Understanding User-Generated Content on Social Media

Dissertation submitted in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy

By

MEENAKSHI NAGARAJAN

(Signature of Student)

_____

2010
Wright State University

WRIGHT STATE UNIVERSITY
SCHOOL OF GRADUATE STUDIES

August 18, 2010

I HEREBY RECOMMEND THAT THE DISSERTATION PREPARED UNDER MY SUPERVISION BY Meenakshi Nagarajan ENTITLED Understanding User-generated Content on Social Media BE ACCEPTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF Doctor of Philosophy.

Amit P. Sheth, Ph.D.
Dissertation Director

Arthur Ardeshir Goshtasby, Ph.D.
Director, Computer Science
and Engineering Ph.D. Program

Andrew T. Hsu, Ph.D.
Dean, School of Graduate Studies

Committee on Final Examination

John M. Flach, Ph.D.

Daniel Gruhl, Ph.D.

Michael L. Raymer, Ph.D.

Shaojun Wang, Ph.D.

Kevin Haas, M.S.

ABSTRACT

Nagarajan, Bala Meenakshi, Ph.D., Department of Computer Science and Engineering, Wright State University, 2010. Understanding User-generated Content on Social Media.

Over the last few years, there has been a growing public and enterprise fascination with 'social media' and its role in modern society. At the heart of this fascination is the ability for users to participate, collaborate, consume, create and share content via a variety of platforms such as blogs, micro-blogs, email, instant messaging services, social network services, collaborative wikis, social bookmarking sites, and multimedia sharing sites.

This dissertation is devoted to understanding informal user-generated textual content on social media platforms and using the results of the analysis to build Social Intelligence Applications.

The body of research presented in this thesis focuses on understanding what a piece of user-generated content is *about* via two sub-goals of Named Entity Recognition and Key Phrase Extraction on informal text. In light of the poor context and informal nature of content on social media platforms, we investigate the role of contextual information from documents, domain models and the social medium to supplement and improve the reliability and performance of existing text mining algorithms for Named Entity Recognition and Key Phrase Extraction.

In all cases we find that using multiple contextual cues together lends to reliable inter-dependent decisions, better than using the cues in isolation and that such improvements are robust across domains and content of varying characteristics, from micro-blogs like Twitter, social networking forums such as those on MySpace and Facebook, and blogs on the Web.

Finally, we showcase two deployed Social Intelligence applications that build over the results of Named Entity Recognition and Key Phrase Extraction algorithms to provide near real-time information about the pulse of an online populace. Specifically, we describe what it takes to build applications that wish to exploit the 'wisdom of the crowds' – highlighting challenges in data collection, processing informal English text, metadata extraction and presentation of the resulting information.

Contents

List of Figures

List of Tables

Acknowledgement

To all the people I have taken so much from and given back so little, thank-you!

My time in graduate school has been an enormously positive and social experience largely because of the outstanding, but more importantly, kind people around me.

This journey would not have started without my advisor, Amit Sheth's gentle, relentless push and unfailing belief in my abilities. For that I am deeply grateful. The collaborative eco-system that he has provided for me and other graduate students is one that has given us utmost freedom in pursuing our research interests. His energy and enthusiasm at work is worthy of emulation. I am grateful to him for throwing all his support and resources behind me during the course of the program and the job hunt process.

I am indebted to Daniel Gruhl and Kevin Haas for introducing me to Social Media (i.e. making me 'cool'!), Business Analytics and Text Mining. The summer I spent at IBM Almaden Research Center helped me lay the foundations for this thesis and was undoubtedly a strategic turning point in my graduate school career. I am also thankful to them for introducing me to the world of broken English – I owe all of my SMS-style vocabulary to you both!

Dan read and critiqued every single paper I wrote, including this dissertation – for that he has my profuse thanks.

Along with Dan and Kevin, Christine Robson, Jan Pieper and Julia Grace have been great collaborators, co-authors and supportive colleagues. I am thankful to them for their time, insights and for making collaborative work so easy and a lot of fun.

Marti Hearst taught me the rigor of scientific investigation. In many ways, Marti's visions for the field of Text Data Mining, her take on the new evolving online media and views on how to do research have strongly influenced me. I am grateful for the time she gave to my paper drafts and research statement, for the crucial letter of recommendation, for all the good advice during the job hunt process and for simply being supportive throughout.

I am particularly grateful for the time I spent working with Amir Padovitz, someone who is

simultaneously brilliant and fun! His ideas and visions for the field of Named Entity Recognition have inspired a lot of my work. Michael Gamon was my introduction to the world of Opinion Mining. His genuine interest in helping beginners like me, and the enthusiasm he has shown for the numerous ideas and projects I have taken to him, have left a lasting impression on me.

I am thankful to my thesis committee members, John Flach, Michael Raymer and Shaojun Wang for their advice and comments. I am particularly grateful for my association with John. His perspective on the emerging social media informed by his vast knowledge of the field of Cognitive Psychology opened the world of computational social science to me.

The summers I spent at Hewlett Packard Labs, IBM Research, UC Berkeley and Microsoft Research were rewarding periods of my graduate program. I have learned a lot from the researchers and fellow interns in each of these organizations. Arif Merchant and Kimberly Keeton gave me the opportunity to work on my first ever industry research project. I have learned a great deal about succinct, eloquent description of complex ideas from Matt Hurst. His in-depth knowledge of a field and simultaneous grasp of the big picture have strongly influenced how I treat a research problem.

Life in graduate school would not have been half as fun without my dear friends and colleagues Cartic Ramakrishnan, Christopher Thomas, Karthik Gomadam, Pablo Mendes and Ajith Ranabahu. I am grateful to each one of them for their time and insights in paper writing and critiquing, and most importantly, for helping me strengthen my research ideas.

Cartic and Christopher were perhaps the most important reason why I considered graduate school. Their grasp of the subject matter has always left me in awe and with the desire to learn more. The discussions I have had with Cartic during the formative years of my thesis greatly enriched my understanding of the field of Text Mining. My gratitude also extends to Cartic and Christopher for being immaculate, passionate chefs – the soul food they cooked always brightened an otherwise dull graduate school day!

Karthik's enthusiasm for technology, research and all things Web has been extremely conta-

gious. I am thankful to him and Ajith for selflessly taking time off their weekends and away from their families to educate me in the world of Web services and for not laughing at me when I did not know what Ajax meant, months after it was invented!

Kunal Verma was my first academic student mentor who taught me the art of effectively probing scientific concepts and communicating technical ideas. I am grateful to him for teaching me the tenets of good writing and for all the advice he gave me during the job hunt process.

My time in graduate school would not have been fulfilling if it were not for my friends outside this world. I am grateful for the infallible support that Drashti Dave and Haritha Muthyala have extended to me through my highs and lows. I have grown a great deal, both personally and academically because of their friendships. I am very thankful to Hemant Purohit for showing me the benefits of slowing down on a daily basis! Kamal Baid and Hemant also have my gratitude for allowing me to be their mentor during their initial years in the graduate program. KK has my gratitude and respect for helping me keep my eye on the ball throughout the dissertation writing process and more importantly, for helping me stay grounded.

Self-belief was a waning resource without the conviction, undivided love and support that my family continues to show to me from half-way across the globe. I will forever be thankful to my sister for exploring her love for Biology and Entomology, so I could learn very early on what I did not want to do!

To my parents Pitchai, Kousi and my dearest grandmother Balu Patty

1. Introduction

Over the last few years, there has been a growing public and enterprise fascination with 'social media' and its role in modern society. At the heart of this fascination is the ability for users to participate, collaborate, consume, create and share content via a variety of social platforms. Blogs, micro-blogs, email, instant messaging services, social network services, collaborative wikis, social bookmarking sites, and multimedia sharing sites are examples of social software platforms that have facilitated this growing fascination with sharing data.

