prof. Srikanth Jagabathula, Prof. Ralph Grishman and Prof. Jinyang Li for playing an important role in shaping my research over the last few years. I would also take this opportunity to thank Leslie, Santiago, Rosemary, Flora, Marian, Liuba for the administrative support during my graduate studies. I am also indebted to my past advisors Prof. Anupam Basu, Prof. Sudeshna Sarkar from IIT Kharagpur and Dr. Kentaro Toyama from Microsoft Research India for their valuable help and motivating me for doing a PhD.

I am deeply thankful to my parents, Subhas and Uttara Chakraborty, for having instilled in me from my childhood an unquenching desire for the pursuit of knowledge, this thesis is a result of which. They have been and will continue to be my role models for what I, as a scholar am striving to become. I am indebted to my brother Suranjan Chakraborty and sister-in-law Radhika Shenoy for their valuable help during this journey. Their support helped me throughout this process, right from the beginning, helping me with the application process to the emotional and financial support whenever required. I am also thankful to my family, my uncles Indrajit Sen and Abhjit Sengupta, aunts Deepa Sen and Alakananda Sengupta, my cousins Anandaroop, Sukhalata, Aparajito, Ananyo; my great aunt Jharna Gourlay and uncle Stephen Gourlay, my parents-in-law Tapan and Ratna Bhaduri, sister-in-law and her family, Tapti

and Anuthosh for their continuous support and inspiration. I would also take this opportunity to offer a special thanks to Prof. Rajkumar Raychaudhuri for instilling in me love for Mathematics, Science and research in general during my formative years.

My friends and co-workers have also been a huge source of encouragement. I would explicitly like to thank – Aditya, Shankar, Ashlesh, Michael, Matt, Christopher, Paul, Alex, Russell, Nguyen, Ashwin, Talal, Varun, Renfei, Ruchi, Tiffany, Nicole, Alex, David, Giorgia, Emilia and Sam for their support. I should also thank Pradipta, Samriddhi and Purbasha for sharing their wisdom about graduate studies and the wise advises and friendly encouragements.

Finally, I would like to express my gratitude to my wife, Toa. Her sacrifices were more and pains no less than mine. She did make sure that my life was as comfortable as possible during this arduous journey. I thank her from the bottom of my heart for being patient and having so much faith in me. I appreciate all her sacrifices to make this thesis possible. I am forever grateful to her and I dedicate this thesis to her.

Abstract

Volatility in critical socio-economic indices can have a significant negative impact on global development. This thesis presents a suite of novel big data analytics algorithms that operate on unstructured Web data streams to automatically infer events, knowledge graphs and predictive models to understand, characterize and predict the volatility of socioeconomic indices.

This thesis makes four important research contributions. First, given a large volume of diverse unstructured news streams, we present new models for capturing *events* and learning spatio-temporal characteristics of events from news streams. We specifically explore two types of event models in this thesis: one centered around the concept of event triggers and a probabilistic meta-event model that explicitly delineates named entities from text streams to learn a generic class of meta-events. The second contribution focuses on learning several different types of *knowledge graphs* from news streams and events: a) Spatio-temporal article graphs capture intrinsic relationships between different news articles; b) Event graphs characterize relationships between events and given a news query, provide a succinct summary of a timeline of events relating to a query; c) Event-phenomenon graphs that provide a condensed representation of classes of events that relate to a given phenomena at a given location and time; d) Causality testing on word-word graphs which can capture strong spatio-temporal relationships between word occurrences in news streams; e) Concept graphs that capture relationships between different word concepts that occur in a given text stream. The third contribution focuses on connecting the different knowledge graph representations and structured time series data corresponding to a socio-economic index to automatically learn *event-driven predictive models* for the given socio-economic index to predict future volatility. We propose several types of predictive models centered around our two event models: event triggers and probabilistic meta-events. The final contribution focuses on a broad spectrum of inference case studies for different

types of socio-economic indices including food prices, stock prices, disease outbreaks and interest rates. Across all these indices, we show that event-driven predictive models provide significant improvements in prediction accuracy over state-of-the-art techniques.

Table of Contents

Acknowledgments	iii
Abstract	v
List of Figures	xii
List of Tables	xiv
1 Introduction	1
1.1 Contributions	5
1.2 Summary of Chapters	5
1.2.1 Inference of Location-specific Critical Trends	5
1.2.2 Summarization Search	6
1.2.3 Extraction of (Key,Value) Pairs from Domain-specific Unstructured Data	7
1.2.4 Event Analytics and Prediction from News Articles	7
1.3 Inference Case Studies	10
1.3.1 Concept Graphs and Comprehension Diagnostics for TextBooks	10
1.3.2 Satellite Image Analytics and Land Change Patterns	11
2 Related Work	12
2.1 Large Text Representational Schemes	12
2.2 News Analytics and Applications	12
2.3 Event Extraction	13

2.4	News Visualization	15
3	Inference of Location-specific Critical Trends	18
3.1	Motivation	18
3.2	System Details	20
3.3	Information Retrieval from the Web	21
3.3.1	Base Document Set	21
3.3.2	Extended Document Set	22
3.3.3	Page Validation	25
3.4	Summarizing Critical Trends	26
3.4.1	Inference Engine	28
3.4.2	Summarization	29
3.5	Evaluation	30
3.5.1	Results and Observation	30
3.5.2	Validity of Information	33
3.6	Summary	36
4	Summarization Search	39
4.1	Introduction	39
4.2	Methodology	41
4.2.1	Text Summarization Engine	41
4.2.2	Image Extraction Engine	44
4.2.3	Aggregation and Presentation	47
4.3	Evaluation	48
4.3.1	Focused Queries from AOL Logs	49
4.3.2	Category Specific Information Queries	49
4.4	Summary	50
5	Extraction of (Key,Value) Pairs from Domain-specific Unstructured Data	52
5.1	Introduction	52
5.2	Problem Statement	54

5.3	Datasets	55
5.4	Unsupervised Method	55
5.4.1	Affinity Graph	58
5.4.2	Identification of Labels-Values Pairs from Affinity Graph	59
5.4.3	Performance	62
5.5	Supervised Learning	63
5.5.1	Conditional Random Fields	65
5.5.2	Training and Extraction of Label-Value Pairs	65
5.5.3	Experiments and Results	67
5.6	Summary	69
6	Event Analytics and Prediction from News Articles	71
6.1	Introduction	71
6.2	Related Work	73
6.3	Problem Statement	74
6.4	Event Models	76
6.4.1	Event Trigger Extraction	78
6.4.2	Event class Model	79
6.4.3	Historical Event Model	83
6.4.4	Topic Based Prediction	84
6.5	Evaluations	85
6.5.1	Data	85
6.5.2	Experiments and Results	86
6.6	Event Phenomenon Graph	93
6.7	Causality Test between Words	94
6.7.1	Word-word Relations	96
6.7.2	Finding Word Relations	99
6.7.3	Granger Causality Test	101
6.7.4	Evaluation	103
6.8	Summary	105

7	Concept Graphs and Comprehension Diagnostics for TextBooks	113
7.1	Introduction	113
7.2	Related Work	114
7.3	Comprehension Burden	115
7.3.1	Properties of Well-written Books	116
7.3.2	Origin of Comprehension Burden	116
7.3.3	Notations	118
7.3.4	Definition of Comprehensive Burden	119
7.3.5	Concepts	120
7.3.6	Relationship between Concepts	121
7.3.7	Significance Scores	121
7.3.8	Key Sections	122
7.4	Evaluation	122
7.4.1	Corpus	123
7.4.2	Diagnostic Tool	123
7.5	Summary	125
8	Satellite Image Analytics and Land Change Patterns	126
8.1	Introduction	126
8.2	Land Pattern Analysis and Food Security	128
8.2.1	Satellite Images	129
8.2.2	West Bengal: A Case Study	130
8.3	Satellite Image Data	130
8.4	Satellite Image Analysis Tool	131
8.4.1	Design	131
8.4.2	Performance	134
8.5	Land Pattern Analysis of West Bengal	138
8.6	Summary	142
9	Conclusions	145

List of Figures

1.1 Overall Flow of the Thesis	2
1.2 Onion Price in India between 2006 and 2012	4
1.3 Currency exchange rate: USD vs INR	4
1.4 Stock Prices of Infosys	4
1.5 Detailed Flow of the Thesis	4
1.6 Events related to the phenomenon of onion price rise	8
1.7 Word causality graph	10
3.1 Overall Architecture of the System	21
3.2 PageRank distribution	34
3.3 Classification of webpages	34
4.1 AOL Queries Results	50
4.2 Category Specific Information Queries Results	50
5.1 Total accuracy (F-measure) for car ads and under each category	62
5.2 Total accuracy (F-measure) for apartment ads and under each category	62
5.3 Linear chain CRF model to annotate ads (from Table 5.1) with keys	65
5.4 Accuracy for each experiment. There were 10 experiments performed on car ads with combinations of different word-window size and feature sets. Y-axis shows the accuracy in percentage.	69

6.1	Prices of crops between 2006 and 2012 represented as weekly average. The spikes represent an increase in price of 10% from the previous value. The horizontal axis represents the weeks between January 2006 and December 2012, where 0 is the first week of Jan 2006 and 364 is the last week of Dec 2012	88
6.2	Stock price fluctuation of 4 Indian companies between Jan 1, 2006 and Dec 31, 2012	88
6.3	Monthly and yearly average accuracy	92
6.4	Accuracy of HIST model with varying past days. x-axis show how many days in the past was considered	92
6.5	Events related to the phenomenon of onion price rise	95
6.6	Causality between two time-series [src: Wikipedia]	101
7.1	Illustration of comprehension burden: Concept c is explained in section 3 ($k_c = 3$) and is also mentioned in sections 1 and 4. A related concept d occurs in sections 2 and 4. As c is explained in section 3, the reader incurs comprehension burden when reading about c in section 1 ($\psi((c, 1) \leftarrow c) > 0$) and about d in section 2 ($\psi((d, 2) \leftarrow c) > 0$), but not in section 4 when encountering c and d $\psi((c, 4) \leftarrow c) = \psi((d, 4) \leftarrow c) = 0$	120
8.1	Image of a location in Angola taken in 2003	128
8.2	Image of a location in Angola taken in 2011	128
8.3	Noise in the dataset	132
8.4	Segmented Image	135
8.5	Superpixels and their corresponding color histograms	136
8.6	Different types of farmland their corresponding color histograms	136
8.7	Texture of farmland (+) and treecover land (o)	137
8.8	Percentage of arable land per year	139
8.9	Comparison between official percentage cropped area (upper) and estimation from the satellite images analysis tool (lower)	140
8.10	Map of West Bengal and district boundaries of southern part of the state	141
8.11	Relationship between agricultural land area and rice production for different districts (2012)	143

List of Tables

3.1	A Segment of the Query List	23
3.2	Keywords extracted from 4 locations	33
3.3	Manual verification of the problems reported in some locations	37
5.1	A sample ad from Craigslist	55
5.2	Different keys and their occurrence in the ads	56
5.3	Corresponding $\langle key, value \rangle$ pairs from Table 5.1 (partial). Desc: Descriptive, Num: Numeric, Bin: Binary	57
5.4	Subset of labels used for car and apartment ads	66
5.5	A subset of the final feature set	68
5.6	Accuracies for the supervised model for car and apartment ads	68
6.1	Model variables and parameters	83
6.2	Performance for Food Prices	89
6.3	Performance for Stock Prices	89
6.4	Performance for Dengue Outbreaks	90
6.5	The top events (triggers) associated with each crop is shown in the order of their likelihood ratio value. This list shows the top 5% of the triggers for each crop.	91
6.6	Associated topics for the triggers from Table 6.5. The topics increase the readability of the event-phenomenon graph. Topic names are manually assigned.	93

6.7	Examples of events extracted from the news data. For each event class the main event triggers are shown in the left column. The main event triggers are assumed to be equally likely so there are no orders. In the right column, subsidiary events are shown in the order in which they are more likely to be associated with the event class	106
6.8	Continuation of Table 6.7	107
6.9	Dependency between event pairs given a time difference δ , computed as normalized point-wise mutual information. δ is in months	108
6.10	Comparison of performance	108
6.11	Examples of causal links obtained from the test for a set of selected words. This table is generated by selecting a set of words and then running g-causal test across all other words. The list of words (in alphabetical order) in the right column represents all the words were the test was positive for the word in the left column	109
6.12	Continuing from Table 6.11	110
6.13	Continuing from Table 6.12	111
6.14	Continuing from Table 6.13	112
8.1	West Bengal at a glance	131
8.2	Accuracy of the tool for different features used	138
8.3	Comparison between arable land and food production	140
8.4	District-wise change in agricultural land between 2000 and 2012	142

Chapter 1

Introduction

Many socio-economic indices have historically shown high volatility in several countries, especially in the developing world. Volatility in important indices, such as, commodity prices, unemployment rate, currency exchange rates etc. can adversely impact the state of the economy of a country. Commodity price instability has a negative impact on economic growth, countries' financial resources, and income distribution, and may lead to increased poverty instead of poverty alleviation. Many countries, including India, derive more than 90% of their export earnings from commodities. Unstable currency exchange rates can also lead to fluctuations in commodity prices. Thus, monitoring these socio-economic indicators and understanding their volatility is an important task for a stable and healthy economy. Despite numerous fluctuations that have been observed decades, we still lack a fundamental understanding why does a particular socio-economic index fluctuate at a given time and location and how do fluctuations across different indices relate to each other. Economists who have studied such problems often rely on predefined economic models to understand and predict volatility of a given socio-economic index. Computational researchers who have studied such problems have primarily used computational modeling techniques to analyze structured time series data. The rapid proliferation of unstructured data streams on the web combined with the emergence of highly sophisticated computational linguistics algorithms in the past decade presents a unique opportunity to understand socio-economic fluctuations. This thesis makes the following fundamental research contribution: *we present a suite of novel big data analytics algorithms that operate on unstructured news streams to automatically infer events, knowledge graphs and predictive*

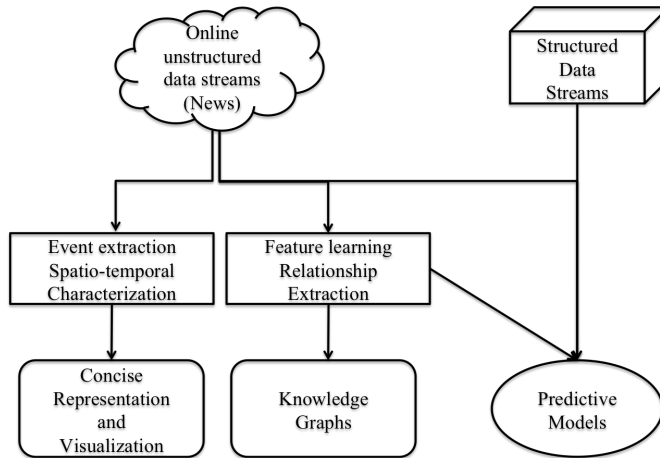


Figure 1.1: Overall Flow of the Thesis

models to – understand, characterize and in specific cases, predict – the volatility of different socio-economic indices.

The web is a repository of large information and most of it is in the form of unstructured text. It can be overwhelming for a user to look for specific information in this large pool of text. Our primary goal in this thesis is to have a precise and succinct representation of this large volume of data. We propose novel ways of representing events by extracting information from these texts. This notion of events is a lower dimension representation of web documents reporting about events and incidents happening around the world, representing the information in a much more precise form. The events represent isolated points in this data. Based on how things have happened over time and location, many relations can be understood that are not obvious. Our second contribution is centered around finding such latent relationships between events. These relations are depicted through the different knowledge graphs proposed in this thesis. Finally, we use the lower dimensional and precise form of representing web text (events) and the latent relationships between them (knowledge graphs) and combine them with observed time-series data of a phenomenon (e.g. fluctuations in socio-economic indicator) to understand what types of events are driving these phenomena and how we can use them to predict these phenomena. This flow is depicted in Figure 1.1.

We motivate the vision of this thesis using simple examples of socio-economic index volatility in the context of India. Figures 1.2, 1.3 and 1.4 shows the variation of three socio-economic indicators in the

last few years. Figure 1.2 shows the price onion in India between 2006 and 2012. Figure 1.3 shows how Indian Rupee has performed against US Dollars in last 4 years and finally Figure 1.4 is stock prices of Indian IT company Infosys. All these figures are a reflection of the unstable nature of India's economy. Fluctuations in onion prices can have adverse effects as it is one of the most consumed food crop in the country. In last 7 years onion prices have seen an increase of almost 400%. Such, instability in the price can affect the farmers, the consumers as well as the government. Rate of Indian Rupees (INR) against U.S. Dollars (USD) have been very unstable as well, the rate has ranged from 45 INR to almost 70 INR in a span of 2 years. India's economy is very dependent upon import and export of goods, which is directly affected by the foreign exchange rates. In this thesis, we try to understand the nature of such fluctuations and its effects by monitoring, in parallel, news events. We explore how Web data can be used to identify these changes, understand the impact and in certain cases predict them. The larger goal is to provide a toolset for early detection and proper management of these problems, using the methods proposed in the thesis.

Broadly, this thesis looks at two different types of data. On one side there is a collection of unstructured news streams and on the other side, a set of structured data of various socio-economic indicators. There are many challenges involved in analyzing these unstructured data. These data are large in volume, noisy and do not follow a uniform structure. Our methods are designed to be tolerant to such noises, efficient in dealing with such huge volumes of data. Information extracted from the unstructured news is represented in a reduced form, using events, visualization frameworks and knowledge graphs for easier interpretation of news. Some of these visualizations and knowledge graphs are designed to be human-readable, so that human experts can perform their own analysis with this information, This gives an opportunity for human users to combine external tools and our data for their analysis. Simultaneously, we provide our own frameworks to use the events, knowledge graphs and combine them with the structured data to build predictive and analytical tools for socio-economic indicators. Another challenge is combining the structured and unstructured data. These two types of data are very different structure and properties. Combining them efficiently and effectively are some of the main challenges addressed in this thesis. Figure 1.5 provides a summary of the different reduced forms of online news and the overall flow of the process through these components and how they interact with each other. The unstructured online data are converted into precise form and into different knowledge graphs. These conversions are used to build frameworks for better visualization and search interfaces for this large volume of data. On the

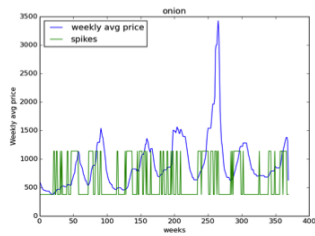


Figure 1.2: Onion Price in India between 2006 and 2012

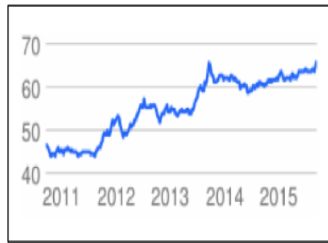


Figure 1.3: Currency exchange rate: USD vs INR

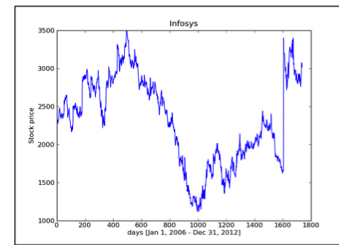


Figure 1.4: Stock Prices of Infosys

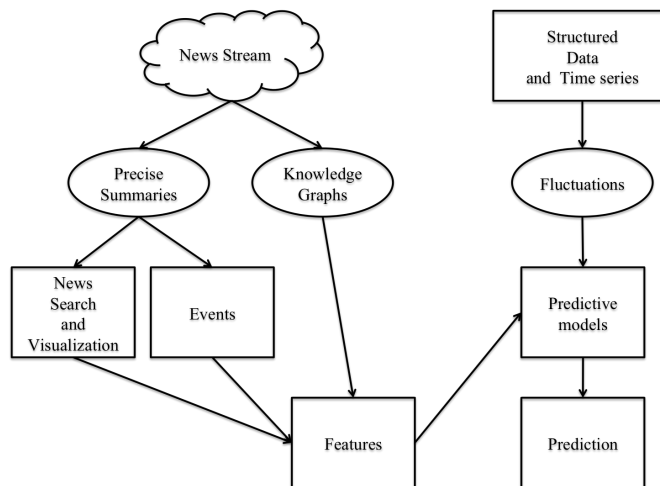


Figure 1.5: Detailed Flow of the Thesis

other hand, the precise form of this data, and the latent relationships are used to extract features. These are effectively combined with the fluctuations in the structured time-series data to learn models that can characterize the fluctuations and subsequently predict the fluctuations.

The larger goal can be achieved by solving four sub-problems – (1) defining a succinct way of representing events from news articles; (2) identifying how events are related to each other; (3) which events are associated with observed fluctuations in an indicator of interest; and (4) how this information can be combined to design predictive models to predict the volatility of a socio-economic indicators. In the remaining part of this chapter we discuss our approach to achieve these steps and at the end we present some inference case studies where our methods were applied and tested.

1.1 Contributions

The main contributions of this thesis centered around addressing these steps:

1. Given a location, time and topic, using the web as the data source, build an automatic inference engine that can identify the critical trends for that location and topic.
2. Process a large corpus of news articles to identify the main events from them and how these events can be classified and represented
3. Building knowledge graphs depicting,
 - (a) Latent relationships between different events and connecting them to external phenomena for prediction
 - (b) human readable relations between news events and external socio-economic indicators (such as, food prices)
 - (c) graphical structure of words appearing in articles to understand causal links between them
 - (d) extracting structures from unstructured domain-specific texts in the form (key, value) pairs
4. Exploiting these relationships to build predictive models for socio-economic indicators
5. Understand and characterize changing patterns in land using satellite images

The following section summarizes the thesis with an introductory summary of each chapter.

1.2 Summary of Chapters

1.2.1 Inference of Location-specific Critical Trends

Given a location, time and topic, using the web as the data source, build an automatic inference engine that can identify the critical trends for that location and topic. In this chapter, we describe the design of a system that mines disparate information sources on the Web to automatically summarize important trends for any specific location and construct a location-specific and topic information portal. We apply this method to monitor key agricultural and climatic trends in India. We have evaluated the system across 605

different districts in India. The results revealed a pan-India scenario of different problem affected areas. The key findings from this work include, around 64.58% of the districts of India suffer from soil related issues and 76.02% have water related problems. We have also manually validated the authenticity of our information sources and validated our summarized results for specific locations with findings in reputed journals and authoritative sources.

1.2.2 Summarization Search

In this chapter we propose a new web search interface that are suitable for regions with infrastructural issues, limiting the access to the Web. Mobile networks in developing regions have been plagued by three basic problems: low bandwidth, high latency and high costs. Due to these factors, mobile web users in these regions are known to have a highly intermittent and relatively non-interactive user experience. In this chapter, we consider the case of mobile search and propose *Summarization search*, a new search paradigm for mobile users where the basic goal is to provide a single summarized and relatively complete search response for every user search query. Summarization search is specifically designed as a relatively non-interactive, one-time quick shot search experience for mobile users where a mobile user can issue *focused queries* for specific information search needs. Our summarization search engine is designed as a meta-search service on top of conventional search services to generate a single summarized and condensed search response based on extracting the “most useful” information from the underlying search result pages. Our system yielded a high accuracy (82-92%) for three classes of focused queries: task oriented queries, category specific queries and focused issue *focused queries* for specific information search needs. queries sampled from the AOL log. We also performed a user study with 400 queries and 30 users. The user study results show that in a scale of 1 (best) to 5(worst), 45% of the cases, users rated the result as '1' and 40% of the cases were rated as '2', with an average rating of 2.09 across all users for all queries. Another task in the user study revealed that for around 55% of queries, users found their information in the summarized result. This demonstrates the effectiveness of the summarization search interface, where users' information requirement is fulfilled with just one operation of submitting the search query.

1.2.3 Extraction of (Key,Value) Pairs from Domain-specific Unstructured Data

One of the larger goal in this thesis is to convert large amount of text available in the web into a precise and summarized form. In this chapter we provide a method that targets specific types of domain-specific text that can be represented using a succinct (key, value) pairs. In this chapter, we focus on the problem of extracting structured labeled data from short unstructured ad-postings from online sources like Craigslist, where ads are posted on various topics, such as job postings, rentals, car sales etc. A fundamental challenge in addressing this problem is that most ad-postings are highly unstructured, short-text postings written in an informal manner with no inherent grammar or well-defined dictionary. In this chapter, we propose unsupervised and supervised algorithms for extracting structured data from unstructured ads in the form of (key, value) pairs where the *keys* naturally represent topic-specific features in the ads. The unsupervised algorithm is centered around building an affinity graph, using the words from a topic-specific corpus of such ads where the edge weights represent affinities between words; the (key, value) extraction algorithm identifies specific groups of words in the affinity graph corresponding to different classes of key attributes. The supervised algorithm uses a Conditional Random Field based training algorithm to identify specific structured (key, value) pairs based on pre-defined topic-specific structural data representations of ads. Based on a corpus of car and apartment ad-postings from Craigslist, the unsupervised algorithm reported an accuracy of 67.74% and 68.74% for car and apartment ads respectively. The supervised algorithm demonstrated an improved performance with accuracies of 74.07% and 72.59% respectively.

1.2.4 Event Analytics and Prediction from News Articles

We propose a novel method of representing news events using the conventional event extraction task parameter – event triggers. The event model is based on the assumption that every news article is about one and only one event and this event is drawn from a larger event class, modeled by similar event triggers. An *event* class represents an abstract grouping of similar events agnostic of spatio-temporal, entity or topic based features. Whereas, an *event* represents an instance of an event class with specific spatio-temporal features along with entities and topics participating in the event. This instance or the occurrence of the event is manifested in a news article. Our model captures the central event (within the headlines or the lead (first) paragraph) as a collection of words/phrases that best describes the main theme of the article,



Figure 1.6: Events related to the phenomenon of onion price rise

called *event triggers*. Subsidiary events in an article are events that are described in the body of the article and are related to the central event; similar to the central event, the subsidiary events are also represented by a collection of words. Our event class model can model any events, provided that there exists at least one article in the corpus that is about that event. So, our model overcomes two main limitations of existing frame works – lack of flexibility and reliance on external knowledge bases and ontologies [73]. As our model is capable of extracting generic events from any corpus and these events are not specific to any domain or class.

Events extracted from news can be used to build a succinct representation of large amount of news. Such a representational framework can have numerous applications. In this chapter, we also show how these events can be used to characterize and predict fluctuations in macro-economic indicators. Many macro-economic indicators are sensitive to real world events. Proper characterization of events can help to identify the events that drive the fluctuations in these indicators. In this chapter, we present models based on our event classes, designed to predict the fluctuations in macro-economic indicators based on event occurrences. We evaluate the models by predicting price fluctuation points of four crops and stock prices of four companies in India using a 7 years of news articles published from India. Experimental results show that our model demonstrated an improvement of 5-10% over baseline systems, including an LDA-based predictive model. Experimental results show that our model can predict sudden fluctuations in food price with an accuracy of 62% and stock prices with an average accuracy of around 64%. There are some additional contributions of this chapter.

Event Phenomenon Graphs

There are many phenomena that have a direct or indirect influence from external events. For example, a sudden drop in a company's stock price can be an effect of a negative news about the company (e.g. the CEO got arrested). Traders, who are interested in a company's stock will be interested to know what events might affect the company's stock prices. For many such phenomena, there might be a set of events that are quite likely to influence its trend. Event phenomenon graph addresses this question – what events are typically linked to a sudden change in a specific variable or index. For a given phenomenon related to a variable (e.g. sudden drop in a company's stock price), such a graph can be built by observing a sufficiently large data from the past containing many occurrences of the phenomenon. Based on the time and the location of the phenomenon, finding the events that are potentially related to the phenomenon can be obtained by finding which events are more likely to occur when the phenomenon was observed than the phenomenon was not observed. An example event-phenomenon graph is shown in Figure 6.5.

Word Causality Graph

In general, studies exploring word-word relations within a corpus are limited to the relations within a sentence or consecutive sentence. There are some models, where association between words are not limited to a sentence or within a local region of a document, sometimes across documents. LDA type topic models is an example of such a model. Where many words are clustered together to form topics. Here, the words relations are defined as how they combine with different proportions to form different topics. We extend the idea of word-word relations into a completely different aspect. We explore how words are related across time. In other words, if we observe a set of words at a particular time t , we ask what are other words that are likely to be observed at $t, t + 1, t + 2$ and so on. We build time-series of the probability of different words appearing across time and compare these time-series to find how words co-vary across time. Word time-series can be used to find correlation or causality and derive different types of temporal relations between words.

We use a news corpus for 7 years between 2006 and 2012, and use the articles published across these years to build a vocabulary and the temporal distribution of that. The time-series of words are built based on the relative frequency of these words for a given day. We apply different methods to find association between words, which includes, correlation coefficients, likelihood ratio, mutual information. All these

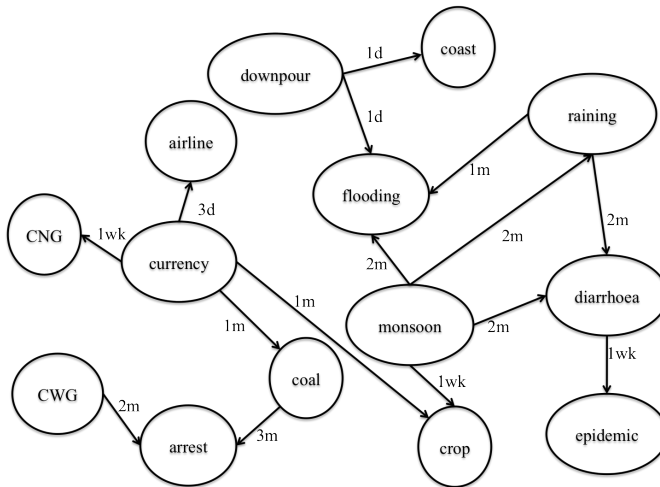


Figure 1.7: Word causality graph

methods can help in finding mutual dependence between words. However, these methods are unable to find causal links between words. We employed a particular causality test, called Granger Causality to exploit the causal links between various word pairs, if exists. Using successful causal pairs of words we constructed a causal word graph that summarizes all the causal pairs found in the corpus. A small part of the larger word causality graph is shown in Figure 1.7. Chapter 6 explains the causal graph in much more details.

1.3 Inference Case Studies

We apply some the ideas presented in this thesis in a couple of real-world applications. In this part of the thesis we present two such applications using other data sources apart from news

1.3.1 Concept Graphs and Comprehension Diagnostics for TextBooks

Good textbooks are organized in a systematically progressive fashion so that students acquire new knowledge and learn new concepts based on known items of information. We provide a diagnostic tool for quantitatively assessing the comprehension burden that a textbook imposes on the reader due to non-sequential presentation of concepts. We present a formal definition of comprehension burden, build a concept graph and an algorithmic approach for computing it. We apply the tool to a corpus of high school

textbooks from India and empirically examine its effectiveness in helping authors identify sections of textbooks that can benefit from reorganizing the material presented.

1.3.2 Satellite Image Analytics and Land Change Patterns

Changing patterns in agricultural land availability is one of the fundamental problems that impacts food security in developing regions like India. Rapid economic growth coupled with increasing populations and changes in climatic patterns are among the main factors impacting availability of agricultural land. In this chapter, we present the design of a satellite image analytics engine that we use to perform a detailed analysis of changes in agricultural land patterns over a 13-year time period (2000-2012) in West Bengal, India, traditionally considered one of the most fertile areas in the world. Our satellite analytics engine can perform a fine-grained analysis of macro-granular satellite images and classify small portions of land in each image into different categories: agricultural, developed, forest and water bodies. Our analytics engine can analyze temporal changes in land patterns and compute the percentage of change in land under each category. Based on detailed food production data gathered in collaboration with the bureau of statistics of West Bengal, we analyze the correlations between changes in agricultural land patterns and corresponding changes in food production (normalized by change in yield patterns). This analytics tool is targeted for government and non-governmental policy makers to analyze land pattern changes and correlate them with food security metrics.

Chapter 2

Related Work

2.1 Large Text Representational Schemes

Recent advances in text mining techniques have produced many sparse, low dimension representational schemes for text documents. Topic models, such as pLSI [39], LDA [12], where a large corpus of documents with a vocabulary $\sim 100,000$ words can be represented using ~ 100 topics. These models have made knowledge acquisition from natural language text easier and more effective. Such representation of documents has been effectively used in online news articles. TAM [47] or TM LDA [98] modeled the temporal aspect of topics. MedLDA [107] uses maximum margin classifier jointly modelled with topics to build predictive models for categorical and continuous variables. Vaca et al [95] used a collective matrix factorization method to track emerging, fading and evolving topics from news streams. Shahaf et al. [83] developed a metro maps like visualization scheme to understand how news articles are connected to each other, which helps in understanding the news better.

2.2 News Analytics and Applications

There have been some works that have news data in building novel applications. Radinsky and Horvitz [73] proposed a framework to predict events from news data, such as predicting disease outbreaks after natural phenomena. The work by Rudin et al [76] involves predicting the next event in a sequentially organized data using association rule mining and Bayesian analysis. Amodeo et al [7] proposed a hy-

brid model to predict future events using a New York Times corpus. FBLG [19] focussed on discovering temporal dependency from time series data and applied it on to a Twitter dataset about the Haiti earthquake. In a similar work by Luo et al [55] showed the correlation between events and time-series data. Hogenboom et al evaluated the effects of rare news events on stock prices and eventually improved the performance of Value-at-Risk (VaR), a popular tool for assessing portfolio risk [40].

Some works have focussed on predicting specific variables from news data, such as stock price. Hagenau et al [35] proposed a new scheme to include context from financial news and market feedback to better predict stock prices. Ming et al [59] used WSJ corpus and sparse matrix factorization to predict stock prices and Gidofalvi [31] also used financial news to predict volatility of stock prices. Si et al [89] used text sentiment for stock price prediction. Other works emphasizing on predicting stock prices include [104][14][25]. Similar works have been proposed for political indicators [28].

2.3 Event Extraction

Specifically, extracting events from news has witnessed a larger number works. There have been two different methods of identifying and extracting events – data-driven and knowledge-driven approaches. Data-driven approaches are commonly used for natural language processing applications. These approaches rely solely on quantitative methods to discover relations. Data-driven approaches require large text corpora in order to develop models that approximate linguistic phenomena within the text. Most of the data driven approaches use statistical and machine learning methods to find association between entities in the text and eventual discovery of event elements. There are several examples found in the literature which use data-driven text mining approaches for event extraction. For instance, in their 2009 work, Okamoto et al. [66] used hierarchical clustering technique to identify local events. Liu et al. [53] used weighted undirected bipartite graphs in addition to clustering to extract key entities and significant events from news items. Clustering techniques are also employed by Tanev et al. [92], in their work on real-time extraction of violence and disaster events. [16] [16] applied word-based statistical text mining in their work. A major fallacy of these data-driven approaches is that they ignore the semantics present in the text mined [41]. However, since these approaches are not based on knowledge, neither linguistic resources, nor expert (domain) knowledge are required.

In contrast to data-driven methods, knowledge-driven text mining is often based on patterns that ex-

press rules representing expert knowledge. It is inherently based on linguistic and lexicographic knowledge, as well as expert knowledge on the domain of the text. Majority of these works employ lexico-syntactic patterns [20][21] or lexico-semantic patterns [22]. The former patterns combine lexical representations and syntactical information with regular expressions, whereas the latter patterns also make use of semantic information. Semantics are usually added by means of knowledge base or linguistic resources [23]. Nishihara et al. [63] extracted personal experiences from blogs by means of three keywords (place, object, and action) that together describe an event, using lexico-syntactic patterns. [29] [29] and Hung et al. [43] also focussed on lexico-syntactic pattern matching for event detection. The work by Xu et al. [100] is another example of using patterns for mining very specific prize award events. They used seed patterns and learned more patterns from the text data. The disadvantage of using knowledge based approaches is the requirement of lexical knowledge and prior domain knowledge. Another disadvantage is that patterns are often hand tuned and are aimed for very specific types of events, making scalability an issue.

There are numerous other data-driven approaches using different techniques and data sources to extract events and apply them in different contexts. Radinsky and Horvitz [72] used a news corpus to extract events predict the occurrence of certain events days before they actually occurred. Similar works include finding similarities in events related to international conflicts [75] or riots [51]. Different types of data have been used for similar event related analysis. For example, using query logs to predict flu trends [33][42] or finding political issues [99], using social media (e.g. Twitter) to predict stock prices [93][88].

The methodology of this thesis has also been motivated by the concepts of topic modeling particularly LDA [12]. There have been numerous variations of LDA proposed, such as supervised LDA (sLDA) [11], correlated LDA [9] MedLDA [107] etc. Some of the LDA based models incorporated temporal information to include temporal dynamics of topics, as a result these models bring “event” like aspect into topics. In general, all these models fall into two categories. The models in the first category do not impose a global distribution assumption about how topics evolve over time. In other words, these models assume that topics change over time depending on their previous conditions following a Markovian process. The examples in this category are Dynamic Topic Model (DTM) [10], proposed by Blei and Lafferty and Continuous Time Dynamic Topic Models (cDTM), proposed by Wang et al [97] , embedding state-space models into topic models. The second category of models usually imposes a global distribution of temporal dynamics. For instance, Wang et al. introduce a beta distribution over timestamps and

incorporate it into the standard topic model. Masada et al. assume a Gaussian distribution over the whole time-line of topics. Based on the two basic categories, other extensions are proposed. For example, Nallapati et al. and Iwata et al. tried to model topic over time using a hierarchical structure.

2.4 News Visualization

New analytics and visualization has been a prominent area in information retrieval and knowledge discovery. Shahaf and Guestrin [82][85] developed a framework to find event chains across news articles to build story lines and building metro map like visualizations. Ahmed et al [5] presented a unified framework to group incoming news articles into temporary but tightly-focused storylines, to identify prevalent topics and key entities within these stories, and to reveal the temporal structure of stories as they evolve. Kim and Oh [48] proposed a similar framework, based on probabilistic topic modeling, for uncovering the meaningful structure and trends of important topics and issues hidden within the news archives on the Web. Central in the framework is a topic chain, a temporal organization of similar topics. Another instance of automatic timeline generation is the work by Yan et al [101][102]. They modeled trans-temporal correlations among component summaries for timelines, using inter-date and intra-date sentence dependencies, and present a novel combination. Allan et al's [6] work focused on temporal summaries of news stories as extracting a single sentence from each event within a news topic, where the stories are presented one at a time and sentences from a story must be ranked before the next story can be considered. Mei and Zhai [56] studied discovering and summarizing the evolutionary patterns of themes in text streams. They presented a general probabilistic method for solving this problem through, discovering latent themes from text; constructing an evolution graph of themes; and analyzing life cycles of themes. Gillenwater et al [32] used threads to find relationships between documents. Threads are coherent chains of documents linking related documents together. Yang et al [103] developed a model-based method for discovering common progression stages in general event sequences. They used a generative model in which each sequence belongs to a class, and sequences from a given class pass through a common set of stages, where each sequence evolves at its own rate. NIFTY [90] is a news meme tracker over a large news corpus to track and cluster the flow of such short texts and phrases. PATTY, [61] a system for learning semantic relationships from the Web. PATTY is a collection of relations learned automatically from text. It aims to be to patterns what WordNet is to words. The semantic types of PATTY relations enable advanced search

over subject-predicate-object data. With the ongoing trends of enriching Web data (both text and tables) with entity-relationship-oriented semantic annotations. Yan et al [101] presented a novel framework for Evolutionary Timeline Summarization (ETS). Given the massive collection of time-stamped web documents related to a general news query, ETS aims to return the evolution trajectory along the timeline, consisting of individual but correlated summaries of each date, emphasizing relevance, coverage, coherence and cross-date diversity. ETS greatly facilitates fast news browsing and knowledge comprehension and hence is a necessity.