The popularity of social media among users and the role that it has played in attracting real-time traffic, enabling large-scale information diffusion and creating tangible effects on participating economies and societies is well known. There are over 100 million active users of MySpace today [1]. Facebook has over 400 million users[2] and Twitter is growing at over 1300% a year[3]. With the ability to rapidly disseminate information, new topics can generate tremendous buzz in a matter of hours and enable applications to tap into the 'wisdom of the crowds' in near real-time.

This change in the landscape of communication on online media from a publisher-oriented media to a more conversational, social media has not only affected what or how we share infor-

[1] http://www.myspace.com/pressroom?url=/fact+sheet/
[2] http://www.facebook.com/press/info.php?statistics
[3] http://blog.nielsen.com/nielsenwire/online mobile/twitters- tweet-smell-of-success/

mation, but also what we seek. In addition to factual information, we are also able to access conversations, opinions and emotions that these facts evoke among other users. We are able to ask questions such as, what are people saying about a news-worthy event or entity? Can we use this information to assess a population's preference? Can we study how these preferences propagate in a network of friends? Are such crowd-sourced preferences a good substitute for traditional polling methods?

This growing need for 'social' information has fueled several innovative computational methods of encoding and analyzing individual and group behavior on the Web. Several of these investigations are grounded in Laswell's theory of Communication: *"who (people) said what (content) to whom (network structures) in what channel with what effect"*; and can be broadly categorized under one or more of these areas of investigation:

(a) Understanding aspects of the user-generated *content*, be it textual data, images, videos or attention metadata such as page visitations or thumbs up, thumbs down votes on an article;

(b) Modeling and observing the *network* within which the content was produced and

(c) Characterizing *individuals and groups* that produce and consume the content.

1.1 Dissertation Focus

This dissertation is dedicated to the first of these three areas – in the computational analysis of *textual* user-generated content on social media platforms for the end goals of understanding what the content is *about*.

'Aboutness' of text is one among several terms used to express certain attributes of a discourse,

text or document. A central component of knowledge organization and information retrieval, the concept of *aboutness* aims to characterize what a document is about, what its content, subject or topic matter are (Wilson (114)).

While user-generated content (UGC), also known as consumer-generated media (CGM) or user-created content (UCC), refers to various kinds of publicly available media content that are produced by end-users [4], the focus of this dissertation is in understanding of *textual* content via the lens of text mining.

Text mining is a field tasked with gleaning meaningful information from natural language text and is loosely characterized as the process of analyzing text to extract information that is useful for particular purposes. The focus of this dissertation is in applying statistical natural language processing and text mining techniques to understand what a piece of user-generated textual content is *about*.

Informal User-generated Content

There is a rich body of literature in processing textual corpora for various end goals (for examples, see Witten (115), Berry (9), Navigli (86), Nadeau and Sekine (84), Pang and Lee (89)). However, certain characteristics specific to UGC introduce new challenges in their analyses.

Communication on social media platforms is inherently less formal and in most cases unmediated. Pockets of self-contained interactions lack explicit context because they are typically conducted between like-minded people that already have a sense of shared context. A large portion of language is also in the Informal English domain – a blend of abbreviations, slang and context

[4] `http://en.wikipedia.org/wiki/User-generated_content`

User-generated content on Twitter during the 2009 Iran Election

show support for democracy in Iran
add green overlay to your Twitter avatar with 1-click - http://helpiranelection.com/

Twitition: Google Earth to update satellite images of Tehran
#Iranelection http://twitition.com/csfeo @patrickaltoft

Set your location to Tehran and your time zone to GMT +3.30.
Security forces are hunting for bloggers using location/timezone searches

User comments on music artist pages on MySpace

Your music is really bangin!

You're a genius! Keep droppin bombs!

u doin it up 4 real. i really love the album.

hey just hittin you up showin love to one of
chi-town's own. MADD LOVE.

Comments on Weblogs about movies and video games

I decided to check out Wanted demo today even though I really did not like the movie

It was THE HANGOVER of the year..lasted forever..
so I went to the movies..bad choice picking GI Jane worse now

Excerpt from a blog around the 2009 Health Care Reform debate

Hawaii's Lessons - NY times
In Hawaii's Health System, Lessons for Lawmakers
Since 1974, Hawaii has required all employers to provide relatively generous health care benefits to any employee who works 20 hours a week or more. If health care legislation passes in Congress, the rest of the country may barely catch up.Lawmakers working on a national health care fix have much to learn from the past 35 years in Hawaii, President Obama's native state.
Among the most important lessons is that even small steps to change the system can have lasting effects on health. Another is that, once benefits are entrenched, taking them away becomes almost impossible. There have not been any serious efforts in Hawaii to repeal the law, although cheating by employers may be on the rise. But perhaps the most intriguing lesson from Hawaii has to do with costs. This is a state where regular milk sells for $\$8$ a gallon, gasoline costs $\$3.60$ a gallon and the median price of a home in 2008 was $\$624,000$ Ñ the second-highest in the nation.

Figure 1.1: Examples of user-generated content from different social media platforms

specific terms which are lacking in regularities and are delivered with an indifferent approach to grammar and spelling. Consequently, traditional content analysis techniques and algorithms that were built for a more formal, contextually rich genre such as newswire, Wikipedia or scientific articles do not effectively translate to informal content on social media.

Figure 1.1 shows examples of user-generated content from popular social media platforms that highlight the informal nature of text in this medium. The difference in the amount of context inherent to the content across platforms is also suggestive of the fact that no one technique will work well for all social media content.

1.2 Dissertation Statements and Contributions

The contributions of this dissertation are in the *aboutness* understanding of text on social media platforms. The work presented in here compensates for the informal nature of user-generated content by examining the usefulness of multiple context cues towards two subtasks of *aboutness* understanding – Named Entity Recognition and Key Phrase Extraction.

The following statements will be made in the course of this dissertation:

1. There is high variability in the content matter, stylistic cues and contextual information within a document on social media that affects the reliability of named entity recognition and key phrase extraction algorithms.

2. Additional contextual information from the social medium and external domain knowledge sources can supplement and improve the reliability and performance of existing algorithms

for named entity recognition and key phrase extraction on informal text from social media. Using multiple contextual cues together lends to reliable inter-dependent decisions, better than using the cues in isolation.

3. Improvements from using multiple and varied types of contextual cues tend to be robust across domains and content of varying characteristics, from micro-blogs like Twitter, social networking forums such as those on MySpace and Facebook, and blogs on the Web.

1.2.1 Broader Impact

The rapid and mass adoption of social media platforms has made available a variety of computationally accessible records of information. Several aspects of user participation on this emerging medium have prompted new tools and methods to facilitate, observe, analyze and promote social interactions. Over the last few years there has been an increasing focus on understanding the online networks that people inhabit; the structure and dynamics of social and information networks and its effect on collective online behavior, information diffusion, expertise sharing, influence formation etc. (Leskovec (69), Wu *et al.* (116), Watts (110)). We are also starting to answer questions about the motivation behind people joining social networks, and gaining insights about the nature of their engagement with the medium, identity formation, peer-to-peer interactions etc. (boyd (13), Back *et al.* (5)).

This dissertation focuses solely on the less thoroughly explored content aspects of user expression on social media platforms. It is one of the first efforts in a systematic investigation into why text on social media is different from the sort of corpora that most text mining applications have focused on; and why automated language understanding tasks need to be revisited.

Building systems for natural language understanding, i.e. interpreting texts in much the same way that a human reader would, is a very difficult problem and requires a large amount of domain and contextual knowledge. Over the last few years, in addition to advances in natural language processing (NLP) applications and frameworks[5], we have also seen an increasing interest from the Semantic Web community in building and exposing machine readable models of a domain (Lee *et al.* (68)).

The *aboutness* understanding algorithms presented in this dissertation take simultaneous advantage of statistical NLP and machine learning algorithms over large corpora and rich models of a domain (ontologies, taxonomies and dictionaries) to demonstrate that despite the inherent informality of social media content, combining multiple contextual cues can yield reliable results for language understanding tasks.

This dissertation is motivated by the fundamental observation that understanding online human social dynamics requires a principled investigation into several participating micro-level variables such as the networks people form, the content they generate and the local contexts they employ. The broader applications of this dissertation are in re-using the findings over the content-level aspects of user expression to explore how they interact with other micro-level variables of a social system. For example, how does the interplay of the *topic* of discussion, *emotional charge* of a conversation, the presence of an expert and connections between participants, together affect emerging social order in an online conversation or explain information propagation in a social network?

[5] LingPipe `http://alias-i.com/lingpipe/`, UIMA `http://uima.apache.org/`, GATE `http://gate.ac.uk/`

1.3 Dissertation Organization

This dissertation document is organized as follows.

Chapter 2 motivates the theory of *aboutness* and the role of context in the semantic interpretation of natural language. This chapter highlights the different modes of context that is generally available to text mining applications operating with social media content and motivates the need to tap into prior domain knowledge for improving performance of text mining algorithms.

Chapter 3 describes the problem of named entity recognition, briefly surveys current trends in NER and details the two proposed algorithms for identification of cultural entities in two different context scenarios – in blogs, that are context-rich environments and user comments on MySpace forums that lack sufficient context. Chapter 4 describes the problem of identifying key topical phrases in text, contrasts the nature of the problem for social media content and presents two algorithms that extract and score topical phrases generated around real-world news-worthy events on two platforms, Twitter and online discussion forums.

In both Chapters 3 and 4, the focus will be on showing how prior domain knowledge and a variety of contextual cues specific to a medium can be used to improve state-of-the-art algorithms.

Chapter 5 showcases two deployed Social Intelligence applications that build over the results of named entity recognition and key phrase extraction algorithms to provide near real-time information about the pulse of an online populace. Specifically, these chapters will describe what it takes to build applications that wish to exploit the wisdom of the crowds, highlighting challenges in data collection, processing informal English text, metadata extraction and presentation of resulting information.

2. Aboutness of Text and The Role of Context

2.1 Characterizing Aboutness

Aboutness, a term aimed at describing what is said in a document, what it is about, its content, subject or topic matter, is a central concept in the fields of knowledge organization and information retrieval both for the human abstraction of material and automatic indexing of content.

The term *aboutness* was introduced in Library and Information Science (LIS) around the 1970s. Fairthorne (Fairthorne (36)) was among the first to define the concept while making a differentiation between 'intentional aboutness' and 'extensional aboutness' of text. He suggested that intentional aboutness is the author's views and intentions of what a document is about, and extentional aboutness is the document aboutness as reflected semantically by actual units and parts of the text.

These aspects of *aboutness* were also echoed in Wilson's analyses of the concept when he outlined the following non-exclusive methods of determining the subject of a document (Wilson

(114)):

(1) identifying the author's purpose in writing the document;

(2) weighting the relative dominance and subordination of different elements in the picture given by reading the document;

(3) grouping or counting the documents' use of concepts and references, and

(4) inventing a set of rules of selection for what are the essential elements (in contrast to the inessential) of the document in its entirety.

Human Comprehension of Content and Aboutness

Perhaps, most relevant to the purposes of an information system is the definition and argument that Hutchins laid out in 1977 (Hutchins (55)). He suggested that the manner in which humans comprehend content and search for information must inform the process of describing *aboutness*.

Human comprehension of content is a complex process. According to Quillian's semantic network theory (Collins and Quillian (27)), readers pre-possess a semantic network of denotative factual and experiential knowledge that is made up of individual units (a unit of thought, parts of a vocabulary, real-world entities etc.) that participate in rich associative relationships with each other. In interpreting the contents of a new document, it has been shown that readers find *aboutness* cues in the initial passages of a document and use their semantic knowledge network to provide the context for its interpretation. In the process of comprehension, they also continuously change or augment their 'state of knowledge' or the semantic network.

Additionally, in consulting an information system in search of a document, a user will typically formulate his needs in terms of what he knows or the knowledge that he pre-possesses. In

other words, he will search using what is part of his semantic network or 'state of knowledge'.

Even though his primary interest might be in the 'new' information of a document, and in the additions and changes that are made to his knowledge map as a result of reading the text, he is typically searching for a document that has some part of his semantic knowledge network as its theme. A reader cannot specify what 'new' information should be in the document or what the rich semantic network of the *information need* is like.

Consequently, Hutchins argues that "the primary aim of indexing (for *aboutness* understanding) is to provide readers with points of contact, leading them from what they know to what they wish to learn" – and should be in terms of the knowledge that most readers of a document pre-possess. *Aboutness* then should be regarded as the pre-supposed knowledge of the text and not necessarily a summary of the total content.

However, since all readers cannot posses equivalent 'states of knowledge', an information system would need decide on a certain level of valid abstraction that will serve the majority of users. While there might be exceptions to this statement when dealing with highly restrictive scientific or organizational documents where readers can be assumed to have an equivalent starting point, the essential features of *aboutness*, Hutchins argued, should strive to capture something that is sufficiently general to the context of most readers' assumed 'states of knowledge'.

Thematic Components of a Document

This leads us to the next important question – what are the parts of a 'generalized' semantic network that relate the document to the context of most readers' assumed 'states of knowledge'?

In document analysis the most important parts of a document's semantic network are considered to be elements of the *theme* that form the knowledge base upon which any 'new' information is built. Typically, thematic elements of a document are bound contextually and textually to the preceding text and assumed as 'given', as opposed to 'new' information.

In linguistics, the topic (or theme) is informally what is being talked about, and the comment (rheme or focus) is what is being said about the topic (Halliday and Matthiessen (46)). The theme or what the text is about, is typically introduced in the initial parts of the document and their surface forms are realized using anaphoric devices such as pronouns or definite articles. The rheme on the other hand, expresses what the author has to say about the thematic elements. Consequently, theme and rheme are often considered as the 'given' vs. the 'new' parts of a document respectively.

For the end goals of indexing for a common state of knowledge, the focus has therefore been on extracting themes or points of contact in text that are a 'given' and are familiar to users and not on the rhemes that potentially present 'new' information that an information seeker might not know to search for.

Subgoals of Aboutness Understanding

A majority of computational methods for identifying theme have focussed on extracting topical elements that represent important areas of text that identify the essential elements in contrast to the inessential ones in the document.

While the level of abstraction chosen for extracting thematic elements depends on the environment and goals of indexing, there are several subgoals of *aboutness* understanding that are

grounded in the field of Information Extraction:

- Named Entity Recognition: recognition of entity names, names of people, organizations, place names etc. that are focal points in a document (Nadeau and Sekine (84)).

- Coreference Resolution: detection of coreference and anaphoric links between text entities. For example, identifying that "International Business Machines" and "IBM" refer to the same real world entity or that the pronoun 'he' in a passage refers to the mention of 'John Smith' in an earlier sentence (School *et al.* (97)).

- Terminology, Key Phrase Extraction: finding the relevant terms in a given document that characterize key topical elements (Turney (107)).

- Lexical Chains, Fact Extraction, Relationship Extraction: identifying a succession of semantically related words in a text that creates a context (lexical chains or phrases) (Barzilay and Elhadad (7)) or extracting relations between entities, such as 'person works for organization'.

These subgoals, individually and collectively in several cases, are aimed at identifying the 'basic theme' of a text – roughly what the authors of those documents presuppose of their readers, and that when indicated as a focal point will in fact point the reader to learn something 'new' about the 'theme' of the document.

2.2 The Role of Context

Fundamental to identifying the core thematic elements in a document is the interpretation of the individual elements in context. The sense or meaning of any thematic element that is extracted is determined principally by its relationship to other elements in the document.

In general, it is challenging, even for humans to resolve the intended meaning of a word in the absence of context. Consider this popular example of the word 'bass', appearing in the following sentences:

(a) I can hear *bass* sounds.

(b) They like grilled *bass*.

The two occurrences of 'bass' clearly denote different meanings. In the absence of contextual information, such as the words co-occurring with the two usages ('hear', 'sounds' vs. 'grilled'), a natural language processing application would find it hard to disambiguate the references to low-frequency tones and a type of fish, respectively.