A main contribution of this thesis is knowledge discovery through news articles, particularly identifying causal dependencies between events. Similar works related to knowledge discovery has also been in the focus. DeepDive [64] is a similar system that does knowledge-base construction (KBC) from hundreds of millions of web pages. DeepDive [64] employs statistical learning and inference to combine diverse data resources and scalable algorithms. REX [27] takes a pair of entities in a given knowledge base as input and efficiently identifies a ranked list of relationship explanations. Furthermore, REX efficiently enumerates and explains the relationship by ranking them considering multiple factors. Jo et al introduced “The Web of Topics”, where topic discovery was combined with the evolution of topics. Although this method was applied on a corpus of scholarly articles, it can easily be extended to topic discovery from news articles.

Extracting information from a huge pool of documents is hard. Having a structural representation in the resulting articles can benefit users in meeting their information needs. There are different methods to find out the a good structure. Capturing relations between individual entities within a collection of news articles [26] may be too fine grained, while relations between large complex and dispersed topics [12] may be too vague. Linear structure have also been proposed to uncover the structure of a story, numerous tools have moved beyond list-output. Many of these approaches boil down to timeline generation. This style of summarization only works for simple stories, which are linear by nature. In contrast, complex stories display a very non-linear structure: stories split into branches, side stories, dead ends, and intertwining narratives. [6][91], Another popular representational choice is a graph [56]. While a graph is surely more expressive than a timeline, these methods offer no notion of path coherence: the edges in the graph are selected because they pass some threshold, or belong to a spanning tree. We believe that the notion of coherent paths facilitates the process of knowledge acquisition for the users. Shahaf [82] which is a combination of multiple linear structures are good for closed stories but cannot tell us what else happening

around the main events and whether those events are linked to the main story

Automatic extraction of information has been a popular area of study in recent years. This topic detection and tracking from online documents, including news articles. LDA[13] and its variations like Dynamic LDA[10] are popular models in this area. There are other models based on LDA that has shown effective results in mining information from online data. Such works include TM LDA [98], which is a temporal topic model capable of tracking topic transitions in social media. Topic chain [49] is another such example and it builds chain of topics with emphasis on long-term topics, short-term topics and shifts in topic focus. A similar work is by [83], which constructs a coherent story by linking news articles for better understanding of news from a huge collection of articles [83]. Another example of a work related to topic transition and evolution is TAM [47]. TAM models trends over continuous time to detect topic co-occurrence distribution over time. Captures topic evolution over time, predicts timestamps of unlabeled documents based on temporal topics etc. Predic Vaca et al [95], focused on discovering and tracking topics from online news streams to find emerging, fading and evolving topics, using collective matrix factorization approach.

There has been numerous works related to prediction of various parameters from news articles. Radinsky et al [74] used a collection of New York Times articles to predict events like drought few days in advance. Other parameter predicted from online news and social media, include stock prices. Zhang and Skiena [104] have used frequency, sentiment polarity, and subjectivity to build sentiment-based market-neutral trading strategy. Other works related to trading and stock price prediction include, [79] [14] [80] [94] [15] [25].

Chapter 3

Inference of Location-specific Critical Trends

In this chapter, we present the design of a simple location-specific inference engine that can automatically crawl web data pertaining to a given phenomenon and extract critical location-specific trends that closely relate to the phenomenon. This chapter provides important motivation to showcase the effectiveness of using unstructured data from the web for understanding spatio-temporal characteristics of important socio-economic indices. We specifically use the case of agriculture in the context of India as an example for the study presented in this chapter. However, the idea can be easily extended to other topics in other regions.

3.1 Motivation

Historically, due to favorable geographical conditions, India's economy has always been dependent on agriculture. Perennial rivers have provided with abundant water, as well as formed fertile basins, both crucial for farming. Consequently, about 55.9% of total land area in India is cultivable [174] and agriculture contributes approximately one-fifth of total gross domestic product (GDP) and it accounts for about 10% of the total export earnings[173]. Also, 58.4% of the population in India is dependent on agriculture[173] for their livelihood.

Various, constantly evolving factors have drastic effects on agriculture, as well as on its influencing parameters, such as soil conditions, water availability, climate etc. Recent studies revealing that agriculture in India is under serious threat due to changing environment [175]. Changing climatic conditions, such as changes in monsoon patterns, is resulting into frequent floods in some regions and droughts in others; Rising sea levels are increasing the salinity of the soil in coastal areas. These changes are leading to low productions in farming. Apart from natural causes man-made changes, like, massive deforestation or increase in environmental pollution can also adversely affect farming. Practices such as, uncontrolled use of chemical fertilizers or pesticides can also decrease the soil fertility, resulting in lower yield in production.

While there is a general awareness of these facts but there is a definite lack of knowledge on how these factors have an effect in agriculture or in particular locations. Hence, most of the problems are not taken care of properly and they continue to cause destruction. A probable remedy would be to acquire knowledge and deduce solutions out of it.

From the above discussion it is clear that, any decline in agriculture can bring upon devastating effects on India's economy. Hence, a system to safeguard agriculture can bring upon positive consequences.

Previously, there have been various attempts to apply Information and Communication Technology (ICT) based solutions for agriculture. The project Digital Green [143] has investigated the option of using videos to disseminate information about agricultural practices and methods among the community. Aavaaj Otalo, is an example of using mobile phones and audio interface to provide information to the farmers [145]. The system aAQUA [144], provides a multilingual web interface for farmers. With the help of aAQUA, farmers can get answers to their queries from remote experts, through rural kiosks. The system AgrIDS [147] is another instance of using information technology based agricultural information dissemination system. AgrIDS is a web-based system, which helps to send agriculture experts' opinions to farmers with the help of human coordinators. They have demonstrated how experts' help through AgrIDS has helped some cotton farmers in India, with better yields and reduced expenditure.

Some similar systems developed or used outside India, include, the web-based application developed by Xin and Hu [146], to assist in agricultural emergency against natural disasters. In this work, the authors used Google Earth application to obtain necessary geographical information to assess disasters and provide a remedy, thereafter. Also, the web decision support system built by Tambour *et al* [148], which focuses on helping farmers with strategic and operational assistance to overcome uncertain environment. Some

works have focused on applying ICT on environment related issues such as water resource management [158]. Data Observation Network for Earth (DataONE) [151] is an initiative to provide, distribute and share information about environment for better understanding and awareness.

3.2 System Details

The overall idea of our system is to leverage the wealth of information available on the Web to automatically construct a location-specific information portal that can summarize the key climatic and agricultural trends in a given location. Our basic system is designed around the following design steps,

- Creating a location specific repository of documents from the Web
 - Identifying target topics and appropriate search queries
 - Downloading the top results using the query set and deriving new features
 - New searches with the derived features
 - Validating the webpages to avoid inauthentic data

- Analyzing the text for identify critical patterns
 - Extracting relevant information from the text
 - Inferencing the main problem areas in the location
 - Summarizing and presenting extracted information and the key trends

For example, if we want to analyze the problems regarding agriculture and climate change in a particular region, say Jabalpur, in Madhya Pradesh, we can search the web for documents about Jabalpur and related topics, using queries, such as, “Jabalpur soil erosion”, “Jabalpur water availability” etc. Relevant information from the downloaded webpages can be extracted and summarized to identify the critical trends regarding the major problems and threats faced in Jabalpur in terms of agriculture and climate change. The overall flow of the system is shown in Figure 3.1.

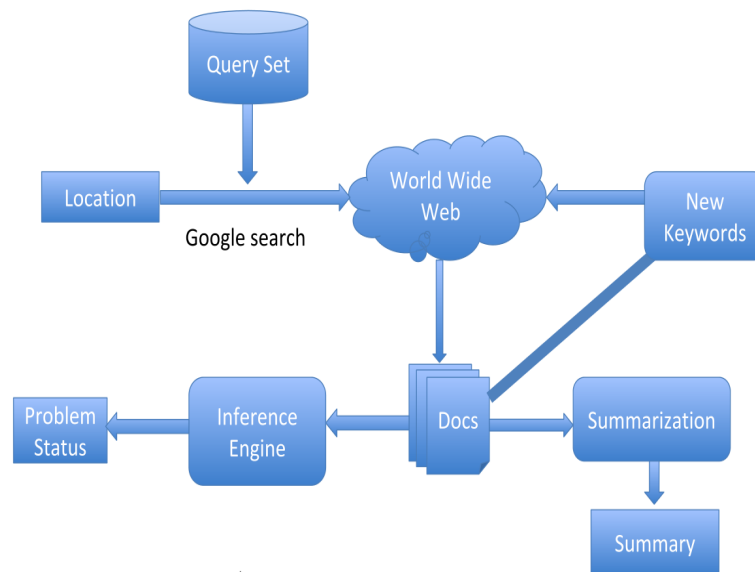


Figure 3.1: Overall Architecture of the System

3.3 Information Retrieval from the Web

This section discusses the process of gathering information of the web. This process has two major steps. Framing a set queries which helped to search for the best resources from the vast pool of documents the Web offers. In the subsequent step some preliminary processing was done to validate the information sources.

3.3.1 Base Document Set

A significant step of this work was to create a channel in the web to get relevant information. In order to obtain such information an important task was to identify some key terms associated with the topic, i.e. agriculture and its dependencies, such as climate. There are various factors on which agriculture is heavily dependent upon. A carefully chosen set of such terms can be used as queries to search and identify meaningful and contextual pages from the Web. Hence, the primary step of this work was to identify the important categories and frame a list of queries which can provide a good description of each category. We identified six major categories influencing agricultural production. These are: soil, water, climate, agricultural practices, crops and pesticides/fertilizers. For a more minute description,

each category was associated with numerous keywords. For example, soil is an important parameter in agriculture. The main characteristics of soil which influence agriculture are soil type, soil fertility, soil moisture, etc. Additionally, there can be some important phenomenon associated with a category. For example, for soil, such a phenomenon can be “soil erosion”. The location, category and a keyword are combined to form the search query. A typical search query to a search engine took the form of “Location + Category + Keyword”. A simple example is “Jhabua + soil + erosion”. The final list had 52 queries across all 6 categories. Table 3.1 shows some of the queries used across the categories.

Note that the set of keywords in this table are only representative and may not be complete. We specifically chose these keywords as one representative sample set of queries to illustrate the power of this approach. This approach can be generalized for a better choice of search queries.

For each query, we downloaded the first 64 documents featured as the top results. So, these 64 documents for each 52 queries served as the initial corpus of documents and a basis for primary analysis. Google API [177] was used as a tool to search the Web.

The process of acquiring documents is summarized below.

Let Q_c be the set of keywords related to the category $c \in C$, where C is the set of 6 categories. For any location Loc ,

For all $c \in C$ do

For all $q \in Q_c$ do

$\{D_c\} \leftarrow searchWeb(Loc + c + q)$

So, $\{D_c\}$ represents a set of documents for category c

The set of documents $\{D\} = \{\{D_{soil}\} \cup \{D_{water}\} \cup \dots\}$ represent the preliminary set of documents after this step.

3.3.2 Extended Document Set

The documents obtained from firing the queries as described in the previous section represent the initial repository. This repository can potentially reveal some critical information which was not actually referred by the preliminary query set. For example, the collection of documents returned by the queries for water in Jhabua, had repeated occurrence of the phrase “fluoride contamination” but the term “fluoride”

Table 3.1: A Segment of the Query List

Categories	Keywords
Soil	Type Quality Fertility Erosion
Water	Ground water level Rivers Dam Irrigation
Climate	Climate Weather Flood Rainfall Drought Deforestation
Agriculture, Crops, Pesticide/ Fertilizers	Crops Crop disease Pesticide Seeds Methods Harvest Fertilizers

was not in the query set. This depicted that the area is suffering from fluoride contamination and this particular problem area was ignored in the primary query set. Thus, “fluoride” can be a new *feature* with respect to the topic in hand.

document set which were outside the query set we used. In order to perform such a task, there is a need to recognize a set of “popular” terms within the document set. For example, a particular set of words appear with a very high frequency that can be termed as important in this context. However, some terms specific to the English language can occur frequently within the text, like “of the”, “and with” etc. Hence, the critical features should have a very high frequency within the documents downloaded but must occur relatively less frequently in a more general place, like in the web.

We used an N-gram based approach to extract these extra features from the initial set of documents. An *N-gram* refers to a collection of N consecutive words in a document. We defined an *N-gram* to represent a *critical trend* associated with a topic if the N-gram had a high rank but was not a very commonly occurring N-gram on the Web. Hence, an N-gram with high Term Frequency (TF) is an indicator of a critical trend. However, frequently occurring *N-grams* could also be just an artifact of the language in which the text is written (for example, the term “of the”). To differentiate those *N-grams* that are important to the topic, we eliminated the N-grams which occurred very frequently in common English texts. The list of the very common N-grams was obtained from the Linguistic Data Consortium dataset. An N-gram with significantly high frequency and not present among the frequent N-grams (from the LDC corpus) is definitely an important phrase within our text corpus. Therefore, this N-gram is one of the features identified.

These extra set of features were included as new keywords and were used to perform an extended search in the web to get additional documents. The new pages downloaded from this second phase of searches were added to the original repository. The process of identifying new features and getting additional documents is summarized below:

For all $c \in C$ do

$\{T\} \leftarrow \text{computeNgrams}(D_c, n)$ [where $1 \leq n \leq 5$]

For all $t \in T$ do

If $\text{freq}(t) \geq \text{freq}_{th}$ and $t \notin LDC$ [where LDC represents the frequent N-grams in LDC corpus]

$\{NewFeatures\} \leftarrow t$

$$\{D_c\} = \{D_c\} \cup searchWeb(Loc + t)$$

End If

End For

End For

At the end of this process, D_c will be an extended set of documents for the category c .

3.3.3 Page Validation

Validation of the information obtained was an essential step as some websites can produce false data. Moreover, analyzing data from the same source can falsely increase the importance of a fact. Hence, to make the information available from our system robust there was an additional step to check for duplicate sources and also their authenticity.

Duplicate Removal Web searches using different but similar query terms might return pages from the same domain. As a result, the same information will be repeated and it will be erroneously considered important for repeated occurrences. However, in reality the same information has been obtained from the same domain and it is a mere repetition of the same facts. Hence, during downloading the pages from the web, the pages were checked for duplicate domains. That is, a page was eliminated if contents from pages from the same domain were selected for a certain number of times previously. Thus, duplicate texts were avoided.

In addition, there was a duplicate checking module which explicitly checked whether duplicate text exists. Application of different semantic, lexical or statistical methods [149][150] can yield better results but we followed a simpler approach for similar text exclusion. We represented the paragraphs using feature vectors. Presence of a feature was represented as 1, and 0 otherwise. Thus, the paragraphs were represented by a string of 1s and 0s. Low distance between paragraphs depicted a similarity between them. Paragraphs whose distances were lower than a threshold were considered for duplicate removal. This step ensured that the information produced by the system was from disparate sources.

Authenticity of the Sites In some websites, like blogs or personal homepages, data can be presented without any authentication. Our system is vulnerable to this kind of incorrect information. One of the simplest way to avoid that was to use the PageRank [179] score of the page. This score gave an estimation

of the reliability and relative importance of a webpage. In our system, we have considered only the pages whose PageRank was greater than 3 out of 10. This simple screening ensured some authenticity in the information retrieved by our system.

3.4 Summarizing Critical Trends

In the previous section, we discussed how information was gathered from the Web. The next step involved identifying the relevant text regions from the documents. In order to realize that, we used the concept of term frequency-inverse document frequency (tf-idf) [178] measurement. The tf-idf is a metric through which the importance of a word in a set of documents can be understood. It is a combination of two separate weights, term frequency (tf), which is a local parameter and inverse document frequency(idf), which is a global parameter. The tf is the frequency with which the word appears in the document and idf is the estimate of the number of documents it appears in. Low value of idf signifies the generality of the term and hence, the low importance in the current context. Higher value of combined tf-idf denotes greater associativity between the word and the document. We were interested in finding the relevant portions within a document. Hence, the concept of inverse document frequency was slightly changed and adapted to find the same measure, but with respect to the paragraphs in the documents. Here, the value of tf-idf signified the associativity of the word with a paragraph in a document.

We calculated the cosine similarity between the query (a combination of the name of the location and the keywords) and the paragraphs of the documents, using the tf-idf value between them. The following formula was used for this computation.

For, a query or a feature q and paragraph p_j

$$sim(q, p_j) = \frac{\sum_{i=1}^T w_{i,j} \cdot w_{i,q}}{\sqrt{\sum_{i=1}^T w_{i,j}^2} \cdot \sqrt{\sum_{i=1}^T w_{i,q}^2}}$$

where T is the number of terms in the query q and $w_{t,p}$ ¹ represents the weight of a query term (t) within the paragraph (p), defined as,

¹This is similar to tf-idf measure but here it is computed w.r.t paragraphs instead of documents

$$w_{t,p} = tf_t \cdot \log\left(\frac{|P|}{|\{t \in P\}|}\right)$$

where, tf_t is the term frequency of t in paragraph p and P is number of paragraphs in the document.

The value $w_{t,p}$ for each query term t denoted the relevance of t in paragraph p . Note that this measure is *inverse paragraph frequency* which is slightly different from the traditional inverse document frequency (IDF) used in information retrieval. Inverse paragraph frequency is the IDF metric applied considering each paragraph as a separate document. This ranking mechanism allowed us to rank at a paragraph granularity to extract the most useful text snippet paragraphs for a given topic and location. The significance of computing cosine similarity ($sim(q, p_j)$) for all query terms t is that it is a quantitative estimate of the similarity between the query and each paragraph of the document. High value of similarity between a query q and a paragraph p_j denotes higher importance of p_j for q . This implies p_j is an important portion of the document for the query q . Hence, p_j should be selected as a part of the information being searched for q .

The decision, of selecting a part of the text as relevant, was based on the similarity value discussed above. If the similarity value exceeded a threshold value then it was considered to be important. The selection criteria was defined as:

$$sim(q, p_j) > Sim_{th}$$

The final value of Sim_{th} was determined after trials with different possible values. The system was evaluated for 10 locations for each experimental value of the threshold. Manually observing the outcomes from these trial runs the final value of Sim_{th} was set to 0.55, which demonstrated relatively better results compared to other values of Sim_{th} . Results were too generalized when the threshold value was less than 0.55. On the other hand, a greater value tended to eliminate important facts. Hence, any information with similarity value greater than Sim_{th} was assumed to have the right amount of details.

3.4.1 Inference Engine

The objective of the inference engine was to identify the main problems in a region with respect to a category from the extracted text of the documents. In other words, provide answers to questions like, “Are there any soil related problems in Jabalpur?” or “Does Sambalpur suffer from water scarcity?”. Inferring such critical facts were based on the following heuristics.

- **Presence of Keywords:** Each topic is associated with a set of keywords. Some of them were framed manually and the rest were automatically extracted from the repository. Presence of such keywords in a text establishes the importance of the sense portrayed by the query terms. For example, from high frequency of the terms “soil erosion” or “fluoride contamination”, the presence of such problems in that location can be deduced. As, we considered N-grams ($N \leq 5$) during feature extraction, repeated mention of “no soil erosion” or “soil erosion was not observed” cannot be mistakenly inferred as soil erosion problems. In such a case the frequency of “no soil erosion” or “soil erosion was not observed” would exceed “soil erosion”.
- **Number of occurrences from disparate sources:** This metric provided an estimate of the popularity of the information. The severity of the problem can be interpreted from the number of disparate sources which reported the problem.
- **PageRank [179] of the pages:** This metric is a quantitative estimation of the reliability of the information source. Verification of the authenticity of the source can help to avoid inclusion of trends incorrectly termed as critical.

The following algorithm was proposed for the inference engine,

Procedure: SummProblems

1. $Keywords_c$ is the set of keywords for category c
[$Keywords_c$ is the collection of both manually constructed and automatically extracted keywords]
2. For all i where $terms_i \in Keywords_c$ do
3. $S \leftarrow \sum_{k=1}^{D_c} [tfidf_k(term_i) + (A_{ik} * PR_k)]$
4. If $S > T_{th}$ then

5. The presence of the problem described by $term_i$ is *true*
6. End if
7. End for
8. Repeat steps 1-7 for all categories $c \in C$

D_c represents the total number of documents gathered for a category c (e.g. soil, water). For example, for soil, D_{soil} is the collection of the documents obtained after firing all soil related queries (such, as soil erosion, soil fertility) etc. $tfidf_k^f(term_i)$ represents the tfidf of $term_i$ with respect to the document D_k . A_{ik} is a binary term and $A_{ik} = 1$ if $term_i$ is present in the document D_k and 0 otherwise. Finally, PR_k is the PageRank value of the webpage from where D_k is obtained. The $\sum(A_{ik} * PR_k)$ represents the sum of PageRank of all the different sources which had these terms (combining PageRank with the number of disparate sources).

The N-gram analysis also helped in avoiding misleading inferences. For example, repeated occurrence of “water shortage” or “soil erosion” can mean presence of these problems in that region. However, the phrases could have been “no water shortage” or “little soil erosion”. An N-gram analysis is resistant to such ambiguities as it involves a larger portion of the text.

3.4.2 Summarization

The final step was to aggregate the whole information gathered from disparate sources and prepare a summarized version of all the relevant information and critical trends and present them to the users. Text summarization involves extracting only the meaningful portions from larger text and prepare a concise form. There are various methods for automatic text summarization [152][153][154]. We adopted a simple *N-gram* based approach.

As discussed in the previous section, we obtained a set of keywords (N-grams) which identified the critical parts of the text. For each location and categories we have set of query keywords (manually framed) and a set of features extracted automatically from the gathered text. Using these key terms the important text regions within the documents were identified and retained them as part of the summarized text. The paragraphs in the documents were ranked on the basis of the occurrences of these terms. The ranking values were computed as,

$$R_p = \sum_{i=1}^n (f_p^i)$$

where, R_p denotes the rank of the paragraph p and f_p^i denotes the frequency of the i^{th} key term in the paragraph p for all the n key terms. The n terms include the predetermined query set as well as the extracted features. Paragraphs with high R_p values are included in the summarized text.

The resulting text is a concise form of the all the text gathered but includes most of the crucial parts. Moreover, this summarized form can be a useful tool for the users to get a quick reference of the important facts.

3.5 Evaluation

In this section, we discuss about the performance of the system. First we discuss the methodology used in this evaluation process. Then we elaborate on some of the key findings from the results returned by the system. Then we talk about the variety and the authenticity of the sources used. Finally, we try to perform a qualitative validation on the information obtained by comparing the findings with established theory and facts published in reputed journals.

3.5.1 Results and Observation

Methodology

The system was applied on 605 districts of India [176], which covers almost whole of the country. For each district, the documents were downloaded and analyzed as discussed in the previous section. The performance of the system was evaluated on the basis of the extracted and summarized information from the results of these 605 districts in India.

Inferencing was based on the method discussed in Section 3.4.1. A location is said to be suffering from a problem (e.g. soil erosion) if the inference module reported the presence of that problem within the document set returned for the location. However, the current version of the inference module is incapable of reporting absence of a problem. If in a location there was no mention of a certain problem the module will cannot assert that the problem is not present there. In such cases the report can be stated as “do not

know” or “cannot decide”. For example, if the system reports 40% of the locations suffer from water scarcity it does not mean the rest 60% is free from water scarcity. It suggests that those 60% of the locations may or may not suffer from water scarcity, the system was unable to infer that.

Results

The summarized text returned by the system revealed some general facts for each location. Like, primary soil type, average temperature, average annual rainfall, normal water availability, major crops, coverage of irrigation etc. In addition, the inference engine reported various problems related to climate and agriculture in these locations.

The system reported that about 64.58% of the districts suffer from soil related problems, mainly soil erosion and infertility and 47.94% of the districts has issues related to crop health, pesticides and fertilizers. The system also reported that 76.02% districts suffer from water related issues, such as water unavailability, droughts, floods etc. This findings are summarized in Figure 5.3.

The pan-India data reveal the graveness of the situation and can act as a serious warning. Almost every region of the country has reported problems regarding soil, water or agricultural issues. This data also signals the necessity of taking immediate steps to prevent further deterioration because any worsening may prove catastrophic.

Particularly threatening picture is obtained from the central part of the country, including the states of Rajasthan, Madhya Pradesh, Chattisgarh, north-west Orissa, Jharkhand and western part of West Bengal. The situation here is far worse than the average national level data. Out **115** districts in this region **103** reported soil erosion, **95** has soil infertility issues, **108** suffer from droughts and only **53** districts has irrigation facilities. Moreover, **86** districts has reported contamination (fluoride or arsenic) in drinking water. These facts vindicate the need for giving special attention to this region. This region is by far the least developed and the most poverty-stricken part of the country. The data revealed from this region sends a strong message to take urgent precautionary steps and to stop further negligence.

A different kind of fact was also reported by the system, how some locations were potentially prone to certain kind of natural calamities. For example, Kutch in Gujrat falls in an earthquake prone zone, proximity to sea makes it vulnerable to cyclones and regular inflows from the sea leads to frequent flooding. Some other such instances include, regular floods in Brahmaputra valley in Assam (district: Nalbari) or devastating cyclones in the coastal districts of Orissa. These calamities cause destruction by damaging

crops as well as affects soil by erosion or increasing salinity (in case of cyclone). Moreover, irrigation is also affected by natural calamities and outbreak of diseases, like malaria are also common.

We also learnt about some location specific factors affecting various parameters. In Faridabad, Haryana, paddy-wheat rotation has caused degradation in soil fertility. Moreover, mine debris blocked feeding channels to drying up lakes, leading to depletion of ground water level. In the regions around Chhatrapur, Madhya Pradesh, numerous ravines created by gully erosion, are heavily under soil loss. Government of Madhya Pradesh has tried to check this soil erosion and expansion of ravines by the means of watershed development and by aerial-seeding for plants like Prosopis, Acacia, and Jatropa in the ravines. In another case, burning is the normal method of rice stubble management in mechanically harvested rice-wheat growing areas of North-West India. This causes air pollution and loss of soil health as well as impacting on animal health.

Similarly, some local success stories also came up within this obtained information. For example, construction of watersheds in Ahemdnagar, helped to fight erosion and drought, both of which are quite prevalent there. Again, Government of Bihar took a positive step when they enacted the Bihar Ground Water act, 2006, which provides mandatory provision of roof top rain water harvesting structures in the building plan in an area of 1000 sq. m. or more. This could potentially resist ground water depletion.

Extraction of Additional Keywords

In Section 3.3.2, we discussed that additional features were extracted from the preliminary set of documents, based on N-grams. Here, in Table 3.2 we present some of the relevant features identified from 4 locations.

From this list it can be inferred that Amreli District in Gujrat and Jhabua in Madhya Pradesh might suffer from fluoride contamination. Further searches based on these terms on the location can reveal more facts about this problem. Again, for Kurukshetra district in Haryana, the major N-grams based keywords identified were related to “ground water”. Hence, we can interpret that the area is suffering from some issues with ground water. For Bharuch district the main keywords were related to Sardar Sarovar dam and floods. This entails that in that area Sardar Sarovar dam is one of the highlighting features there may be some problems related to the dam and frequent floods. This demands further investigations.

The keyword identification module also reported some features like, “Government of Haryana” (for Kurukshetra, Haryana) or “the gulf of Cambay” (for Bharuch, Gujrat). These are contextual features, as

Table 3.2: Keywords extracted from 4 locations

Location	Keywords
Amreli	fluoride levels in drinking water fluoride content in thriving chemical industry
Bharuch	the sardar sarovar dam flood dams have drowned cause disastrous flood dams
Kurukshetra	with ground water pollution which affect ground water
Jhabua	of fluoride exposure levels of fluoride effects of fluoride

Kurukshetra is in the state of Haryana and Bharuch is close to the gulf of Cambay. However, this kind of features are not very relevant in this context and can add noise to the inference mechanism. To avoid such issues, these kinds of keywords need to be excluded. Hence, a future improvement would be to devise a method which would identify and omit such keywords.

3.5.2 Validity of Information

There are chances that false information might penetrate through the system. In other words, how authentic is the data reported by the system? For example, if the system reports that a location is suffering from soil related problems, is it true or a false alarm? To ensure the authenticity of the information made available by our system, we followed two approaches. First, we analyzed the nature of the websites from which the system got its information. Then, we tried to analyze the information qualitatively. We compared the system's findings with scholarly articles on similar topics published in government reports, reputed journals and magazines. It is assumed that these articles carry authentic and accurate information. By finding the similarity between the accounts of these scholarly articles and our system's findings, we verified the authenticity of the facts produced by our system.

Quality of the Information Sources

Here we present some statistics related to the webpages used by the system for retrieving information. These data show the kind of websites used by the system for information retrieval, the authenticity of the webpages, uniqueness of the sources etc. The results are summarized in the following figures.

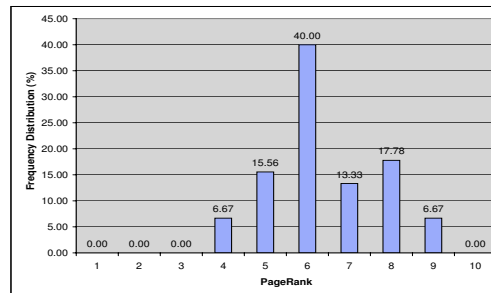


Figure 3.2: PageRank distribution

PageRank Distribution Figure 3.2 shows the PageRank [179] distribution of the webpages used by the system. The average PageRank is **6.4/10**. This is a fairly high value. Usually, pages belonging to reputed institutions which are frequently visited have PageRank value as high as that. Hence, it can be inferred that the system used quite reliable web pages for the information extraction.

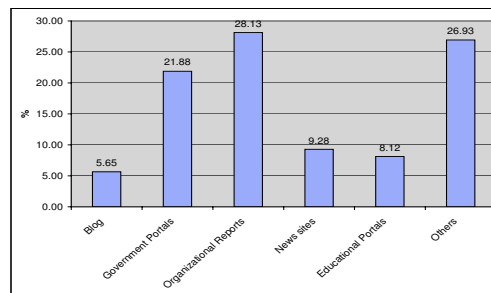


Figure 3.3: Classification of webpages

Diversity of Websites Figure 3.3 depicts the various web sources used by the system and their share of contribution. From the figure it can be seen that the major share of the information were taken from sites like, governmental portals, organization sites, news articles and educational portals. These are quite reliable sources of information and it can be assumed that the information obtained by the system through these sites are authentic.

Main Contributing Sites Apart from the data discussed above, we also tried to identify the top contributing sites. They are as follows,

www.indiaenvironmentportal.org.in, en.wikipedia.org, www.fao.org, agricoop.nic.in, www.tribuneindia.com, planningcommission.nic.in, www.indiawaterportal.org, www.the hindu.com

The list includes some of the specialized environment and agricultural portals from India and also some leading Indian news sites.

Furthermore, we tried to check the number of distinct domains the system used for the whole process. We found that about **83.94%** of the sites were distinct, i.e. only 16.06% pages were repeated during the information retrieval process. This fact proves that most of the information gathered by the system has been from very disparate sources and chances of duplication of facts are very low.

Manual Validation of Information Retrieved

Some of the findings of the system were validated using articles from journals on related topic. For example for many locations, the data presented in Figure 5.3 and other findings from the system were verified from articles published in journals like Journal of Soil and Water Conservation, Journal of Indian Academy of Science, Journal of Hydrological Sciences, Proceedings of International Union of Geophysics and Geodesy, etc . Similarities were found between these scholarly articles and the system's results.

For example, the system reported high fluoride contamination in the water of Jhabua district in Madhya Pradesh. From the system's output,

“In tribal areas of Jhabua district, in which some of the villages were affected by Guinea worm (water borne disease) and high fluoride content in drinking water”

The same fact has been validated from a report published by Central Ground Water Board, Ministry of Water Resources, Government of India [156]. Similarly, other outputs from the system were manually validated by comparing them from different scholarly articles or organizational reports. Table 3.3 shows some of the location-specific problems reported by the system which were verified through this process.

From the above discussion, it can be inferred that the information produced by the system is authentic and can be found mentioned in various scholarly articles or government reports. As, the system is only capable of retrieving and processing html files from the Web and the articles used for manual validation were all in pdf format, there are no chances that the system found the information from the same source. This claim further vindicates the authenticity of the system's findings.

However, there can be more rigorous validations of the system's output. This can be done in variety of ways. For example, evaluating the system's performance by domain experts and taking an opinion score, manually gathering related data from the Web and other sources and finding the similarity between the outcomes of the manual mode and the system's output etc. Such further validations will greatly enhance the system's reliability.

3.6 Summary

This work is based on the premise that information dissemination and awareness can help in preventing major devastation in agriculture due to environment related issues. Our system incorporates a novel scheme where localized information on various agriculture and environment related topics are retrieved from the Web and processed to provide concise information from which critical trends on the topic for a location can be interpreted.

The work has been evaluated by obtaining such information for 605 districts in India. The results demonstrated the some effectiveness of the system in achieving the desired goals. Fetched information revealed various facts about the regions, major areas of concern and their vulnerability to different factors.

These results clearly shows that web data can be a useful resource to make inference about key trends given a topic and a location. However, there are certain drawbacks in the approach adapted in this work. Firstly, the data collection method introduces noise. The topic specific data were collected from the Web using focused queries. This method can retrieve relevant information, however, lower ranked pages for a query can often contain irrelevant information. This might lead to inaccurate deduction by the system.

Table 3.3: Manual verification of the problems reported in some locations

Location	Reported problem	Verified from
Jhabua	soil, water contamination	Govt. report [156]
Vellore	water scarcity	Govt. report [157]
Bidar	crop, water	Govt. report [160]
Malda	soil erosion	Journal article[155]
Haryana	soil quality	Journal article[159]
Nalbari	flood	Govt. report[161]
Bharuch	crop disease	Govt. report[165] Tech. Report[169]
Korba	air pollution, soil	Conf. article[164]
Kurnool	rainfall	Journal article[167]
Kozhikode	ground water	Govt. report[172]
Ratnagiri	drought	Tech. Report[170]
Bareilly	crop production	Tech. Report[171]
Sikkim	soil erosion	Govt. report[162] Journal article[163]
Gaya	fertilizer, soil quality	Tech. Report[168]

Similarly, location-specific queries might not always fetch pages about the exact location. Thus, this might lead to having results about many other locations. Finally, there were no control about the time. As a result a web search query might fetch outdated information. The system was built on the assumption that these factors were not true. Hence, the next logical step is to collect data from reliable sources, where the information is up to date. Also, the documents are clearly marked with location and time so that these parameters can be controlled. Considering all these factors, we narrowed down the data sources to be reliable sources, where all these criteria are met. The remainder of the thesis is about expanding this idea, using more sophisticated methods and reliable data to infer similar trends for socio-economic indicators and other applications.

Chapter 4

Summarization Search

4.1 Introduction

In this chapter, we present a new interface for web search by presenting summarized content of news based on the results returned by a commercial search engine. Using any conventional search engine to search the news presents the user with a list of pages related to the user query. We contend that this conventional search model does not work well for searching online news articles. We further argue that conventional model is has additional disadvantages in developing regions due to poorer infrastructure. Building and expanding infrastructure for networks have suffered from various problems, like difficulties in licensing appropriate portions of the spectrum (for wireless networks), negotiating land use, navigating bureaucracy or corruption [224]. In places with existing network infrastructure, the network suffers from poor coverage, limited connectivity, high round trip delays [223], Intermittent connections, and limited bandwidth [220]. Studies have shown, improving upon these limitations can bring upon significant improvement in web access and usefulness of the web in developing regions [218]. Apart from infrastructure and network related issues, cost is also an important factor in developing regions. Unlike developed countries, subscribers do not pay a fixed value for unlimited data transfer, the operators usually charge on a pay-per-use basis [219]. Hence interactive applications such as search for news from mobile phones can be extremely slow and frustrating to the users. Several prior efforts have proposed important optimizations at different network layers including the design of new transport protocols [225], novel prefetching

techniques [222] [220] [223], delay tolerant network solutions [222] [224] etc.

We propose *Summarization Search for News* as a new paradigm for news search. The basic objective of *Summarization Search* is to provide a one-round search abstraction with the goal of enhancing the utility of a single and concise search response for a user search query. Summarization Search operates as a layer on top of conventional search engines and leverages existing search engines to obtain search results for user queries but aims to build a summarization layer for users that can summarize all the search results in a condensed response for the user. The goal is to either directly answer the user query or to direct the user to the right search result that may best contain detailed information corresponding to information user is looking for. In the event where the information sought by the user is not present in any of the search result pages, one round search is fundamentally not feasible and the user needs to modify the search query.

The search query model resembles the standard web search interface from a user stand-point where the search query is a collection of keywords. We use natural language processing techniques to identify *critical sections* in the result pages returned by a search engine and provide a condensed summary of these critical sections in the summarization search response. The summary contains the important portions from the resulting pages with respect to the query, ignoring less relevant parts. Moreover, in many cases graphical representation of information is as important as textual information and can provide important additional information to the end-user, especially for better judgment on selecting result links for further exploration. Our summarization engine extracts images relevant to the query and the summarized text from the result pages returned and presents these images along with the summarized text to the user. This approach significantly expands beyond the definition of using 1-line text snippets coupled with web pointers where the user needs to browse through multiple result pages and potentially perform multiple rounds of search queries to determine the appropriate response. In summarization search, users can have a look at the most critical sections of page according to the query in the form of summarized text and this summarized text can itself contain the answer to query or can provide sufficient information to better select the appropriate link for exploration. In contrast to the snippets, where it only helps the user in deciding whether to visit the actual page, summarization search potentially provides that information from the target page into the search result page.

4.2 Methodology

In this section, we focus on the methodology used to obtain a condensed response following a web search query submitted by a user. As a precursor, our summarization engine uses the Google Search API to download the the top 64 (configurable) search result pages. The key goal of the summarization engine is to perform detailed text analysis of all the result pages to prepare a condensed summary page across all the search result pages. Final search response is condensed version of the original text retaining only the critical portions from the larger text, relevant images and corresponding pointers to the individual search result pages, if user feels the need to click on any result page. The three core design components of the summarization search engine are: (a) Text Summarization Engine which extracts relevant summaries of each page; (b) Image Extraction Engine which extracts relevant images from the page; (c) Aggregation and Presentation layer - condense the summaries across pages along with relevant images. Next, we describe the scope of each component individually.

4.2.1 Text Summarization Engine

Text Summarization is a process of creating a concise version of a larger text preserving the main theme of the original text. There are different ways of summarizing a text. Two popular methods used in the NLP literature are Extraction based summarization and abstraction based summarization [211]. Our text summarization uses the extraction based summary approach which involves creating the summary by using exact sentences from the original text. This involves identifying the key sentences/phrases from the original text and using them to form the summary. Our summarization is different from standard extraction based summarization task. Unlike the standard task the extraction in our case is driven by the query terms. The summarized form is supposed to highlight the portion of the documents which have high relevance with respect to the query terms. Our extraction based summarization algorithm has two steps: (a) identification of key terms in a document w.r.t. the query terms; (b) identification of key portions of a documents using the key terms identified in the previous step. The simplest way to perform such a task is to consider only those sentences which contain the query terms. However, such an aggressive summarization approach can significantly reduce the accuracy of the task. Therefore, it is important to identify other phrases within the text which carry significant information. Sentences containing these

terms will help understand the information rich portions within the text and subsequently extract them to prepare the summary.