Automatic semantic interpretation (or understanding the meaning) of natural language, irrespective of the end goals, is characterized by complex interdependent decision-making that requires large amounts of prior background knowledge. Most natural language processing and text mining applications rely on multiple context cues, some of which can be categorized as follows:

Word Associations: In line with the Firthian tradition that "You shall know a word by the company it keeps" (Firth (39)), a common cue used in the interpretation of a word is looking at words that co-occur with it in a corpus. The distribution of words such as, 'saw', 'theater', 'imax', 'plot' etc. found around the words 'Twlight', indicate that the word is used in reference to the movie. Finding such cues in the same sentence as the word 'Twilight', in the same paragraph or elsewhere in a document are indicative of stronger to weaker cues for deducing lexical semantics (meaning of word).

Linguistic cues: The linguistic structure of a sentence as defined by the rules of a language also

serve as strong contextual cues. For polysemous words (those that have multiple meanings), knowing the part-of-speech tag assignment aids in disambiguating the sense with which it is used in the sentence. For example, a word tagged as an adjective is indicative of an opinion-laden expression. Similarly, the presence of noun phrases (NP) in patterns such as "NP such as NP" indicates a hyponym relationship between the two phrases, as in, 'The bowlute such as the Bambarandang' (Hearst (51)).

Syntactic and Structural cues: Structural cues and markups that are internal to a document or a Web page also give additional contextual information. For example, the <title> tag in an html document indicates that the enclosed text summarizes what the document is about. Certain entity types such as dates, phone numbers and email addresses have a regularity in their syntax that can be exploited in their identification. Other structural cues such as topic directories and explicit hyperlinks between documents or Web pages also provide additional context that improves text mining results (Chakrabarti (21)).

Knowledge Sources: With the recent renewed interest from the Semantic Web community in modeling machine-processable domain knowledge, text mining approaches are tapping into contextual cues that are external to a document or corpus. The fundamental idea being that the conceptualization of a domain or language is a valuable prior in processing information about the domain and that such factual knowledge sources encode the 'state of knowledge' in a domain that most readers are likely to pre-suppose. Examples include word or concept definitions in dictionaries (Wordnet (Miller *et al.* (80)), LIWC(Pennebaker and Francis (91))), description of concepts and linking of related articles on Wikipedia pages, descriptions of concepts in domain Ontologies such as the Gene Ontology (Gene Ontology Consortium (43)), or the MusicBrainz Ontology (Swartz (100))

for the music domain etc.

Metadata: Information about the author of the content, his/her age, gender, the date and time associated with a piece of content, the location from where the content originated, are all useful metadata providing additional context to text processing applications. These cues are especially useful when there is insufficient content available to applications. Consider this short forum message, "This is sad news about the King" that is, in fact, a reference to Michael Jackson, the 'King of Pop'. Knowing that the content was generated around the time of his death is valuable in associating the word 'King' with Michael Jackson and not Martin Luther King Jr.

2.2.1 The Formality of Language

One of the fundamental issues when studying the role of context is to determine the degree of context-dependence in a given communication situation. All communication refers to the context to some degree, but in some situations, context will be more critical to understanding than in others (Heylighen (52)). In low-context situations, communication is explicit, stating the facts exactly and in detail; while in high-context situations, communication is implicit, and information is conveyed more by the context than by the verbal expression (Hall (45)).

Quantifying context-dependence in language can tell us how much people rely on explicitly stated context in a medium. More importantly, it can inform how much contextual information is available to automated methods for semantic interpretation.

In order to quantify context-dependence in communication, it is valuable to turn to the notion of the *formality of language*, a fundamental dimension of linguistic communication introduced by

Heylighen and Dewaele (Heylighen and Dewaele (53)). By analyzing the degree of contextual-dependence of words belonging to different grammatical categories, they defined a fundamental dimension of variation, going from the high-context pole ('contextual') where information is conveyed more by the context than by the verbal expression, to the low-context one ('formal'), where communication is rather explicit and overt.

They devised a score that measured the formality of text based on the proportions of deictic and non-deictic words. Words with a deictic function are those that require reference to the spatio-temporal or communicative context to be understood. Pronouns, verbs, adverbs, and interjections are examples of words in this category. Non-deictic words are the class of words that do not normally vary with changes in context. Nouns, adjectives, prepositions and articles are examples of words in this category. The frequency of usage of non-deictic words is then expected to increase with the increasing formality of a text.

They defined the formality of language using the following equation:

$$\text{Formality Score} = (\text{noun frequency} + \text{adjective freq.} + \text{preposition freq.} + \text{article freq.} - \text{pronoun freq.} - \text{verb freq.} - \text{adverb freq.} - \text{interjection freq.} + 100)/2$$

The frequencies here are expressed as percentages of the number of words belonging to a particular category with respect to the total number of words in a document. The score will then vary between 0 and 100% so that the more formal the language excerpt, the higher the formality score is expected to be.

Using this proposed metric for formality, the authors and others found that academic writing and national broadcast reports were more formal than conversations or speech, where much of the

shared context was not brought to the content (Heylighen and Dewaele (53); Nowson *et al.* (87)).

Nat Broadcast Reportage	62.2
Informational writing	61
Academic Social Science	60.6
Writing	58
Professional Letters	57.5
Non Academic Social Science	56.9
Broadcasts	55
Blog corpus	53.3
Scripted Speech	53
Email Corpus	50.8
Critic Music reviews metacritic.com	**50.13**
Yahoo Personals AboutMe	**50.10**
MySpace AboutMes	**50.07**
MySpace - comments on Artist Pages	**50.06**
Prepared speeches	50
Personal Letters	49.7
Twitter	**49.46**
Facebook Wall Posts	**48.2**
Imaginative writing	47
Fiction Prose	46.3
Interviews	46
Unscripted Speeches	44.4
Spontaneous speech	44
Conversations	38
Phone Conversations	36

Figure 2.1: Formality scores of text from various genre

2.3 Communication on Social Media Platforms

Using the same metric for measuring formality of language on social media platforms, I found that the formality scores of user-generated content on social media platforms tended to be more similar to formality scores of speech than writing (see bolded genres in Figure 2.1).

A lower score indicates use of a higher proportion of deictic words (those that require context to be understood) and a lower proportion of non-deictic words (those that do not change with context); indicating that content on social media, on some platforms more so than others, are generally context-poor.

It is not surprising then, that traditional content analysis techniques and algorithms that were built for a more formal, contextually rich genre such as news, Wikipedia or scientific articles do not effectively translate to all types of UGC on social media given that communication predominantly relies more on context than on explicitly stated expressions.

In order to appreciate why communication on social media platforms is less formal, it is useful to look at the properties of the medium and the environment in which users communicate.

- **Informal and Non-policed:** Communication on social media platforms is geared toward interpersonal communication and is inherently less formal. A large portion of language is consequently in the Informal English domain - a blend of abbreviations, slang and context specific terms lacking in regularities and delivered with an indifferent approach to grammar and spelling. This along with the unmediated nature of most platforms lends to data of variable quality.

- **Creativity and Variability:** The variability in the production of language is more pronounced on social media than on other mediums like newswire or scientific articles because of the sheer volume and diversity of participants. A significant portion of users of social media platforms are also in the teen demographic and engage in very creative forms of online expression. Responses to location attribute fields, such as "where do you live? – best place on earth", or the use of slang, as in, "Your song is wicked bad", are fairly common expressions and require special treatment in the automatic interpretation of their parts.

- **Shared Context:** Conversations on social media are typically pockets of self-contained interactions between like-minded people where an author is generally speaking to a known

audience that already has a sense of shared context. Consequently, the generated content lacks explicit context, leaving room for ambiguity in their interpretation.

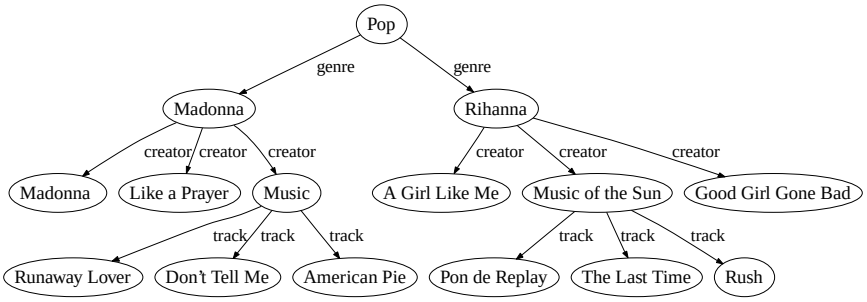
- **Medium Protocols:** Every platform differs in the protocol for expression that it enforces. In some cases, the social medium does not allow for elaborate expression. Twitter, a recent popular micro-blogging platform limits user expression to 140 byte-long content, consequently limiting the amount of contextual information available to systems. Online question-answering and discussion forums on the other hand encourage user discussion, and increase chances of off-topic discussions. The structure of an online conversation on certain platforms is also arbitrary at best. While threads of conversations are valuable context indicators, they are not uniformly traceable across all platforms.

2.4 *Aboutness* Understanding in Informal Text

While there are fewer regularities, higher variability and generally lower context in verbal expression on social media platforms, there are several context indicators such as the structural, syntactic metadata and domain specific information (see Section 2.2) that can complement text mining techniques for *aboutness* understanding.

The focus of this thesis is in aggregating context cues for the semantic interpretation of informal text towards two end goals of *aboutness* understanding: Named Entity Recognition and Key Phrase Extraction.

This work examines the usefulness of several types of contextual cues – those that are internal to a corpus such as word associations; local and global structural and syntactic cues such as pattern




```
I went to <artist id=89>Madge's</artist> concert last night.
<artist id=262731>Rihanna</artist> is the greatest!
I love <artist id=357688>Lily's</artist> song <track id=8513722>smile</track>.
```

Figure 2.2: Snapshot of MusicBrainz, a knowledge base of facts in the music domain and examples of in-line annotations of artist names in user-generated content.

similarities or similar URLs; cues from the social medium such as space and time metadata; and cues external to the corpus in the form of prior knowledge about a domain.

The aim is to demonstrate that these cues when encoded as features can be used together to make reliable inter-dependent decisions that exceed the perfomance than any of them in isolation for the tasks of named entity recognition and key phrase extraction on text from social media.

Of particular focus in this thesis is the use of machine processable conceptualizations of a domain that house a rich knowledge base of facts about the entities in that domain. MusicBrainz for example, is a knowledge base of instances, metadata and relationships in the music domain, that as of 2007 contained 281,890 artists and 4,503,559 distinct artist/track pairs (see Figure 2.2 for a snapshot). It is a non-trivial problem to identify what parts of a domain model are relevant or how the use of attributes of an entity can actually be operationalized to improve a given information extraction end task.

This thesis presents two elegant approaches that combine the bottom-up approach of statistical corpus analysis and the top-down view of what a domain model informs about the statistical

word distributions in a corpus.

As will be shown in the rest of the thesis, particularly for the task of Named Entity Recognition, word distributions studied in light of prior knowledge lend to powerful characterizations of the task, reduce the prohibitive computational costs associated with deep NLP applications and show reliable improvements over baselines that do not account for prior domain knowledge.

Figure 2.2 summarizes the *aboutness* understanding tasks, the types of contextual cues used and user-generated content over which the algorithms proposed in this thesis have been evaluated.

1. Aboutness Understanding Task: Named Entity Recognition
Domain: Identifying Album, Track, Artist names in the Music Domain
User-generated Content: MySpace, Twitter, characterized by short sentences. Example, "I love Lily's song smile"
Context Sources used as Features:
Word Associations: Distribution of sentiment expressions (love, like, hate) and domain words (music, concert, album) in content.
Syntactic Cues: Part-of-speech tags of tokens, Syntactic typed dependencies. For example, a nsubj(loved-8, Smile-5) relationship implies that Smile is the nominal subject of the expression loved.
Page level cues: Artist page URL
Word-level Cues: Capitalization, In quotes
Knowledge sources: MusicBrainz, a knowledge base of instances, metadata and relationships in the music domain. See Figure X for an example.
Resulting Metadata: Annotations against MusicBrainz - "I love <artist id=357688>Lily's</artist> song <track id=8513722>smile</track>."

2. Aboutness Understanding Task : Named Entity Recognition
Domain Focus: Identifying Movie names
User-generated Content: Weblogs, characterized by sufficient context in verbal expression.
Example, "It was THE HANGOVER of the year..lasted forever.. so I went to the movies..bad choice picking 'GI Jane' worse now"
Context Sources used as Features
Word Associations: Distribution of domain (movie, imax, saw, theatre) and entity related words in content, blog URL, title of page etc.
Syntactic Cues: Part-of-speech tags of tokens
Page level cues: Blog, post URL, Title
Word-level Cues: Capitalization, In quotes
Knowledge sources: Infobox definitions for a movie entity from Wikipedia. See Figure X for an example.
Resulting Metadata: Annotations against Wikipedia - "It was THE HANGOVER of the year..lasted forever.. so I went to the movies..bad choice picking <movie id=2434>GI Jane</movie> worse now."

3. Aboutness Understanding Task : Topical Key Phrase Extraction
User-generated Content: Twitter, characterized by 140 byte-long sentences. Example, "RT @WestWingReport: Obama reminds the faith-based groups 'we're neglecting 2 live up 2 the call' of being R brother's keeper on #healthcare"
Weblogs, discussion forums, heavy on inter-personal communication and off-topic chatter. Example, "help, I need to buy this photoshop software today to familiarize myself with the project. I ate eggs last night when I went to Merrill Lynch and I have food poisoning now."
Context Sources used as Features
Word Associations: Distribution of topic categorizations (hashtags such as #healthcare) and co-occurrence based association strengths of n-gram phrases.
Syntactic Cues: Part-of-speech tags of tokens used to favor noun phrases
Word-level Cues: Capitalization, In quotes
Medium specific metadata: Spatial and temporal metadata of content.
Resulting Metadata: Extracting key phrases, for example, "I need to buy the <keyphrase>photoshop software</keyphrase> to familiarize myself with the project. I ate eggs last night when I went to Merrill Lynch and I have food poisoning now."
Sample descriptive, topical phrases in discussion around the 2009 health-care debate reform on Twitter - 'president obama', 'faith based groups', 'soylent green', 'hospice rates posted' etc.

Figure 2.2: Thesis Contributions - *Aboutness* understanding tasks and use of varied types of contextual cues

3. Named Entity Recognition in Informal Text

One of the subtasks of *Aboutness* understanding is identifying thematic elements in text, i.e., identifying portions of text that refer to objects or 'entities' in the world in order to give the reader a sense for what the document contents are *about*.

Named entity recognition (NER) (also known as entity identification and entity extraction) is a subtask of information extraction that seeks to locate and classify atomic elements in text into predefined categories such as the names of persons, organizations, locations, expressions of times, quantities, monetary values and percentages [1]. Identifying mentions of real-world entities in text serves a variety of tasks such as contextual information retrieval, browsing, summarization, tracking and trending their mentions etc.

The focus of this thesis is in the identification of a particular class of rather ambiguous named entities that we call *Cultural entities*. Cultural entities in text are references to real-world entities that are artifacts of culture, such as movie names, book names, video games etc., but are often simply some number of words from the language (The Lord of the Rings, Up, Crash, Twilight,

[1] nerwikipage

Wanted, Today), which makes their identification in text challenging.

The approach proposed in this thesis also deviates from the typical approach of treating NER as a sequential prediction problem, i.e., finding a sequence of words in text that might indicate a named entity and further identifying the type of entity. In this thesis, I take a *spot and disambiguate* approach to NER, where the goal is to spot a known entity (or its variant) in text and subsequently resolve its semantic ambiguity for accurate entity identification.

To this end, we present two disambiguation-intensive algorithms that use local and global natural language cues and sources of prior domain knowledge as contextual information to achieve robust NER across domains and content types.