To identify important key terms in a document we use a natural language parser to parse each sentence. This step breaks down each sentence into noun phrases, verb phrases and prepositional phrases etc. At the leaf level, the tree represents each word of the sentence along with their parts-of-speech tag. The parse tree of the following sentence is shown below

“More than 1,000 Occupy Wall Street protesters have blocked cargo trucks at busy US west coast ports.”

(ROOT
(S
(NP
(QP
(XS (JJR More) (IN than))
(CD 1,000))
(NNP Occupy) (NNP Wall) (NNP Street) (NNS protesters))
(VP (VBP have)
(VP (VBN blocked)
(NP (NN cargo) (NNS trucks))
(PP (IN at)
(NP (JJ busy) (NNP US) (NN west) (NN coast) (NNS ports))))))
(.)))

In the parse tree NP, VP and PP represents noun, verb and prepositional phrases respectively. NP, JJ, VP etc. are parts of speech tags of the words. We parse each sentence and consider only the noun phrases and the verbs from each sentence. The noun phrases are the most informative parts of the sentences and the verbs, which depict the actions have relevance too. After the parsing and extraction of relevant portions of the sentences, as a second step of eliminating unnecessary words, we used a list of English stopwords [200] and the final list of words were taken as the candidate set of words. A simple measure of popularity of a term is the term frequency. As the target set of documents has resulted from a web search query, words with high frequency are contextually important. We normalize the frequency of every candidate term t_i across all documents using:

$$P_{corpus}(t_i) = \frac{n(t_i)}{\sum_i n(t_i)} \quad (4.1)$$

In spite of taking care of eliminating commonly used English terms, a frequency based approach can include words that are still artifact of the language. Hence, to eliminate any such bias another metric was introduced. The new metric was the probability of the term occurring in the Web. For a given term, the Microsoft Ngram Web Service ¹ was used to obtain its web probability. Let $P_{web}(t_i)$ represent this measure for term t_i . High value of the web probability represents the popularity of the term. This means it has relatively less importance in this context because it tends to appear very frequently in any large English corpus of text. On the other hand, a term with comparatively lower web probability but has higher probability in the search result text means it has more relevance in the current text. Hence, it should be given more importance. So, the net importance score of a term, $Imp(t_i)$ appearing in the search corpus is given by:

$$Imp(t_i) = P_{corpus}(t_i)/P_{web}(t_i) \quad (4.2)$$

This score is similar to TF-IDF but in this case all the documents is a focus set of documents resulting from a query. Hence, many important terms might have high document frequency and using TF-IDF weights will lower their scores. After computing the importance score for each individual term, their score was normalized across the the terms in the following way,

$$Imp_{norm}(t_i) = \frac{Imp(t_i)}{\sum_i Imp(t_i)} \quad (4.3)$$

The normalized score represent the relative importance of a particular term compared against all other terms. Manually inspecting the terms and the distribution of the importance scores for the text generated from different queries, we observed empirically that usually the top 20% of the mass represents a meaningful set of terms. Hence, we consider the top 20% of the terms to represent the candidate term set of the document². Hence, any sentences containing these terms are important sentences in the text and should be included as part of the summarized text.

Let $V = \{t_1, \dots t_m\}$ represent the set of chosen candidate terms. To evaluate the importance of the sentence s_j , we computed the similarity between each sentence of the text with the set of key terms (along

¹<http://research.microsoft.com/en-us/collaboration/focus/cs/web-ngram.aspx>

²In all the cases the original query terms were also found to occur in this set

with their importance score) from the text. The score of each sentence is defined as,

$$Score(s_j) = \sum_{t_i \in s_j, V} w_t \times Imp_{norm}(t_i) + \sum_{t_q \in s_j, Q} w_q \times Imp_{norm}(t_q) \quad (4.4)$$

where $t_q \in Q$ is a query term belonging to the query Q . Query terms and non-query terms are assigned different weights, where the weight for a query term (w_q) is greater than the weight of a non-query term (w_t). This ensures that the sentences containing query terms are given additional weights. The extracted summary is a collection of sentences keeping two constraints – maximizing the sentence scores while minimizing the size. The sentence score is a monotonically increasing function, so adding more sentences will increase the score, hence the entire document will have the maximum score. However, this will defeat the purpose of creating the summary. The optimum summary should have maximum score while minimum size. Our method uses the “elbow rule“, which involves incrementally inserting a sentence in descending order of score into the summary until the increase in score is nominal or below a threshold. At his point only the size of the summary is increasing without sufficient information content. In the final summary, the sentences are placed in the order in which they appeared in the original document. The algorithm summarizing the entire process in presented below.

Here, the summarization process highlights the critical sections from each document with respect to the query and the final text presented is a collection of the critical summaries from every document. Thus, in the summarized result the boundaries of the documents are preserved. An extension to this approach is creating a single summary from multiple pages among the top results. This extension will result in much less exploration required by the user. This will result in the user viewing a condensed version of all the top result pages instead of going through the summaries individually, thus further reducing the time spent to search for the information.

4.2.2 Image Extraction Engine

The process for extracting relevant images pertaining to a query is not an easy task in a web page especially since the visual layout of a webpage may have little resemblance to the actual DOM structure of a page. Ideally, the image from a result page is extracted only if it is relevant to search query and if the position of image is in close proximity to the summarized text generated by NLP text summarization engine for that particular page; however, identifying the close proximity between the text and its corresponding images may not be an easy task. We use a combination of two parsing approaches to extract

Algorithm 1 Summarization Search

procedure SUMMARIZE(query = q , document = d)

for each $t_i \in V$ **do**

 compute $Imp_{norm}(t_i)$ [Eq 4.3]

end for

for each $s_j \in d$ **do**

 compute $Score(s_j)$ [Eq 4.4]

▷ the terms in query q is used in Eq 4.4

end for

$\mathcal{S} \leftarrow \text{sort}([\forall s_j \in d], \text{key} = \text{Score}(s_j), \text{order} = \text{Desc})$

$\text{summ_score} \leftarrow 0$

$\text{summary} \leftarrow \text{empty_list}$

for $s_j \in \mathcal{S}$ **do**

if $\Delta \text{summ_score}_{k-1:k} \leq \epsilon$ **then**

 break

else

$\text{summ_score} \leftarrow \text{summ_score} + \text{Score}(s_j)$

$\text{summary.append}(s_j)$

end if

end for

$\text{summary} \leftarrow \text{sort}([\forall s_j \in \text{summary}], \text{key} = j, \text{order} = \text{Asc})$

▷ j = sequence no. of the sentence in document d

 return summary

end procedure

images relevant to a given query: (a) Open Graph Protocol Approach; (b) parsing the DOM structure.

Open graph protocol (OGP): The Open Graph Protocol enables any web page to become a rich object in a social graph (<http://ogp.me>). The web pages conforming to OGP defines basic metadata such as title, type, image and url using predefined OGP tags. Following is the OGP markup for “The Rock” movie page on IMDB:

```
    <html prefix="og: http://ogp.me/ns#" />
  <head>
    <title>The Rock (1996)</title>
    <meta property="og:title" content="The Rock" />
    <meta property="og:type" content="video.movie" />
    <meta property="og:url" content="http://www.imdb.com/
    title/tt0117500/" />
    <meta property="og:image" content
    ="http://ia.media-imdb.com/images/rock.jpg" />
    ...
  </head>
</html>
```

This metadata defines how this particular web page wants to be represented in the social graph and thus represents accurate information about the webpage. The image tag refers to the image which best represents the page. This metadata can be different for every webpage of a single domain. For example, if a news website conforms to OGP then it may define metadata for all news articles differently as the title and image for every news article will be different. Hence, if the result set of the user query includes some content from this embedded web page (which appears as an OGP object), then the corresponding image is the best candidate image in conjunction with the summarized information. The advantage of using this approach is to simply check whether the webpage is conforming to OGP or not. However, not every site may follow the OGP protocol.

Parsing the DOM: If a webpage does not conform to OGP then the image needs to be extracted by parsing the DOM structure of webpage and finding the image which is relevant to query and is in

close proximity to the summarized text of that webpage. The objective is to find images in the DOM structure around text which has been included as part of the summary by the text summarization engine. The proximity of an image with the summarized text is measured by finding the difference in position of both elements in the DOM Structure. Once we locate the relevant element in the DOM structure, the siblings of the text-element are navigated to search for images. If an image exists among the siblings set, then a score is assigned to image equivalent to the number of navigational steps. If none of the siblings of text-element contains an image, then we navigate through the siblings of the parent of text-element. If image is located among siblings of parent then we mark the image and assign it a score as in a similar fashion. If image could not be located even in the siblings of parent element then we abort our search for relevant images. The same process will be repeated for all texts that are part of summarized text.

Finally, image with the lowest score is selected to represent the content of webpage since the one with the lowest score is the closest to the text included in summary. The downside to this approach is that relevant images can be missed if they are not present among the siblings of the text-element or siblings of parent of the text-element but we found anecdotally that this approach seems to effectively extract relevant images in most cases.

4.2.3 Aggregation and Presentation

Given the summarized text and images from the individual search result pages, we create a single large document contain all the summarized texts. The aggregation step runs the same text summarization algorithm described earlier on this aggregate document of web page summaries. This step is essentially extracting a multi-document summary across the important paragraphs from individual webpages. The end-result of this step is the top paragraphs from each document that seem most related to the query is identified and collected together to form the multi-document summary. Corresponding to this final filtered set of paragraphs, we order them in the same order as to how the search engine ranked the individual pages and annotate the paragraphs with the corresponding images extracted by the image extraction engine. Each paragraph is also directly linked with the title and the URL of the result page. In case the information summarized is not adequate for the user, the user can use the links to directly navigate to the corresponding result page.

4.3 Evaluation

The goal of the summarization search engine is to maximize the information delivery from a web search query by minimizing the size of the data transaction and the number of user interaction. In developing regions, this can improve user experience as, any interaction with the web suffers from high delay due to poor infrastructure. Moreover, the summarization search engine is designed as a layer on top of any standard web search engine. It uses the ranking, indexing, and searching techniques of the underlying search engine. Thus, the aim of this evaluation is different from any standard information retrieval techniques. The effectiveness of the summarization search engine is evaluated by measuring how fast it can provide the users with results and whether it is decreasing the number of interactions between the user and the web.

To evaluate our system, we need an appropriate set of focused queries. We defined three different classes of focused queries for our evaluation:

Focused queries sampled from the AOL query log: Based on the publicly available AOL query log, we took a small sample of 150 focused queries. In these queries the user intent was clear.

Category specific focused queries: Popular search engines publish a broad array of categories or information topics online. We chose 25 popular categories from this set and 3-4 queries per category.

Task oriented queries: We considered a range of day-to-day activities of users and defined simple information need tasks around these activities. We chose 50 task-oriented settings where a user may search for specific type of information and the corresponding queries.

To evaluate our system for each query, we defined upfront "what is the explicit information need" from a user standpoint and our goal was to test if the summarized search response satisfied the information need or not. There are two basic ways for defining success: (a) If the summarized page has a summarized text that satisfies the information need; (b) The summarized text provided sufficient information for the user to determine the appropriate link to obtain the information in one click from the result page. In a more general notion, the number of explored pages per search query is a measure how successful the summarized result pages were. It also shows how fast a user gets the information once submitted a query. Reducing the amount of data transaction is another goal of this new layer. All this metrics provide a good analysis on whether, this search interface can be effective in developing regions, by reducing total amount of data transacted as well as lowering the number of interaction. Thus, a web search is not bottlenecked

by network related issues prevalent in developing regions.

Finally, we conducted a user study to collect a set queries from users, asking from what types of queries they typically submit from mobile devices. The evaluation of the summarization search interface using these queries were done using another round of user study.

In the remainder of the section, we present our findings from these experiments.

4.3.1 Focused Queries from AOL Logs

The objective of summarization engine is to provide information enough to conclude search in one round or at-least offer information enough for user to select the right link for further exploration. Search responses generated for these queries were analyzed to check whether summarization search engine achieve its objective just described. The results of AOL Queries are shown in Figure 4.1. Out of 150 queries, results of 56 queries retrieved information that user was looking for and thus no further exploration was required. Search response of 76 queries contained enough information to enable user to select the most appropriate webpage for that particular search query and search response of 18 queries could not provide any relevant information. In summary, for the AOL sample query logs, for 37% of the queries, no further browsing was required and in total for 88% of the queries, the appropriate search response could be achieved within a single click and the search could be concluded in one round of interaction. For a significant fraction of the 12% of queries which were not satisfied, the summarized page and the result pages did not accurately capture the information need we had set upfront for the query.

4.3.2 Category Specific Information Queries

We chose a broad array of popular categories of information determined by search engines such as Sports, Movies, Health etc. Within each category, we created a small sample of 3-4 queries and create a pool of 90 queries as part of this category-specific query dataset. Here again, the information need for each query was defined in advance and the search responses to the queries were analyzed to check if they conform to the objective of the summarization search engine. The results of Category Specific Information Queries are shown in Figure 4.2. Out of 90 queries, results of 45 queries retrieved information that user was looking for and thus no further exploration was required. Search response of 40 queries contained enough information to enable user to select the most appropriate webpage for that particular search query and

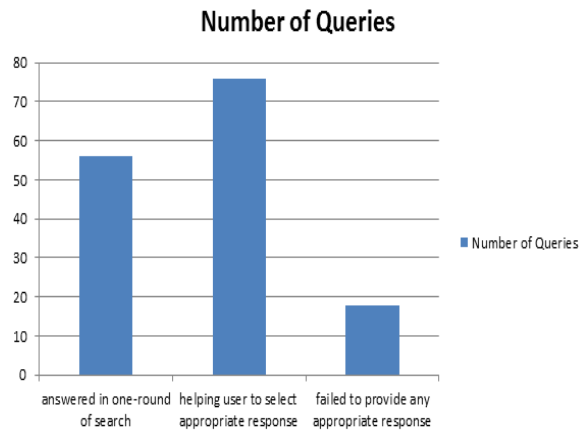


Figure 4.1: AOL Queries Results

search response of 5 queries could not provide any relevant information. For these queries, 50% of the times no further browsing was required. In other words, 94% of the times search was concluded in one interaction or user just needed to browse one more webpage. Like result of queries from AOL Logs, these results assert that summarization search paradigm enhances the utility of web-search for mobile users.

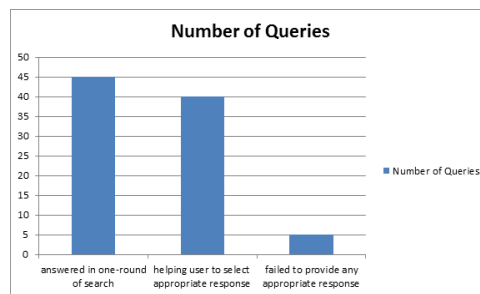


Figure 4.2: Category Specific Information Queries Results

4.4 Summary

In this chapter, we presented *Summarization Search* as a new search abstraction for mobile users and show how this abstraction could be more effective in answering to web queries in developing regions, where web access suffers from a variety of infrastructure related issues. The summarization search was designed

to mimic the conventional web search queries; in this case, our system provides a condensed summary of the relevant paragraphs and images from the search result pages. The condensed summary generation algorithm leverages automatic text summarization to extract and summarize a collection of search result pages for a query. Here, we also observed that adding images to the structured response can enhance the usability and utility of the system. In our evaluations across a diverse set of queries, roughly 89% of the queries in one or maximum two rounds of search interactions. We conducted a user study to further evaluate the summarization search interface. Around 85% of the queries received an average rating of 1 or 2 in a scale of 5 (1 is best) and around 55% of the queries received a majority rating of 1 from all users. These results vindicates the fact that, in developing region, summarization search can be a useful tool to search the web using mobile devices, because it requires less data transactions and lesser number of interactions to meet the users' information needs, thus circumventing the important limitations of web access in those regions. In the following chapter, we present a similar concept of summarizing text for better search experience. We focus on specific types of websites and content, where there is an inherent structure and try to automatically extract that structure for a more succinct representation of the text.

Chapter 5

Extraction of (Key, Value) Pairs from Domain-specific Unstructured Data

5.1 Introduction

This chapter aims to address the problem of extracting structured information from domain-specific unstructured data in the form of $(key, value)$ pairs where a *key* represents a specific attribute about the underlying domain with a corresponding *value*. This problem is largely relevant in the content of the current Web where large volumes of user-defined information in online portals like Craigslist are in the form of unstructured short-message ad-postings. These ad-postings are written in an informal style without a well defined dictionary or grammar and the quality of the textual data tends to be highly variable and noisy; in addition, different users may use different abbreviations and formats for conveying the same information. While humans can easily interpret such ad-postings, it is hard for an automated tool to perform data analysis on them. To enhance machine analysis of such unstructured postings, a critical problem to address is to be able to convert these ad-postings into structured data with defined keys and values.

Given a corpus of ads under a topic (such as car ads from Craigslist), our goal is to extract a structured feature-based representation for all the ad-postings. Specifically, we want to convert any unstructured ad to a set of $(key, value)$ pairs where *key* refers to a specific feature corresponding to the topic and *value* represents specific descriptive information for that feature. Solving this problem has several important

practical ramifications including enabling a range of advanced search options which can be tailored for topic-specific features; currently many of the advanced search options for unstructured ads are constrained only for specific features (such as cost [price, rent etc.], model, year) with simple field extractors. Our approach can enable the users to search using more specific terms and constraints as well as improve the quality of the results based on existing options.

Our problem is different in spirit from prior work on extracting structured information from unstructured text [34][36][57][24] due to our focus on unstructured ads and the task of extracting (key,value) pairs from these postings. While specific prior work [34][57] have also examined unstructured ad-postings, the underlying focus of these works have been different from our work. The focus of [34] was to extract field structures from unstructured texts using small amounts of prior knowledge while[57] proposed building a relational database from such texts using external knowledge bases.

This chapter proposes an unsupervised and a supervised algorithm for (key, value) extraction from unstructured ads. In most ads the object advertised is described using some standard features of the object. For example, for apartment ads such features include apartment size, apartment rent, location, number of bedrooms etc. Every ad under a specific topic contains such a generic template and a particular ad is a full description of those features but presented without any proper format. We try to capture this inherent structure in the form of (key,value) pairs, where *keys* are the features (e.g. apartment rent) and *values* are a specific value (e.g. \$2000) as advertised. The converted structured form of an ad is represented as a set of various (key,value) pairs. The unsupervised algorithm constructs a word affinity graph, where the edge weights represent the affinities between words as measured by the mutual information metric between word pairs. We define three specific classes of keys in the unsupervised algorithm: binary keys, numeric keys and descriptive keys. Binary keys represent specific features where the value is a binary output on whether the feature is present or not. Numeric features involve keys where the value represents a numeric output. Descriptive keys are ones where the key represents a broad category with a possible set of values (e.g., color of a car). The identification of the keys and the values from this graph are determined by analyzing specific affinity patterns of words in the graph with their neighbors.

The supervised approach is trained on a manually annotated training set, where we explicitly assume that the set of keys for a given topic is known previously. We implement a Conditional Random Field (CRF) based method to annotate the ad terms with descriptive keys. This supervised model computes the best sequence of keys – from a predetermined set of topic-specific keys which can best describe the ads –

given the word sequence of an ad. Although, a supervised approach is costlier and difficult to generalize over variety of topics, it demonstrated better performance.

We applied the unsupervised approach on a corpus of 12,984 ads on cars and 10,784 apartment ads downloaded from Craigslist. Evaluating on a manually annotated test set, the unsupervised method achieved an accuracy of 67.74% for cars and 68.74% for apartment ads. The supervised approach was trained on a manually annotated training set of 600 ads from each topic. The supervised algorithm yielded an accuracy of 74.07% for car ads and 72.59% for the apartment ads.

5.2 Problem Statement

A collection of domain-specific text contains repeated mention of attributes specific to the domain. All these attributes can be represented as a list of $(key, value)$ pairs, where the *key* represents the attribute and *value*, some typical values of this attribute. Based on this assumption, given $S = \{S_1, S_2, \dots, S_n\}$ where each S_i is an unstructured text on a specific topic t . The end goal is to convert each S_i into a set of $\{< key_{ik}, value_{ik} >\}$, where each key_{ik} represents some feature or attribute of the domain and $Value_{ik}$ its value. This means that the original text is approximately represented using K keys and their corresponding values in a tabular structure. Here, we consider Craigslist ad as an example of such an unstructured domain-specific text and build, evaluate the model using a large corpus of ads from Craigslist, across different topics (e.g. cars, apartments). An example of such a representation of the ad in Table 5.1 is shown in Table 5.3.

We assume that there are three different groups of keys. They can be *descriptive*, *binary* and *numeric*. Descriptive keys are those who can have numerous values, binary keys have only two values, *yes/no* or *present/not present*. Finally, numeric keys have numerical values. Table 5.1 shows a typical ad from Craigslist. Here, color can be a descriptive key whose value in this case is black. Other possible values are blue, red, silver etc. Power window, AM/FM Radio are examples of binary keys. As they are mentioned in this ad, their values are yes, else it would have been no. For some other possible binary key, say “power seats”, the value in this case is “no”. Finally, miles, price are examples of numerical key and their values are mentioned close to their occurrences in the ad.

From this example it can also be seen that for descriptive features, the keys sometimes appear in the ad (e.g. Transmission) and on other occasions it does not (e.g. car model or make). For binaries, presence

Table 5.1: A sample ad from Craigslist

Great car for our New England weather, 2004 BMW 325xi, Color Black, 114k Miles, 4 Door, All Wheel Drive, Automatic Transmission, Alloy Wheels, Fog lamps, Sun/Moon Roof, Air Conditioning, Cruise Control, Heated Seats, Leather Seats, Power Door Locks, Power Mirrors, Power Windows, Rear Defrost, AM/FM Radio, CD Player, Keyless Entry, Trip/Mileage Computer, Driver Air Bag, Passenger Air Bag, price \$11,488

of the key determines the value. And, for numerical keys, usually the key and the their values both are present in the ad. Different ads follow different formats. Often the keys with same meaning is expressed using different terms, e.g. **price of the car** can be mentioned using the terms *price*, *cost*, *offer* etc.

5.3 Datasets

To implement and evaluate our methods, we applied them on a corpus of Craigslist ads on 2 topics, cars and apartment rentals. We collected 12,984 cars ads and 10,784 apartment rental ads from Craigslist. We downloaded these ads directly from the Craigslist website ¹ belonging to the cities of Boston, New York, Chicago and Los Angeles. We used around 80% of the ads to build the models and kept the rest to test them. The test set was manually labeled to measure the performance.

5.4 Unsupervised Method

The unsupervised method proposed in this chapter is a graph-based approach. The graph constructed using the words in the ads – as vertices – capture the relationships between the words, i.e. an edge between the vertices shows the affinity between them and thus, the constructed graph is called *Affinity Graph*. This approach is devised on the assumption that across topics, the keys can be classified into

¹www.craigslist.org

Table 5.2: Different keys and their occurrence in the ads

Label	Type	Example
Color	Numeric	Grey with black interior shiny red paint the color is black Red with black racing stripe
Price	Desc	Best Offer \$5000 Cost \$2952 Asking \$2200 firm value is 3000\$ price for quick sale 3500 \$
Miles	Numeric	Under 62,000 miles approx 170k miles has 144,000 miles
Power steering	Binary	new power steering Power Steering, Cruise ... Control Power Steering - Power ... Brakes

Table 5.3: Corresponding $\langle key, value \rangle$ pairs from Table 5.1 (partial). Desc: Descriptive, Num: Numeric, Bin: Binary

Label (Type)	Value
Make (Desc)	BMW
Color (Desc)	black
Miles (Num)	114,000
Transmission (Desc)	automatic
Alloy wheels (Bin)	yes
Air conditioning (Bin)	yes
Leather Seats (Bin)	yes
Poor window (Bin)	yes
AM/FM radio (Bin)	yes
CD player (Bin)	yes
Price (Num)	11,488

3 different categories: descriptive, binary and numeric. The affinity graph is designed in a way that these different classes of keys can be easily detected from the graph. Hence, we propose a deterministic rule-based inference mechanism to detect the set of topic-specific keys from the affinity graph and their corresponding values.

5.4.1 Affinity Graph

The entire ad corpus can be represented as a set of K unique terms or words $W = w_1, w_2, \dots, w_k$. Here, each w_i can be a key or a value belonging to one of the classes, descriptive, binary or numeric. Intuitively, a key-value pair, belonging to a particular class will demonstrate a strong relationship between themselves. We design a graphical structure that can capture these inter word relationships. We construct a graph $G_{aff} = (W, E)$, we call it Affinity Graph, where the vertices correspond to each w_i in W and edges are constructed between vertices when the corresponding words in the ad demonstrate a strong relationship. The edge weight between two words w_i and w_j is computed as the mutual information between them defined as,

$$MI(w_i, w_j) = \log \frac{p(w_i, w_j)}{p(w_i)p(w_j)}$$

where, $p(w_i)$ is the probability of the term w_i occurring in an ad and the joint probability of two terms is the probability of the two terms occurring next to each other (adjacent) in the corpus. Here, w_i are all the words appearing in the entire ad corpus excluding the stop words and very frequently appearing words. Also, all numeric terms in the ads are replaced by a variable vertex w_{NUM} , instead of representing the exact numeric value.

And edges between w_i, w_j are defined as,

$$e(w_i, w_j) \in E \text{ for } \{w_i, w_j\} \in W, \\ \text{if } MI(w_i, w_j) \gg 0$$

with the edge weight,

$$w_{e(w_i, w_j)} = MI(w_i, w_j)$$

As, mutual information is symmetric, the edges in the affinity graph are undirected. By construction, the Affinity graph will be a collection of disjoint subgraphs.

5.4.2 Identification of Labels-Values Pairs from Affinity Graph

The vertices in the affinity graph are of 3 types: descriptive, binary and numeric, based on the orientation of the terms within the affinity graph. The hypothesis behind this clustering is based on the following claims,

- Descriptive keys will form a star-shaped component in the graph, where the center term is the key and the terms connected to it are its values
- Numeric keys will have a strong relationship with the w_{NUM} vertex. As the values of a numeric key is non standard and not fixed (e.g. price of a car is not a fixed entity, whereas for descriptive keys, like *color* of a car, have fixed values), this approach can only identify the keys. The value for a particular numeric key has to be extracted from individual ads
- Binary keys will form strongly related components (in terms of edge weights) and all the terms in that component will have similar weights. Binary keys do not have separate values. The presence of the binary keys is indicative of their values

Based on these assumptions, we propose an algorithm which takes the affinity graph as input and classifies each vertex into one of the following 3 classes, *descriptive*, *binary* or *numeric*. The proposed algorithm is a deterministic algorithm and is presented as Algorithm 2.

The classification of the vertices into the three classes is based on the degree of the vertices. As per the hypotheses, the different types of keys will have different kinds of orientation in the graph. For example, if a vertex has many edges, the vertex is more likely to be a descriptive key. This makes the association between the words (or an edge between the words) key indicator in identifying the (key,value) pairs. So, the graph needs to be pruned to eliminate the edges which do not demonstrate sufficient affinity between the words. An edge $e(w_i, w_j)$ was pruned if $MI(w_i, w_j) > \lambda_{threshold}$. The $\lambda_{threshold}$ is determined empirically by observing the edge weights of sampled word pairs. Two samples were taken; one with known cases of high affinity and the other with no associations between the pairs. We took the means and

the standard deviation of both the distributions. Assuming that both are normally distributed, we took the $\lambda_{threshold}$ as the average of the data point value at the 95th percentile of the weak association distribution and point at the 5th percentile of the strong association distribution. If $qnorm(p, \mu, \sigma)$ represents the quantile function of a normal distribution with mean μ and standard deviation σ then,

$$\lambda_{threshold} = \frac{qnorm(0.05, \mu_s, \sigma_s) + qnorm(0.95, \mu_w, \sigma_w)}{2}$$

where $\mu_s, \mu_w, \sigma_s, \sigma_w$ are the means and the standard deviations of the strong and the weak association edge-weight distributions. Using this threshold edge weight value, we pruned the affinity graph to have more well defined components. The association in the new components is stronger and well-defined, eliminating all the weak associations.

The classification of the words into various *keys* and *values* can be done by detecting the orientation of the new components and is done by computing the conditional probability of a vertex t given its neighbors for all the vertices in a component (line 4-5 in Algorithm 2). The cumulative score (line 6) for a vertex is the sum of this conditional probability from all its neighbors. If the conditional probabilities $P(w_i|w_j) \approx P(w_j|w_i)$ means that whenever w_i or w_j occurs they occur together. Hence, they together constitute a binary key. This concept can be generalized to include binary keys containing two or more words.

On the other hand for descriptive keys, the key words (e.g. color of a car) should have comparatively large number of neighbors and the corresponding value words (e.g. black, blue etc.) should only be associated with the key word. Hence, if $P(w_i|w_j) \gg P(w_j|w_i)$ then w_j occurs only with w_i but w_i can occur with other terms. This translates into w_i is a descriptive key and w_j is one of the possible value term as occurred in the corpus.

Finally, the numeric keys can be identified if $P(w_i|w_{num}) \approx P(w_{num}|w_i)$, which means that if a term only occurs with a numeric entry in the ads, then that term is a numeric key. Three lists are created $Label_{Desc}$, $Label_{Bin}$ and $Label_{Num}$, where all the corresponding keys are stored. The function returns these lists at the end.

Algorithm 2

```
1: procedure GETKEYVALUE
2: Input: Affinity graph
3: Output: Set of keys and their values
4:    $Labels_{Desc} \leftarrow \{\}; Labels_{Bin} \leftarrow \{\}; Labels_{Num} \leftarrow \{\}$ 
5:   for each  $w_i$  in  $W$  do
6:      $score(w_i) = 0$ 
7:   end for
8:   for each  $w_i$  in  $W$  do
9:     for each  $x$  in  $neighbor(w)$  do
10:       $P(x|w) = \frac{1}{deg(w)}$ 
11:       $score(x) = score(x) + P(x|w)$ 
12:    end for
13:    for each  $e(w_i, w_j)$  in  $E$  do
14:      if  $score(w_i) \gg score(w_j)$  then
15:         $Labels_{Desc}.add(w_i); Value[w_i] \leftarrow w_j$ 
16:      elseif  $score(w_i) \ll score(w_j)$ 
17:         $Labels_{Desc}.add(w_j); Value[w_i] \leftarrow w_i$ 
18:      end if
19:      if  $score(w_i) \approx score(w_j)$  then
20:         $Labels_{Bin}.add(w_i)$ 
21:         $Labels_{Bin}.add(w_i, w_j)$ 
22:      end if
23:      if  $score(w_i) \approx score(w_{num})$  then
24:         $Labels_{Num}.add(w_1)$ 
25:      end if
26:    end for
27:  end for
28:  return  $Label_{Desc}, Label_{Bin}, Label_{Num}$ 
end procedure
```

5.4.3 Performance

The graph-based method was applied on a corpus of 10,000 Craigslist ads on cars and 8,000 ads on apartment rentals to learn a set of keys (descriptive, binary and numeric) for the two different topics. The set of keys learned from the training were applied on a set of 2,984 car and 2,784 apartment rental ads to evaluate the performance. (key,value) pairs from test sets were manually extracted beforehand and used as a golden set. The golden set was created by a human annotator, who manually inspected the test sets and identified all the descriptive, binary and numeric labels. To evaluate the unsupervised approach, the affinity graph constructed during the training phase was used to extract the (key,value) pairs from the testing set. The extracted key-value pairs were compared against the manually crafted golden set and the performance was calculated using precision-recall values. The F-value computed from the precision and recall values for the car and apartment ad sets are shown in Figure 5.1 and 5.2 respectively.

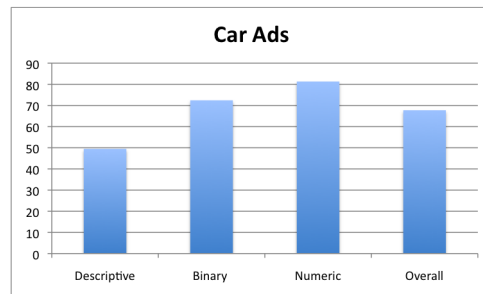


Figure 5.1: Total accuracy (F-measure) for car ads and under each category

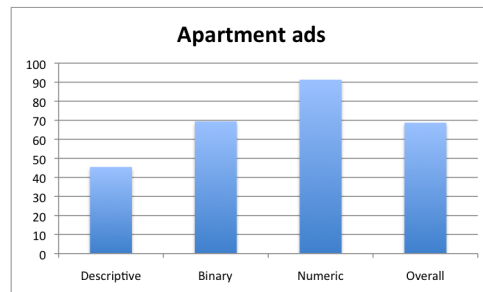


Figure 5.2: Total accuracy (F-measure) for apartment ads and under each category

This unsupervised approach gave an accuracy of 67.74% for car ads and 68.74% for the apartment ads.

The error rate was comparatively lower for numerical and binary keys but it was higher for descriptive keys. The reason behind this low accuracy is mainly due to the fact that often the descriptive key terms are not mentioned within the text. As an example, for car ads the ‘value’ of the ‘key’ ‘color’ is usually mentioned, like, black, silver but the actual name of the key (in this case ‘color’) does not appear in the text. As a result, in many cases descriptive keys are misclassified as binary keys. On the other hand, some binary keys with same words in them are classified as descriptive instead of binary. An example of such an error is ‘power window’, ‘power brakes’, ‘power steering’ etc. Instead of classifying them as separate binary keys, the algorithm classified ‘power’ as a descriptive key and ‘window’, ‘brakes’ and ‘steering’ as its values. This happened because the word ‘power’ was in all of them and satisfied the condition of being a descriptive key.

The goal of this work is to build a workable system which can convert unstructured ads into a structured tabular form. Hence, a supervised method can be a better approach where the keys are assigned after being trained on an annotated training corpus. This will also eliminate the problem of *descriptive keys* not appearing in the text because such keys will appear in the training set as tags. In the next section, we describe a Conditional Random Field based supervised learning method to solve the same problem but with a new assumption that the set of keys for a given topic is known beforehand.

5.5 Supervised Learning

There are certain aspects of the data which were not properly considered in the unsupervised approach. Particularly, for the cases where the (descriptive) keys do not appear in the text (e.g. the key “color”). Applying a supervised method trained on a manually annotated training set solve this problem, as the unknown keys can be included as annotating labels.

The problem statement slightly changed in the supervised learning method but the end goal of this work essentially remain the same. The problem statement in the supervised approach is,

Given a sequence of words from the ads $\{x_1, x_2, \dots, x_k\}$ what is the best hidden key sequence $\{y_1, y_2, \dots, y_k\}$ that describes the observed word sequence. In other words, what key sequence maximizes the conditional probability,

$$P(y_1, y_2, \dots, y_k | x_1, x_2, \dots, x_k) \quad (5.1)$$

Here, the set of keys $Y = y_1, y_2, \dots, y_n$ is known in prior. Once the sequence of words in an ad is automatically labeled by this model, the original ad can be converted into a tabular structure using the keys on one side and the corresponding words from the ads on the other, keeping it same with our original end goal.

The ads are usually written in an informal manner often using syntactically incorrect grammar, incomplete sentences and incorrect spellings. However, there is an inherent sequential aspect to these ads. If a word is assigned a label then the next word is more likely to have the same label. This property of the text was not taken into account in the unsupervised approach. There are other properties of the data, which the unsupervised approach did not consider, e.g. dealing with unknown words. If a particular word is not found in the training corpus the unsupervised method failed to classify it properly. All these properties make the problem similar to other NLP tasks such as POS-tagging or Named-entity recognition. Model like HMM, CRF have been quite popular in dealing with such NLP problems. Past works have shown that Conditional Random Fields (CRF) demonstrate better performance for these tasks compared to other models like HMM, Maximum Entropy [81][69]. Considering this fact and the nature of the problem in hand, we decided to employ a CRF based approach for this problem.

Moreover, a variety of features can be included in a CRF based model. In our context, including a large number of features can accurately model the irregularities in our data. Particularly, linguistic features can help in dealing with ambiguity and unknown words. Also, it enables the use of previous words and keys as features, which can model the sequential aspect of the data. Implementing the model using CRF has other advantages, as well. CRF considers the entire key sequence while training. This results into optimizing the model parameters with respect to the entire key sequence. However, this makes the optimization slightly more costly but it increases the accuracy. Also, this approach introduces some options in building the model. Figure 5.3 shows how CRF can be used to annotate the ad from Table 5.1. From this figure we see that the keys are quite localized. They tend to occur next to each other. This property can be used as a feature to improve the accuracy. Across many ads, the sequence of keys also follow a pattern, i.e. users tend to talk about the model, color, mileage, features following a pattern. All these properties of the text make CRF a good choice for the task in hand.

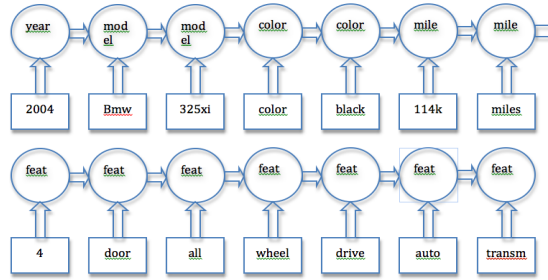


Figure 5.3: Linear chain CRF model to annotate ads (from Table 5.1) with keys

5.5.1 Conditional Random Fields

Conditional Random Fields (CRF) are discriminative probabilistic models (Lafferty et al., 2001) used for labeling sequential data. Given an input sequence of words from the ads $x = (x_1, x_2, \dots, x_n)$ and an output sequence of keys $y = (y_1, y_2, \dots, y_n)$, the model is defined as follows:

$$P(Y|X) = \frac{1}{Z(x)} \exp\left(\sum_i^N \sum_k^K \lambda_i f_i(y_{i-1}, y_i, x_i, i)\right) \quad (5.2)$$

where, $f_i(y_{i-1}, y_i, x_i, i)$ is the feature function comprising of current word, current key and the previous key. λ_i are the model parameters whose values are estimated from the learning process. $Z(x)$ is the normalizing factor and it is a function of x , as CRF computes the conditional probability instead of joint probability between x and y .

After the training process and parameter estimation, given an ad, the learned model tries to find a sequence of y_i which can best describe the observed sequence of words (x) from the ad.

5.5.2 Training and Extraction of Label-Value Pairs

600 ads per topic were randomly selected from the corpus described in the Datasets section to be used as a training set. Each word in the training set was manually tagged with a label (some of the labels used is shown on Table 5.4). The ads contained some words which are mostly exclamatory and have limited relevance. These words are not required to be a part of the structured ad. Such irrelevant words in the ads

Table 5.4: Subset of labels used for car and apartment ads

Apartment Labels	Car Labels
location	model
address	year
size	price
contact	size
rent	feature
bedrooms	miles
baths	colour
kitchen	phone
floor	email
phone	engine
email	interior
	problems
	condition

were tagged using a special key “null”. An additional 200 ads were selected and similarly annotated to be used as the test set.

Various features were used to build the model. Some of the features were textual, i.e. the words or some linguistic feature of the words and some were binary, expressing some properties of the words whose value is either 0 or 1. A subset of the features used is shown in Table 5.5. These features were added in the model using the feature function of CRF $f_i(y_{i-1}, y_i, x_i, i)$ (Equation 5.2). The model was trained on the annotated set to learn the parameter λ_i . The learned value of this parameter was later used to get the key sequence for a new ad.

Two separate models were built for the two different topics, having the parameter sets Λ_{cars} and $\Lambda_{apartments}$. The learned models for a topic can be applied on a new unstructured ad on that topic to generate a labeled version of the ad. In the labeled ad, the keys can be extracted along with the corresponding words assigned to that key. The extracted pairs are presented as a table which is the structured form of the unstructured ad.