First, we cover preliminaries relevant to the task of NER and subsequently describe the two algorithmic contributions made by this thesis.

3.1 Preliminaries

The end goals of most NER systems is to add structure to unstructured blocks of text, such as to the following news headline.

> <ENAMEX TYPE="PERSON">Hillary Clinton</ENAMEX> did the morning news circuit talking about <ENAMEX TYPE="COUNTRY">Afghanistan</ENAMEX>. [2]

The metadata resulting from the task of NER can act as a proxy for a document's content and aid applications in information retrieval, indexing, searching, interlinking between documents,

[2] In this example, the annotations have been done using so-called ENAMEX tags that were developed for the Message Understanding Conference in the 1990s.

tracking mentions in documents, text mining etc.

3.1.1 NER is Challenging and Expensive

Entity recognition is a challenging and knowledge-intensive task. Consider personal names for instance. It is generally impossible to distinguish personal names from other kinds of names in the absence of context or domain knowledge – for example, 'May' (person vs. month vs. tentative word); 'Washington' (person vs. location); 'Dr. Pepper' (person vs. product); 'David Copperfield' (person vs. book title) etc. Contextual cues of the kinds that were described in Section 2.2 play a central role in the identification and disambiguation of named entities in text.

Over the last 20 years, the community has made significant progress in this field with several state-of-the-art NER systems for English producing near-human performance. For example, the best system entering the MUC-7 contest scored 93.39% of F-measure while human annotators scored 97.60% and 96.95% (Chinchor (24)) for identifying Person, Location and Organization named entities.

In 2007, Nadeau and Sekine documented approaches and successes in the NER space in a very comprehensive survey paper (Nadeau and Sekine (84)). Here, I borrow from their work to outline three aspects of NER that are relevant to the concepts in this thesis.

3.1.2 Entity Types

In the expression named entity, the word named restricts the task to those entities for which one or many rigid designators, such as words in textual expression, refer to the named object in every pos-

sible world in which the object exists (Kripke (65)). For example, Ford or Ford Motor Company, both refer to the automotive company created by Henry Ford in 1903.

The three most studied entity types are all specializations of 'proper names': names of persons (PER), locations (LOC) and organizations (ORG). Several sub-categories of these entities have also been investigated; such as 'politician', 'teacher' for person entities and 'city', 'country' for location entities. In addition, time and quantity entities have also received significant attention. At least two hierarchies of named entity types have been proposed in the literature. BBN categories, proposed in 2002, are used for Question Answering and consist of 29 types and 64 subtypes [3]. Sekine's extended hierarchy, proposed in 2002, comprises 200 subtypes [4].

3.1.3 Techniques and Approaches

Broadly, NER systems can be classified along one of two types: rule-based systems and learning systems. Rule-based systems are knowledge-engineering intensive approaches where experienced language engineers inspect entities occurring in text and use their intuition to handcraft a generic set of rules that can be used in subsequent entity identification. An example of such a rule would be one that indicates that two capitalized words appearing in sequence are indicative of a person's first and last name.

The recent trend in NER identification is the use of supervised machine learning approaches as a way to automatically induce entity recognition and classification rules using features associated with positive and negative examples of entities appearing in text.

[3] http://www.ldc.upenn.edu/Catalog/docs/LDC2005T33/BBN-Types-Subtypes.html
[4] http://nlp.cs.nyu.edu/ene/

The most common approach to the named entity identification problem in supervised learning (SL) methods has been that of sequence prediction – where the goal is to assign a label to each element in a sequence. For instance, in a sentence like "Paula Adbul gave a speech at the show." that contains one named entity 'Paula Abdul', the goal would be to assign the label 'PERSON' to the entire phrase and not to the individual words.

At the heart of all SL approaches is a system that reads a large annotated corpus where entities are marked and creates disambiguation rules based on discriminative features associated with the entities for the goal of predicting entities in previously unseen text. In other words, the learner has to generalize from the presented data to unseen entities in a reasonable way. One of the serious bottlenecks to SL approaches therefore is the availability of training data with accurate annotations from which to learn the rules.

A variant of SL methods that are termed semi-supervised learning (SSL) approaches address exactly this bottleneck by not requiring a lot of training examples. SSL methods rely on bootstrapping techniques that involve a small degree of supervision, in that they start with a set of highly relevant seed words. For example, a system aimed at learning organization names starts with a small number of organization names provided to it by the user. Then, the system searches for sentences that contain these names and learns patterns such as 'company X located in city'; 'person works for X' that are common to the seed examples. Then, it tries to find other instances of organization names appearing in similar patterns, thereby learning new entity names. The learning process is also reapplied to the newly identified names, so as to discover new relevant contexts. Performance of SSL methods for learning lexicons have been shown to rival that of SL in many cases (for examples, see Brin (16), Collins and Singer (29), (94), Cucchiarelli and Velardi (31)).

An alternate to SL and SSL approaches are unsupervised approaches to entity identification where the goal is to eliminate human supervision. Motivated by the availability of very large collections of texts, efforts in this space include the use of clustering techniques to gather entities from documents clustered based on context similarity, relying on statistics computed on a large unannotated corpus such as distributional similarity measures (e.g., (Turney (104)), (Etzioni *et al.* (34))), and using lexical resources such as WordNet (e.g., (Alfonseca and Manandhar (3))) to seed patterns or contexts in which entities occur etc.

3.1.4 Feature Space for NER

Fundamentally, NER hinges upon recognition and classification rules that are learned from features associated with positive and negative examples of entities appearing in text.

In the learning phase, each named entity is considered as an independent learning instance. Its features reflect properties of this individual instance, such as its type, frequency, corpus level statistics or domain specific properties. Features of all learning instances are encoded as feature vector abstractions. The learner or any rule-based system then operates over the feature vectors to learn recognition ('capitalized words are entity candidates') and classification rules ('entity candidates of length greater than 3 words is organization').

Encoding text as feature vectors involves several pre-processing steps such as sentence segmentation (splitting text using sentence boundaries), part-of-speech tagging (to encode grammatical features), tokenization (breaking a sentence into meaningful units such as words or n-grams), stemming to reduce inflected words to their root form (encoding the words runs, running, ran as their base form run) etc.

The central focus of NER systems is figuring what features to extract from text and what the efficacy of a feature is for identifying the entity type under focus. Nadeau and Sekine outline the most commonly used features for NER along three broad categories (Nadeau and Sekine (84)):

- **Word-level** features that are related to the character encoding in words and describe features such as word case, presence of quotes, numeric characters, punctuation etc.
- **Document and corpus-level** features typically go beyond word-level features and are defined over document content, structure and statistics over large collections of documents (corpora). Examples include features that capture the frequency of an entity's occurrence in a corpus, the appearance of an entity at the beginning of a paragraph etc.
- **List look-up** features also referred to as gazetteer, dictionary, lexicon or domain knowledge features are typically derived from information outside of a document or corpus. If an entity appears in a list of movies it lends certain confidence that at least one of its types will be that of a movie when the entity is identified in text. Models of a domain that encode entities and their relationships in the real-world (Christopher Nolan is the director of the movie The Dark Knight) facilitate richer feature definitions.

Table 3.9 adapted from Nadeau and Sekine's survey (Nadeau and Sekine (84)) shows examples of features in each of these categories.

3.1.5 Evaluation Metrics

NER systems are expected to be robust across multiple domains (e.g., politician and entertainment person names), and across a diverse set of documents (e.g., historical texts, news articles, patent

Features	Examples
	Word-Level Features
Case	Starts with a capital letter, word is all upper case, word is mixed case (e.g., ProSys, eBay)
Punctuation	Ends with period, has internal period (e.g., St., I.B.M.), internal apostrophe, enclosed in quotes, contains hyphen or ampersand
Digit	Digit pattern (two-digit and four-digit numbers can stand for years) cardinal and ordinal, roman number, word with digit (e.g., W3C, 3M)
Character	Possessive mark, first person pronoun, Greek letters
Morphology	Prefix, suffix, singular version, stem, common ending
Part-of-speech	proper name, verb, noun, foreign word (e.g., My/PRP$ dog/NN also/RB likes/VBZ eating/VBG bananas/NNS./.)
Functions over words	Alphabetical, non-alphabetical characters, n-gram, lower case, upper case version, patterns, token length, phrase length
	List-lookup/ Knowledge-derived Features
General List	General dictionary (common nouns), Stop words (e.g., the, of), Capitalized nouns (e.g., January), Common abbreviations
List of entities	Organization, government, airline, educational; First name, last name, celebrity; Astral body, continent, country, state, city
List of entity cues	Typical words in organization (e.g., , inc., associates, ltd.); Person title, name prefix, post-nominal letters; Location typical word, cardinal point
List of emoticons	Informal expressions indicative of preference, emotion (e.g., :(, !)
Models of a domain or language	Wordnet: Words in English organized by their conceptual-semantic and lexical relations (Miller *et al.* (80)), Gene Ontology: a controlled vocabulary of terms for describing gene product characteristics and gene product annotation data (Gene Ontology Consortium (43)).
	Document and Corpus Features
Multiple occurrences	Other entities in the context, Word associations, Upper-case and lower-case occurrences, Anaphora, co-reference
Local Syntax	Enumeration, Position in sentence, in paragraph, and in document, Word presence in page Url
Meta-information	Uri, email header, XML section, bulleted/numbered lists, tables, figures
Statistics over a Corpus	Word and phrase frequency, co-occurrences, multi-word statistics, distributional similarity measures (e.g., TFIDF(Salton and Buckley (96)), Mutual Information (Fano and Hawkins (37)) , (Church and Hanks (25)))

Table 3.1: Adapted from Nadeau and Sekine's organization of common NER features (Nadeau and Sekine (84))

applications, webpages etc.). Several initiatives such as the CoNLL-2003 [5] and the MUC-7 [6] shared tasks concentrate on a set of focus named entities (e.g., persons, locations, organization names) and offer training and test datasets for participants to evaluate their NER system.

Fundamentally, systems are evaluated based on how their output compares with the output of human linguists who identify and classify named entities in text. Systems can be scored at different levels of granularity, for example, if word boundaries were identified correctly, if the type of the entity identified by the system was accurate etc. The interested reader should refer to scoring techniques used for MUC, IREX, CONLL and ACE shared NER tasks.

The most common mathematical definitions for evaluating how a system's performance measures against a human-annotated, gold standard for a statistical classification task like NER are as follows:

- Precision: The number of true positives (correct answers) divided by the total number of answers produced (i.e. the sum of true positives and false positives, the number of elements labeled as a named entity)

- Recall: The number of true positives (correct answers) divided by the total number of possible correct answers (i.e. the sum of true positives and false negatives, the total number of elements that are named entities)

- F-measure: The weighted harmonic mean of precision and recall, where both precision and

[5] `http://www.cnts.ua.ac.be/conll2003/ner/`
[6] `http://www-nlpir.nist.gov/related_projects/muc/proceedings/ne_task.html`

recall are weighted evenly in this equation below.

$$F = 2 * \frac{\text{precision} * \text{recall}}{\text{precision} + \text{recall}}$$

- Accuracy: The number of correct classifications (both true positives and negatives) divided by the total number of classifications made by the system.

$$Accuracy = \frac{\text{number of true positives} + \text{number of true negatives}}{\text{numbers of true positives} + \text{false positives} + \text{false negatives} + \text{true negatives}}$$

Depending the requirement of the applications, these metrics can be appropriately tuned in NER systems. For example, if an application is aimed at tracking entity mentions on the Web, precision might be sacrificed in lieu of high recall. Applications that send out email alters every time the name of an organization is mentioned in the news, will typically aim for higher precision.

3.2 Thesis Focus - Cultural NER in Informal Text

Recognition of named entities is an important and often difficult task to perform in social media content, where general conventions of writing (e.g., capitalization, use of quotations around named entities) are typically relaxed. Accurate identification of named entities also becomes more challenging when there is insufficient context surrounding the discourse; and the language used is in the Informal English domain – a blend of abbreviations, slang and context dependent terms delivered with an indifferent approach to grammar and spelling.

3.2.1 Entity Type - Cultural Named Entities

The focus of this thesis is the extraction of a particular class of entities called Cultural Entities – entities that refer to artifacts of culture, for example, names of movies, TV shows, songs and book titles. This type of entity is most prevalent in content on social media and is currently the focus of a burgeoning area of research.

In addition to referring to multiple real-world entities (e.g. 'The Lord of the Rings' can refer to multiple instances of movies, different video games and a number of novels), Cultural entities are harder to extract because of their use of fragments from everyday language. For example, the movie or book 'Twilight' is also used in reference to the time of day; movies 'Wanted', 'Up' are common parts of speech in English. The song 'Yesterday', could refer to the previous day, a Beatles song (one of 897 songs with that title), or a movie (there are three productions so named).

Moreover, unlike person or location type named entities, possible senses of Cultural entities and the contexts surrounding them tend to change frequently and in relation with current events. Consider the movie 'Star Trek' that is popular as a movie, a TV series and a media franchise but also features in the food cuisine sense. The movie 'The Dark Knight' has in recent times been used in reference to President Obama and his health care reform.

Consequently, the process of enumerating all distinct senses of a Cultural entity does not scale, let alone labeling their varied occurrences in training data.

This poses a significant problem for supervised learning approaches that rely on positive and negative training examples of entity instances in order to learn general recognition rules.

Table 3.2 shows examples that highlight challenges in Cultural movie NER in Informal so-

'It was THE HANGOVER of the year..lasted forever.. so I went to the movies.. bad choice picking <movie id='2331'>'GI Jane'</movie> worse now.'
'I decided to check out Wanted demo today even though I really did not like the movie'

Table 3.2: User posts discussing Cultural entities, highlighting challenges in their identification.

cial media content. In the presence of case information, and contextually relevant words such as 'movies', it is non-trivial to rule out the phrase 'The Hangover' as a non-movie entity.

The focus of this thesis is in developing and evaluating techniques and algorithms for identifying Cultural named entities in Informal text. However, the observations related to sense ambiguities and the contributions made in Cultural NER in light of these challenges are not limited to Cultural entities by any means. Organization names such as Amazon and Apple present similar challenges; so do products like Eclipse and Gladiator (movie names but also names of Nike's shoes).

3.2.2 The 'Spot and Disambiguate' Paradigm

Unlike the predominant approach to NER which is sequential labeling of words found in text, in this thesis we take a *'Spot and Disambiguate'* approach. The basic idea is that we already know what entities and types we wish to find in documents. We begin by trivially spotting or locating the entities (and their variants) in free text and subsequently focus on disambiguating whether the entity mention indeed matches the entity type we are interested in or not.

As a concrete example, assume the NER system has at its disposal a lexicon of movie names that it is interested in identifying in weblogs. Table 3.2 shows an example of a user post that contains phrases that are also movie names. In the *'Spot and Disambiguate'* paradigm, the first

step is to extract these potential candidates ('The Hangover', 'GI Jane' and 'Wanted') by matching what appears in text with entries in the lexicon, and subsequently disambiguate their usage and label only 'GI Jane' as a valid reference to a movie entity.

Formally, the disambiguation step task can be regarded as a *binary classification* problem where each candidate entity (found by a naive spotter) can be labeled 1 if it is a named entity in the target sense of interest (movies in this case); and -1 if it is not.

The algorithms proposed in this thesis fall under the *supervised learning* class of methods, where the NER system learns discriminatory features associated with labeled positive and negative training entity examples and attempts to predict entity types in previously unseen text.

3.2.3 Two Approaches to Cultural NER

In this thesis, we present two approaches to Cultural NER, both addressing different challenges in their identification. Cultural entities display two characteristic challenges related to their sense or meanings – certain Cultural entities are so commonly used that they tend to have multiple senses in the same domain. The music industry is a great example of this scenario where popular themes feature in several track/album titles of different artists. Table 3.3(a) shows examples of such cases – for example, there are more than 3600 songs with the word 'Yesterday' in their title.