5.5.3 Experiments and Results

The model was trained on a training set of 600 ads. An additional 200 ads were annotated in the same way to evaluate the model. The size of the training and testing set was small due to the high cost of building such sets. The final results are reported based on the accuracy computed on the test set of 200 ads². In Figure 5.4 the X axis shows the performance for different experiments. We performed 10 different experiments where various combinations of features (including word window sizes) were used.

The baseline model is defined as where the key is solely dependent upon the current word ($f_i(y_i, x_i, i)$). This model gave an accuracy of 48.23% in the car ad test set. We increased the window size of words to include the previous word along with the current ($f_i(y_i, x_{i-1}, x_i, i)$). The performance increased to 56.75% and by making the window as (x_{-1}, x_0, x_{+1}) took the accuracy to 60.12%. Finally, experimenting with window sizes, we found the best performance (accuracy of 67.45%) with a window of $(x_{-2}, x_{-1}, x_0, x_{+1}, x_{+2})$.

We analyzed the error cases and added some more features based on some surface characteristics of the words. These features were mostly binary. Example of such features include, *is there a digit in the word*, *is there a symbol in the word* etc. Including the digit feature improved the performance by almost 0.6. However, the symbol feature did not add to the accuracy. Based on this observation, we added features involving common symbols, like, '\$', '-'. Inclusion of these features proved to be effective and the accuracy rose to 73.66%.

Finally, we included features which looked into deeper characteristics of the words. Using regular expressions, the features looked into whether the word looks like a phone no, email address, any unit (e.g. square feet/sq. ft. or cc/litre of car engines etc.) Experiments conducted with these features reported an accuracy of **74.07%**. Repeated experiments beyond this point did not improve the accuracy significantly. All the accuracies reported here based on car ad test set 5.4, which we used as the development set. The feature set which reported the highest accuracy in the development set, achieved an accuracy of **72.59%** on the apartment test set. The variation of accuracy with the performed experiments is summarized in Figure 5.4 and Table 5.6. A subset of the final feature set is shown in Table 5.5.

²All the accuracies presented in the section are for the car ads. Because, the car test set was used as the development set. A summary of the results is presented in Figure 5.4 and 5.6

Table 5.5: A subset of the final feature set

Feature	Description
x_0	Current word
$x_{-2}, x_{-1}, x_{+1}, x_{+2}$	previous and following two words from the current word
digit in w_0	whether the current word has digits
'\$' in w_0	whether the current word has '\$' sign
'-' in w_0	whether the current word has '-' sign
phone pattern in w_0	whether the current word matches with a phone no. pattern
email pattern in w_0	whether the current word matches with a email id pattern

Table 5.6: Accuracies for the supervised model for car and apartment ads

Cars	Apartment Rental
74.07%	72.59%

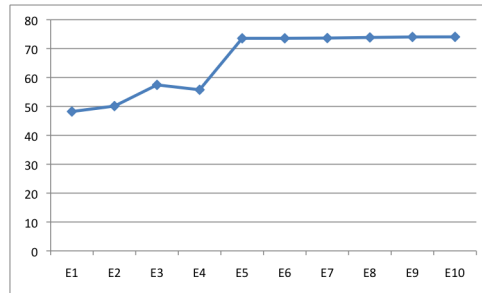


Figure 5.4: Accuracy for each experiment. There were 10 experiments performed on car ads with combinations of different word-window size and feature sets. Y-axis shows the accuracy in percentage.

5.6 Summary

In this chapter, we presented a solution to the problem of structuring unstructured online ads with a $\langle key, value \rangle$ style representation. We proposed a graph-based unsupervised algorithm which gave a performance with an accuracy of 67.74% for cars and 68.74% for apartment ads downloaded from Craigslist. We also presented an alternative supervised learning algorithm where we used CRF to compute the most probable label sequence given an observed sequence of words in an ad. The supervised algorithm achieved an accuracy of 74.07% and 72.59% respectively for car and apartment ads. Lower accuracies in the unsupervised method can be attributed to the fact that there are some aspects to the problem which are very difficult to model in an unsupervised method. Implementation of the supervised algorithm actually shows that some of the shortcomings of the unsupervised method can be reduced by the supervised method.

We are currently exploring the possibilities to further enhance the accuracy of our algorithms. One possible extension to the unsupervised method can be to introduce some probabilistic learning and inference mechanism. Such a method can overcome the failure cases of the deterministic approach currently employed. For the supervised approach, the task described in this chapter slightly differs with similar tasks such as, POS tagging and NER. In those tasks, every consecutive words are usually assigned different labels. In this problem, consecutive words are very likely to have the same label. Using some other model like Semi-markov CRFs can potentially increase the performance. Similarly, using a parser or a chunker can help to identify the consecutive but related words (e.g. “power windows”), which can avoid

assigning different labels to each of them. Another approach to improve the accuracy for the supervised method can be to increase the annotated training set from 200 to a larger number. An alternative to this can be to explore the possibility of a semi-supervised [77] approach: using a small amount of prior information on a given topic, can we further increase the accuracy? This method can reduce the cost of manually annotating huge number of ads, on the other hand perform better compared to a completely unsupervised approach. Finally, processing huge number of ads can be slow to process serially. Thus, adapting a distributed approach can improve the performance in terms of time. Storing the affinity graph using some distributed storage can help in parallelize the testing part and reduce time considerably.

Chapter 6

Event Analytics and Prediction from News Articles

6.1 Introduction

This chapter focuses on three different problems – (1) a novel way of extracting and representing events from news articles, (2) find relations between these events and specific external phenomenons; and (3) uses this novel event model and the combination of external phenomenons to predict certian characteristics in the external phenomenon. This chapter builds upon the basic observation that real world events, which manifest themselves in unstructured text in news and social media, can provide strong signals of the underlying factors that influence fluctuations in macro-economic indicators [40]. We aim to automatically extract real-world events from news sources and learn the relationships between event occurrences and variations in macro-economic indices. Our larger goal is to automatically derive event-driven predictive models that can be used to infer and potentially predict future fluctuations in specific macro-economic indicators. At an abstract level, the question we address is: Given a structured time series about a specific macro-economic index of interest and given a large corpus of real world events that can be extracted from news sources, can we learn a predictive model for the macro-economic index by connecting it with the appropriate events that relate closely to it.

Estimation of different macro-economic indices relies on information from multiple sources [37].

However, in studies aimed at estimating such variables [62], analysis and forecasting are usually done using structured data sources, considering only a handful of such factors, which are also chosen manually. There may be unknown factors playing an important role in the indicators' volatility or a combination of multiple factors leading to a sudden fluctuation of the indices. Tracking down events from around the world can provide useful insights on the volatility of these indicators. Much of the information about these real-world events are in the form of unstructured text streams, such as, news, blogs, social media etc. There have been a broad array of research where news sources have been mined to predict real-world events. Radinsky and Horvitz [73], Amodeo et al [7] used news corpus to predict future events. There are works that have used financial news to predict stock prices Ming et al [59] Gidofalvi [31] or political indicators [28].

Our work fundamentally differs from prior work on three fronts: (a) we explicitly assume no knowledge about the exact events that affect fluctuations of a specific index and aim to automatically discover them; (b) use any external knowledge base or data to build the model; (c) restrict the prediction to specific variables. Given a large collection of events and a specific index of interest, we aim to derive a condensed index-specific event-based predictive model for inferring future trends in the index. Existing efforts to derive econometric models for understanding fluctuations in macro-economic indices make one fundamental assumption is that the underlying variables of the model are known in advance. In addition, most known macro-economic index prediction mechanisms that have leveraged news content have primarily relied on pre-defined domain-specific features or market sentiment extraction mechanisms to predict variations in specific indices [37].

To achieve the larger goal of predicting volatility in certain macro-economic indicators, we propose a novel event model to represent events surfacing in the news media to be used as signals for the prediction. The event model is based on the assumption that every news article is about one and only one event and this event is drawn from a larger event class. An *event* class represents an abstract grouping of similar events agnostic of spatiotemporal, entity or topic based features. Whereas, an *event* represents an instance of an event class with specific spatio-temporal features along with entities and topics participating in the event. This instance or the occurrence of the event is manifested in a news article. Our model captures the central event (within the headlines or the lead (first) paragraph) as a collection of words/phrases that best describes the main theme of the article, called *event triggers*. Subsidiary events in an article are events that are described in the body of the article and are related to the central event; similar to the central

event, the subsidiary events are also represented by a collection of words. Our event class model can model any events, provided that there exists at least one article in the corpus that is about that event. So, our model overcomes two main limitations of existing frame works – lack of flexibility and reliance on external knowledge bases and ontologies [73]. As our model is capable of extracting generic events from any corpus and these events are not specific to any domain or class. Subsequently, the predictive model based on this event model is also not restricted to predicting any domain-specific variable. We evaluate our model by predicting sudden changes in a couple of macro-economic variables but the model can be extended for any types of external variables. We also propose another model, which incorporates past event information along with present values to predict the current value of a variable.

We evaluate our event driven predictive models to predict the sudden fluctuations in food prices for 4 crops, stock prices of 4 companies and dengue outbreaks in India based on events extracted from 7 years of news articles. We compare the results with a naive version of our event model along with a LDA based predictive model. Experimental results show that our model can predict fluctuations in food price with an accuracy of 62% and stock prices with an average accuracy of around 64%. In predicting both the variables, the event driven predictive model outperformed the baseline models by 5-10%. For food price prediction, the model incorporating *past event* information demonstrated the best performance, followed by the event driven model. On the other hand, for stock prices, the event driven method outperformed all the other models. This observation shows the difference in volatility of these two variables – events that last longer has more effect on food prices and also there is a delay in this effect, whereas the effect on stock prices is more immediate and sudden.

6.2 Related Work

Recent advances in text mining techniques have produced many sparse, low dimension representational schemes for text documents. Topic models, such as pLSI [39], LDA [12], where a large corpus of documents with a vocabulary $\sim 100,000$ words can be represented using ~ 100 topics. These models have made knowledge acquisition from natural language text easier and more effective. Such representation of documents has been effectively used in many applications. Particularly, online news items have been a popular source for mining events using these abstractions. TAM [47] or TM LDA [98] modeled the temporal aspect of topics. MedLDA [107] uses maximum margin classifier jointly modelled with topics to

build predictive models for categorical and continuous variables. Vaca et al [95] used a collective matrix factorization method to track emerging, fading and evolving topics from news streams. Shahaf et al. [83] developed a scheme to connect related news articles to understand news better.

There have been some works that have focussed on relationships and dependencies between topics/events or prediction of events. Radinsky and Horvitz [73] proposed a framework to predict events from news data. The work by Rudin et al [76] involves predicting the next event in a sequentially organized data using association rule mining and Bayesian analysis. Amodeo et al [7] proposed a hybrid model to predict future events using a New York Times corpus. FBLG [19] focussed on discovering temporal dependency from time series data and applied it on to a Twitter dataset about the Haiti earthquake. In a similar work by Luo et al [55] showed the correlation between events and time-series data. Hogenboom et al evaluated the effects of rare news events on stock prices and eventually improved the performance of Value-at-Risk (VaR), a popular tool for assessing portfolio risk [40]. Some works have focussed on predicting specific variables from news data, such as stock price. Hagenau et al [35] proposed a new scheme to include context from financial news and market feedback to better predict stock prices. Ming et al [59] used WSJ corpus and sparse matrix factorization to predict stock prices and Gidofalvi [31] also used financial news to predict volatility of stock prices. Other works emphasizing on predicting stock prices include [104][14][25]. Similar works have been proposed for political indicators [28].

In most of these works, events and topics have just been used as a tool for knowledge acquisition or information extraction. To the best of our knowledge there has been no previous attempt to combine such events or topics from unstructured text streams with structured data to characterize and forecast macro-economic indices. There are a few specific works aimed at predicting stock prices [59][31][89] but in these studies, text sentiment was used for the prediction. In this chapter, we propose a framework, where we connect events extracted from text data (news articles) to predict external variables, driven by the assumption that there exists a dependency between the variables and real-world events. Our framework is generic, unlike many existing works that have targeted specific variables such as stock prices. p

6.3 Problem Statement

We assume that news events can influence fluctuations in certain external variables like macro-economic indicators. Our objective is to learn a model that can predict fluctuations in macro-economic indicators,

based on events happening across the world. Given a corpus of news articles D , where at time t there are t_n news articles in the collection and that set is denoted as D^{t_n} . We build a mapping $\phi : D^{t_n} \rightarrow R^{t_k}$ that maps each t_n news articles appearing at time t to t_k events with varying probability. The training involves m labeled examples of the form $R = (R_{t_1}, R_{t_2}, \dots, R_{t_m})$, where each R_{t_u} has the form $R_{t_u} = (\phi_{t_1}^u, y_{t_1}^u)$ and each $\phi_i^u = \phi(d_{t_i})$ denotes the events at time t_i manifested in the articles appearing on that day and y_t is a binary variable whose value is 1 if a spike in the macro-economic indicator is observed at time t and 0 otherwise. We will learn a function, parametrized by a weight vector $\theta \in R^k$:

$$f_{\theta}(\phi) = \theta^T \cdot \phi \quad (6.1)$$

Based on this function the response variable (denoting a fluctuation) is assigned a value:

$$y_{t_i} = \begin{cases} 1 & \text{if } f_{\theta}(\phi) \geq \epsilon \\ 0 & \text{otherwise} \end{cases}$$

The contribution of this chapter is

1. Design the mapping ϕ , which converts a corpus of n news articles into a set of k event classes (\mathbf{x}), where $n \gg k$ and each article d_i is sampled from the event classes and belongs to only one event class x_j in \mathbf{x} . The design of this mapping is explained in detail in section 6.4
2. Learn the function $f_{\theta}(\phi_t)$ with parameters θ , to predict volatility of a macro-economic indicator based on the events happening at time t . In the remainder of the section, we shall elaborate on how we design this function f_{θ} .

Let y is the external variable whose value we wish to predict, and y_t be a binary variable denoting whether there was a fluctuation in its value at time t . We can observe the values of y_t within the time range $t = 1, \dots, T$. Let, $\mathbf{x} = x^1, x^2, \dots, x^k$ are k possible events happening in the same time period (1... T). We would like to predict the value of y_t , from observing $\mathbf{x}_{1:t}$ and $y_{1:t-1}$. The general statement of the problem is,

$$P(y_t | \mathbf{x}_t; \theta) \quad (6.2)$$

So, we would like to predict whether there will be a sudden change in the index's value based on what events have just occurred. We assume that all the events x^1, \dots, x^k are independent to each other. The joint probability of the model can be expressed as,

$$P(y_t, \mathbf{x}_t) = p(y_t)p(\mathbf{x}_t|y_t) = p(y_t)p(x_t^1|y_t)p(x_t^2|y_t)\dots p(x_t^k|y_t) \quad (6.3)$$

where each $x_t^i \in [0, 1]$ denotes the probability of event i at time t . So, the posterior probability of y_t given the observed events can be represented as,

$$P(y_t|\mathbf{x}_t) = p(y_t) \prod_{i=1}^k p(x_t^i|y_t) \quad (6.4)$$

The prior is assumed to be drawn from a uniform distribution, i.e. all the events are equally likely. Then for the likelihood term ($p(\mathbf{x}_t|y_t)$) we assume a gaussian distribution. The likelihood term for $y_t = 1$ – when a major change in the value of the indicator is observed – is defined as,

$$p(\mathbf{x}_t|y_t = 1) = \frac{1}{\sqrt{2\pi\sigma_1^2}} e^{-\frac{x-\mu_1}{2\sigma_1^2}} \quad (6.5)$$

where, $\theta_1 = \{\mu_1, \sigma_1\}$ are the parameters for this case. Similarly, for the case $y_t = 0$ has the parameters $\theta_0 = \{\mu_0, \sigma_0\}$. From a data, where we can observe the value of y_t between $t = 1, 2, \dots, T$ and all the events from the same time period, we can learn the parameters μ, σ . Finally, the value of y_t can be estimated from observing the events at time t using maximum a posteriori estimation. If y' denotes a value of y_t , where $y' = 1$ denotes a sudden change and $y' = 0$ otherwise, then,

$$y' = \arg \max_{y \in \{0,1\}} p(y_t; \mu_{y'}, \sigma_{y'}) \prod_{i=1}^k p(x_t^i|y_t; \mu_{y'}, \sigma_{y'}) \quad (6.6)$$

In all of these above equations, we denote the probability of an event i occurring at time t as x_t^i .

6.4 Event Models

News articles report about events happening around the world. So, a collection of news articles can be used as a repository to mine real-world events. There are many works that have used news articles as sources to extract events. In such works events were interpreted and represented in different ways. Radinsky et al [73] used storylines extracted from news articles to be represented as events, for example

– *storm in Rwanda, flood in Africa* etc. In another representational scheme, events were represented as a triplet of objects (or actors), actions and time [71]. Automatic Content Extraction (ACE) has specifications in their event extraction tasks, where 8 types and 33 sub-types of events are defined using *entities, event triggers, times, event mention argument* etc.[52]. In some cases domain-specific event types are targeted and triggers are extracted correspondingly. For example, for finance related events, the main types are – *hiring, firing, resignation, acquisition, loss, profit-up, profit-down* etc [40].

In this chapter, we propose a novel probabilistic model to represent events extracted from a corpus of news articles. Based on the structure of majority of news articles, we assume that a news article is a description of one and only one event. Using this assumption, we try to model an event using certain properties as provided by the news article. We also assume that within a time period there is a finite number of event types – we call each type an *event class*. A news article is sampled from this set of event classes and is an instance of that class – this instance is called an *event*. An event class is a broad description of an event. For example, "accident" is an event class, a specific occurrence of an accident is an event, involving a location, topics (e.g. car accident or an air crash) and values of other properties of an event classes, such as, actors, objects etc. The main property of the event class is the *event trigger*, which is the primary action that describes the event class. In the preceding example, trigger is "accident" but there can be other words or phrases that can act as triggers for the same class – e.g. crash, collision, rammed etc.

Definition 1. Event class: An event class represents a type of event encapsulating similar themed events. Event classes are labeled by synonymous event triggers. There is a finite set of event classes that comprehensively describes all real-world events featured in a large corpus of news articles. An **event** is a sample from one of the event classes and a news article is an observable instance of that event with specific time and location, along with other artifacts of an event like, topics, entities etc.

In majority of the cases, the title of the article is a one-line summary of the main event covered in the article. For example, the title appearing in New York Times after the bombing at the Boston marathon – "Blasts at Boston Marathon Kill 3 and Injure 100" – the words – **blasts, kill** and **injure** summarize the main event covered in the article, and are the *event triggers*. The body of the article covers this central (main) event, in addition to other events, topics and entities associated with the central event of the article. Usually the first paragraph of the article (also known as the Lead Paragraph) contains the most information

about the main story of the article [65]. So the *event triggers* presented in the title and the lead paragraph of the article represent the central theme of the article. Hence, we put a special tag on these event triggers and call them *event class triggers*. They are defined as,

Definition 2. Event class triggers are a set of words in the title of a news article that describes the main story of the article. These triggers are sampled from a cluster of triggers, where each cluster represents an event type or an event class.

K event class triggers can be represented as $\{X_1, X_2, \dots, X_k\}$, where X_i represents one event class as a collection of synonymous triggers. In the example headline presented above the event class can be labeled as,

$$X_{blasts} = \{blast, explosion, bombing, \dots\}$$

All the trigger words appear in an event class with a uniform probability and a particular trigger can be replaced with another one without any loss of information. An article also mentions other events. For example, an article describing a bomb blast, the main event trigger can be “blast” or “explosion”. The article has also mentioned the number or *deaths* due to the explosion, which in this case is a subsidiary event of “explosion”. Thus, two important features of an article is the event class triggers and a set of subsidiary event triggers. Subsidiary events are defined as,

Definition 3. Subsidiary events is the set of events mentioned in an article in addition to the main event of the article. It represents the additional events likely to happen along with the main event.

Example of subsidiary events are “deaths” and “injuries” after a blast, which are also mentioned in the article covering a blast. So, an event class can be characterized by the event class triggers as well as the set of subsidiary events that are *likely* to co-occur with the event.

An *event class* can have at most one main event but a mixture of subsidiary events. Hence, an event class is represented as a combination of one main event and a mixture of subsidiary events. An *event* is an instance of one of these classes manifested in a news article, where the event occurrence is mentioned with specific location, objects etc.

6.4.1 Event Trigger Extraction

Automatic Content Extraction (ACE) has a specific task for event extraction from news. ACE specifications has defined *event trigger* as the words in a sentence that specifies the occurrence as well as the type

of the event. For example, in the news article headline – *FIFA Officials Arrested in Corruption Case* – the word *arrested* is the trigger word for the event mentioned in the article. Usually event triggers are verbs or nouns present in the sentence that describes some notion of action. Event triggers can have different forms - verbs (*Traders protest over FDI in retail*), nouns (*Burglary in police station leaves cops red-faced*) and sometimes the verb or noun themselves cannot form the trigger but a combined phrase is the trigger (*Number of AIDS patients go up in MP*).

In standard event extraction task, triggers are extracted at a sentence level to understand the type of event presented in the sentence. Our goal is to understand the best event type that describes the entire article. Our assumption is that every article is about one event, hence, identify which triggers in the article collectively describe the event in the article. Typically, a news article is organized as follows – a title or a headline with a one line overview of the main event; first paragraph of the article contains a brief description of the main event; the rest of the article presents details of the event with follow-up actions of this specific event. Based on this standard flow of a news article, our assumption is, the trigger appearing in the title of the article are representative of the main event, and *the verb present in the title is the main event trigger*. However, this verb cannot be a stop-word verb, such as, “is”, “have”, “said” etc. We used a list of stop-words from a standard NLP dataset to identify a set stop-word verbs ¹.

6.4.2 Event class Model

The event model presented in this chapter is based on the assumption that every news article is about one central event. This central event is drawn from an event class, which is a broad description of the event presented in the article. Whenever an event occurs it belongs to one of these broad event classes. News articles report this event with additional descriptions like locations, objects, people associated with the event. These additional descriptors are variable and will change across different occurrences of the same type of event. The event class encapsulates the constant factors associated with a particular type of event. This includes, (1) event triggers that provides a comprehensive description of the events within the class (the event class triggers) and (2) a set of subsidiary events that are likely to co-occur with the event in class but are not the same. Like in the previous example, for the event class “blasts”, the triggers that describe events from that class are – “blast”, “explosion” etc. On the other hand, events like – “death”,

¹<http://snowball.tartarus.org/algorithms/english/stop.txt>

“injury” are closely associated but not directly linked. Hence, “death”, “injury” are subsidiary events of event class “blast” and are very likely to happen irrespective of location, people, object of the blast.

To build the event class model, the main steps are to detect the event classes and the corresponding triggers; and for each class the likely subsidiary events. The construction of this model follows a pipelined process with the first step being identifying the class triggers. Any trigger is extracted from the text following the method described in 6.4.1. To identify the triggers which define an event class, we go back to our earlier assumption that the main story of an article is described within the title and the lead paragraph [65]. Hence, the triggers extracted from these parts are potential candidates of class triggers. The idea is to cluster similar articles (describing similar events) and obtain the class triggers that are used to describe events from the same event class. For example, articles about a bomb explosion are likely to have triggers like, *explosion, blast, bombing* etc. So, in our model an event class is represented by synonymous event triggers.

The similarity between clusters are computed using word embedding based on deep neural net based language model [58]. This technique embeds a word from a large corpus of text into a vector space where words appearing in very similar contexts are placed within a vicinity. The vector form of these words can be used to cluster words having similar meanings or used in very similar context. The event triggers extracted from the news corpus were embedded in this vector space and clustered using the vector as features. Thus, triggers appearing in similar contexts are placed in the same cluster, in other words, triggers describing similar events are represented by the members of the cluster. These clusters represent the event classes in our model and each cluster is an event type represented by the cluster members. To get the optimum number of clusters, we varied the number of clusters and used the “elbow” method to get the best number clusters. If SSE_k represents the sum squared error for a particular cluster configuration with number of clusters k , we iterated for different values of k (varying between 50 and 1000 with an increment of 50) until the following condition was met.

$$|SSE_{k-1} - SSE_k| < \epsilon$$

where ϵ is the limiting threshold. This method yielded 250 types of events extracted from a 7 year news corpus. At a given time t , the events identified from the news articles appearing at t can be used to compute the change in a response variable (y_t) using the expression in Eq 6.6. The detailed implementation of creating the event class clusters is shown in Algorithm 3. This algorithm outputs the event classes with member triggers and the best value of K , the number of event classes.

Algorithm 3 Event Class

```
procedure EVENT_CLASS(News Articles (D))  
  for each  $d \in D$  do  
     $\mathcal{E} \leftarrow$  Extract event triggers from Title and Lead Paragraph  
  end for  
   $W2V(\mathcal{E}) \leftarrow$  Word_embedding( $\mathcal{E}$ )  
   $SSE \leftarrow 0$   
  for  $k \leftarrow 50 : 1000$  do  
     $C.k \leftarrow$  KMeans( $W2V(\mathcal{E}), k$ )  
    if  $|SSE - \text{sum\_squared\_error}(C.k)| < \epsilon$  then  
      return ( $C.k, k$ )  
    else  
       $SSE \leftarrow \text{sum\_squared\_error}(C.k)$   
    end if  
  end for  
end procedure
```

Algorithm 3 generates the k event classes where each class contains "similar" event triggers. We denote the event classes as $C = \{C_1, C_2, \dots, C_k\}$ and the fixed number of m triggers as $\{u_1, u_2, \dots, u_m\}$. As stated previously, our assumption is that every news article has *one* underlying event. For article d , the event class is represented as C_d . The article d is an instance or an occurrence of the event represented by the class C_d .

If there are D articles in the corpus, every article $d \in D$ is assumed to be a union of two disjoint sets: $U_d \cup W_d$. Where, each $u_{d,i} \in U_d$ represents the set of event triggers present in d . The rest of the non-trigger words are represented as W_d . W_d represents other features of the document, such as, topics, entities, locations, people etc. W_d can be considered as a description of the event, like *where* it happened, *when* it happened, *people*, *organization* involved etc but is independent to the event class. In other word, an event can occur involving any topic, entity or location. Our goal, is to identify which event class (C_d) the article d belongs to, by observing the words belonging to U_d and their positions.

The generative process of the articles from event classes is as follows. The main event is sampled from a multinomial distribution with parameter λ .

$$\pi_d \sim \text{Multinomial}(\lambda)$$

where λ is the prior probability distribution of the event classes. Then for each trigger word i in the article d , sample uniformly from the triggers belonging to sampled event class

$$u_{d,i} | \pi_d \sim \text{Uniform}(\pi_d)$$

On the other hand, the subsidiary events act like topics in a document but it is only generating the subsidiary event triggers. Hence, the generation of this part of the article is similar to LDA topic model [12] and as follows,

$$\theta_d \sim \text{Dirichlet}(\alpha)$$

$$Z_{d,j} \sim \text{Multinomial}(\theta_d)$$

$$u_{d,j} \sim \text{Uniform}(Z_{d,j})$$

Now, whether a trigger word in the article d , is drawn as a main event ($u_{d,i}$) or subsidiary event ($u_{d,j}$), is decided through a biased coin with probability for main event (p_i) less than the probability of subsidiary event (p_j)². The posterior is given by,

$$p(\pi_{1:D}, \theta_{1:D}, \mathbf{Z}_{1:D} | \lambda, \alpha, \mathbf{u}_{1:D,i}, \mathbf{u}_{1:D,j})$$

where \mathbf{Z} , \mathbf{u}_i , \mathbf{u}_j are vectors. In this model, we can observe the variables, u_i and u_j , also the multinomial (λ) and the dirichlet (α) parameters are assumed to be known. The rest of the variables are hidden. The posterior distribution is used to infer these variables given the news corpus. We used Markov Chain Monte Carlo approximation method with Gibbs sampling to compute the posterior.

²Because the subsidiary event triggers are likely to be more in number than the main event triggers

Table 6.1: Model variables and parameters

Variables	Description
D	No. of articles in the corpus
$C_{1:k}$	Event classes, represented as a set of event triggers describing the class
$U_{1:m}$	Event triggers extracted from D
λ	Event class prior denoting event from which classes are more likely
π_d	the event class from which article d is generated
α	Dirichlet prior for subsidiary events
θ_d	The proportion of event classes as subsidiary events in article d
$Z_{d,j}$	Which subsidiary event produced the j^{th} trigger in article d

6.4.3 Historical Event Model

So far, we have only considered the current events, i.e. to predict the value of y_t , we consider the event occurring at time t . However, in reality, an event might have a delayed effect on y . There might be a delay of time δ between the event occurring and its effect on y . To interpret y_t as being tracked by historical occurrences of news events, we use a parameter δ and look at text of articles occurring at times $t - (\delta + 1), \dots, t - 1$. We fit a model on $x_{i,t-j}$ that best approximates y_t :

$$y_t = \alpha_0 + \sum_{i=0}^N \sum_{j=0}^{\delta} \alpha_{i,j} x_{i,t-j} + \epsilon_t \quad (6.7)$$

Let, $\chi(t)$ denote the vector of length $N\delta + 1$ given by $(1, ni, t - ji, j)$ and α denote $(\alpha_0, \{\alpha_{i,j}\})$ for $i = 1, \dots, N$ and $j = 0, \dots, \delta + 1$. We rewrite this in vector notation as:

$$y_t = \alpha \cdot \chi(t) + \epsilon_t \quad (6.8)$$

While this fully captures the relationship between y_t and co-occurring text in news articles, it is somewhat unwieldy given that N can be very large. The crucial simplification we proposed earlier in this chapter is to approximate $\chi(t)$ using the event model, reducing the dimension of the data. Let m be a parameter of our choice. Let $g : R^{N\delta} \rightarrow R^K$ be a mapping of vectors $\chi(t)$ in $R^{N\delta}$ into R^K . We formulate the model from (6.8) below:

$$y_t = \hat{\alpha}.g(\chi(t)) + \epsilon_t \quad (6.9)$$

where, $\hat{\alpha} = \hat{\alpha}_0, \dots, \hat{\alpha}_m$ is $(K + 1)$ -dimensional vector of coefficients. The reduction of $g : R^{N\delta} \rightarrow R^K$ is achieved through dimensionality reduction using the event model.

In this representation scheme, the current historical trends across text features are not implicitly represented. Hence, we add the immediate past trends of the features within the representation of the model. For every feature f –appearing at time t – we add its value for past δ time points. For example, when t represents a particular day, we add the values of feature f for the time points $t - \delta, t - \delta + 1, \dots, t - 1$, along with the value at time t . This way the model combines the immediate past with the present to capture temporal trends in the events. The resulting events will have word distributions covering words appearing at the present along with the past. We expect this model to detect, at specific time steps, dominant trends that a global trained model may have missed.

6.4.4 Topic Based Prediction

A set of topics learned from a large corpus of documents represents the main *themes* presented in the corpus. For news articles – which can be assumed as a source of events – topics can be thought of the main themes of the events covered by a news corpus. Moreover, every news article also mentions the time of an event described in the article. We define topic based event as the probability of i^{th} topic at time t , represented as, $\{x_t^i\}$ ($i \in \{1, \dots, K\}$). According the LDA topic model, a document is a mixture of topics appearing with different proportions. If d_t is the set of news articles appearing at time t , the proportion of i^{th} topic at time t is defined as,

$$x_t^i = \max_{d \in d_t} (\theta_{i,d}) \quad (6.10)$$

where, $\theta_{d,i}$ represents the proportion of topic i in article d appearing on day t . On a day t the topic-based

events are represented by: $\{x_t^1, x_t^2, \dots, x_t^K\}$. We chose K to be 100. The values of x_t^i computed from Eq 6.10 are used in Eq 6.4 to build the LDA based predictive model.

6.5 Evaluations

We evaluate our event-based predictive models by predicting the prices of 4 crops and stock prices of 4 separate companies and dengue outbreak in Delhi, India based on the events extracted from a corpus of news articles. Our model's performance was compared with other baseline systems. Food price and stock price were predicted with a maximum accuracy of around 62% and 67% respectively and dengue outbreak with an accuracy of almost 70%. For all the variables our event models demonstrated much better performance compared to the other systems, with an improvement of around 5-10%. In the rest of the section, we will provide a brief description of the data and the experiments performed for this evaluation.

6.5.1 Data

For the experiments – we used three different types of data – a corpus of news articles, food price data and stock price data. The time period of our experiments is from January 1, 2006 till December 31, 2012. All our data were collected within this period of 7 years. All the data collected are from India.

For the event extraction, we used a corpus of news articles published by Times of India³ between 2006 and 2012. This corpus had 700,863 articles. Each article in this corpus had – title, main text, publishing time, and the location from where the article was published.

The food price data collected for the experiments were collected from the website of the Ministry of Agriculture of Government of India⁴. The Directorate of Marketing and Inspection under the ministry publishes daily prices of different crops. The daily prices include minimum, maximum and modal (the rate at which maximum sale was done) prices of these crops across many different markets in the country. In this work, we focused on four crops – onion, potato, rice, wheat – which are among the most consumed agricultural produces in India. We collected the daily modal prices of these crops and computed the average across different markets to represent the daily price of the crop in India. We predict the spike in

³timesofindia.indiatimes.com/archive.cms

⁴<http://agmarknet.nic.in/>

price instead of the absolute price. A spike in price at time t defined as, if the percentage change from the price at $t - 1$ is greater than equal to 10%: $y_{spike} = 1$ if $(y_t) - (y_{t-1})/y_{t-1} \geq 0.10$

The advantage of this procedure is that it normalizes for some effects like inflation. The weekly average price and the spikes observed in the price for the four crops are shown in Figure 8.6. The price data was smoothened over a 7 day period and the price for time t is computed as a function of $t - 1$.

For the stock price prediction task, we collected the daily stock price data for 4 Indian companies – Wipro, Infosys, Tata Coffee and Tata Motors. These companies are listed in the Bombay Stock Exchange (BSE) and data was collected from the BSE website⁵. The website lists the daily opening, closing, highest and lowest price of a company stock. Similar to the price data, for the stocks, we used the daily maximum price and computed the change based on whether there was a 10% change compared to the previous day’s value. However, unlike food prices, here we considered both increase and decrease in price. A snapshot of this data is shown in Figure 6.2.

6.5.2 Experiments and Results

In this section, we provide qualitative evaluation of our event class model and quantitative analysis on the predictive accuracy of the model in predicting two external variables – food and stock prices.

Qualitative Analysis of Event Model

The design and the goals of the event model presented in this chapter is different from existing event extraction tasks. Thus, it is difficult to evaluate a model like this as there are no gold standards to evaluate such a model. So, we evaluated the model qualitatively on a corpus of news articles. Our data consists of the archive of Times of India between the years 2006 and 2012. Times of India is one of the most circulated English national daily published from India.

There were **250** distinct events classes extracted from the corpus (from Algorithm 3). These event classes represent broader category of events and every news article belongs to one of these classes. Table 6.7 and 6.8 presents a subset of event classes that were extracted along with a sample of subsidiary events. Here, we present top 6 trigger words associated with the event. The corpus contained articles mostly from India, hence, some words extracted are specific to India. For example, *aila* was a cyclone that hit eastern

⁵<http://www.bseindia.com/markets/equity/EQReports/StockPrcHistori.aspx>

coast of Indian in 2009, *dharna* is a Hindi word for protest or agitation. From this list we see that for every event, the subsidiary event is actually showing some associated events. Like, the event class related to disease outbreak, most of the subsidiary events are about what follows an outbreak – *treatment, admitting* patients to hospitals and plans for *prevention*. However, there are some subsidiary events that are precursor to the event – *announce, nominate* happens before election. The present version of the model is incapable of differentiating between the subsidiary events that happens *before* or *after* an event. A future direction of this work can be to classify the subsidiary events to these 2 classes. This will provide better insights to the events class.

Prediction Accuracy

Our dataset consists of 7 years (2557 days) of data. Using the event model, we extracted the daily events across these 7 years and use them to predict spikes in food prices and sudden changes in daily stock prices. The metric used to evaluate the model is how accurately the model can predict the changes. The accuracy is computed as the ratio of number of correct predictions to the total number of data points. We evaluate the performance of our event model across different baselines and other text mining methods described in Section 6.4.

Our event model is built using a pipeline process, involving extraction of event triggers as the first step, adding subsidiary events at a later stage. So, one of the first experiment was conducted to see the advantage of adding the subsidiary events to the event model. So, the performance was compared against a naive event trigger based model (TRIGG) used as a baseline. We also compared the performance against LDA based predictor (section 6.4.4). We also compared the efficiency of adding subsidiary events in our model against adding topics. So, we implemented another model by combining TRIGG and LDA [TRIGG+LDA] to observe the specific benefits of the subsidiary events. Finally, we evaluate all these models with another model which incorporates past events and the current events to predict the change in the current value of the indicators. We call this model HIST (6.4.3). The results of all these experiments are summarized in Tables 6.2 , 6.3 and 6.4 for food prices and stock prices respectively.

Some of the events that showed strong association with food price fluctuation are shown in Table 6.5 for each crop separately. We see some commonalities across crops, like – disease outbreaks, violence and movements, natural calamities, festivals etc. However, there are some unique events for each crop. For example, transportation strike or any incident affecting transportation system has a more profound effect

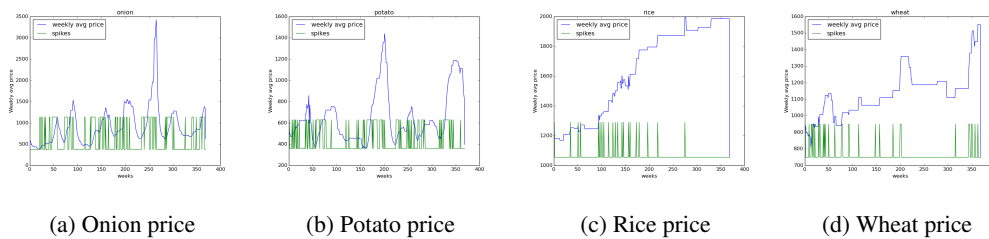


Figure 6.1: Prices of crops between 2006 and 2012 represented as weekly average. The spikes represent an increase in price of 10% from the previous value. The horizontal axis represents the weeks between January 2006 and December 2012, where 0 is the first week of Jan 2006 and 364 is the last week of Dec 2012

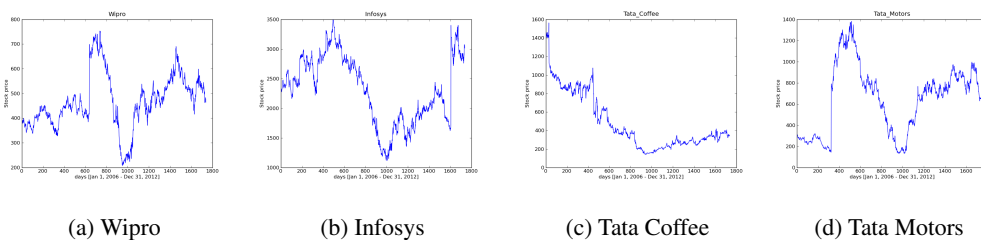


Figure 6.2: Stock price fluctuation of 4 Indian companies between Jan 1, 2006 and Dec 31, 2012

on wheat and potato (events: attack on railways, railway strike etc.). This is probably due to the fact that wheat and potato are not cultivated uniformly across the country and heavily relies on transportation for proper distribution. Any event that affect transportation of these crops has an effect on the supply, leading to a rise in price. On the other hand, festivals have effects only on onion and wheat. These two crops are more consumed during festivals and around that time demand increases, thus increasing the price.

Table 6.2, 6.3 and 6.4 summarize the performance of event driven predictive models against other baselines and popular text mining techniques. Table 6.3 presents the performance of all the models in predicting the price of 4 crops. All the models showed their best performance in predicting spikes for all crop combined. Among the individual crops, onion price prediction showed the best results, followed by rice, wheat and potato. A possible explanation of this observation is that price volatility of onion is covered more in the news articles compared to the other crops, as onion price has more adverse effect compared to the other crops. Comparing against the different models we see that HIST has the best performance.

Table 6.2: Performance for Food Prices

	Potato	Wheat	Rice	Onion	All
TRIGG	0.509	0.512	0.514	0.520	0.535
LDA	0.431	0.432	0.471	0.488	0.492
TRIGG+LDA	0.539	0.544	0.548	0.553	0.592
TRIGG+SUBS-TRIGG	0.567	0.569	0.579	0.592	0.599
TRIGG+SUBS-TRIGG+HIST	0.603	0.613	0.615	0.618	0.621

Table 6.3: Performance for Stock Prices

	Wipro	Infosys	Tata Motors	Tata Coffee
TRIGG	0.636	0.641	0.541	0.604
LDA	0.433	0.439	0.473	0.413
TRIGG+LDA	0.632	0.573	0.543	0.597
TRIGG+SUBS-TRIGG	0.679	0.662	0.593	0.611
TRIGG+SUBS-TRIGG+HIST	0.622	0.553	0.509	0.551

Although our event model (TRIGG+SUBS-TRIGG) demonstrated marginally poorer performance, the reason behind better performance by HIST is that events driving food price volatility are not sudden and often they have a delayed or prolonged effect. The events which are found to have a strong association with food price rise, such as, government policies, festivals, strikes take some time to have an effect. For stock prices, events have more immediate effect. Hence, HIST was not the best performing model. The TRIGG+SUBS-TRIGG model demonstrated the best performance in predicting stock prices.

In both the cases, the model TRIGG+SUBS-TRIGG outperformed TRIGG and TRIGG+LDA. This observation vindicates the value addition of the subsidiary events. Subsidiary events are showing better performance with triggers than topics. Poor performance of topics (LDA) is due to the ubiquitous nature of the topics. Almost all topics appeared almost every day and the per day topic distribution was approximately uniform. On the other hand, events are much more sparse and displayed a more non-uniform distribution over time. Due to this difference in their property, events displayed more discriminating

Table 6.4: Performance for Dengue Outbreaks

	Dengue
TRIGG	0.610
LDA	0.575
TRIGG+LDA	0.637
TRIGG+SUBS-TRIGG	0.662
TRIGG+SUBS-TRIGG+HIST	0.691

power than topics and hence, better in predicting external variables. For dengue outbreak (6.4) prediction we see a reflection of the food price prediction results. In absolute terms, it had the best performance (~70%) across all three socio-economic indicators. Again, the HIST model demonstrated the best performance vindicating the fact that events that potentially affect dengue has a delayed effect instead of immediate, unlike for stock prices.

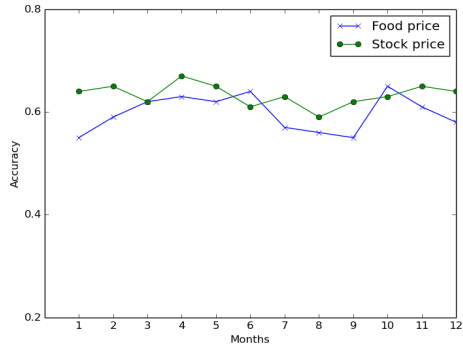
A finer analysis on the EVENT model showed some characteristics in predicting these variables. Figure 6.3a shows the average accuracy of this model in predicting food and stock prices for each month across the 7 years. For food price we see that performance goes down in the middle of the year (Jul-Sep) and again in December and January. This is due to inherent properties of food price where it displays some seasonal effect. July and August are peak monsoon months and September often suffers from a lasting effect of the monsoon, including floods, disease outbreaks etc. Similarly, during winter (Dec-Jan) prices tend to go up. As this is a natural process, this fluctuations are not driven by events, hence, event-driven prediction failed. Tuning our model to remove such seasonal effects will increase the accuracy for these months. On the other hand, stock price prediction did not show much variation across months. However, if we extend this analysis to yearly average accuracy, we observe an opposite effect. Figure 6.2 summarizes the stock values of the 4 targeted companies between the years 2006 and 2007. All the four companies, showed a common characteristic, the prices went up in 2007 and then went down in the middle of 2008 and continued till 2009. This happened due to some changes in the world-wide economy. As, these incidents were well-covered in the news media, our event-predictive models could capture these well. As a result, we see a rise in the accuracy in predicting the stock prices during these years. This

Table 6.5: The top events (triggers) associated with each crop is shown in the order of their likelihood ratio value.

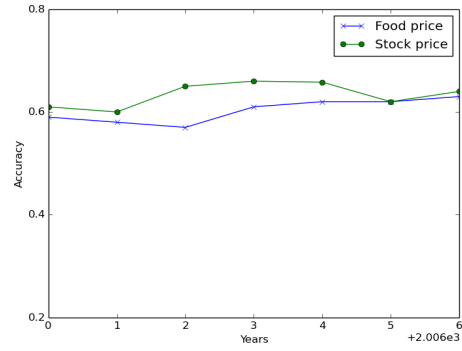
This list shows the top 5% of the triggers for each crop.

Crop	Associated Triggers
Onion	Strikes/agitation/disturbances, increase , arise/occur exported, scam, celebrated flooding, importing, outburst, electing, paralysed, polluted
Potato	Strikes/agitation/disturbances, exported, scam, prevented scam, blockage, probing, hike, storing, raining
Wheat	strikes, importing, rise/hike, stabilize, dominated procure, rafting, sacked, save, disruption, gain
Rice	exported, scam, strikes, celebrated, riding, visited, hail, damaged, rise, acquire, transfer, join

observation is presented in Figure 6.3b. However, we see no such effect for food prices. The rise in both variables' prediction accuracy near the end can be attributed to the fact that Times of India increased the count of online articles gradually. In 2006, the average daily articles were around 200, which increased to 600 in 2011-12. This means more coverage of events, hence, better event-driven prediction. Finally, we performed a more minute analysis on the HIST model. HIST demonstrated the best performance for food price prediction. This lead to the conclusion that events that drive food price has a more delayed and long-lasting effect. We performed an experiment to observe what is the best time delay of these events. We varied the time delay (δ in Eq 6.7) and observed how the accuracy varied with δ . The result of this experiment is shown in Figure 6.4. For all four crops the best performance is demonstrated when δ is either 3 or 4, which means that incorporating past events from the last 3/4 days maximizes the accuracy of predicting the current price. This leads to the conclusion that food prices can be predicted upto 4 days in advance.



(a) Average monthly accuracy of EVENT model across 7 years



(b) Average accuracy of EVENT model for each year [2006-07] topics/events

Figure 6.3: Monthly and yearly average accuracy

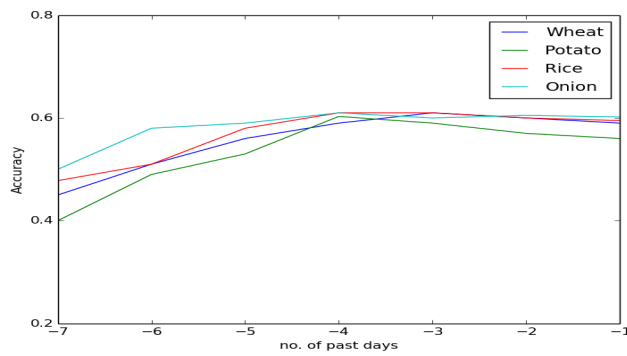


Figure 6.4: Accuracy of HIST model with varying past days. x-axis show how many days in the past was considered

Table 6.6: Associated topics for the triggers from Table 6.5. The topics increase the readability of the event-phenomenon graph. Topic names are manually assigned.

Crop	Topics/Events
Onion	Transport strike, fuel price increase, disease outbreak govt. policies (import/export), scams and scandals festivals, natural calamities
Potato	Transport strike and labor unrest, govt. policies (export) scams and scandals, electricity tariff hike, temperature rise
Wheat	strikes, govt. policies electricity shortage/disruption, Punjab temperature rise
Rice	govt. policies, scams and scandals, strikes religious festivals, Tourism (visits), rise disease cases

6.6 Event Phenomenon Graph

In the previous section, we presented a novel approach to extract events from news corpus. In this section, we describe how we can identify events that are closely associated with a particular phenomenon. Given a set of events \mathbf{x}_K , we need to find a subset of M events, \mathbf{x}_M that have a strong association with a phenomenon y^c , such as observed fluctuation in a socio-economic indicator c . The event phenomenon graph can be constructed using this relationship between news events and a phenomenon. A event-phenomenon graph will have $M + 1$ nodes, where one of the node represents a phenomenon and the rest M nodes are the events related to the phenomenon. For the predictive model, we use these M events as the features to predict the corresponding phenomenon using Eq 6.4.

To build this graph for various phenomenon, specifically to monitor price fluctuation of different crops, we use time series data for these crops and find the times where a fluctuation has been observed. A fluctuation is observed at time t , if y_t^c is at least 10% higher than y_{t-1}^c . We find associated events for y^c based on the co-occurrence of events at all t , when fluctuations have been observed for y^c . We use

likelihood ratio test for to compute this association. For any event $x_i \in x_K$, whether x_i is related to y_c can be found by computing the ratio of two hypotheses: whether x_i and y_c are independent [$p(y^c|x_i) = p(y^c|\neg x_i)$] and whether they are dependent [$p(y^c|x_i) \neq p(y^c|\neg x_i)$]. Repeating this test for all $x_i \in x_K$, we identify the events (triggers) that have a high value of this ratio. Table 6.5 shows these for each crop in the order of their likelihood ratios. These triggers represent the top 5% of the events based on this test. We take these triggers to form the set x_M , which denotes the set of events that have a stronger association with y_c . This set is used as features for the prediction and results are shown in Figure 8.6.

This set can produce good results in predicting fluctuations in food prices as they empirically show strong association with the observed fluctuations. However, a strong motivation for the event-phenomenon graph is providing a human readable interface for understanding the relations between events and an external phenomenon. The triggers can depict some kind of action but without further information these associations might look vague. Hence, we increase the readability of the event-phenomenon graph by including associated topics with these triggers. The associated topics increase the readability by providing some insights to the action given by the triggers. We follow the same method for the TRIGG+LDA model for this purpose. This is done only to increase the quality of the event-phenomenon graph, in spite of TRIGG+LDA not being the best performing predictive model. The event-phenomenon graph using this method for onion is shown in Figure 6.5. The motivation for this step is to better understand what information is given by the generic triggers, such as, *hike*, *rise*, *celebrate*, *exported* etc. Including the topics can provide better insights like, *what is on the rise*, *what has hiked*, *what is being celebrated* etc. Table 6.6 summarizes these events with associated topics for a better understanding of the event-phenomenon relationships by providing finer information. Table 6.6 provides much more insights, for example, it is precisely stating *transport* related strike, or rise in *disease cases*. Thus, combining topics with triggers can increase the readability and help understand precisely what events and topics are associated with the phenomenon.

6.7 Causality Test between Words

In the previous sections, we have presented how to extract events from news articles ways of connecting them with each other or some external phenomenon to build predictive models. This is an example of relation extraction from a large corpus of documents. We also showed that what events are closely



Figure 6.5: Events related to the phenomenon of onion price rise

associated with sudden fluctuations in food prices, stock prices and dengue cases. The question we ask in this section is – *are these events actually driving the fluctuations?*. In other words, can we establish a causal link between them. In this chapter, we explore this question with respect to word-word relations. In other words, if we observe a set of words at a particular time t , we ask what are other words that are likely to be observed at $t, t + 1, t + 2$ and so on. We build time-series of the probability of different words appearing across time and compare these time-series to find how to words co-vary across time. Word time-series can be used to find correlation or causality and derive different types of temporal relations between words.

We use a news corpus for 7 years between 2006 and 2012, and use the articles published across these years to build a vocabulary and the temporal distribution of that. The time-series of words are build based on the relative frequency of these words for a given day. We apply different methods to find association between words, which includes, correlation coefficients, likelihood ratio, mutual information. All these methods can help in finding mutual dependence between words. However, these methods are unable to find causal links between words. We employed a particular causality test, called Granger Causality to exploit the causal links between various word pairs, if exists. Using successful causal pairs of words we constructed a causal word graph that summarizes all the causal pairs found in the corpus.

We evaluate our concept of temporal word relationships with first providing some anecdotal examples of word pairs. We also compare the performance of the different methods by performing experiments on a manually labeled test set. Our findings show that all of methods demonstrate high recall but comparatively

lower precision. And, among other methods likelihood ratio showed the best results. Finally, we showed an example of the causal graph and evaluated the causal relations using .

6.7.1 Word-word Relations

Online news can be viewed as a stream of articles appearing through time. Every article again is a stream of words. So, a collection of news articles published over a time period can be viewed as a stream of words. Many statistical text mining models assumes that every word is independent to each other and the altering word order does not affect any thing. This might be true for words appearing within the boundaries of a document, the chronological ordering of words might carry useful information. There are other models which look at ordering of words, such as ngram model. However, all these models look word ordering or word-word relationships within a document. As news streams produce words with time there might be latent relationships between words that go overlooked in middle of vast amount of data.

In this chapter we address this issue of finding latent relationships between words across documents and across time. The premise of this idea is that there are topics and events that generate these news articles. Every such higher order entities produce certain/specific words. It is well researched that many events and topics are related to each other, so, the underlying words should also be related. The question we are addressing in this chapter is, given a pair of words w_i and w_j and a time difference $\delta = \{0, 1, 2, \dots\}$, how likely it is that w_i and w_j are related to each other for different values of δ . In other words, we would like to derive function,

$$\mathcal{F}(w_i, w_j, \delta)$$

where, \mathcal{F} can measure the relatedness of w_i and w_j . Various queries can be answered with \mathcal{F} . For example, given a word w_i , what are the other words that are related to w_i for different values of δ . In other words,

$$W_\delta = \operatorname{argmax}_{w_k \in V} \mathcal{F}(w_i, w_k, \delta)$$

for different values of δ . This method gives us a list of words for a given δ that are likely to show up in news after δ days from w_i showing up in the news. Another variation of this representation is to find out the value of δ that maximizes the co-occurrence of a specific pair of words w_i and w_j , i.e.

$$\Delta_{w_i, w_j} = \operatorname{argmax}_{\delta} \mathcal{F}(w_i, w_j, \delta)$$

All these values help us to understand one basic question, given the temporal distribution of words, which words are related to each temporally and observing one word, when is the other one most likely to occur. For all these queries the required data is time-series of different words and the methodology involve finding the best way to design \mathcal{F} .

Time Series of Words

Typically, a corpus of documents used in NLP or any other text mining research is a collection of documents, where the inter-document links are not considered. The documents are assumed to be independent to each other. However, in some corpus the documents can have some links, or a way to organize them together. For example, a news corpus, where every document has a date and time of publication. Using this information news articles can be organized chronologically. This can be extended to the words appearing in the articles. In doing so, we get a chronological order of every word appearing in the vocabulary. This can be used to build time series data of words appearing in the news article corpus.

Suppose, V represents the vocabulary of a news corpus \mathcal{D} . Let w_i is the i^{th} word in V . The net term frequency of w_i can be distributed over time based on the timestamps of the articles in which w_i appeared. So, if we represent the time range of a news corpus as vector of size T , where t_0 represents the first day of the corpus and t_T the last day, then we can represent each $w_i \in V$ as a timeseries. These time-series will represent the temporal distribution of the words. If the time-series of w_i is represented as $\mathcal{T}(w_i)$ of size T then each element in $y_i^t \in \mathcal{T}(w_i)$ will represent some weight (e.g. frequency) showing the prominence of w_i at time t . Numerous analysis can done using such time-series, such as finding temporal dependence between words appearing over time. There are many variation of the score of w_i at in the time-series. It can be raw frequency, where y_i^t at time t is the total frequency of w_i on that day. A variation of this can be a binary version where time-series value at time t is 1 if the frequency of w_i is greater than a threshold or 0 otherwise. This version is a crisper representation of the time distribution of w_i .

Word Relations in Documents

The Probabilistic Reduction Hypothesis claims that words are more reduced when they are more predictable or probable. There are many ways to measure the probability of a word. This section discusses a number of local measures that we have studied, although we will mainly report on two measures: the conditional probability of the target word given the preceding word and the conditional probability of the target word given the following word. The simplest measure of word probability is called the prior probability. The prior probability of a word is the probability without considering any contextual factors (prior to seeing any other information). The prior probability is usually estimated by using the relative frequency of the word in a sufficiently large corpus. For a word w_i , the prior probability is defined as: $C(w_i)/N$, where $C(w_i)$ is the total count of w_i in the corpus and N is the total number of words. This probability is a measurement of the word's relative importance without any context. On the other hand, joint and conditional probabilities of two words are often used to measure relationships between two words, often to measure how two words are likely to occur consecutively. The difference between the conditional and joint probability is that the conditional probability controls for the frequency of the conditioning word. For example, pairs of words can have a high joint probability merely because the individual words are of high frequency (e.g., *of the*). The conditional probability would be high only if the second word was particularly likely to follow the first. Most measures of word cohesion, such as conditional probability and mutual information, are based on such metrics which control for the frequencies of one or both of the words.

Collocation is based on this concept, where word pairs and phrases are computed where the entire phrase is very likely given the appearance of the individual words. Another application of word-word relations is statistical language modeling (SLM). In SLM, the main issue is to find out the next likely word given a sequence of words. There are many variations of SLM, such as, unigram, bigram, trigram where it looks into a sequence of one, two and three words respectively to predict the next likely word to complete a valid sentence in a language. So, SLM word word relations are limited to consecutive occurrence in a sequence or in a sentence. Such existing initiatives in finding word relationships were limited to the boundaries of a document. Inter-document relations were not explored much. In this chapter, we provide a broader perspective of word relations across documents and across time.

6.7.2 Finding Word Relations

There are many statistical methods to identify whether two entities are related. Given time-series of two words, numerous metrics and methods can help to find whether the two words are related or not. Correlation coefficient is among such methods. Pearson's correlation coefficient can be used to find how two word time-series co-vary. Correlation can be computed on a pair of word time-series to identify how much they co-vary. This measure can give us an estimate of the likelihood that of one word appearing by observing the occurrence of another word. In addition, we used several other methods used to find word-word association in the past. In the rest of the section, we discuss those methods in some more details.

Likelihood Ratio Test

Each of the two competing models, the null model and the alternative model, is separately fitted to the data and the log-likelihood recorded. The test statistic (often denoted by D) is twice the difference in these log-likelihoods:

$$D = -2 \ln \left(\frac{\text{likelihood for null model}}{\text{likelihood for alternative model}} \right)$$

The model with more parameters will always fit at least as well (have an equal or greater log-likelihood). Whether it fits significantly better and should thus be preferred is determined by deriving the probability or p-value of the difference D . Where the null hypothesis represents a special case of the alternative hypothesis, the probability distribution of the test statistic is approximately a chi-squared distribution with degrees of freedom equal to $df_2 - df_1$. [2] Symbols df_1 and df_2 represent the number of free parameters of models 1 and 2, the null model and the alternative model, respectively.

Here is an example of use. If the null model has 1 parameter and a log-likelihood of -8024 and the alternative model has 3 parameters and a log-likelihood of -8012 , then the probability of this difference is that of chi-squared value of $-2(-8024 - (-8012)) = 24$ with $3 - 1 = 2$ degrees of freedom. Certain assumptions [3] must be met for the statistic to follow a chi-squared distribution, and often empirical p-values are computed.

The likelihood-ratio test requires nested models, i.e. models in which the more complex one can be transformed into the simpler model by imposing a set of constraints on the parameters. If the models

are not nested, then a generalization of the likelihood-ratio test can usually be used instead: the relative likelihood.

Mutual Information

Mutual information (MI) of two random variables is used to measure how the two variables are dependent to other. MI is more general and determines how similar the joint distribution $p(X,Y)$ is to the products of factored marginal distribution $p(X)p(Y)$. MI is the expected value of the pointwise mutual information (PMI). The most common unit of measurement of mutual information is the bit. Here, we are interested only in MI of discrete time-series data of different words, hence MI between two discrete variables X and Y is computed as,

$$I(X;Y) = \sum_{y \in Y} \sum_{x \in X} p(x,y) \log \left(\frac{p(x,y)}{p(x)p(y)} \right),$$

where $p(x,y)$ is the joint probability distribution function of X and Y , and $p(x)$ and $p(y)$ are the marginal probability distribution functions of X and Y respectively. Intuitively, mutual information measures the information that X and Y share: it measures how much knowing one of these variables reduces uncertainty about the other. For example, if X and Y are independent, then knowing X does not give any information about Y and vice versa, so their mutual information is zero. At the other extreme, if X is a deterministic function of Y and Y is a deterministic function of X then all information conveyed by X is shared with Y : knowing X determines the value of Y and vice versa. As a result, in this case the mutual information is the same as the uncertainty contained in Y (or X) alone, namely the entropy of Y (or X). Moreover, this mutual information is the same as the entropy of X and as the entropy of Y . (A very special case of this is when X and Y are the same random variable.)

Mutual information is a measure of the inherent dependence expressed in the joint distribution of X and Y relative to the joint distribution of X and Y under the assumption of independence. Mutual information therefore measures dependence in the following sense: $I(X; Y) = 0$ if and only if X and Y are independent random variables. This is easy to see in one direction: if X and Y are independent, then $p(x,y) = p(x) p(y)$, and therefore:

$$\log \left(\frac{p(x,y)}{p(x)p(y)} \right) = \log 1 = 0.$$

Moreover, mutual information is nonnegative (i.e. $I(X;Y) \geq 0$; see below) and symmetric (i.e. $I(X;Y) = I(Y;X)$).

All the methods discussed in this section compute how the two time series are dependent to each other. However, the direction of this dependence is not apparent. One way to get this direction is by looking at the chronological ordering of the words, as they appear in the corpus. However, that still might not give a strong notion of causality links between the word pairs. Finally, use causality tests on these word time-series to find causal links between the word pairs. We experiment with a popular test called Granger causality test.

6.7.3 Granger Causality Test

Given two time-series, the Granger causality test is a statistical test that checks whether the second time-series is more effective in predicting the first one than using just the first. Compared to correlations, causality can be used to predict the future values of a time series using prior values of another time series. According to Granger causality, if a signal X_1 "Granger-causes" (or "G-causes") a signal X_2 , then past values of X_1 should contain information that helps predict X_2 above and beyond the information contained in past values of X_2 alone. Its mathematical formulation is based on linear regression modeling of stochastic processes (Granger 1969). More complex extensions to nonlinear cases exist, however these extensions are often more difficult to apply in practice.

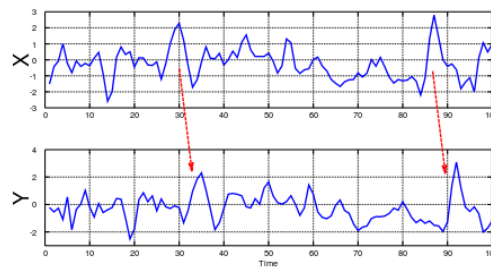


Figure 6.6: Causality between two time-series [src: Wikipedia]

A time series X is said to Granger-cause Y if it can be shown, usually through a series of t-tests and F-tests on lagged values of X (and with lagged values of Y also included), that those X values provide statistically significant information about future values of Y .

Formulation

The causality relationship based on two principles:

1. The cause happens prior to its effect.
2. The cause has unique information about the future values of its effect.

The hypothesis for identification of a causal effect X of Y on is:

$$P[Y(t+1) \in A | \mathcal{I}] \neq P[Y(t+1) \in A | \mathcal{I}_{-X}(t)] \quad (6.11)$$

where A is an arbitrary non-empty set, \mathcal{I} and \mathcal{I}_{-X} respectively denote the information available as of time t in the entire universe, and that in the modified universe in which X is excluded. If the above hypothesis is accepted, we say that X Granger Y causes .

Let y and x be two time series. To test the null hypothesis that x does not Granger-cause y , one first finds the proper lagged values of y to include in a univariate autoregression of y :

$$y_t = a_0 + \sum_{i=1}^m a_i y_{t-i} + \epsilon_t \quad (6.12)$$

Next, the autoregression is augmented by including lagged values of x :

$$y_t = a_0 + \sum_{i=1}^m a_i y_{t-i} + \sum_{j=p}^q b_j x_{t-j} + \epsilon_t \quad (6.13)$$

One retains in this regression all lagged values of x that are individually significant according to their t -statistics, provided that collectively they add explanatory power to the regression according to an F -test (whose null hypothesis is no explanatory power jointly added by the x 's). In the notation of the above augmented regression, p is the shortest, and q is the longest, lag length for which the lagged value of x is significant.

The null hypothesis that x does not Granger-cause y is not rejected if and only if no lagged values of x are retained in the regression.

However, Granger causality is not necessarily true causality. If both X and Y are driven by a common third process with different lags, one might still fail to reject the alternative hypothesis of Granger causality. Yet, manipulation of one of the variables would not change the other. Indeed, the Granger

test is designed to handle pairs of variables, and may produce misleading results when the true relationship involves three or more variables. A similar test involving more variables can be applied with vector autoregression.

Causal Graph

Granger causality can also provide the time lag for which the causality is found to be optimum [321]. This information can be used to build a word causality graph, showing the causal links between word pairs. The causality graph can be defined as, $\mathcal{G}(V, E)$, where, the vertices V are words from the news corpus, and E are the edges between the words, depicting causality. An edge in E , e_{ij} is defined as a directed link between the word w_i to w_j annotated by an edge weight, defined as, α_{ij} denoting the time gap between the time-series of w_i and w_j for which the causality is shown to be found maximum.

6.7.4 Evaluation

Data

Our words were extracted from a news corpus of 7 years, appearing between 2006 and 2012. The corpus had 700,863 articles with 699,754 unique words after removing stopwords and numerals. However, this list has a huge number of very rare words appearing very few times in the corpus. On the other hand, there are many words very frequently appearing the corpus that potentially will co-occur with many other words without having any relationship. So, we filtered the list of words to include not very rare and very frequent words. We included words that appear at least in 1000 articles and removed the top 5% of the words. This process reduced the list of words to a list of size 2356 words.

Results

In this chapter, we try to find relationships, including causality to build a word-word graph. It is difficult to evaluate such a method as there are no standardized tests, metrics or dataset to evaluate such a task. We start the evaluation of finding meaningful pairs of words by providing some anecdotal evidences. Table 6.11-6.14 shows a list of 28 words manually chosen and the corresponding word pairs to which G-Causal test was positive. The words in left column of these tables were randomly chosen. For each word, we ran the G-Causal test for all the other words in the vocabulary and for all the words, where the

test returned positive are listed in the right hand column. A lot of noise can be observed in the result tables. However, certain word pairs are meaningful. For example, the word “currency” leads to frequent appearance of “airline” in 3 days, “CNG” (compressed natural gas) in a week, “coal”, “crop” in a month. Similarly, “monsoon” leads to disease like “diarrhoea” in 2 months. Also “flooding” in 2 months whereas, “downpour” leads to “flooding” in a day. The granger causality is able to draw directed edges between words. The meaningful pairs are a few among other noisy pairs. Hence, our future work for this chapter will focus on narrowing down the results with better precision. The other methods used in this chapter can only find association between words, i.e. draw undirected edges. Next we evaluate all the different methods used to find word relations. Again we start with some anecdotal examples of word-pairs found by these methods, followed evaluating their performance. The examples are shown in Table 6.9

We applied the different techniques to find relations between words discussed in Section 6.7.2. The objective is find out the best method to find word relations from a corpus of text. We manually identified 35 pairs of words that are known to have dependencies between them. The task is to identify the method that can best identify these pairs. We applied three methods on our news corpus – Pearson’s correlation coefficient; likelihood ratio; mutual information. We used the method on the word pairs from the test set to see how many among them has been included as a valid and related word pair. The experiment was designed as follows – for every word pair in the test set, we pick a random word. We take all other words in the vocabulary and apply the methods (correlation, mutual information etc.) to see how many related word pairs are computed for the chosen word and among them how many are among the test set pairs. The test set has the form $T = \{(w_1, w_2), (w_1, w_3), (w_4, w_5) \dots (w_m, w_n)\}$, where each (w_i, w_j) represents a pair of related word pair. For the experiment, we pick one word w_i at random from all the pairs $(w_i, w_j) \in T$ and compare w_i with all other words in the corpus vocabulary (V), such as $w_k \in V$, where $w_k \neq w_i$, and test if w_j tests positive as a related pair with w_i . We evaluated the performance using precision-recall metrics. Table 6.10 summarizes the performance of these methods. Across all methods precision has been low compared to recall. This observation can attributed to the fact that all these methods extracted plenty of word pairs showing high association given a word from the test set. So, the precision values were low across all the methods. On the other hand most of pairs in test set, were found to be positive by these methods, thus the recall values were relatively high. The performance of likelihood ratio test was best among all these methods. A plausible explanation is that likelihood ratio has been found to work well in sparse cases. As many of these words had a sparse distribution in the corpus,

likelihood ration performed better in this case.

6.8 Summary

This chapter presented a way to extract and represent events from news articles. First part of the chapter focused on how to detect events from news articles and given a corpus of news articles within a time period, provide answers to questions like, when and where an event has occurred. All the events were handled independently and the inter-dependencies of events were completely ignored. In the second part, the inter-dependency of events are explored based on their spatio-temporal properties. This concept is extended by determining the relations between news events and external phenomenons that can be easily observed. We use these events and the dependencies to predict the fluctuations in three such phenomenons – food price, stock prices and dengue outbreaks in India. In the final part of the chapter a new kind of relationship is presented. This part tried to explore causal relationships at a word level to identify pairs of words that potentially have a cause-effect relationship. This word-level relationship is represented as a word graph, where the edges represent causal relationships between word pairs tested positive using Granger’s causality tests.

Table 6.7: Examples of events extracted from the news data. For each event class the main event triggers are shown in the left column. The main event triggers are assumed to be equally likely so there are no orders. In the right column, subsidiary events are shown in the order in which they are more likely to be associated with the event class

Event triggers	Subsidiary event triggers
molest,kill,eliminate,abduct, manhandle,kidnap	(including:0.071),(denied:0.048),(eliminated:0.048),(killed:0.048) (left:0.048),(set:0.048),(chopped:0.024),(elected:0.024), (escalating:0.024),(estimated:0.024),(expressed:0.024),
accused,suspects,killers,kingpin, conspirators,masterminded	(arrested:0.03),(found:0.013),(told:0.013),(raped:0.01),(filed:0.009), (registered:0.009),(alleged:0.009),(claimed:0.008),(including:0.008),
supporting,allies,backing, marxists,criticising,tacit	(added:0.062),(activists:0.031),(advised:0.031),(armed:0.031), (arrested:0.031),(attended:0.031),(concerned:0.031),(concerted:0.031), (engaged:0.031),(extending:0.031),(found:0.031),
drought,flood,worst, tsunami,situation,cyclone	(provide:0.027),(pump:0.027),(added:0.013),(adding:0.013),(aired:0.013), (allocated:0.013),(announced:0.013),(apathy:0.013),(arrive:0.013), (aila:0.013),(assumed:0.013),(beating:0.013),(changing:0.013)
campaigning,canvassing,campaigned, mayoral,pitching,lobbying	(ensure:0.048),(campaigning:0.038),(premises:0.029),(reserved:0.029), (canvassing:0.019),(closed:0.019),(conducted:0.019),(contesting:0.019), (including:0.019),(leaving:0.019),(prohibited:0.019),(taking:0.019),
capture,decode,recreate, propagate,arouse,ignite	(managed:0.075),(make:0.05),(project:0.05),(ruled:0.05), (alleged:0.025),(appealed:0.025),(attacked:0.025),(based:0.025), (bored:0.025),(capture:0.025),(caste:0.025),(change:0.025),
gained,emerged,lost, boosted,transformed,demonstrated	(purchased:0.094),(exported:0.062),(lift:0.062),(ranging:0.062), (reap:0.062),(added:0.031),(attached:0.031),(availed:0.031), (districts:0.031),(enabled:0.031),(fallen:0.031),(growing:0.031)
blast,bomb,malegaon, explosions,bakery,defusing	(arrested:0.038),(injured:0.03),(sought:0.03),(accused:0.023),(told:0.023), (made:0.019),(picked:0.019),(added:0.015),(demanded:0.015),(file:0.015), (killed:0.015),(occurred:0.015),(found:0.011),(involved:0.011),
protest,demonstration,protests, agitation,dharna,strike	(held:0.016),(staged:0.016),(demanded:0.014),(added:0.013),(pay:0.013), (protest:0.012),(decided:0.01),(told:0.01),(alleged:0.007),(died:0.007), (proposed:0.007),(protesting:0.007),(submitted:0.007),(assured:0.006),

Table 6.8: Continuation of Table 6.7

Event triggers	Subsidiary event triggers
kidnapped,abducted,molested, gangraped,raped,murdered	(called:0.069),(found:0.052),(lodged:0.052),(told:0.052),(abducted:0.034), (added:0.034),(left:0.034),(raped:0.034),(returned:0.04),(approached:0.017), (booked:0.017),(demanded:0.017),(dropped:0.017),(fighting:0.017),
injured,wounded,killed, attacked,abducted,grievously	(died:0.024),(found:0.016),(identified:0.015),(deceased:0.014),(accused:0.014), (hit:0.014),(arrested:0.013),(including:0.013),(rushed:0.012),(sustained:0.01), (added:0.01),(lost:0.009),(admitted:0.009)
negotiate,align,coordinate, merge,cooperate,integrate	(ruled:0.094),(decided:0.062),(directed:0.062),(filed:0.062),(give:0.062), (hear:0.062),(observed:0.062),(pending:0.062),(registered:0.062),(citing:0.031), (considered:0.031),(declined:0.031),(divorce:0.031),(giving:0.031)
winter,rains,monsoon, showers,summer,fog	(coupled:0.132),(coordinate:0.053),(distributed:0.053),(improved:0.053), (making:0.053),(predict:0.053),(provide:0.053),(added:0.026),(adopted:0.026), (approved:0.026),(chance:0.026),(collaborating:0.026),(contributing:0.026)
paralysed,disrupted,crippled, unaffected,waterlogged,jammed	(told:0.036),(bed:0.018),(speak:0.018),(started:0.018),(accused:0.014), (paralysed:0.014),(joined:0.011),(suffered:0.011),(thrown:0.011), (arrested:0.007),(brought:0.007),(facilitate:0.007),(fed:0.007),(fell:0.007),
elected,nominated,unopposed, delimited,councilors,uncontested	(elected:0.16),(added:0.019),(held:0.019),(submit:0.019),(won:0.019), (working:0.019),(affected:0.013),(announced:0.013),(attended:0.013), (called:0.013),(carry:0.013),(created:0.013),(ensure:0.013),(formed:0.013),
accident,mishap,incident, explosion,dharamkata,stampede	(injured:0.025),(found:0.022),(hit:0.019),(added:0.016),(claimed:0.016), (driving:0.013),(killed:0.013),(told:0.013),(brought:0.009),(carried:0.009), (caused:0.009),(coming:0.009),(deceased:0.009),(donated:0.009),
drowned,electrocuted,capsized, molested,rescued,perished	(drowned:0.059),(confirmed:0.034),(feared:0.025),(released:0.025),(told:0.025), (cleaning:0.017),(damaged:0.017),(demanded:0.017),(flowing:0.017), (identified:0.017),(set:0.017),(trapped:0.017),(':0.008),(accused:0.008),
repealed,enforced,scrapped, abolished,curbed,derecognised	(demanded:0.074),(demanding:0.074),(abolish:0.037),(allowed:0.037), (began:0.037),(belonging:0.037),(belt:0.037),(benefit:0.037), (carried:0.037),(directed:0.037),(distributed:0.037), (don:0.037),(give:0.037),(hiked:0.037),(impose:0.037)

Table 6.9: Dependency between event pairs given a time difference δ , computed as normalized point-wise mutual information. δ is in months

Event-pairs	$\delta = 0$	$\delta = 1$	$\delta = 2$	$\delta = 3$
flood-dengue	0.28	0.709	0.872	0.325
monsoon-malaria	0.082	0.54	0.901	0.808
baishakhi-harvest	0.893	0.414	0.120	0.131
narmada-protest	0.780	0.651	0.28	0.119
eid-ramadan	0.838	0.686	0.489	0.294

Table 6.10: Comparison of performance

Method	Precision	Recall
Correlation	0.451	0.787
Mutual information	0.551	0.759
Likelihood ratio	0.613	0.827

Table 6.11: Examples of causal links obtained from the test for a set of selected words. This table is generated by selecting a set of words and then running g-causal test across all other words. The list of words (in alphabetical order) in the right column represents all the words were the test was positive for the word in the left column

Words	Causal Links
airline	ahluwalia, amendment, ayurvedic, bench, boxing, caste, cleanliness, constituencies, curfew, democratic, donors, entire, fever, forensic, goa, hospitals, investigation, judge, km, literacy, margao, mla, narendra, olympics, patna, politics, projects, rajat, regulatory, road, sdo, ship, speech, support, terminal, transporting, vacancies, vote, york,
cng	abu, afghanistan, akash, anticipatory, assam, averted, banks, bharat, bjp, bridge, businessmen, cattle, chemical, cm, companies, constructing, cricketing, dave, denied, dies, donors, economies, engineer, explosives, festivities, food, gangster, gold, hanged, hindustan, illegal, inspector, irrigation, judgement, khalid, labour, legislator, ludhiana, manifesto, matthew, mills, molested, musharraf, newspaper, osmania, parrikar, petroleum, policing, presidency, projects, raid, raped, released, rings, sabha, sanjeev, season, shekhawat, sinha, squad, streets, supremo, taxi, timesgroup, treasury, uday, varanasi, vihar, wales, worli,
coal	ambassador, arrest, cag, corporation, envoy, goa, iran, liberation, navy, prime, safdarjung, transport,
crop	airport, ballot, ceremony, cpm, dismissal, festival, gods, injuries, kochi, mahatma, mohali, olympics, pranab, ramadoss, screen, stadium, transport, west,
currency	affairs, airline, amarnath, australia, bill, bush, charges, cng, coal, collector, corporators, crop, defaulters, elected, fast, games, homemaker, inspector, joshi, kozhikode, logic, maoist, moradabad, navy, ongc, placards, prithviraj, raining, republic, russia, service, speedy, tankers, trichy, victims, yeddyurappa,
cwg	arab, arrest, bond, chhatrapati, dhananjay, festivities, hunger, karim, media, obama, prayer, resignation, sikhs, torched, yrs,

Table 6.12: Continuing from Table 6.11

Words	Causal Links
dengue	agitation, aids, bellary, boats, budgetary, burns, camp, campaign, economically, government, judgement, mercury, nuclear, pressure, rathod, soldiers, tripura,
diarrhoea	amendments, banerjee, breast, ceo, communist, cycles, dock, epidemic, expense, gangster, heat, industry, kalmadi, left, martial, mukul, nominated, pervez, priyanka, raza, sabha, shells, storm, tmc, vhp, zaheer,
downpour	address, autopsy, breath, city, coast, december, emergency, flooding, government, investors, lakh, milind, october, project, rs, silva, temperatures, vajpayee, yrs,
drought	admission, andhra, bureaucrat, business, canada, china, chinese, city, congress, illiterate, mirror, rawat, terrorism,
electricity	ahmedabad, assembly, atomic, bhutan, booming, budget, city, coal, corporation, heat, lankan, nobel, qaida, slowdown, varun,
flood	activist, applications, bangla, bokaro, census, coast, criminal, denied, encroachment, fireworks, gondia, icc, investors, judicial, landing, mahatma, mithun, national, parties, pressure, race, residence, sangam, shootout, subhash, terrorism, vandana, yadav,
gdp	abusive, ahmed, animal, assaulted, banning, bikes, broker, carnage, cellphone, cheque, child, chronic, comfortable, conservative, crackdown, darbhanga, deprived, disease, drug, entire, farming, flames, gautam, guntur, homemaker, infected, iraq, judicial, kmc, laptop, literary, mahatma, meal, mortem, national, ore, parrikar, plea, presidents, protest, rajeev, reforms, rice, salaries, seat, shoots, speech, streets, surgeons, tdp, tilak, trees, urdu, villagers, wheel,

Table 6.13: Continuing from Table 6.12

Words	Causal Links
holi	agitation, anticipatory, ban, bsf, chargesheet, compensation, councillor, deployment, divisions, expenses, forests, grocery, hrd, international, judicial, lajpat, lucknow, mbbs, natarajan, panel, plans, prizes, rashtriya, revise, sabha, sewer, state, survive, threaten, tumkur, vigilance,
irrigation	assam, charred, durga, hindus, maoist, monsoon, naxal, radical, rain, rainfall, road, showers, slowdown, reports, subscribers,
land	arrested, bill, budgetary, humidity, coalition, climate, died, force, iranian, mangalore, nashik, poor, road, suffering, victim,
monsoon	afzal, attack, bhushan, burst, convicts, crop, diabetes, diarrhoea, fervour, flooding, heart, hunger, igi, kingdom, maoists, muslim, ongc, pramod, raining, religion, secunderabad, snow, talk, violence,
naxalite	aamir, address, aircraft, ansari, atal, bangalore, bhubaneswar, border, burn, casualties, cheap, civic, colleges, congested, councillor, dalits, defaulters, dgp, disease, earth, empowerment, expense, farm, fleet, fury, glasses, haq, himanshu, ill, institute, irrigation, journalist, kids, labour, lend, madhya, maoists, milind, monitor, muslim, naidu, nature, navin, naxals, odisha, ore, parel, payment, policeman, pregnant, produce, proximity, railways, rash, reign, result, routes, samiti, score, shakti, slowdown, ssp, suburbs, survival, telegraph, tourism, treaty, union, verify, visakhapatnam, wheeler, yield,
petroleum	abduction, acquisition, agriculture, animal, assembly, battling, boards, burst, business, city, charges, chennai, colombo, corporation, dam, deployment, distribute, elected, eyes, fireworks, flooding, gautam, guwahati, housing, inspectors, jayant, kant, lakh, loans, maoists, mineral, msedcl, newspaper, pacify, petition, prevent, puja, raining, rane, reports, rivers, scope, shootout, spain, struggling, taluk,
price	accused, aziz, cag, corp, dismissal, festivities, illegal, khalid, lakh, land, mercury, obama, projects, roads, tariff, sydney, victims,

Table 6.14: Continuing from Table 6.13

Words	Causal Links
rainfall	akali, bobby,dam, darjeeling, fdi, flu, lakh, odisha, nations, rain, religion, slogans, virus, uttar,
rice	academic, airline, ap, agriculture, bamboo, bill, bipasha, burns, chatterjee, coast, construct, dam, development, economy, expressway, flood, gauge, hanged, indiranagar, jalandhar, kejriwal, lalgarh, madhya, matthew, nano, olympic, patronage, prajapati, prosecutor, ranchi, retail, sales, service, spiritual, supporter, terrorists, trs, vigilance, winter,
rioting	acres, ativist, ahluwalia, allies, apollo, attacks,banned, bhatt, bones, burning, castes, chef, cm, communal, controversies, crackdown, daniel, democracy, died, dropping, encounter, extortion, film, force, gehlot, greenery,headquarters, hot, indore, interrogated,japanese, kapadia, kozhikode, legislators, madurai, maoists, molestation, murder, nations, nurse, pakistani, pawar, politicians, prime, protests, railways, ratna, remove, rising, samiti, seal, shekhawat, sikhs, speedy, students, swaraj, temple, torture, tripathi, vacation, vijayawada, western, youths,
river	alliance, churchill, died, flu, kochi, mns, pilots, rathore, rain, slowdown, virus,
soil	alliance, bali, budget, cold, dam, democracy, festive, fox, heat, humidity, italian, kidney, mahindra, molestation, nris, nature, salem, spain, survival thiruvananthapuram, virus,
strike	admission, attacks, blast, chennai, protests, reports, sex, strikes, tour, visited, consulate, diamonds, federation, gurdwara, indonesia, kadapa, law, mittal, navy, pervez,
tariff	assam, attack, bill, buddhadeb, cm, corporation, democracy, empowerment, engineer, finance, hike, irani, kumar, matthew, national, organs, pressure, rallies, score, statues, train, winter,
vegetable	afghanistan, budget, christmas, chandrababu, chennai, coalition, cold, constituency, devotees, farms, gadkari, hindutva, israeli, killed, mahatma, mercury, narendra, omar, precautionary, ram, rss, ship, students, treatment, wedding,

Chapter 7

Concept Graphs and Comprehension Diagnostics for TextBooks

7.1 Introduction

In the previous chapters, we have shown how different types of web data can be used to extract knowledge and be used for different applications. In this chapter we focus on a very specific domain and data related to that domain. Education is known to be a key determinant of economic growth and prosperity [297, 319]. While the issues in devising a high-quality educational system are multi-faceted and complex, textbooks are acknowledged to be the educational input most consistently associated with gains in student learning [314]. Textbooks are the primary conduits for delivering content knowledge to the students and the teachers base their lesson plans primarily on the material given in textbooks. Considerable research has gone into investigating what makes for good textbooks [289, 293, 298, 305, 318]. There has also been work on designing ideal textbook and guidelines and checklists have been proposed for assessing the quality of a textbook [283, 286, 309]. Most education researchers concur that the good textbooks are organized in a systematically progressive fashion so that students acquire new knowledge and learn new concepts based on known items of information [282, 301, 309]. Unfortunately, many textbooks suffer from what Harriet Tyson-Bernstein calls the mentioning problem [313] that causes concepts to be encountered before they have been adequately explained and forces students to randomly knock around in

the book [286]. Indeed, the extensive survey in [284] reports a number of empirical studies in which learning from textbooks was successfully improved by rewriting the text to enhance comprehension. We propose a diagnostic tool for authors to enable them to mine the content of a textbook to quantitatively assess the comprehension burden that a particular organization of the textbook can impose on the reader due to non-sequential presentation of concepts. A textbook with large comprehension burden makes it harder for the reader to understand the textbook material. We introduce the notion of comprehension burden, present a formal definition, and provide an algorithmic approach for computing it. We evaluate the proposed methodology over a corpus of Indian textbooks that demonstrates its effectiveness for identifying books and sections of textbooks that can benefit from reorganizing the material presented.

7.2 Related Work

The question of what factors influence understandability of a text has intrigued researchers for a long time. An early comprehensive investigation, dating back to 1935 [294], identified two principal sets of factors. The first set pertains to individual differences amongst readers, such as levels of intellectual capacity, reading skills, attitudes and goals, previous experiences, and personal interests and tastes. The second set relates to the readability of the material, which in turn depends on format (page layout, appearance, etc.), organization (headings, indexes, flow, etc.), style (linguistic structural elements, tone of the writer, etc.), and content (theme, nature of the subject matter, etc.). Much of the readability research has focused on the style category because of perceived relative importance of stylistic variables and the fact that stylistic variables are easier to operationalize [296]. We refer the reader to the survey in [290] for overview of readability research. Sherman is considered to be the first to use statistical analysis for analyzing readability in 1890s. By counting average sentence length, he showed how sentence-length averages had shortened over time [310]. The first readability formula, a weighted index of vocabulary complexity, is attributed to the work of Lively and Pressley in 1923 [302]. Since then over two hundred formulas have been developed for measuring the difficulty of reading. Some popular formulas include Flesch Reading Ease Score, Flesch-Kincaid Grade Level, Dale-Chall Grade Level, Gunning Fog Index, SMOG Index, Coleman-Liau Index, and Automated Readability Index. The readability formulas have come under criticism because of their purported low validity from the perspective of psycholinguistic theories [285] and there have been efforts to develop new approaches for predicting reading difficulty, for example, by

using statistical language modeling techniques and linguistic features [288, 300] and by devising domain-specific readability measures [320]. While this body of readability research focuses on measuring the difficulty of reading of a text in isolation, our focus is on studying the organization and presentation of concepts in the entire textbook. In linguistics, cohesion refers to connections between sentences, whereas coherence refers to the connectedness of the ideas [317]. Cohesion provides a sense of flow from sentence to sentence and the principle of cohesion states that one must start a sentence with old information and end it with new information. The principle of coherence states that to make a series of individual sentences into a coherent passage, one must focus the topics of those sentences on a limited number of concepts. Discourse parsing techniques are used to build a structural representation of a text which reflects the semantic relationships among basic textual units [295, 303, 307, 311]. We build upon these ideas in our definition of comprehension burden, where we focus on flow from concept to concept. The cognitive load theory proposes that poorly designed instructional materials place cognitive overload on learners as working memory is limited. This cognitive overload impairs schema acquisition, later resulting in a lower performance [304, 306]. The theory distinguishes between three types of load: intrinsic, extraneous and germane. Intrinsic load is a function of the complexity of the content rather than instructional design. Extraneous load on the other hand is not inherent within the content, but depends on how the instructional designer has structured and presented information. Intrinsic and extraneous loads are additive. Germane load is the remaining free capacity in working memory that may be directed toward schema acquisition. Thus, a well-designed book increases the learning capacity of a learner. We make use of these ideas in abstracting the properties of well-written books. In [280], a probabilistic decision model has been proposed for identifying those sections of a textbook that are not wellwritten. The decision model is based on the syntactic complexity of the writing and the notion of the dispersion of key concepts mentioned in the section. However, each section is treated independently and the flow of writing across different sections is not taken into account. By contrast, the focus of the present chapter is on determining whether the entire textbook is organized in a systematically progressive fashion.

7.3 Comprehension Burden

We begin by enunciating properties of well-written textbooks abstracted from education literature. We then describe our model of a textbook and how a poorly written book can impose comprehension burden

on the reader. Finally, after introducing some notations used in the chapter, we formally define comprehension burden of a textbook.

7.3.1 Properties of Well-written Books

PROPERTY 3.1: **FOCUS:** Each section in a well-written textbook explains very few concepts.

PROPERTY 3.2 **UNITY.** For each concept in a well-written textbook, there is a unique section that best explains the concept.

PROPERTY 3.3 **SEQUENTIALITY.** Concepts in a well-written textbook are discussed in a sequential fashion, that is, a concept is adequately explained prior to occurrences of this concept or any related concept.

PROPERTY 3.4 **PRIORITIZATION.** In a well-written textbook, the tie for precedence in presentation between two mutually related concepts is broken in favor of the more significant of the two.

7.3.2 Origin of Comprehension Burden

Although textbooks are typically organized into chapters which are subdivided into logical sections, for the purposes of this chapter, we find it sufficient to model a textbook simply as a sequence of sections. Assume that the textbook has been written in such a way that each section explains one or very few concepts, that is, each of these concepts is discussed in depth (Property 3.1). But a section can also mention other concepts, that is, each of these concepts is referenced but not explained in detail. For each concept, there is a key section in the book that best explains it (Property 3.2). Thus a reader who has gone through the key section corresponding to a concept has comprehended the concept, while a reader who has not yet read the key section has a vague comprehension of the concept. A concept could be related to other concepts, and hence significant for understanding other concepts. A textbook imposes comprehension burden on the reader if the concepts are not presented in a sequential fashion. In particular, for a concept c , an occurrence of a related concept d in a section preceding the key section for c imposes comprehension burden on the reader since the reader has not comprehended c yet and hence faces difficulty in understanding d . Note that d could be same as c , in which case the comprehension burden is imposed on the reader who encounters a mention of c in an earlier section whereas c has been explained properly in a later section (Property 3.3). Suppose the reader encounters d in section i prior to explanation of c in its key section

kc. The more significant the explanation of c in the section kc is for understanding other concepts in the book, the greater chance the reader cannot understand d in section i without thoroughly understanding c and hence larger the comprehension burden on the reader while reading section i . Further, the more significant d in section i is for understanding other concepts in the book, the larger is the comprehension burden since the reader may not be able to follow subsequent material that is based on the discussion of d in section i (Property 3.4). For illustration, consider a book consisting of three sections, each of which contains three mutually related concepts, c_1 , c_2 and c_3 . Unity demands that each concept has a unique key section where it is explained. Focus requires that a section explains very few concepts. Assume that a section can be the key section for only one concept. Suppose that c_1 is the most significant and c_3 is the least significant of the three. Prioritization implies that the author will explain c_1 in section 1, followed by c_2 in section 2, and finally c_3 in section 3. Concept c_1 obeys Sequentiality but not the other two. Its mention in any of the later sections does not incur comprehension burden as the reader has by now comprehended it. A mention of c_2 in section 1 will incur comprehension burden as it is explained only in section 2. However, its mention in section 3 will not incur comprehension burden. Similarly, any mention of c_3 in a section earlier than section 3 will incur comprehension burden. As the above example demonstrates, there is a trade-off to be made since the properties interact with each other. In Although textbooks are typically organized into chapters which are subdivided into logical sections, for the purposes of this chapter, we find it sufficient to model a textbook simply as a sequence of sections. Assume that the textbook has been written in such a way that each section explains one or very few concepts, that is, each of these concepts is discussed in depth (Property 3.1). But a section can also mention other concepts, that is, each of these concepts is referenced but not explained in detail. For each concept, there is a key section in the book that best explains it (Property 3.2). Thus a reader who has gone through the key section corresponding to a concept has comprehended the concept, while a reader who has not yet read the key section has a vague comprehension of the concept. A concept could be related to other concepts, and hence significant for understanding other concepts. A textbook imposes comprehension burden on the reader if the concepts are not presented in a sequential fashion. In particular, for a concept c , an occurrence of a related concept d in a section preceding the key section for c imposes comprehension burden on the reader since the reader has not comprehended c yet and hence faces difficulty in understanding d . Note that d could be same as c , in which case the comprehension burden is imposed on the reader who encounters a mention of c in an earlier section whereas c has been explained properly in a later section (Property 3.3). Suppose

the reader encounters d in section i prior to explanation of c in its key section k_c . The more significant the explanation of c in the section k_c is for understanding other concepts in the book, the greater chance the reader cannot understand d in section i without thoroughly understanding c and hence larger the comprehension burden on the reader while reading section i . Further, the more significant d in section i is for understanding other concepts in the book, the larger is the comprehension burden since the reader may not be able to follow subsequent material that is based on the discussion of d in section i (Property 3.4). For illustration, consider a book consisting of three sections, each of which contains three mutually related concepts, c_1 , c_2 and c_3 . Unity demands that each concept has a unique key section where it is explained. Focus requires that a section explains very few concepts. Assume that a section can be the key section for only one concept. Suppose that c_1 is the most significant and c_3 is the least significant of the three. Prioritization implies that the author will explain c_1 in section 1, followed by c_2 in section 2, and finally c_3 in section 3. Concept c_1 obeys Sequentiality but not the other two. Its mention in any of the later sections does not incur comprehension burden as the reader has by now comprehended it. A mention of c_2 in section 1 will incur comprehension burden as it is explained only in section 2. However, its mention in section 3 will not incur comprehension burden. Similarly, any mention of c_3 in a section earlier than section 3 will incur comprehension burden. As the above example demonstrates, there is a trade-off to be made since the properties interact with each other. In

7.3.3 Notations

Before formally defining comprehension burden, we introduce some notations. Let $S = 1, 2, \dots, n$ denote the set of sections in a given textbook. Let C denote the set of concepts in the book. For each concept $c \in C$, let k_c denote the key section for understanding the concept. For each concept $c \in C$, denote the set of concepts related to it by $R(c)$. Note that $R(c)$ includes c . Let $\lambda(c, i)$ denote the significance score of concept c in section i for understanding other concepts in the entire book. Let $d \in R(c)$ be a concept related to c . We denote by $\psi((d, i) \leftarrow c)$ the comprehension burden imposed on the reader while reading about d in section i due to c being necessary for understanding d but c being explained in a later section in the book. We say that $\psi((d, i) \leftarrow c)$ is the comprehension burden for concept d in section i attributed to concept c . Similarly, denote the comprehension burden for concept d attributed to concept c over all sections by $\psi(d \leftarrow c)$, the comprehension burden on all concepts attributed to concept c by $\psi(c)$, the

comprehension burden of section i by $\psi(i)$, and the total comprehension burden of the textbook by \mathcal{B} .

7.3.4 Definition of Comprehensive Burden

We now present the formal definition of comprehension burden for a given textbook, assuming that the book satisfies the FOCUS and UNITY properties.

1. Given a concept $c \in C$ with key section k_c , and a concept $d \in R(c)$ occurring in section i , define the comprehensive burden for concept d in section i attributed to concept c as

$$\psi((d, i) \leftarrow c) = \begin{cases} f(\lambda(d, i), \lambda(c, k_c)) & \text{if } i < k_c \\ 0 & \text{if } i \geq k_c \end{cases}$$

where f is a monotonically increasing function in two variables satisfying $f(x, y) < f(y, x)$ whenever $x > y$.

2. Given a concept $c \in C$ and a concept $d \in R(c)$, define the comprehension burden for concept d attributed to concept c as

$$\psi(d \leftarrow c) = \sum_{i \in S} \psi((d, i) \leftarrow c)$$

3. Define the comprehension burden attributed to concept $c \in C$ as

$$\Psi(c) = \sum_{d \in R(c)} \psi(d \leftarrow c)$$

4. Define the comprehension burden of section $i \in S$ as

$$\Psi(i) = \sum_{d \text{ in section } i} \sum_{c: d \in R(c)} \psi((d, i) \leftarrow c)$$

5. Define the comprehension burden of the textbook as

$$\mathcal{B} = \sum_{c \in C} \Psi(c) = \sum_{i \in S} \Psi(i)$$

The key aspect of Definition 3.5 is to quantify $\psi((d, i) \leftarrow c)$. Occurrence of d in kc or a section following it does not impose any comprehension burden on the reader since the reader would have understood c before encountering d , and hence $\psi((d, i) \leftarrow c)$ is non-zero only when $i < k_c$. See Figure

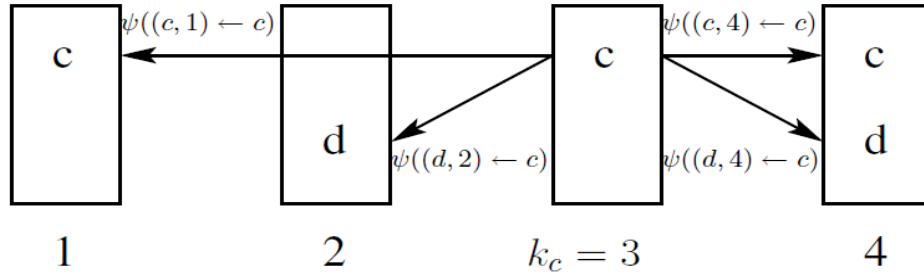


Figure 7.1: Illustration of comprehension burden: Concept c is explained in section 3 ($k_c = 3$) and is also mentioned in sections 1 and 4. A related concept d occurs in sections 2 and 4. As c is explained in section 3, the reader incurs comprehension burden when reading about c in section 1 ($\psi((c, 1) \leftarrow c) > 0$) and about d in section 2 ($\psi((d, 2) \leftarrow c) > 0$), but not in section 4 when encountering c and d ($\psi((c, 4) \leftarrow c) = \psi((d, 4) \leftarrow c) = 0$).

7.1 for an illustration. Burden $\psi((d, i) \leftarrow c)$ depends monotonically on the significance score of concept d in section i and the significance score of concept c in section k_c , and has been expressed as a monotonically increasing function $f(\lambda(d, i), \lambda(c, k_c))$. As the reader incurs comprehension burden every time she encounters an occurrence of d in a section prior to k_c , we sum $\psi((d, i) \leftarrow c)$ over all sections to obtain $\psi(d \leftarrow c)$. The comprehension burden attributed to concept c , $\Psi(c)$ is obtained by summing $\psi(d \leftarrow c)$ over all concepts related to c . The comprehension burden of section i , $\Psi(i)$ is obtained by summing $\psi((d, i) \leftarrow c)$ over all concepts that d is related to and then over all concepts in section i . The comprehension burden of the textbook is obtained by summing the comprehension burden attributed to each concept, or equivalently by summing the comprehension burden of each section.

sectionComputing Comprehension Burden

Computation of comprehension burden of a textbook requires the following inputs: (i) concepts in the book, (ii) relationship between concepts, (iii) the significance score for each concept in each section, and (iv) the key section for every concept. We describe next the computation of each of these inputs.

7.3.5 Concepts

We define concept phrases to be terminological noun phrases. We first form a candidate set of phrases using linguistic patterns, with the help of a part-of-speech tagger. We adopt the pattern AN^+ , where A

refers to an adjective and N a noun, which was found to be particularly effective in identifying concept phrases. Examples of phrases satisfying this pattern include cumulative distribution function, fiscal policy, and electromagnetic radiation. The initial set of phrases is further refined by exploiting complementary signals from different sources. First, WordNet [291], a lexical database is used to correct errors made by the part-of-speech tagger. Next, both malformed phrases and very common phrases are eliminated, based on the probabilities of occurrences of these phrases on the Web, obtained using Microsoft Web N-gram Service [315]. The reason for eliminating common phrases is that they would be already well understood.

7.3.6 Relationship between Concepts

We first attempted to induce relationships between concepts by mapping concept phrases to Wikipedia articles and use the link structure between the Wikipedia articles to infer relationship between concepts. We discovered the following issues. Many Wikipedia articles have asymmetric hyperlink structure, plausibly due the encyclopedic nature of Wikipedia: there are relatively less links from articles on specialized topics to articles on more general topics. For instance, the Wikipedia article titled Gaussian surface mentions electric field 11 times but does not have a link to the latter. Furthermore, while Wikipedia provides good coverage for universal subjects like Physics and Mathematics, it has inadequate coverage for concepts related to localedependent subjects such as History. We, therefore, derive the relationship between concepts directly from textbooks using co-occurrence. More precisely, we defined $R(c)$ to be the set of concepts (including c) that co-occur with c in at least e sections such that both c and the co-occurring concept c occur within a window of size l in each of these e sections. The requirements of co-occurrence in multiple sections and co-occurrence within a window size ensure that we only consider concept pairs that are significantly related to each other. We used $e = 2$ and $l = 500$ words, after confirming through sensitivity analysis that the results were not sensitive to these parameter choices.

7.3.7 Significance Scores

The significance score $\lambda(c, i)$ is a measure of how significant is the description of concept c in section i for understanding other concepts in the book. One possible approach would be to define the significance score in terms of the relative frequency of the concept phrase in the section, for example, $\lambda(c, i) = (freq(c, i)) / (\sum_{1 \leq i < n} \sum_{c \in C} freq(c, i))$. A problem with such a definition is that it does not differentiate

between two concept phrases c_1 and c_2 with the same frequency in a given section. However, concept c_1 may be related to many concepts that occur in other sections in the book and hence more significant for understanding the entire book, while c_2 may be relevant only for the current section. We, therefore, define the significance score of a concept phrase in a section taking into account: (a) how frequent is the concept in the section, and (b) how many concepts are related to it. The significance score of a concept c in section i is defined as

$$\lambda(c, i) = \pi(\text{freq}(c, i), |R(c)|)$$

where π is a monotonically increasing function in two variables.

7.3.8 Key Sections

We use the intuition that the key section for a concept will have high significance score for that concept. Thus, we algorithmically obtain the key section for a concept by comparing its significance scores in different sections. For each concept $c \in C$, we set the key section for c to be

$$k_c = \text{argmax}_{1 \leq i \leq n} \lambda(c, i)$$

this choice is equivalent to selecting the section where the concept is most frequent. We remark that multiple alternatives exist for computing the above inputs and our model of comprehension burden can admit multiple such choices. For example, the significance score computation can benefit by including anaphoric references to a concept when computing its frequency in a section.

7.4 Evaluation

We next present a diagnostic tool we have built based on the notions just introduced that allows an author to understand the sources of comprehension burden in a textbook. This tool helps authors accomplish the following goals: (1) identify and investigate sections with large burden, and (2) identify and probe concepts that impose large burden.

7.4.1 Corpus

We studied the characteristics of our tool over a corpus of Indian high school textbooks published by the National Council of Educational Research and Training (NCERT). We selected this corpus because millions of students study from these books every year and these books were readily available online. We applied the tool to eleven books from grades IX-XII, covering four broad subject areas: Sciences,

Social Sciences, Commerce, and Mathematics. In [279], we described how to quantify the properties in 3.1 and measured the extent to which these properties are followed in the above books. Here our focus is on measuring the comprehension burden in these books. In applying the tool, we instantiate function f as $f(\lambda(d, i), \lambda(c, kc)) = \lambda(c, kc)$ for computing comprehension burden. This choice satisfies the characterization of f has the interpretation that the comprehension burden for concept d in section i attributed to c is proportional to the significance of c in its key section.

7.4.2 Diagnostic Tool

Burden of Sections

The author of a textbook might like to first determine sections where the readers incur large comprehension burden, so that she can prioritize such sections for revision. Hence our tool presents to the author an overview of the book with the sections ordered in the decreasing order of burden. Figure 2 illustrates this page for Grade XII Economics book. From this page, the author can observe that though this book consists of twenty five sections, 75% of the burden arises from just ten sections. Further, Section 4.1 titled Ex Ante and Ex Post has the largest burden. Having identified a section with large burden, the author might next like to understand why the reader incurs large burden in the section. With this goal in mind, our tool allows navigation to a page created for each section. Figure 3 shows this page for the section, Ex Ante and Ex Post.

This page lists the concepts occurring in the section in the decreasing order of their burden. This page also helps the author to determine whether each of these concepts is explained or mentioned in the section, and for a concept that is mentioned, whether it has already been explained in an earlier section or not. Using this information, the author can identify those concepts that are mentioned in the section prior to their explanation in the book. For such a concept, the author may choose to include greater explanation

in this section or an earlier section where it occurs. The author can observe that the section Ex Ante and Ex Post contains a large number of distinct concepts, of which only a small fraction is explained in this section. About half of the remaining have been explained in earlier sections but the remaining half are explained in later sections. Moreover, 75% of the burden for the section arises from half of the concepts. Next the author may drill down to understand what causes burden for each of these concepts. Hence our tool permits navigation to a page dedicated to each concept occurring in the section. This page provides the distribution of the significance score of the concept across different sections in the book. It also shows the related concepts needed for understanding this concept, in the decreasing order of burden they impose on this concept. Figure 4 shows the page for the concept aggregate demand which has the largest burden in the section, Ex Ante and Ex Post. The author learns that this concept is mentioned in this section, prior to its explanation in a later section. Further the author sees that the large burden for aggregate demand is caused by the presence of many related concepts (such as exchange rate, final goods and government spending) which are explained only in later sections.

Burden Attributed to Concepts

The author may also examine the book along an orthogonal dimension by investigating concepts that impose large burden across the book. Such diagnosis is beneficial since the author may choose to include a glossary of such concepts. For this purpose, our tool presents an analysis of the book with respect to concepts present in the book, ordered in the decreasing order of burden imposed by them. Figure 5 shows this analysis for Grade XII Economics book. The author can see that concepts such as aggregate demand, exchange rate, final goods and national income impose the largest burden on the reader, accounting for nearly 60% of the total burden in the book. Moreover, 80% is attributed to just 10% For a high burden concept, the author might like to understand its impact on related concepts. This information might help the author decide whether this concept should be explained in an early section in the book. Hence our tool lets the author navigate to a page for each concept, which lists the burden imposed by this concept on its related concepts. Figure 6 illustrates this page for the concept aggregate demand. By probing into concepts related to this concept and their distribution across sections, the author can infer that the burden can be reduced by explaining aggregate demand in a section prior to occurrences of many of its related concepts.

Differences between Books

We next present some comparative observations from applying our tool to books belonging to three different subjects: Grade XII Economics, Grade X Science, and Grade XII History. These books also had different organizational structure. Figure 7 gives a statistical overview of the sections in these books. It shows the distribution of burden across different sections in a book, as well as the distribution of concepts across different book sections. Note that the X-axis refers to the section number (ordered over the entire book) and that these books have different number of sections. Grade XII History book consists of chapters covering different periods of Indian history, ranging from 3000BC to the 20th century. Our manual inspection revealed that the last few chapters in this book pertain to interrelated topics such as British rule and Indian freedom struggle, while the chapters in the initial two thirds of the book mostly discuss disjoint time periods, and hence disjoint concepts. As a result, the burden for this book arises from very few sections occurring in the last few chapters. Grade X Science book consists of independent modules, corresponding to Biology, Chemistry, Physics, and Environmental Sciences in that order. However certain concepts such as electric current and carbon dioxide are shared across modules, and are mentioned in Biology module prior to their explanation in later parts of the book. Further sections in Biology module contain a large number of concepts. Consequently the burden is concentrated in the initial part of the book. By contrast, Grade XII Economics book pertains to the single theme of macroeconomics, with different sections sharing related concepts, and hence the burden is spread out across more sections. We also observe that sections with large burden tend to have a large number of concepts.

7.5 Summary

This chapter represents our attempt to expand the scope of data mining by considering a new application area mining textbooks for identifying sections and concepts that can benefit from reorganizing the material presented. Towards this goal, we introduced the notion of comprehension burden, presented a rigorous definition as well as an algorithm for computing it, and provided a diagnostic tool for authors to quantitatively assess the comprehension burden that a textbook imposes on the reader due to non-sequential presentation of concepts. We applied the tool to a corpus of high school textbooks, currently in active use in India. Using the tool, we were able to isolate high-burden sections and concepts.

Chapter 8

Satellite Image Analytics and Land Change Patterns

8.1 Introduction

Agricultural land availability is undergoing dramatic changes across the globe. This phenomenon is more rampant in the developing world where rapid economic growth and increasing population is resulting in unplanned development. The loss of arable area is estimated to be 1-21% in South America and around 18% in Africa [142]. The major causes identified for this decline are: (1) rapid urbanization of these countries including industrialization; (2) the migration of farmers to cities resulting in the sale of farmland for non-agricultural development, a trend that has escalated in the past few years due to rising real estate prices. Loss of arable land has a direct impact on food security. Most developing regions are also predominantly agrarian economies and changes in arable land can significantly impact food production and availability. Apart from urbanization and industrialization, changes in climatic patterns and other environmental factors are also resulting in degradation of farmlands and eventual disappearance. There are reports that Sahara desert is expanding southwards at an alarming rate[116]. Rising sea levels are increasing salinity of soil and decreasing productivity of land [125]. In countries like unpredictable monsoon is also harming production and land quality.

Loss of arable land is also a well documented phenomena in developed regions around the world,

especially in North America and Europe. Unlike developing regions, land usage is well-documented in most developed regions at fine-grained granularities. For example, from 1982 to 2007, more than 23 million acres of agricultural land was converted to developed land [111][112], with each state losing significant areas of farmland. Similarly, Germany lost five million acres of its Utilized Agricultural Area (SAU) between 1960 and 2010, a decline of almost 11 percent. France has about 50 percent of its land used for agricultural activities in 2010 and it is also declining [108]. Most developed countries have traditionally maintained detailed electronic records to monitor change in land patterns over 5-10 decades. In contrast, such fine-grained data is often found lacking in developing regions.

In this chapter, we propose the design of an automated satellite image analytics tool that can leverage publicly available satellite image data sources to provide a fine-grained longitudinal analysis of changes in land pattern in a given region. Our goal is to design a data analytics system that can understand the longitudinal relationship between changes in agricultural land availability patterns in a given small geographic area and its corresponding impact on food production. This chapter is specifically contextualized for the region of West Bengal, traditionally considered one of the most fertile areas in the world being in the delta of the Gangetic plains. We used a corpus of satellite images gathered from Google Earth, which maintains updated repository of satellite images along with archives of older images across the globe. Based on detailed food production data gathered in collaboration with the bureau of statistics of West Bengal, we analyze the correlations between changes in agricultural land patterns and corresponding changes in food production at fine-grained district granularities.

The key building block of our analytics tool is a satellite image analysis engine that can analyze potentially noisy satellite images and provide fine-grained classification of regions within each image into different categories such as: arable land, water body, developed land, forest etc. Given historical data about the same location, the image analysis engine can provide a detailed analysis of land pattern changes. Figure 8.1 is an example of a land in Angola in 2003 and a developed version of the same land in 2011 is shown in Figure 8.2. Our engine can detect such changes at different location granularities (small region, district, state level etc.). In the case of West Bengal, we obtained data over a 13 year time period from 2000-2012 and could track land evolution over this entire time period. We correlate this land change pattern with food production data over the same time period gathered by the Bureau of Statistics in the government. This tool can be helpful to policymakers to monitor the changes in the land pattern and take appropriate steps if any drastic changes are noticed.

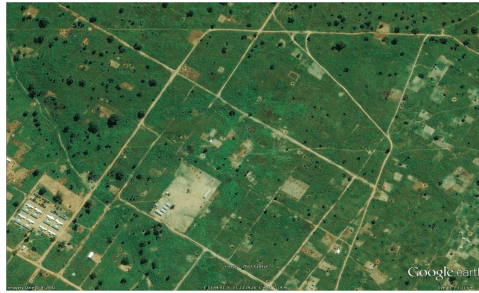


Figure 8.1: Image of a location in Angola taken in 2003



Figure 8.2: Image of a location in Angola taken in 2011

8.2 Land Pattern Analysis and Food Security

In this section, we provide the brief context and motivation for the problem addressed in this chapter. We begin by providing the context and the importance of the problem before providing specifics of the specific data analysis study presented in the chapter.

Context: Food security is emerging as one of the biggest problems that human populations may face in the upcoming century due to growing populations, urbanization and reduction in arable land. Reduction in arable land is a relatively hard task to reverse and food statistics around the world clearly indicate that agricultural land patterns are on the decline, and in recent times the rate has accelerated.

The problem of decrease in agricultural land is particularly an important question for developing countries which are predominantly agrarian economies. Given unprecedented population growth, coupled with only a relatively modest growth in agricultural yields, any minor changes in land can have catastrophic consequences on the food security for the population in such country. While western countries have relied on large scale imports, developing countries do not have the economic horsepower to rely on imports for

tackling food deficiencies.

A critical metric that needs to be monitored for food security in agrarian economies is agricultural land availability. While developed nations are known to have much more detailed records on land change patterns, the relative documentation of such data in developing regions is often lacking due to the lack of fine grained data. In many developing regions, data is not frequently updated or not complete. Policy decision-making is often done based on stale or incomplete data and important phenomena such as disappearance of arable land often go unnoticed.

Even if policy makers are aware of the decline in arable land, the second problem is the lack of knowledge of the underlying causes behind this disappearance. On many occasions, land acquisition for development is happening illegally [110]. Local authorities are unaware of this change and at what rate it is happening. Apart from this, there might be reasons for which land is losing productivity [133] due to bad agricultural practices or environmental factors. The exact transformation of agricultural land to what other types can provide valuable clues to the solution of the problem. For all these factors, land pattern statistics at regular intervals are essential.

8.2.1 Satellite Images

Advantage of using satellite images is that they can give a clear picture of the state of the land. Over the years, the quality of these images have greatly improved providing rich information about the surface area of different regions. Processing these images can reveal the present status of a region. Moreover, having access to historical versions of such images, short and long term changes in a region can be easily tracked, in terms of land patterns as well environmental factors, such as surface temperature, humidity etc. How historical satellite images can detect changes is evident from Figure 8.1 and 8.2. This images are taken from Angola in the year 2003 and 2011 respectively. In the 2003 it shows a green patch of land, probably used for agriculture, with marks of being converted into developed land. Within 8 years the entire green land has turned into an urban area. So, the main contribution the satellite images can have is to track the change.

In addition, satellite images can provide an account of of what type of changes happened in a region. A piece of agricultural land disappearing might not always be a very useful observation to tackle the problem. On the other hand, identifying the exact change can provide more insights. An existing agricultural

land can change to some other land type – acquired for urban or industrial development. Or, due to some reason the land's quality has deteriorated and hence, it is turning into barren land. Being able to track this change can provide enough information for a policy maker to deal with the situation. In first case, the solution is to deal with illegal and forcible acquisitions and in the second case come up with sustainable solution to protect existing farmlands. In conclusion, these steps can not only protect agricultural land, also ensure food security and stabilize food prices for long term sustainability.

8.2.2 West Bengal: A Case Study

In this chapter, we contextualize our study of the agricultural land availability problem for the state of West Bengal in India. West Bengal lies in the north Indian Gangetic Plains, as a result fertile alluvial soil is abundant in the state. Hence, agriculture is the predominant driving force of the economy of the state. Table 8.1 shows some facts of the state. The reason we chose West Bengal is due its relatively smaller area and its dominance in agriculture. Based on the data presented in Table 8.1 it is clear that that the economy of the state is heavily dependent upon agriculture. Agriculture contributes to about 24% of the state's domestic product and with such a high percentage of the labor force engaged in agriculture, it is apparent any decline in agriculture will have a drastic effect on the state, its economy as well food security. One of the goal of this chapter is to use the tool to estimate land pattern statistics of West Bengal and observe what changes have occurred in the last 10 years.

In this chapter, we try to analyze the agricultural land availability problem by building a satellite image analysis tool that can classify satellite images into various categories. In addition, it can monitor the changes over time and report *what is changing* and *how it is changing*. Subsequently, we use the data produced by the tool to compare changing land patterns with officially collected government data on food production at a district granularity.

8.3 Satellite Image Data

We created our dataset from the satellite image repository of Google Earth (GE). GE has freely available satellite images from across the world, including an archive of historical images. Our data was collected from the eastern Indian state of West Bengal. We collected satellite images of West Bengal from the GE repository for 7 different years, starting from 2000 till 2012 at an interval of 2 years. For each year we had

Table 8.1: West Bengal at a glance

Area	88,752 sq km
Population (2011)	91,347,736
Population density (2011)	1000/sq. km
Rural population (2011)	72%
Area under agriculture (2012)	5,666,000 h.a.
Labor force engaged in agriculture	67%

7,505 images covering the land area of the entire state. Each image captured were rectangular in shape with a fixed height and width. All the images were captured at an elevation of 11 KM, to make sure they are not too blurry (if captured from too close to the ground) or missing details (if captured from a very high elevation). A typical image is shown in Figure 8.1. We used a subset of this data to train and build the model.

One of the key challenges in processing the satellite image is the variability in the quality of images. Google Earth images do not have consistent quality. Particularly, for older images the quality is quite poor and blurry. Second, the precision of images presented by Google Earth can historically vary. Third, in many images the land area is not visible in some parts due to completely blanked out images or due to cloud cover. Finally, the background color across images is not consistent which makes it difficult to classify land into different categories. Examples of such noisy images are shown in Figure 8.3.

8.4 Satellite Image Analysis Tool

8.4.1 Design

The design of the tool has two parts - (1) using a sample of the collected images as a training set to build a classifier that can classify the region covered by any satellite image into various category - arable, urban, forest, water body and (2) use this trained tool to monitor land patterns across any region given the name of the region or latitude-longitude coordinates.

The image classification is done using K-means. K-means algorithm can analyze various features



(a) A blurry image



(b) Green field with different color



(c) Cloud cover

Figure 8.3: Noise in the dataset

of an object and assign them into clusters with similar items. We chose K-means as it does not require any annotated data. It can work directly on the extracted features of the objects. In our case, we used K-means to cluster the similar looking satellite images into the same clusters, i.e. all the images of arable land should go into one cluster, similarly for other types, urban, tree forest etc.

K-means clustering algorithm starts with a parameter K , which is a predetermined number denoting the number of clusters. In our case $K = 4$ and the clusters are,

1. Arable
2. Tree-covered
3. Water body
4. Developed

However, a single frame in a satellite image covers a large portion of land and they typically have more than one category of land. Fig 8.4 show a typical satellite image from our dataset and this image has agricultural land, tree-covered areas, as well as water bodies. Thus, it is not accurate to assign a single category to this image. Thus, every image needed to be broken into segments or superpixels, where each superpixel represents a homogeneous area in the image potentially belonging to only one of the categories mentioned above. There are many related works that involve segmenting images and extracting superpixels [123] [130] [114][141], we chose Markov Random Field based approach [137] to extract the superpixels. In this method the conditional distribution of a pixel is determined by the 8 neighboring pixels. A graphical model is constructed where every node is a pixel and edges are drawn between adjacent pixels. The edge potentials define how similar they are or whether they belong to the same superpixel or not. Based on some annotated superpixels, the model learned the edge potentials and from learned model, the superpixels were extracted from the larger image. Figure 8.4 shows the resulting superpixels after segmentation. Here, each superpixel represents a uniform region within the image. These superpixels are the atomic units of the images and they are classified into various land types to get the land pattern statistics of a region. The input is:

1. K : no. of clusters
2. D : data, in the form of n data points $\{x_1, x_2, \dots, x_n\}$, where each x_i is a f sized vector (f is the number of features)

The data D is used to learn the model parameters, which in this case are K points, $\{\mu_1, \mu_2, \mu_3, \mu_4\}$ representing the centroid of each cluster. In our case, $K = 4$ and each x_i is a satellite image represented in the feature space. The objective is to minimize the distance of every point x_i with the cluster centroids $\mu_{1:K}$. If c_i is the cluster x_i is assigned to and $\mu_{c(i)}$ is the cluster centroid of the cluster c_i then the following objective function is minimized to obtain optimum clustering.

$$\min J(\mu, x, c) = \frac{1}{n} \sum_{i=1}^n \|x_i - \mu_{c(i)}\|^2$$

We used Euclidean distance in an M (no. of features) dimensional space to compute the distance between x_i and $\mu_{c(i)}$. We used color histograms as a feature. Color histogram of an image gives the frequency distribution of different pixels belonging to a particular shade. This decision is based on the intuition, an arable land should have high frequency of pixel green shades, whereas, a water body will have more blue or gray colored pixels. We started with greyscale images with 256 distinct colors, hence we had 256 features. We applied the K-means algorithm to the superpixels and the average number of superpixels per image was 500 and for every year we had 7505 images. So, for each year we processed around 3,752,500 image segments. For a given year the proportion of land under any category is computed as,

$$\theta_k^{year} = \frac{\sum_i \alpha_{ik}^{year}}{\sum_{[1:K]} \alpha_{[1:K]}^{year}}$$

where, α_{ik}^{year} is the total number of superpixels in the image segments i which are assigned to the cluster k . Monitoring θ_k^{year} for different years can give an estimation of how land area under k has changed over the year.

However, given the characteristics of the dataset, images belonging to same category can have vastly different feature values. For example, the color of an arable land can have different shades of green as well as in many cases shades of brown. Using greyscale versions of the image cannot capture this variation. Hence, we experimented with different feature sets and methods to empirically find the optimum solution to this problem.

8.4.2 Performance

Our entire dataset contain 7505 images per year across 7 years. Extraction of superpixels resulted in an average 500 segments per image. For the development of tool, we randomly selected 5000 superpixels across all years to train the model and another 500 as a cross validation set, which were manually



Figure 8.4: Segmented Image

annotated. We experimented with different kinds of feature-set and computed the accuracy on this cross-validation set. The accuracy is computed as the percentage of the total number of cases where a superpixel was assigned to the correct cluster compared to the entire cross validation set.

Our first approach was based on using greyscale color histograms as the features. We ran K-means clustering algorithm on these features. This method gave very low accuracy of 38.28%. This is due to the fact that the satellite images had very similar colors for many different objects. Moreover, in greyscale the objects looked alike and color histograms could not discriminate against different types of objects (e.g. tree cover, open fields, buildings). Next we tried a similar approach but with RGB color information separately instead of converting them to greyscale. This increased the performance to an accuracy of 51.04%.

Analyzing the errors from this purely color histogram based K-means approach, we observed that each of the cluster had many false positives from different categories. That is the cluster for farmland had images from tree cover land, water bodies etc, Based on this observation we decided to implement a hierarchical clustering method. So, we started with a large value of K and in the following step we re-clustered the images from the existing clusters and merge the similar clusters. Starting with $K = 20$ and building a 2 level hierarchical clustering method increased the accuracy to 70.93%. This method could distinguish some of the noisy images, such as cloud cover ones. The pale color of the cloud covered images correctly identified.

Different experiments based on color histograms could only provide a maximum accuracy of 70.93%. A closer look at the color distribution of the images reveal that there are not much difference between the

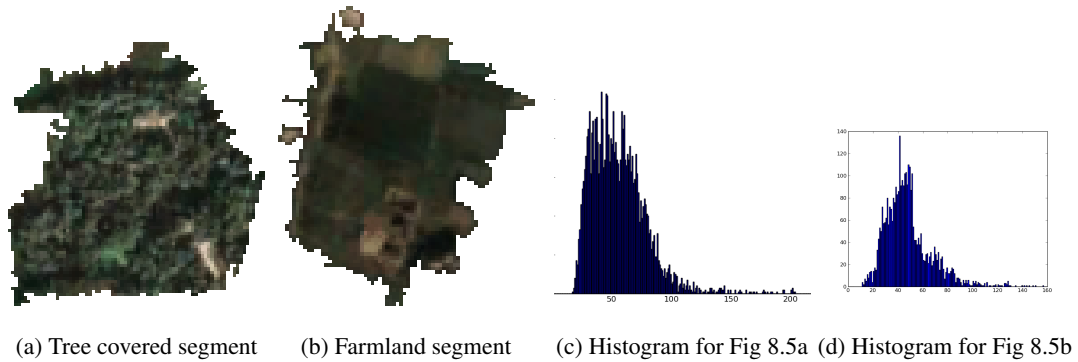


Figure 8.5: Superpixels and their corresponding color histograms

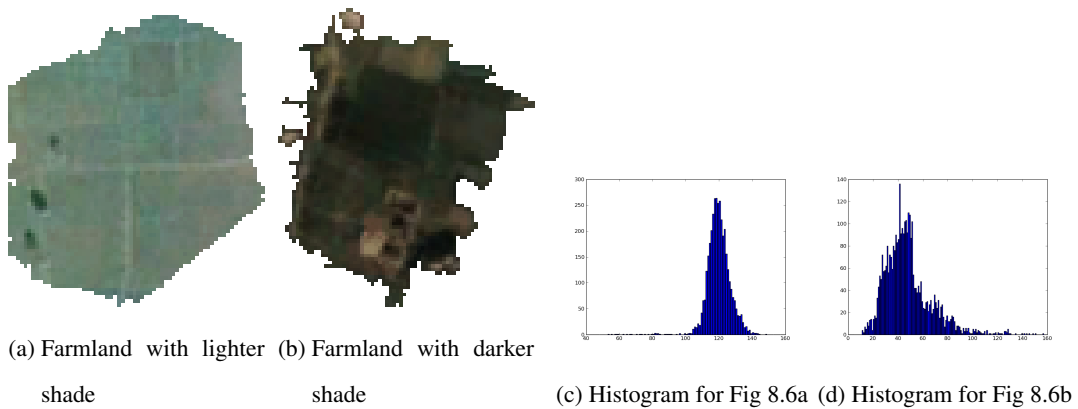


Figure 8.6: Different types of farmland their corresponding color histograms

histograms across different categories. Figure 8.5a is a super pixel of a tree covered land area and Figure 8.5b represents farmland and their corresponding color histograms are shown in Figure 8.5c and 8.5d respectively. Due to similar surface color in the satellite images the color distributions are very similar. On the other hand, different superpixels depicting the same type of land pattern has very different color distributions (Figure 8.6). Hence, only looking at the color histograms cannot differentiate different land patterns.

However, the texture of the images are very different for different land types. Irrespective of colors farmlands tend have a smoother texture compared to other categories. We added Grey Level Co-occurrence Matrix (GLCM) based features to include the texture of the images. Combining this feature along with color histograms increased the accuracy to 83.35%. Figure 8.7 shows the plot of correlation

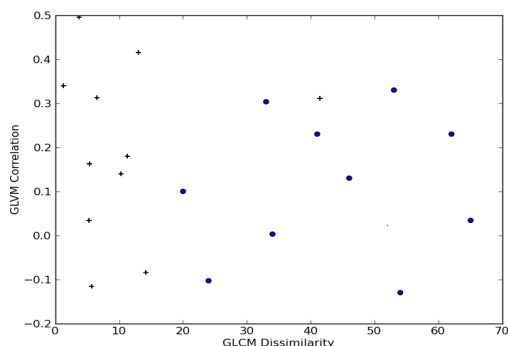


Figure 8.7: Texture of farmland (+) and treecover land (o)

vs dissimilarity of two types of land patterns. Clearly this feature has more discriminatory property compared to using just color histograms. Table 8.2 summarizes the performance of the satellite image analysis tool based on different features.

Finally, we performed some post classification modification called *local area correction*. This is based on the assumption that any type of land is continuous. In all the previous methods the superpixels were treated separately assuming they are independent to each other. However, many of these superpixels are adjacent to each other and they tend to belong to the same class. For example, agricultural land is continuous and any break in between is unlikely. That is, if there is a superpixel classified as developed or tree covered where all its neighboring superpixels belong to agricultural land then there might have been some error. Subsequently, we incorporated this assumption that all the land types are continuous and a superpixel will be more likely to belong to the same class as its neighbors in north, east, south and west directions. As the superpixels are irregular in shape, they do not form a grid-like structure so that 8 neighbors can be considered. Instead we used 4 neighbors in the north, east, south, west directions. For any superpixel, if its distance from the assigned class centroid is further than the class mean distance, then its 4 neighbors are checked. If most of the neighbors belong to the same class then current superpixel is assigned to neighbors' class. This method increased accuracy by more than 3%. This is due to the lack of uniformity in the colors of the satellite images. Often a stretch of continuous land (belonging to same category) has been depicted with different colors and textures. As a result, simple clustering method classified them into different clusters. Example of such a distortion can be seen in Figure 8.4. Near the top left corner of the image, the stretch of arable land has been shown using green as well as yellowish

Table 8.2: Accuracy of the tool for different features used

Feature-set	Accuracy
Greyscale color histogram	38.28%
RGB color histogram	51.04%
Hierarchical clustering	70.93%
Texture (GLCM)	78.41%
Texture + Greyscale	79.09%
Texture + RGB	83.35%
Texture + RGB + local area correction	86.78%

color. Hence, they were assigned to different clusters. Local area correction could handle such exceptions and ultimately assign the correct labels.

8.5 Land Pattern Analysis of West Bengal

After the development of the tool, we applied it to the satellite images collected from the Indian state of West Bengal. Our goal was to analyze the variation of agricultural land in that state across several years. We collected satellite images covering entire area of the state for 7 years between 2000 and 2012. The tool was applied on this data to estimate the percentage of arable land in all these years and observe the change in percentage across years. For a year, the total number segments under arable land category was aggregated to estimate the percentage of arable land for that year.

The estimated percentage of arable land as computed by our tool is shown in Figure 8.8. We see that there has been a decline in agricultural land in the state between these years. Although, the percentage slightly rose between 2000 and 2002, according to the estimate there has been a drop of 2.0% between 2000 and 2012.

We wanted to compare our findings with the food production statistics published by the *Bureau of Applied Economics and Statistics* affiliated to the *Government of West Bengal* [109]. The bureau publishes various statistics about the state in their annual Economic Review journal. The different food production

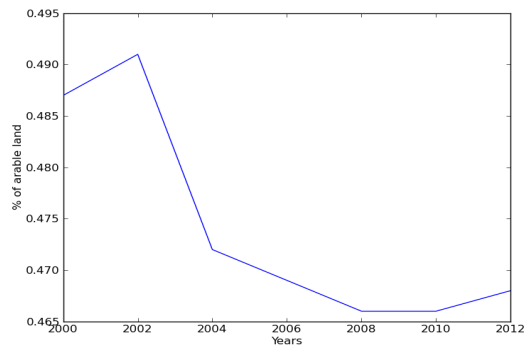


Figure 8.8: Percentage of arable land per year

related metrics we collected are,

- Land area under rice
- Land area under wheat
- Net rice production
- Net wheat production
- Net cropped area
- Agricultural area index
- Agricultural production index
- Cropping intensity index

Our goal was to validate our findings as well understand the implications of reduction in arable land and food production from these data. To validate the estimate given by the tool we compared the result with the cropped area published in the official report. The comparison of the official and our computed values are shown in Figure 8.9. Although our findings do not exactly match with the official figures, we see that the trends in both the plots have similarity.

To understand the implication of decline in arable land, we compared our findings with the food production data published by the government. We computed the correlation coefficients to see how arable

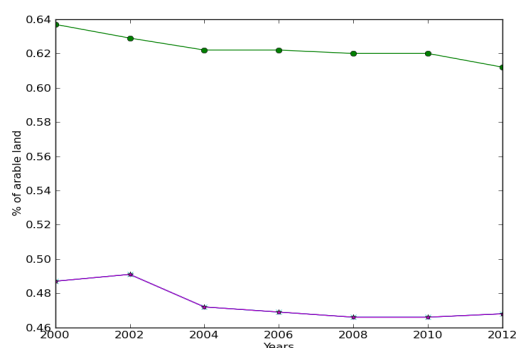


Figure 8.9: Comparison between official percentage cropped area (upper) and estimation from the satellite images analysis tool (lower)

Table 8.3: Comparison between arable land and food production

Index	Correlation coefficient	p-value
Net rice production	0.63	0.12
Net wheat production	0.65	0.11
Agricultural area index	0.52	0.23
Agricultural production index	0.77	0.03*
Cropping intensity index	-0.63	0.12

land area can affect food production. The results are summarized in Table 8.3. The results clearly indicates that in a region reduction in arable land has a positive correlation with food production. However, in this particular case we observe that *cropping intensity* has a negative correlation with land area. This may be due to the fact that cropping intensity is a measure of production per area. Thus, when area reduces and decline in production is comparatively lower, the value of cropping intensity increases.

Apart from looking at the entire state as a whole, we have also analyzed different districts of the state separately. In India, districts are the second level administrative boundaries in each state. The state of West Bengal is divided into 19 districts, including the urban district of Kolkata, which is also the state's capital. In this chapter we have focused on 12 districts in the southern part of the state, excluding the

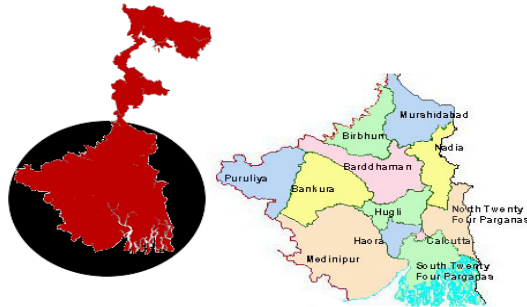


Figure 8.10: Map of West Bengal and district boundaries of southern part of the state

district of Kolkata. Figure 8.10 shows the map of West Bengal and district boundaries of the region considered in this study. The southern part of the state is part of the large Gangetic plain and conditions are well suited for agriculture.

Our satellite images were labeled with the latitude-longitude coordinates of the location from where they were extracted. Using a GIS database we identified the district from where the image was extracted. Then for each such cluster we computed the year-wise share of each land type and produced a district level data of land pattern and its changes. The percentage change in agricultural land for these districts is summarized in Table 8.4. The figure in the table shows the change between year 2012 and 2000, as produced by our tool.

We see that in some districts the disappearance of agricultural land is higher than others. Interestingly, the districts with higher rate of disappearance, such as, Howrah, Hooghly are close to the urban district of Kolkata. Due to increasing population of the city, more land from the neighboring districts are being taken up for urban development. Similar arguments can be applied to relatively high rate of disappearance of agricultural land in the Burdwan district. Burdwan is a very populous district with industrial towns such as Asansol and Durgapur. On the other hand, in some districts, such as, Purulia and Bankura where we observe a increase in agricultural land. These districts are in the western part of the state and where industrialization initiatives are limited. Also, the population density is relatively lower.

Similar to the state-level data, we tried to find the relationship between arable land and food production in the districts as well. We did this comparison between the land share and rice production in these districts. The choice of rice was due to the fact that it is the most produced crop and the most consumed

Table 8.4: District-wise change in agricultural land between 2000 and 2012

District	Change in agricultural land (%)
Bankura	6.1
Burdwan	-1.44
Birbhum	-2.36
Midnapore ¹	0.231
Howrah	-1.76
Hooghly	-5.3
24 Parganas (North)	-0.36
24 Parganas (South)	-2.11
Nadia	2.09
Murshidabad	-2.20
Purulia	1.48

food in this region. The result of this comparison is shown in Figure 8.11. The figure shows a positive relationship between agricultural land (computed by the tool) and rice production in these districts. Both the values have been normalized to a score between 0 and 1. The scatter plot shows an approximate linear relationship between land area under agriculture and rice production. Hence, even in the districts we see that land area is changing over time and this change is showing positive relationship with food production.

8.6 Summary

In this chapter we have described a satellite image analysis tool that can process Google Earth satellite images to classify land area in a region into 4 classes. In addition, the tool accesses the older images of the region and compute the changes in the land pattern. The image classification engine of the tool demonstrated an accuracy of around 87%. The image processing part of the tool was developed mostly using existing methods and image features. Some techniques were implemented in the image classification part to deal with the data-specific noises in the images. However, further improvement in the accuracy can

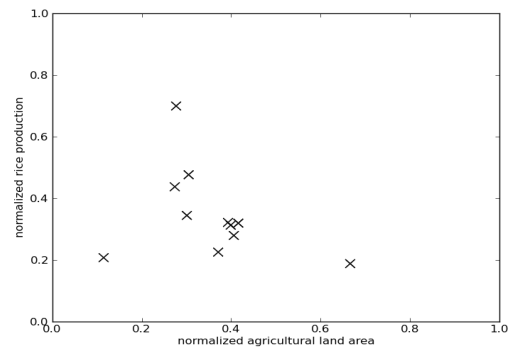


Figure 8.11: Relationship between agricultural land area and rice production for different districts (2012)

greatly increase the tool’s usability and reliability. So, one direction of the future work is to improve the system performance specifically for satellite images used and goals aimed in this work. Another approach in the future versions of this work would be to incorporate datasets from other sources, with less noise, better coverage and more frequent updates.

Another contribution of this work was to provide land related statistics, particularly information about how land pattern have changed between 2000 and 2012 in the Indian state of West Bengal. In this chapter, we have presented the estimation of how agricultural land has changed over this time period. We have also showed some preliminary analysis of how changing agricultural land is affecting food production. Based on these early findings we are in the process of collaboration with Government of West Bengal by deploying our tool at their units, offering our findings and in exchange including their expertise and additional data to improve the system’s functionalities. We are also planning to deploy our tool in Ghana at the Ministry of Food and Agriculture (MoFA) field offices. MoFA field workers can greatly benefit from this tool by obtaining data about the area under their supervision. There is a shortage of field workers recruited by MoFA and thus, each worker has a huge area to supervise. Monitoring the land, which normally done manually by visiting the area in person, is slow and cumbersome. This tool can provide a positive impact on their operations by reducing time and increasing accuracy in their data. Another future feature can further benefit them if the tool can access updated satellite images of the region at a more frequent intervals.

Apart from producing agriculture related information, the increased impact of the tool can be realized by applying it to other applications. This tool can be used – at its present form – to monitor water bodies.

This can help in two different applications - firstly, detecting unlawful filling of ponds and lakes for development, and secondly, monitor the impact of rising water levels in coastal areas or disappearance of land due to river erosion. Generally, this tool can be a useful apparatus for policy makers and law-enforcement agencies to detect illegal constructions on protected or vulnerable land, provided that it has access to supporting data.

Chapter 9

Conclusions

In this thesis, we explored how news articles can be used to extract events and build knowledge graphs to represent the inter-dependencies of events for different types of analytics. In this thesis we presented two types of event models and five different ways to build knowledge graphs at different granularity levels. These knowledge graphs can be used in different manual or automatic analysis of news data. One of the main application of the event models and knowledge graphs are building predictive models for external variables and indicators. We show how predictive models can be built using news events and its relationships (through knowledge graphs) for practical applications.

This thesis makes the following key research contributions:

1. Provide a method to identify critical trends in a location with respect to a topic using web data
2. Analysis of news articles to extract events
3. Building knowledge graphs depicting,
 - (a) Latent relationships between different events and connecting them to external phenomenons for prediction
 - (b) human readable relations between news events and external socio-economic indicators (such as, food prices)
 - (c) graphical structure of words appearing in articles to understand causal links between them
 - (d) extracting structures from unstructured domain-specific texts in the form (key, value) pairs

4. Exploiting these relationships to build predictive models for socio-economic indicators
5. Understand and characterize changing patterns in land using satellite images

One of main emphasis of this thesis was to use unstructured web data to extract events, knowledge graphs and predictive models for socio-economic indicators. We would like to strengthen this contribution by exploring new methods for this part of the thesis. We would like to look at different types of news data and methods to have more accurate and stronger claims. There many aspects of the data which were ignored in this thesis. For example, *bias* of a news source. Many media houses have affiliations and interests that may lead in publishing specific types news items or the presentation mode is different. Such biases might send out wrong signals leading to inaccurate analysis. In future, we would like to address this issues by introducing a *bias factor* using which a news source can be normalized. For the entire thesis, we have focused solely on English language media. However, on many occasions a vernacular source may have better coverage, particularly at the regional level. One of our future initiative will be to use language independent techniques to include multilingual news sources. Finally, we plan to use a different types of news source – citizen journalism platforms. Our conviction is that such platforms will help in gathering much more complete information, but this also involves dealing with the limitations and drawbacks of such news sources.

Bibliography

- [1] Economic review and statistical appendix, department of statistics and programme implementation, government of west bengal, 2000-2012.
- [2] Land lost, singur farmer said no to compensation, commits suicide. <http://archive.indianexpress.com/news/land-lost-singur-farmer-said-no-to-compensation-commits-suicide/31930/>, 2007. [Online; accessed Feb-2015].
- [3] E. Acar, S. A. Camtepe, M. S. Krishnamoorthy, and B. Yener. Modeling and multiway analysis of chatroom tensors. In *Intelligence and Security Informatics*, pages 256–268. Springer, 2005.
- [4] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Susstrunk. Slic superpixels compared to state-of-the-art superpixel methods. *IEEE Trans. Pattern Anal. Mach. Intell.*, 34(11):2274–2282, Nov. 2012.
- [5] A. Ahmed, Q. Ho, J. Eisenstein, E. Xing, A. J. Smola, and C. H. Teo. Unified analysis of streaming news. In *Proceedings of the 20th International Conference on World Wide Web, WWW '11*, pages 267–276, New York, NY, USA, 2011. ACM.
- [6] J. Allan, R. Gupta, and V. Khandelwal. Temporal summaries of new topics. In *Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR '01*, pages 10–18, New York, NY, USA, 2001. ACM.
- [7] G. Amodeo, R. Blanco, and U. Brefeld. Hybrid models for future event prediction. *CIKM '11*, pages 1981–1984, 2011.

- [8] A. Balasubramanian, N. Balasubramanian, S. J. Huston, D. Metzler, and D. J. Wetherall. Findall: A local search engine for mobile phones. In *Proceedings of the 8th international conference on Emerging networking experiments and technologies*, pages 277–288. ACM, 2012.
- [9] D. M. Blei and J. Lafferty. Correlated topic models. In *NIPS*, 2006.
- [10] D. M. Blei and J. D. Lafferty. Dynamic topic models. *ICML '06*, pages 113–120. ACM, 2006.
- [11] D. M. Blei and J. D. McAuliffe. Supervised topic models. In *NIPS*, 2007.
- [12] D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent dirichlet allocation. *J. Mach. Learn. Res.*, 3:993–1022, Mar. 2003.
- [13] D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent dirichlet allocation. *J. of machine Learning research*, 3:993–1022, 2003.
- [14] J. Bollen, H. Mao, and X. Zeng. Twitter mood predicts the stock market. *Journal of Computational Science*, 2(1):1–8, 2011.
- [15] J. Boudoukh, R. Feldman, S. Kogan, and M. Richardson. Which news moves stock prices? A textual analysis. 2013.
- [16] P. Capet, T. Delavallade, T. Nakamura, A. Sandor, C. Tarsitano, and S. Voyatzi. A risk assessment system with automatic extraction of event types. volume 288 of *IFIP Advances in Information and Communication Technology*, pages 220–229. Springer, 2008.
- [17] S. Chava, R. Ennaji, J. Chen, and L. Subramanian. Cost-aware mobile web browsing. *IEEE Pervasive Computing*, (3):34–42, 2012.
- [18] J. Chen and L. Subramanian. Interactive web caching for slow or intermittent networks. In *Proceedings of the 4th Annual Symposium on Computing for Development*, page 5. ACM, 2013.
- [19] D. Cheng, M. T. Bahadori, and Y. Liu. Fblg: A simple and effective approach for temporal dependence discovery from time series data. *KDD '14*, pages 382–391, 2014.
- [20] H.-w. Chun, Y.-s. Hwang, and H.-C. Rim. Unsupervised event extraction from biomedical literature using co-occurrence information and basic patterns. *IJCNLP'04*, 2005.

- [21] P. Cimiano and S. Staab. Learning by googling. *SIGKDD Explor. Newsl.*, 6(2):24–33, Dec. 2004.
- [22] K. B. Cohen, K. Verspoor, H. L. Johnson, C. Roeder, P. V. Ogren, W. A. Baumgartner, Jr., E. White, H. Tipney, and L. Hunter. High-precision biological event extraction with a concept recognizer. *BioNLP '09*, pages 50–58, 2009.
- [23] H. Cunningham, Y. Wilks, and R. J. Gaizauskas. *Gate - a general architecture for text engineering*, 1996.
- [24] G. Druck, G. Mann, and A. McCallum. Learning from labeled features using generalized expectation criteria. *SIGIR '08*, pages 595–602, 2008.
- [25] J. E. Engelberg and C. A. Parsons. The causal impact of media in financial markets. *The Journal of Finance*, 66(1):67–97, 2011.
- [26] C. Faloutsos, K. S. McCurley, and A. Tomkins. Fast discovery of connection subgraphs. In *Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '04, pages 118–127, New York, NY, USA, 2004. ACM.
- [27] L. Fang, A. D. Sarma, C. Yu, and P. Bohannon. Rex: Explaining relationships between entity pairs. *Proc. VLDB Endow.*, 5(3):241–252, Nov. 2011.
- [28] Y. Fang, L. Si, N. Somasundaram, and Z. Yu. Mining contrastive opinions on political texts using cross-perspective topic model. *WSDM '12*, pages 63–72.
- [29] D. Ferrucci and A. Lally. Uima: An architectural approach to unstructured information processing in the corporate research environment. *Nat. Lang. Eng.*, 10(3-4):327–348, Sept. 2004.
- [30] B. Fulkerson, A. Vedaldi, and S. Soatto. Class segmentation and object localization with superpixel neighborhoods. In *Computer Vision, 2009 IEEE 12th International Conference on*, pages 670–677, Sept 2009.
- [31] G. Gidofalvi. Using news articles to predict stock price movements, 2001.
- [32] J. Gillenwater, A. Kulesza, and B. Taskar. Discovering diverse and salient threads in document collections. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language*

- Processing and Computational Natural Language Learning*, EMNLP-CoNLL '12, pages 710–720, Stroudsburg, PA, USA, 2012. Association for Computational Linguistics.
- [33] J. Ginsberg, M. Mohebbi, R. Patel, L. Brammer, M. Smolinski, and L. Brilliant. Detecting influenza epidemics using search engine query data. *Nature*, 457(7232):1012–1014, 2008.
- [34] T. Grenager, D. Klein, and C. D. Manning. Unsupervised learning of field segmentation models for information extraction. *ACL '05*, pages 371–378, 2005.
- [35] M. Hagenau, M. Liebmann, and D. Neumann. Automated news reading: Stock price prediction based on financial news using context-capturing features. *Decision Support Systems*, 55(3):685 – 697, 2013.
- [36] A. Haghighi and D. Klein. Prototype-driven learning for sequence models. In *Proceedings of the Main Conference on Human Language Technology Conference of the North American Chapter of the Association of Computational Linguistics*, HLT-NAACL '06, pages 320–327, Stroudsburg, PA, USA, 2006. Association for Computational Linguistics.
- [37] R. Heakal. Explaining the world through macroeconomic analysis. 2012.
- [38] G. Hinton. Training products of experts by minimizing contrastive divergence. *Neural Computation*, 14:2002, 2000.
- [39] T. Hofmann. Probabilistic latent semantic indexing. *SIGIR '99*, pages 50–57, 1999.
- [40] F. Hogenboom, M. de Winter, F. Frasincar, and U. Kaymak. A news event-driven approach for the historical value at risk method. *Expert Systems with Applications*, 42(10):4667 – 4675, 2015.
- [41] F. Hogenboom, F. Frasincar, U. Kaymak, and de Jong Franciska. An overview of event extraction from text. *Derive 2011*, pages 48–57, 2011.
- [42] A. Hulth, G. Rydevik, and A. Linde. Web queries as a source for syndromic surveillance. *PloS one*, 4(2):e4378, 2009.
- [43] S.-H. Hung, C.-H. Lin, and J.-S. Hong. Web mining for event-based commonsense knowledge using lexico-syntactic pattern matching and semantic role labeling. *Expert Syst. Appl.*, 37(1):341–347, Jan. 2010.

- [44] S. Isaacman and M. Martonosi. Potential for collaborative caching and prefetching in largely-disconnected villages. In *Proceedings of the 2008 ACM workshop on Wireless networks and systems for developing regions*, pages 23–30. ACM, 2008.
- [45] S. Isaacman and M. Martonosi. The c-link system for collaborative web usage: A real-world deployment in rural nicaragua. In *Workshop on Networked Systems for Developing Regions*, 2009.
- [46] S. Isaacman and M. Martonosi. Low-infrastructure methods to improve internet access for mobile users in emerging regions. In *Proceedings of the 20th international conference companion on World wide web*, pages 473–482. ACM, 2011.
- [47] N. Kawamae. Trend analysis model: trend consists of temporal words, topics, and timestamps. In *Proceedings of the fourth ACM international conference on Web search and data mining*, pages 317–326. ACM, 2011.
- [48] D. Kim and A. Oh. Topic chains for understanding a news corpus. In A. Gelbukh, editor, *Computational Linguistics and Intelligent Text Processing*, volume 6609 of *Lecture Notes in Computer Science*, pages 163–176. Springer Berlin Heidelberg, 2011.
- [49] D. Kim and A. Oh. Topic chains for understanding a news corpus. *Computational Linguistics and Intelligent Text Processing*, pages 163–176, 2011.
- [50] J. D. Lafferty, A. McCallum, and F. C. N. Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. *ICML '01*, pages 282–289, 2001.
- [51] K. Leetaru. Culturomics 2.0: Forecasting large-scale human behavior using global news media tone in time and space. *First Monday*, 16(9), 2011.
- [52] S. Liao and R. Grishman. Using document level cross-event inference to improve event extraction. *ACL '10*, 2010.
- [53] M. Liu, Y. Liu, L. Xiang, X. Chen, and Q. Yang. Extracting key entities and significant events from online daily news. *IDEAL '08*, pages 201–209, 2008.

- [54] M.-Y. Liu, O. Tuzel, S. Ramalingam, and R. Chellappa. Entropy rate superpixel segmentation. In *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, pages 2097–2104, June 2011.
- [55] C. Luo, J.-G. Lou, Q. Lin, Q. Fu, R. Ding, D. Zhang, and Z. Wang. Correlating events with time series for incident diagnosis. *KDD '14*, pages 1583–1592, 2014.
- [56] Q. Mei and C. Zhai. Discovering evolutionary theme patterns from text: An exploration of temporal text mining. In *Proceedings of the Eleventh ACM SIGKDD International Conference on Knowledge Discovery in Data Mining*, *KDD '05*, pages 198–207, New York, NY, USA, 2005. ACM.
- [57] M. Michelson and C. A. Knoblock. Creating relational data from unstructured and ungrammatical data sources. *J. Artif. Int. Res.*, 31(1):543–590, Mar. 2008.
- [58] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean. Distributed representations of words and phrases and their compositionality. *NIPS '13*, pages 3111–3119.
- [59] F. Ming, F. Wong, Z. Liu, and M. Chiang. Stock market prediction from wsj: Text mining via sparse matrix factorization. *ICDM '14*, 2014.
- [60] MySQL. Stopwords. <http://dev.mysql.com/doc/refman/5.5/en/fulltext-stopwords.html>, 2014. [Online; accessed Sep-2014].
- [61] N. Nakashole, G. Weikum, and F. Suchanek. Discovering and exploring relations on the web. *Proc. VLDB Endow.*, 5(12):1982–1985, Aug. 2012.
- [62] S. Nallareddy and M. Ogneva. Predicting restatements in macroeconomic indicators using accounting info, 2014.
- [63] Y. Nishihara, K. Sato, and W. Sunayama. Event extraction and visualization for obtaining personal experiences from blogs. In *Symposium on Human Interface and the Management of Information.*, pages 315–324, 2009.
- [64] F. Niu, C. Zhang, C. R., and J. Shavlik. Deepdive: Web-scale knowledge-base construction using statistical learning and inference. *VLDS '12*, 2012.

- [65] B. O'Connor and D. Bamman. Computational Text Analysis for Social Science: Model Assumptions and Complexity. *public health*, 2011.
- [66] M. Okamoto and M. Kikuchi. Discovering volatile events in your neighborhood: Local-area topic extraction from blog entries. AIRS '09, pages 181–192, 2009.
- [67] A. M. Okun. Economics for policymaking. 2004.
- [68] T. Philpott. New research: synthetic nitrogen destroys soil carbon, undermines soil health. <http://grist.org/article/2010-02-23-new-research-synthetic-nitrogen-destroys-soil-carbon-undermines/>, 2010. [Online; accessed Feb-2015].
- [69] D. Pinto, A. McCallum, X. Wei, and W. B. Croft. Table extraction using conditional random fields. SIGIR '03, pages 235–242, 2003.
- [70] D. R. Radev, E. Hovy, and K. McKeown. Introduction to the special issue on summarization. *Comput. Linguist.*, 28(4):399–408, Dec. 2002.
- [71] K. Radinsky, S. Davidovich, and S. Markovitch. Learning causality for news events prediction. WWW '12, 2012.
- [72] K. Radinsky and E. Horvitz. Mining the web to predict future events. WSDM '13, pages 255–264, 2013.
- [73] K. Radinsky and E. Horvitz. Mining the web to predict future events. WSDM '13, pages 255–264. ACM, 2013.
- [74] K. Radinsky and E. Horvitz. Mining the web to predict future events. In *Proceedings of the sixth ACM international conference on Web search and data mining*, pages 255–264. ACM, 2013.
- [75] D. Richards. Political complexity: Non linear models of politics. *J. Artificial Societies and Social Simulation*, 5(1), 2002.
- [76] C. Rudin, B. Letham, and D. Madigan. Learning theory analysis for association rules and sequential event prediction. *J. of Mach. Learn. Rsrch*, 14:3441–3492, 2013.

- [77] S. Sarawagi and W. W. Cohen. Semi-markov conditional random fields for information extraction. In *In Advances in Neural Information Processing Systems 17*, pages 1185–1192, 2004.
- [78] A. Schick, M. Bauml, and R. Stiefelhagen. Improving foreground segmentations with probabilistic superpixel markov random fields. In *Computer Vision and Pattern Recognition Workshops (CVPRW), 2012 IEEE Computer Society Conference on*, pages 27–31, June 2012.
- [79] R. Schumaker and H. Chen. Textual analysis of stock market prediction using financial news articles. 2006.
- [80] R. P. Schumaker and H. Chen. Textual analysis of stock market prediction using breaking financial news: The azfin text system. *ACM Transactions on Information Systems (TOIS)*, 27(2):12, 2009.
- [81] F. Sha and F. Pereira. Shallow parsing with conditional random fields. *NAACL '03*, pages 134–141, 2003.
- [82] D. Shahaf and C. Guestrin. Connecting the dots between news articles. In *Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '10*, pages 623–632, New York, NY, USA, 2010. ACM.
- [83] D. Shahaf and C. Guestrin. Connecting the dots between news articles. In *Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 623–632. ACM, 2010.
- [84] D. Shahaf and C. Guestrin. Connecting two (or less) dots: Discovering structure in news articles. *ACM Trans. Knowl. Discov. Data*, 5(4):24:1–24:31, Feb. 2012.
- [85] D. Shahaf, C. Guestrin, and E. Horvitz. Trains of thought: Generating information maps. In *Proceedings of the 21st International Conference on World Wide Web, WWW '12*, pages 899–908, New York, NY, USA, 2012. ACM.
- [86] D. Shahaf, J. Yang, C. Suen, J. Jacobs, H. Wang, and J. Leskovec. Information cartography: Creating zoomable, large-scale maps of information. In *Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '13*, pages 1097–1105, New York, NY, USA, 2013. ACM.

- [87] M. Shashanka, B. Raj, and P. Smaragdis. Probabilistic latent variable models as nonnegative factorizations. *Computational intelligence and neuroscience*, 2008, 2008.
- [88] J. Si, A. Mukherjee, B. Liu, Q. Li, H. Li, and X. Deng. Exploiting topic based twitter sentiment for stock prediction. In *ACL (2)'13*, pages 24–29, 2013.
- [89] J. Si, A. Mukherjee, B. Liu, S. J. Pan, Q. Li, and H. Li. Exploiting social relations and sentiment for stock prediction. *EMNLP '14*, pages 1139–1145, 2014.
- [90] C. Suen, S. Huang, C. Eksombatchai, R. Sasic, and J. Leskovec. Nifty: A system for large scale information flow tracking and clustering. In *Proceedings of the 22Nd International Conference on World Wide Web, WWW '13*, pages 1237–1248, Republic and Canton of Geneva, Switzerland, 2013. International World Wide Web Conferences Steering Committee.
- [91] R. Swan and D. Jensen. Timemines: Constructing timelines with statistical models of word usage.
- [92] H. Tanev, J. Piskorski, and M. Atkinson. Real-time news event extraction for global crisis monitoring. *NLDB '08*, pages 207–218, 2008.
- [93] P. Tetlock. Giving content to investor sentiment: The role of media in the stock market. 62:1139–1168, 2007.
- [94] P. C. Tetlock. Giving content to investor sentiment: The role of media in the stock market. *The J. of Finance*, 62(3):1139–1168, 2007.
- [95] C. K. Vaca, A. Mantrach, A. Jaimes, and M. Saerens. A time-based collective factorization for topic discovery and monitoring in news. In *Proceedings of the 23rd international conference on World wide web*, pages 527–538. International World Wide Web Conferences Steering Committee, 2014.
- [96] S. Wang, H. Lu, F. Yang, and M.-H. Yang. Superpixel tracking. In *Computer Vision (ICCV), 2011 IEEE International Conference on*, pages 1323–1330, Nov 2011.
- [97] X. Wang and A. McCallum. Topics over time: a non-markov continuous-time model of topical trends. In *KDD*, 2006.

- [98] Y. Wang, E. Agichtein, and M. Benzi. Tm-lda: efficient online modeling of latent topic transitions in social media. In *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 123–131. ACM, 2012.
- [99] I. Weber, V. R. K. Garimella, and E. Borra. Mining web query logs to analyze political issues. *WebSci '12*, 2012.
- [100] F. Xu, H. Uszkoreit, and H. Li. Automatic event and relation detection with seeds of varying complexity. In *AAAI Workshop Event Extraction and Synthesis*, pages 12–17, 2006.
- [101] R. Yan, L. Kong, C. Huang, X. Wan, X. Li, and Y. Zhang. Timeline generation through evolutionary trans-temporal summarization. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, EMNLP '11, pages 433–443, Stroudsburg, PA, USA, 2011. Association for Computational Linguistics.
- [102] R. Yan, X. Wan, J. Otterbacher, L. Kong, X. Li, and Y. Zhang. Evolutionary timeline summarization: A balanced optimization framework via iterative substitution. In *Proceedings of the 34th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '11, pages 745–754, New York, NY, USA, 2011. ACM.
- [103] J. Yang, J. McAuley, J. Leskovec, P. LePendou, and N. Shah. Finding progression stages in time-evolving event sequences. In *Proceedings of the 23rd International Conference on World Wide Web*, WWW '14, pages 783–794, New York, NY, USA, 2014. ACM.
- [104] W. Zhang and S. Skiena. Trading strategies to exploit blog and news sentiment. *ICWSM*, 2010.
- [105] M. Zheleva, P. Schmitt, M. Vigil, and E. Belding. The increased bandwidth fallacy: performance and usage in rural zambia. In *Proceedings of the 4th Annual Symposium on Computing for Development*, page 2. ACM, 2013.
- [106] D. Zhou, D. Zhong, and Y. He. Event trigger identification for biomedical events extraction using domain knowledge. *Bioinformatics*, 20:1587–1594, Jun 2014.
- [107] J. Zhu, A. Ahmed, and E. P. Xing. Medlda: Maximum margin supervised topic models for regression and classification. *ICML '09*, pages 1257–1264, 2009.

- [108] Every single day, over 550 acres of agricultural land, the equivalent of four average-size farms, are disappearing in France. http://www.momagri.org/UK/agriculture-s-key-figures/Every-single-day-over-550-acres-of-agricultural-land-the-equivalent-of-four-average-size-farms-discretionary{-}{ }are-disappearing-in-France_1057.html. [Online; accessed Feb-2015].
- [109] Economic review and statistical appendix, department of statistics and programme implementation, government of West Bengal, 2000-2012.
- [110] Land lost, Singur farmer said no to compensation, commits suicide. <http://archive.indianexpress.com/news/land-lost-singur-farmer-said-no-to-compensation-commits-suicide/31930/>, 2007. [Online; accessed Feb-2015].
- [111] Yale Environment 360: Most productive U.S. farmland disappearing at fastest rate. <http://e360.yale.edu/digest/most-productive-us-farmland-disappearing-discretionary{-}{ }at-fastest-rate-report-says/2627/>, 2010. [Online; accessed Feb-2015].
- [112] Incredible shrinking farmland. <http://grist.org/sustainable-farming/2011-11-07-incredible-shrinking-farmland/>, 2011. [Online; accessed Feb-2015].
- [113] B. Abelson, K. R. Varshney, and J. Sun. Targeting direct cash transfers to the extremely poor. In *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '14, pages 1563–1572, New York, NY, USA, 2014. ACM.
- [114] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Susstrunk. Slic superpixels compared to state-of-the-art superpixel methods. *IEEE Trans. Pattern Anal. Mach. Intell.*, 34(11):2274–2282, Nov. 2012.
- [115] F. Adimi, R. P. Soebiyanto, N. Safi, and R. Kiang. Towards malaria risk prediction in Afghanistan using remote sensing. *Malaria Journal*, 9(125), May 2010.

- [116] BBC. Desertification - expansion of the sahara desert. <http://www.bbc.co.uk/programmes/p00xbj4w>. [Online; accessed Feb-2015].
- [117] A. Bretto, A. Ducournau, and S. Rital. A hypergraph reduction algorithm for joint segmentation and classification of satellite image content. In I. Bloch and J. Cesar, RobertoM., editors, *Progress in Pattern Recognition, Image Analysis, Computer Vision, and Applications*, volume 6419 of *Lecture Notes in Computer Science*, pages 38–45. Springer Berlin Heidelberg, 2010.
- [118] P. Ceccato, S. Connor, I. Jeanne, and M. Thomson. Application of geographical information systems and remote sensing technologies for assessing and monitoring malaria risk. *Parassitologia*, 47(1):81–96, Mar 2005.
- [119] R. Dahmani, A. Fora, and A. Sbihi. Monitoring the proliferation of slums through gis and satellite image processing in the rural common of sidi taibi. *International Journal of Engineering Research and Development*, 10(8), Aug 2014.
- [120] D. Dioufa, A. Nianga, J. Brajardb, M. Creponb, and S. Thiriab. Retrieving aerosol characteristics and sea-surface chlorophyll from satellite ocean color multi-spectral sensors using a neural-variational method. *Remote Sensing of Environment*, 130:74–86, Mar 2013.
- [121] A. Ducournau and A. Bretto. Random walks in directed hypergraphs and application to semi-supervised image segmentation. *Computer Vision and Image Understanding*, 120:91–102, Mar 2014.
- [122] A. Ducournau, A. Bretto, S. Rital, and B. Laget. A reductive approach to hypergraph clustering: An application to image segmentation. *Pattern Recognition*, 45(7), July 2012.
- [123] B. Fulkerson, A. Vedaldi, and S. Soatto. Class segmentation and object localization with superpixel neighborhoods. In *Computer Vision, 2009 IEEE 12th International Conference on*, pages 670–677, Sept 2009.
- [124] J. Hielkema and F. L. Snijders. Operational use of environmental satellite remote sensing and satellite communications technology for global food security and locust control by fao: The artemis and diana systems. *Acta Astronautica*, 32(9):603–616, Sep 1994.

- [125] M. Hossain. Global warming induced sea level rise on soil, land and crop production loss in bangladesh. In *World Congress of Soil Science, Soil Solutions for a Changing World*, 2010.
- [126] S. Huber, R. Fensholt, K. Rasmussen, S. Proud, and A. Anyamba. Improving early warning systems for food security in africa with geostationary earth observation data. *IOP Conference Series: Earth and Environmental Science*, 6:–, 2009.
- [127] C. F. Hutchinson. Uses of satellite data for famine early warning in sub- saharan africa. *International Journal of Remote Sensing*, 12(6):1405–1421, 1991.
- [128] M. Kelly, S. D. Blanchard, E. Kersten, and K. Koy. Terrestrial remotely sensed imagery in support of public health: New avenues of research using object-based image analysis. *Remote Sensing*, 3(11):2321–2345, Oct 2011.
- [129] B. W. Khan MM, Mock NB. Composite indicators for famine early warning systems. *Disasters*, 16(3):195–206, Sep 1992.
- [130] M.-Y. Liu, O. Tuzel, S. Ramalingam, and R. Chellappa. Entropy rate superpixel segmentation. In *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, pages 2097–2104, June 2011.
- [131] A. Mohan, C. Papageorgiou, and T. Poggio. Example-based object detection in images by components. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 23(4):349–361, Apr 2001.
- [132] A. M. Noor, V. A. Alegana, P. W. Gething, A. J. Tatem, and R. W. Snow. Using remotely sensed night-time light as a proxy for poverty in africa. *Population Health Metrics*, 6(5), 2008.
- [133] T. Philpott. New research: synthetic nitrogen destroys soil carbon, undermines soil health. <http://grist.org/article/2010-02-23-new-research-synthetic-nitrogen-destroys-soil-carbon-undermines/>, 2010. [Online; accessed Feb-2015].
- [134] J. A. Quinn, W. Okori, and A. Gidudu. Increased-specificity famine prediction using satellite observation data. In *Proceedings of the First ACM Symposium on Computing for Development*, ACM DEV '10, pages 18:1–18:6, New York, NY, USA, 2010. ACM.

- [135] K. R., S. R., and S. N. Satellite image resolution enhancement using contrast limited adaptive histogram equalization. *Digital Image Processing*, 6(3), 2014.
- [136] A. F. Rahman, D. Dragoni, K. Didan, A. Barreto-Munoz, and J. A. Hutabarat. Detecting large scale conversion of mangroves to aquaculture with change point and mixed-pixel analyses of high-fidelity modis data. *Remote Sensing of Environment*, 130:96–107, Mar 2013.
- [137] A. Schick, M. Bauml, and R. Stiefelhagen. Improving foreground segmentations with probabilistic superpixel markov random fields. In *Computer Vision and Pattern Recognition Workshops (CVPRW), 2012 IEEE Computer Society Conference on*, pages 27–31, June 2012.
- [138] C. S. Thomson MC. The development of malaria early warning systems for africa. *Trends Parasitol*, 17(9):438–445, Sep 2001.
- [139] A. C. Vibrans, R. E. McRoberts, P. Moser, and A. L. Nicoletti. Using satellite image-based maps and ground inventory data to estimate the area of the remaining atlantic forest in the brazilian state of santa catarina. *Remote Sensing of Environment*, 130, 2013.
- [140] P. Viola and M. Jones. Rapid object detection using a boosted cascade of simple features. In *Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on*, volume 1, pages I–511–I–518 vol.1, 2001.
- [141] S. Wang, H. Lu, F. Yang, and M.-H. Yang. Superpixel tracking. In *Computer Vision (ICCV), 2011 IEEE International Conference on*, pages 1323–1330, Nov 2011.
- [142] X. Zhang and X. Cai. Climate change impacts on global agricultural land availability. *Environmental Research Letters*, 6(1):–, 2011.
- [143] Gandhi, R., Veeraraghavan, R., Toyama, K. and Ramprasad V. 2007. Digital Green: Participatory Video for Agricultural Extension, Information and Communication Technology for Development. 2007
- [144] Ramamritham, K., Bahuman, A., Duttgupta, S. 2006. Innovative ICT Tools for Information Provision in Agricultural Extension, Information and Communication Technology for Development. 2006

- [145] Patel, N., Agarwal, S., Rajput, N., Kumar, A., Nanavati, A., Dave, P., Parikh, T. S. 2008. Experiences Designing a Voice Interface for Rural India, IEEE Workshop on Spoken Language Technology for Development. 2008
- [146] Xin, J., Hu, P. 2008. Web Services, Portals and Internet Applications Managing Agricultural Emergency Resources through Information Mashups. J.I of Information Technology in Agriculture, Vol 3(1), 2008
- [147] Reddy, P. K. and Ankaiah, R. 2005. A framework of information technology-based agriculture information dissemination system to improve crop productivity, Current Science, vol. 88, Num.12 2005
- [148] Tambour, L., Houles, V., Cohen-Jonathan, L., Auffray, V., Escande P., and Jallas, E. 2009. Design of a Model-Driven Web Decision Support System in Agriculture: Scientific Models to the Final Software, Advances in Modeling Agricultural Systems, 2009
- [149] Metzler, D., Dumais, S. T. and Meek, C. 2007. Similarity Measures for Short Segments of Text. European Conference on Information Retrieval, 2007
- [150] Sahami, M. and Heilman, T. A web-based kernel function for measuring the similarity of short text snippets. WWW 2006, pp 377-386, 2006.
- [151] <https://dataone.org/> [Last accessed: July, 2010]
- [152] Fattah, M. A. and Ren, F. 2008. Automatic Text Summarization. World Academy of Science, Engineering and Technology 37 2008
- [153] Kan, M., Klavans, J. 2002. Using librarian techniques in automatic text summarization for information retrieval. 2nd ACM/IEEE-CS joint conference on Digital libraries. 2002. pp 36-45.
- [154] Neto, J. L., Freitas, A. A., and Kaestner, C. 2002. Automatic Text Summarization Using a Machine Learning Approach . Advances in Artificial Intelligence, Lecture Notes in Computer Science, Springer. Vol 2507.

- [155] Iqbal, S. 2010. Flood and Erosion Induced Population Displacements: A Socio-economic Case Study in the Gangetic Riverine Tract at Malda District, West Bengal, India. *Journal of Human Ecology* 30(3).
- [156] Ground Water Quality in Shallow Aquifers of India. Central Ground Water Board, Ministry of Water Resources, Government of India. Faridabad, 2010
- [157] Balakrishnan , T. 2009. District Groundwater Brochure Vellore District, Tamil Nadu. Central Ground Water Board, Min. of Water Resources, Govt of India. 2009
- [158] Pereira, A. G., Rinaudo, J., Jeffrey, P., Blasques, J., Quintana, S. C., Courtois, N., Funtowicz, S., and Petit, V. 2003. Ict Tools To Support Public Participation In Water Resources Governance and Planning: Experiences From The Design And Testing Of A Multi-Media Platform. *J. of Environmental Assessment Policy and Management*. Vol. 5(3). 2003. pp. 395-420
- [159] Datta, K. K., Jong, C. 2002. Adverse effect of waterlogging and soil salinity on crop and land productivity in northwest region of Haryana, India. *Agricultural Water Management* Volume 57(3), 2002.
- [160] Gore, P. G., Prasad, T., Hatwar, H. R. 2010. Mapping of Drought Areas over India. National Climate Centre Research Report No. 12/2010
- [161] Nalbari District Disaster Management Database, 2010
- [162] Soils of Sikkim. Forest Environment & Wildlife Management Department, Government of Sikkim, 2010.
- [163] Pathak, A. Cultivation of Large Cardamom in Sikkim. *Journal of Ishani*, Vol 2(6), 2008
- [164] Singh, G., Pal, A. K., Tiwari, A. 2007. Air Pollution and its Impact on Social Spectrum with Special Reference to Korba Coalfield of Chattisgarh. *Intl. Conf. on MSECCMI*, New Delhi, 2007
- [165] Final District Agriculture Plan: Bharuch, Agriculture and Co-operation Dept, Govt of Gujrat, 2007
- [166] Disaster Management in Maharashtra. *Envis Newsletter*, Environment Department, Govt. of Maharashtra. Vol 1, 2006.

- [167] Systematic Evaluation of IWDP and DPAP Programmes: A Study of Kurnool District, AP. Journal of LNRMI, 2010.
- [168] Jha, T.N., Viswanathan, K.U. 1999. Problems and Prospects of Agricultural Development in Bihar. NABARD, 1999.
- [169] Srinivasa, N., Mallik, B., Gowda, C.C. 2004. *Aceria cajani* (Acari : Eriophyidae) Transmitted Pigeonpea Sterility Mosaic Disease. All India Network Project on Agricultural Acarology. Department of Entomology, University of Agricultural Sciences, Bangalore, 2004
- [170] Gadekar, M. Tracking the Drought-II Ratnagiri: Water scarcity amidst plenty. InfoChange News & Features, 2003. [<http://infochangeindia.org>]
- [171] Singh, R.P., Islam. Z. 2010. Land Use Planning In Western Uttar Pradesh: Issues And Challenges. Recent Research in Science and Technology 2010, 2(9)
- [172] Joji, V.S. 2009. Ground Water Information Booklet Of Kozhicode District, Kerala. Central Ground Water Board, Min. of Water Resources, Govt. of India, 2009
- [173] <http://india.gov.in/sectors/agriculture/index.php>
- [174] <http://www.nih.ernet.in/water.htm>
- [175] <http://www.teriin.org/coping/index.htm>
- [176] http://en.wikipedia.org/wiki/List_of_districts_of_India
- [177] <http://code.google.com/apis/ajaxsearch/>
- [178] <http://en.wikipedia.org/wiki/Tf%E2%80%93idf>
- [179] <http://en.wikipedia.org/wiki/PageRank>
- [180] K.R.McKeown,J.L.Klavans,V.Hatzivassiloglou, R. Barzilay, and E. Eskin. Towards multidocument summarization by reformulation: progress and prospects. In Proceedings of AAAI, 99
- [181] D. R. Radev, H. Jing, and M. Budzikowska. Centroid-based summarization of multiple documents: sentence extraction, utility-based evaluation, and user studies. In Proceedings of NAACL-ANLPWorkshop on Automatic summarization 2000,

- [182] D. K. Evans, K. Mckeown, and J. L. Klavans. Similarity-based multilingual multi-document summarization. *IEEE Transactions on Information Theory*, 49, 2005.
- [183] J. Kupiec, J. Pedersen, and F. Chen. A trainable document summarizer. *SIGIR*, 1995
- [184] C. Aone, M. E. Okurowski, J. Gorlinsky, and B. Larsen. A trainable summarizer with knowledge acquired from robust NLP techniques, pages. 1999.
- [185] K. M. Svore. Enhancing single-document summarization by combining ranknet and third-party sources, 2007.
- [186] J. M. Conroy and D. P. O'leary. Text summarization via hidden markov models. *SIGIR 2001*
- [187] M. Kamvar and S. Baluja. A large scale study of wireless search behavior: Google mobile search. *CHI 2006*
- [188] M. Kamvar, M. Kellar, R. Patel, and Y. Xu. Computers and iphones and mobile phones, oh my!: a logs-based comparison of search users on different devices. *WWW 2009*
- [189] J. Yi, F. Maghoul, and J. Pedersen. Deciphering mobile search patterns: a study of Yahoo! mobile search queries. *WWW 2008*
- [190] T. Heimonen and M. Kaki. Mobile findex: supporting mobile web search with automatic result categories. *MobileHCI 2007*
- [191] K. Church, J. Neumann, M. Cherubini, and N. Oliver. SocialSearchBrowser: a novel mobile search and information discovery tool. *IUI 2010*
- [192] K. Church and B. Smyth. Who, what, where and when: a new approach to mobile search. *IUI 2008*
- [193] M. Jones, G. Buchanan, R. Harper, and P. Xech. Questions not answers: a novel mobile search technique. *CHI 2007*
- [194] A. Karlson, G. Robertson, D. Robbins, M. Czerwinski, and G. Smith. FaThumb: a facet-based interface for mobile search. *CHI 2006*
- [195] J. Chen, L. Subramanian, and E. Brewer. SMS-based web search for low-end mobile devices. *MobiComm 2010*

- [196] Google SMS. <http://www.google.com/sms>.
- [197] Open Graph Protocol. <http://www.ogp.me/>
- [198] Yahoo One Search. <http://mobile.yahoo.com/onesearch>.
- [199] Windows Live Mobile. <http://home.mobile.live.com/>.
- [200] <http://dev.mysql.com/doc/refman/5.5/en/fulltext-stopwords.html>
- [201] <http://www.smartinsights.com/mobile-marketing/mobile-marketing-analytics/mobile-marketing-statistics/>
- [202] Chen, Jay and Subramanian, Lakshminarayanan and Li, Jinyang. RuralCafe: web search in the rural developing world. WWW. 2009
- [203] Yan, Tingxin and Kumar, Vikas and Ganesan, Deepak. CrowdSearch: exploiting crowds for accurate real-time image search on mobile phones. MobiSys. 2010
- [204] Teevan, Jaime and Ramage, Daniel and Morris, Merredith Ringel. # TwitterSearch: a comparison of microblog search and web search. WSDM. 2011
- [205] Amini, Shahriyar and Brush, AJ and Krumm, John and Teevan, Jaime and Karlson, Amy. Trajectory-aware mobile search. CHI. 2012
- [206] Svore, Krysta M and Teevan, Jaime and Dumais, Susan T and Kulkarni, Anagha. Creating temporally dynamic web search snippets. SIGIR. 2012
- [207] Teevan, Jaime and Collins-Thompson, Kevyn and White, Ryen W and Dumais, Susan T and Kim, Yubin. Slow Search: Information Retrieval without Time Constraints. HCIR. 2013
- [208] Kamvar, Maryam and Baluja, Shumeet. A large scale study of wireless search behavior: Google mobile search. CHI. 2006
- [209] Arter, David and Buchanan, George and Jones, Matt and Harper, Richard. Incidental information and mobile search. MobileHCI. 2007

- [210] Teevan, Jaime and Cutrell, Edward and Fisher, Danyel and Drucker, Steven M and Ramos, Gonzalo and André, Paul and Hu, Chang. Visual snippets: summarizing web pages for search and revisitation. CHI. 2009
- [211] Radev, D. R., Hovy, E., and McKeown, K. (2002). Introduction to the special issue on summarization. Computational Linguistics., 28(4)
- [212] Yining Wang, Liwei Wang, Yuanzhi Li, Di He, Wei Chen, Tie-Yan Liu. 2013. A Theoretical Analysis of NDCG Ranking Measures. In Proceedings of the 26th Annual Conference on Learning Theory (COLT 2013)
- [213] T.Joachims, L.Granka, B.Pan, H.Hembrooke, and G.Gay. Accurately interpreting clickthrough data as implicit feedback. In *Proc. SIGIR*, pages 154–161, Salvador, Brazil, 2005.
- [214] T.Joachims, L.Granka, B. Pan, H.Hembrooke, F.Radlinski, and G.Gay. Evaluating the accuracy of implicit feedback from clicks and query reformulations in web search. *ACM Trans. Inf. Syst.*, 25(2), Apr. 2007.
- [215] Kalervo Jarvelin, Jaana Kekalainen: Cumulated gain-based evaluation of IR techniques. *ACM Transactions on Information Systems* 20(4), 422-446 (2002)
- [216] Christopher D. Manning, Prabhakar Raghavan, Hinrich Schtze (2009). An Introduction to Information Retrieval. Cambridge University Press. pp 158-162. 2009
- [217] Y. Zaki, J. Chen, T. Pötsch, T. Ahmad, and L. Subramanian. Dissecting web latency in ghana. In *Proceedings of the 2014 Conference on Internet Measurement Conference*, pages 241–248. ACM, 2014.
- [218] M. Zheleva, P. Schmitt, M. Vigil, and E. Belding. The increased bandwidth fallacy: performance and usage in rural zambia. In *Proceedings of the 4th Annual Symposium on Computing for Development*, page 2. ACM, 2013.
- [219] S. Chava, R. Ennaji, J. Chen, and L. Subramanian. Cost-aware mobile web browsing. *IEEE Pervasive Computing*, (3):34–42, 2012.

- [220] J. Chen and L. Subramanian. Interactive web caching for slow or intermittent networks. In *Proceedings of the 4th Annual Symposium on Computing for Development*, page 5. ACM, 2013.
- [221] D. Iland and E. Belding. Emergenet: robust, rapidly deployable cellular networks. *Communications Magazine, IEEE*, 52(12):74–80, 2014.
- [222] S. Isaacman and M. Martonosi. Potential for collaborative caching and prefetching in largely-disconnected villages. In *Proceedings of the 2008 ACM workshop on Wireless networks and systems for developing regions*, pages 23–30. ACM, 2008.
- [223] A. Balasubramanian, N. Balasubramanian, S. J. Huston, D. Metzler, and D. J. Wetherall. Findall: A local search engine for mobile phones. In *Proceedings of the 8th international conference on Emerging networking experiments and technologies*, pages 277–288. ACM, 2012.
- [224] S. Isaacman and M. Martonosi. The c-link system for collaborative web usage: A real-world deployment in rural nicaragua. In *Workshop on Networked Systems for Developing Regions*, 2009.
- [225] S. Isaacman and M. Martonosi. Low-infrastructure methods to improve internet access for mobile users in emerging regions. In *Proceedings of the 20th international conference companion on World wide web*, pages 473–482. ACM, 2011.
- [226] William Thies, Janelle Prevost, Tazeen Mahtab, Genevieve T. Cuevas, Saad Shakhshir, Alexandro Artola, Binh D. Vo, Yuliya Litvak, Sheldon Chan, Sid Henderson, Mark Halsey, Libby Levison, and Saman Amarasinghe. Searching the World Wide Web in Low-Connectivity Communities. In *WWW*, 2002.
- [227] Yasir Zaki, Thomas Ptsch, Jay Chen, Lakshmi Subramanian, Carmelita Grg Adaptive Congestion Control for Unpredictable Cellular Networks ACM Special Interest Group on Data Communication (SIGCOMM) 2015.
- [228] Jay Chen, Lakshmi Subramanian, Janardhan Iyengar, and Bryan Ford TAQ: Enhancing Fairness and Performance Predictability in Small Packet Regimes Proceedings of the European Conference on Computer Systems (EuroSys) 2014.
- [229] Aruna Balasubramanian, Brian Levine, Arun Venkataramani. DTN Routing as a Resource Allocation Problem. ACM Sigcomm, August 2007

- [230] Bloomberg. Daily historical time-series of Federal Funds Target Rate and Futures prices and volumes for Federal Funds Rate and US Treasuries. March 25, 2015.
- [231] E. Boukus and J. Rosenberg. The information content of FOMC minutes. <http://ssrn.com/abstract=922312>.
- [232] A. Cofnas. *Sentiment indicators: renko, price break, kagi, point & figure —what they are and how to use them to trade*. Hoboken, NJ: Wiley. 2010.
- [233] D. Danker and M. Luecke. Background on FOMC meeting minutes. *Federal Reserve Bulletin* 91(2). Spr 2005.
- [234] T. Durden. FOMC minutes word cloud and key word count. *Zero Hedge*. Aug 30, 2011. Available at: <http://www.zerohedge.com/news/fomc-minutes-word-cloud-and-key-word-count>.
- [235] M. Farka and A. Fleissig. The effect of FOMC statements on asset prices. *Intl Rev Appl Econ* 26(3). May 2012.
- [236] K. Fitz-gerald. The huge economic indicator everyone misses. *Money Morning*. Mar 25, 2014. Available at: <http://moneymorning.com/2014/03/25/huge-economic-indicator-everyone-misses/>
- [237] Y. Goldberg and O. Levy. word2vec explained: deriving Mikolov et al.'s negative sampling word-embedding method. Unpublished manuscript, February 14, 2014. Available at: <http://arxiv.org/pdf/1402.3722v1.pdf>
- [238] R. Gürkaynak, B. Sack, and E. Swanson. Do actions speak louder than words? The response of asset prices to monetary policy actions and statements. *Intl J Central Banking* 1(1). May 2005.
- [239] S. Hansen, M. McMahon, and A. Prat. Transparency and deliberation within the FOMC: a computational linguistics approach. Unpublished manuscript. Feb 2015.
- [240] S. Hendry and A. Madeley. Text mining and the information content of Bank of Canada communications. Bank of Canada Working Paper 2010-31. Nov 2010.
- [241] S. Kennedy. Word inflation accelerating at Fed justifies investor confusion. *Bloomberg*. Sep 18, 2014.

- [242] G. Maddala. *Limited-dependent and qualitative variables in econometrics*. Econometric society monographs no. 3. Cambridge: Cambridge University Press, 1983.
- [243] T. Mikolov, K. Chen, G. Corrado, and J. Dean. Efficient estimation of word representations in vector space. *CoRR*, abs/1301.3781, 2013
- [244] T. Mikolov, I. Sutskever, K. Chen, G. Corrado, and J. Dead. Distributed representations of words and phrases and their compositionality. In *Advances in Neural Information Processing Systems 26: 27th Annual Conference on Neural Information Processing Systems 2013*.
- [245] A. Oyedele. This is how often the Fed talks about employment and inflation. *Business Insider*. Feb 18, 2015. Available at: <http://www.businessinsider.com/fomc-minutes-on-unemployment-and-inflation-2015-2> .
- [246] Prattle Analytics. prattle-analytics.com. 2014.
- [247] C. Rosa. Words that shake traders: the stock markets reaction to central bank communication in real time. *J Empirical Finance* 18(5). Dec 2011.
- [248] C. Schonhardt-Bailey. *Deliberating American monetary policy: a textual analysis*. Cambridge, MA: MIT Press. Nov 2013.
- [249] P. Tetlock. All the news that's fit to reprint: do investors react to stale information? *Rev Fin Stud* 24(5). May 2011.
- [250] U.S. Department of the Treasury. Daily Treasury yield curve rates, 1990-2015. Accessed on 3 Apr 2015. Available at: <http://www.treasury.gov/resource-center/data-chart-center/interest-rates/>
- [251] U.S. Federal Reserve Bank. About the FOMC. <http://www.federalreserve.gov/monetarypolicy/fomc.htm>. 2015.
- [252] U.S. Federal Reserve Bank of St. Louis. Federal Reserve Economic Data. Accessed on 3 Apr 2015. Available at: <http://research.stlouisfed.org/fred2/>
- [253] U.S. Federal Reserve System Board of Governors. Federal Open Market Committee. Accessed Apr 2015. Available at: <http://www.federalreserve.gov/monetarypolicy/fomc.htm>

- [254] U.S. Federal Reserve System Board of Governors. The Federal Reserve System purposes and functions. Washington, DC: Board of Governors of the Federal Reserve System. Jun 2005.
- [255] M. Wynne. A short history of FOMC communication. *Economic Letter* 8(8), Sep 2013.
- [256] R. Zadeh and A. Zollman. Predicting market volatility from Federal Reserve Board meeting minutes. Unpublished manuscript. 2009.
- [257] Hulth A, Rydevik G, and Linde A, Web queries as a source for syndromic surveillance, *PloS one* 4 (2009), no. 2.
- [258] Althouse BM, Ng YY, and Cummings DAT, Prediction of dengue incidence using search query surveillance., 5 (2011), no. 8.
- [259] Chew C and Eysenbach G, Pandemics in the age of twitter: Content analysis of tweets during the 2009 h1n1 outbreak, *ACM Trans. Program. Lang. Syst.* 5 (2010), no. 11, e14118.
- [260] Freifeld CC, Mandl KD, Reis BY, and Brownstein JS, Healthmap: global infectious disease monitoring through automated classification and visualization of internet media reports, *J Am Med Inform Assoc* 15 (2008), no. 2, 150157.
- [261] Emily H. Chan, Vikram Sahai, Corrie Conrad, and John S. Brownstein, Using web search query data to monitor dengue epidemics: A new model for neglected tropical disease surveillance, *PLoS Negl Trop Dis* 5 (2011), no. 5, e1206.
- [262] Das D, Metzger K, Heffernan R, Balter S, Weiss D, and et al., Monitoring over-the-counter medication sales for early detection of disease outbreaks U new york city, *MMWR Morb Mortal Wkly Rep* 54 (2005), no. suppl, 41U 46.
- [263] Mykhalovskiy E and Weir L, The global public health intelligence network and early warning outbreak detection: a canadian contribution to global public health, *Canadian Journal of Public Health* 97 (2006), no. 1, 4244.
- [264] Eysenbach G, Infodemiology and infoveillance: framework for an emerging set of public health informatics methods to analyze search, communication and publication behavior on the internet, *J Med Internet Res* 11 (2009), no. 1, e11.

- [265] Eysenbach G and Khler C, Health-related searches on the internet, *JAMA* 291 (2004), no. 24, 2946.
- [266] Janaina Gomide, Adriano Veloso, Wagner Meira, Virgilio Almeida, Fabricio Benevenuto, Fernanda Ferraz, and Mauro Teixeira, Dengue surveillance based on a computational model of spatio-temporal locality of twitter., *Proceedings of the ACM WebSci11*, June 14-17 2011, Koblenz, Germany., 2011.
- [267] Tolentino H, "scanning the emerging infectious diseases horizon-visualizing promed emails using epispider", *International Society for Disease Surveillance Annual Conference*, 2006.
- [268] Ginsberg J, Mohebbi MH, Patel RS, Brammer L, Smolinski MS, and Brilliant L, Detecting influenza epidemics using search engine query data, *Nature* 457 (2009), no. 5, 10121014.
- [269] Bork KH, Klein BM, Milbak K, Trautner S, Pedersen UB, and Heegaard E, Surveillance of ambulance dispatch data as a tool for early warning, *Euro Surveill.* 11 (2006), no. 12, 229233.
- [270] V Lampos and N Cristianini, Nowcasting events from the social web with statistical learning., *ACM Transactions on Intelligent Systems and Technology* (2011).
- [271] Besculides M, Heffernan R, Mostashari F, and Weiss D, Evaluation of school absenteeism data for early outbreak detection, new york city, *BMC Public Health* 5 (2005), 105.
- [272] Lawrence C. Madoff, Promed-mail: An early warning system for emerging diseases, *Clinical Infectious Diseases* 39 (2004), no. 2, 227232.
- [273] Lawrence C. Madoff, David N. Fisman, and Taha Kass-Hout, A new approach to monitoring dengue activity, 5 (2011), no. 5.
- [274] Beatty ME, Stone A, Fitzsimons DW, Hanna JN, Lam SK, Vong S, Guzman MG, Mendez-Galvan JF, Halstead SB, Letson GW, Kuritsky J, Mahoney R, and Margolis HS, Best practices in dengue surveillance: a report from the asia-pacific and americas dengue prevention boards., *Asia-Pacific and Americas Dengue Prevention Boards Surveillance Working Group*.
- [275] Collier N, Doan S, Kawazoe A, Goodwin RM, Conway M, and et al., Biocaster: Detecting public health rumors with a web-based text mining system, *Bioinformatics* 24 (2008), 2940U 2941.

- [276] Khan SA, Patel CO, and Kukafka R, Godsn: Global news driven disease outbreak and surveillance, AMIA Annual Symposium Proceedings, 2006.
- [277] Yih WK, Teates KS, Abrams A, Kleinman K, Kulldorff M, and et al., Telephone triage service data for detection of influenza-like illness, PloS one 4 (2009), e5260U e5260
- [278] Freifeld CC, Mandl KD, Reis BY, Brownstein JS (2007) HealthMap: Global infectious disease monitoring through automated classification and visualization of Internet media reports. J Am Med Inform Assoc
- [279] R. Agrawal, S. Chakraborty, S. Gollapudi, A. Kannan, and K. Kenthapadi. Quality of textbooks: An empirical study. In ACM DEV, 2012.
- [280] R. Agrawal, S. Gollapudi, A. Kannan, and K. Kenthapadi. Identifying enrichment candidates in textbooks. In WWW, 2011.
- [281] R. Agrawal, S. Gollapudi, K. Kenthapadi, N. Srivastava, and R. Velu. Enriching textbooks through data mining. In ACM DEV, 2010.
- [282] B. B. Armbruster and T. H. Anderson. Content area textbooks. Technical report, Reading Education Report No. 23, BBN, 1981.
- [283] B. B. Armbruster and T. H. Anderson. Producing considerate expository text: Or easy reading is damned hard writing. Technical report, Reading Education Report No. 46, BBN, 1984.
- [284] B. K. Britton, A. Woodward, and M. R. Binkley. Learning from Textbooks: Theory and Practice. Routledge, 1993.
- [285] B. Bruce, A. Rubin, and K. Starr. Why readability formulas fail. IEEE Transactions on Professional Communication, PC-24, 1981.
- [286] M. Chambliss and R. Calfee. Textbooks for Learning: Nurturing Childrens Minds. Wiley-Blackwell, 1998.
- [287] R. Clark, F. Nguyen, and J. Sweller. Efficiency in learning: Evidence-based guidelines to manage cognitive load. Pfeiffer, 2006.

- [288] K. Collins-Thompson and J. P. Callan. A language modeling approach to predicting reading difficulty. In HLT-NAACL, 2004.
- [289] E. K. Dishner. Reading in the Content Areas: Improving Classroom Instruction. Kendall/Hunt, 1992.
- [290] W. DuBay. The principles of readability. Impact Information, 2004.
- [291] C. Fellbaum. WordNet: An electronic lexical database. MIT Press, 1998.
- [292] J. Gillies and J. Quijada. Opportunity to learn: A high impact strategy for improving educational outcomes in developing countries. USAID Educational Quality Improvement Program (EQUIP2), 2008.
- [293] S. R. Goldman. Learning from text: Reflections on the past and suggestions for the future. Discourse Processes, 23, 1997.
- [294] W. Gray and B. Leary. What makes a book readable. University of Chicago Press, 1935.
- [295] B. Grosz and C. Sidner. Attention, intentions, and the structure of discourse. Computational linguistics, 12(3), 1986.
- [296] R. Guillemette. Predicting readability of data processing written materials. ACM SIGMIS Database, 18(4), 1987.
- [297] E. A. Hanushek and L. Woessmann. The role of education quality for economic growth. Policy Research Department Working Paper 4122, World Bank, 2007.
- [298] E. B. Johnsen. Textbooks in the Kaleidoscope: A Critical Survey of Literature and Research on Educational Texts. Scandinavian University Press, 1992.
- [299] J. S. Justeson and S. M. Katz. Technical terminology: Some linguistic properties and an algorithm for identification in text. Natural Language Engineering, 1(1), 1995.
- [300] R. Kate, X. Luo, S. Patwardhan, M. Franz, R. Florian, R. Mooney, S. Roukos, and C. Welty. Learning to predict readability using diverse linguistic features. In COLING, 2010.

- [301] D. Kieras and C. Dechert. Rules for comprehensible technical prose: A survey of the psycholinguistic literature. Technical Report TR-85/ONR-21, University of Michigan, 1985.
- [302] B. Lively and S. Pressey. A method for measuring the vocabulary burden of textbooks. *Educational Administration and Supervision*, 9(73), 1923.
- [303] D. Marcu. *The theory and practice of discourse parsing and summarization*. MIT Press, 2000.
- [304] F. Paas, A. Renkl, and J. Sweller. Cognitive load theory and instructional design: Recent developments. *Educational Psychologist*, 38(1), 2003.
- [305] J. M. Pawlowski, K. C. Barker, and T. Okamoto. Quality research for learning, education, and training. *Reading and Writing Quarterly*, 10(2), 2007.
- [306] J. Plass, R. Moreno, and R. Brunken. *Cognitive load theory*. Cambridge University Press, 2010.
- [307] L. Polanyi and R. Scha. A syntactic approach to discourse semantics. In *COLING*, 1984.
- [308] E. Pollock, P. Chandler, and J. Sweller. Assimilating complex information. *Learning and Instruction*, 12(1), 2002.
- [309] R. Seguin. The elaboration of school textbooks. Technical report, ED-90/WS-24, UNESCO, 1989.
- [310] L. Sherman. *Analytics of literature: A manual for the objective study of English prose and poetry*. Ginn and Company, 1893.
- [311] S. A. Thompson and W. C. Mann. Rhetorical structure theory: A framework for the analysis of texts. *IPRA Papers in Pragmatics*, 1(1), 1987.
- [312] K. Toutanova, D. Klein, C. D. Manning, and Y. Singer. Feature-rich part-of-speech tagging with a cyclic dependency network. In *NAACL-HLT*, 2003.
- [313] H. Tyson-Bernstein. A conspiracy of good intentions: Americas textbook fiasco. Technical report, Council for Basic Education, Washington, 1989.
- [314] A. Verspoor and K. B. Wu. Textbooks and educational development. Technical report, World Bank, 1990.

- [315] K. Wang, C. Thrasher, E. Viegas, X. Li, and P. Hsu. An overview of Microsoft Web N-gram corpus and applications. In NAACL-HLT, 2010.
- [316] A. N. Whitehead. The organisation of thought. *Proceedings of the Aristotelian Society*, 17, 191617.
- [317] S. Witte and L. Faigley. Coherence, cohesion, and writing quality. *College Composition and Communication*, 32(2), 1981.
- [318] A. Woodward, D. L. Elliott, and C. Nagel. *Textbooks in School and Society: An Annotated Bibliography and Guide to Research*. Garland, 1988.
- [319] World-Bank. *Knowledge for Development: World Development Report: 1998/99*. Oxford University Press, 1999.
- [320] J. Zhao and M. Kan. Domain-specific iterative readability computation. In *ACM JCDL*, 2010.
- [321] Diebold, Francis X. (2001). *Elements of Forecasting* (2nd ed.). Cincinnati: South Western. p. 254