Connecting mentions of such entities in free text to their actual real-world references is rather challenging, especially in light of poor contextual information. If a user post mentioned the song 'Merry Christmas', as in, "This new Merry Christmas tune is so good!"; it is non-trivial to disambiguate its reference to one among 60 artists who have covered that song.

(a) Multiple Senses in the same Music Domain	
Bands with a song "Merry Christmas"	60
Songs with "Yesterday" in the title	3,600
Releases of "American Pie"	195
Artists covering "American Pie"	31
(b) Multiple senses in different domains for the same movie entities	
Twilight	Novel, Film, Short story, Albums, Places, Comics, Poem, Time of day
Transformers	Electronic device, Film, Comic book series, Album, Song, Toy Line
The Dark Knight	Nickname for comic superhero Batman, Film, Soundtrack, Video game, Themed roller coaster ride

Table 3.3: Challenging Aspects of Cultural Named Entities

On the other hand, there are Cultural entities that span multiple domains. The phrase, 'The Hangover' is a named entity in the film and music domain. Movies that are based on novels or video games are great examples of such cases of sense ambiguity. Resolving the mention of 'Wanted' in Figure 3.2 as a reference to the video game entity (and not the movie reference) is a challenging task in light of the information available at hand.

In this thesis, we present solutions for Cultural NER for the above two scenarios across multiple domains and text types. We examine two approaches:

1. When a Cultural entity tends to have *multiple senses* in the *same domain*: The focus of this approach is on user-generated content from a single domain (e.g., Music) and in resolving the multiple senses of the entity within the domain (e.g., same track covered by multiple artists in the Music domain). The goal is to annotate the mention of the entity with its referent modeled in a domain knowledge base (e.g., MusicBrainz). For example, in the post below, the goal would be associate the track 'Smile' with artist Lily Allen's work and not with that of other artists who also have songs with the same title. Also, see Figure 2.2 for examples of representation of semantic annotation of entities.

"I love <artist id=357688>Lily's</artist> song <track id=8513722>smile</track>".

2. When a Cultural entity tends to have *several popular senses* across *multiple domains*. The focus here is to look at an uncharacterized data distribution, i.e., user-generated content that is not restricted to any particular domain and disambiguate the mention of an entity among its many senses to identify if it is mentioned in the target sense of interest. For example, if we are interested in identifying movie entities, the goal would be to annotate 'Twilight' as an appropriate movie entity only in the first post below.

   "I am watching Pattinson scenes in <movie id=2341>Twilight</movie> for the nth time."

   "I spent a romantic evening watching the Twilight by the bay.."

Common to both approaches however, is the paradigm in which the solutions are set. Both tasks are approached as a 'spot' and subsequent 'binary disambiguation' problem.

3.2.4 Feature Space for Cultural NER

The goal of the proposed NER algorithms is to *improve* existing state-of-the-art NE learning classifiers for the identification of Cultural entities in Informal text. The contributions of this work are in identifying and using multiple context-inducing features that will be effective for NER in Informal text that generally lacks sufficient context and regularities (see Section 2.2.1 for a discussion on the subject of formality).

In line with the features summarized earlier and in Nadeau and Sekine's survey (Nadeau and Sekine (84)), the proposed algorithms also exploit word-level, knowledge and corpus level features for Cultural NER. Novel contributions of this work are in taking advantage of recently

popular domain models in extracting relevant knowledge-level features for NER while paying close attention to the sense ambiguity challenges posed by Cultural entities.

Consider features extracted using word associations for entity identification cues. Given the nature of Cultural entities (prevalent multiple senses), word association or list-look up features tend to be noisy because statistically significant word co-occurrences are not necessarily strongly related to the target sense of the named entity. For example, statistically significant co-occurrences in a corpus, such as 'game' and 'reboot', 'software' or 'graphics' associate with the video game sense of a 'Transformers' and not the movie. The words 'pictures' and 'romantic' are not necessarily discriminative of the movie 'Twilight' because they appear just as frequently with Twilight in the 'time of day' sense (e.g., "I spent a romantic evening watching the Twilight.."; "These Twilight pictures are rare..").

A central focus of this thesis therefore is in accounting for the *meaning* or *sense* of a named entity (as evidenced by a prior domain model and information in a corpus) in the process of extracting knowledge and corpus-level features.

As will be shown in the rest of this chapter, focusing on sense-relatedness measures in addition to statistical co-occurrence features, such as the TFIDF score of a word, allows us to model the semantic distributional space of a named entity in an elegant fashion. Along with word-level syntactic, structural and linguistic features, the sense-focused corpus and knowledge-level features result in statistically significant and reliable improvements over baseline features for NER, across domains (e.g., music and movie entity types) and document types (e.g., MySpace forums and Weblogs text).

Title: Peter Cullen Talks Transformers: War for Cybertron

Recently, we heard legendary **Transformers** voice actor Peter Cullen talk not only about becoming an hero to millions for his portrayal of the heroic Autobot leader, Optimus Prime, but also about being the first person to play the role of video game icon Mario. But today, he focuses more on the recent **Transformers** video game release, War for Cybertron.

Following are some excerpts from an interview Cullen recently conducted with Techland. On how the Optimus Prime seen in War for Cybertron differs from the versions seen in other branches of the franchise and its multiverse...

Figure 3.1: Showing excerpt of a blog discussing two senses of the entity 'Transformers'

3.3 Cultural NER – Multiple Senses across Domains

Our first contribution to Cultural NER is in the scenario where a Cultural entity appears in multiple senses across domains. When a NER system does not know the domain-level characterization of the data, i.e., it is not clear whether the document is from the 'movie' or 'video game' domain, disambiguating the mention of an entity that in reality, has multiple, ambiguous senses becomes challenging. In this approach, we focus on identifying Cultural entities in weblogs, a platform of expression that tends to attract discussion around varied topics and domains. The excerpt below from a blog mentions the Cultural entity 'Transformers' in its two senses. The goal of our work is to achieve fine-grained labeling of entity mentions, as a movie in the first and as a video game in the second mention.

3.3.1 A Feature Based Approach to Cultural NER

In this work, we propose a new feature that represents the *complexity of extracting* particular entities. We hypothesize that knowing how hard it is to extract an entity is useful for learning better

entity classifiers. With such a measure, entity extractors become 'complexity aware', i.e. they can respond differently to the extraction complexity associated with different target entities.

Suppose that we have two entities, one deemed easy to extract and the other more complex. When a classifier knows the extraction complexity of the entity, it may require more evidence (or apply more complex rules) in identifying the more complex entity compared to the easier target. Consider concretely a movie recognition system dealing with two movies, say, 'The Curious Case of Benjamin Button' a title appearing only in reference to a recent movie, and 'Wanted', a segment with wider senses and intentions. With comparable signals a traditional NER system can only apply the same inference to both cases whereas a 'complexity aware' system has the advantage of treating these cases differently, even though other signals are equal.

We also claim that such a measure is particularly useful for identifying entities that are traditionally hard to extract and which are commonly found in informal texts such as weblog posts and social media in general.

3.3.2 Problem Definition

The task before us is to extract a feature that acts as prior in a binary classifier, assisting it in determining whether a spotted candidate entity is an entity of interest or not. For example, if we are interested in extracting a list of movie names, our proposed feature (for every entity in the list) will assist entity classifiers in deciding if a candidate movie mention spotted in text is indeed a valid movie entity or not.

The proposed feature reflects the 'complexity' of extracting the entity in a target sense (movie,

in this case). Quantifying this feature corresponds to identifying how much support there exists for the target sense of the entity in the corpus. In other words, measuring the context surrounding the entity in a corpus, related to the sense of interest - a task related to the problem of Word Sense Disambiguation (WSD) (Navigli (86)). While WSD techniques aim to disambiguate n senses of an entity, our goal is to extract a measure for a selected target entity type (or sense) and as an ingredient (or a feature) within an entity classification model. An additional difference between our work and WSD techniques is the use of open world assumptions, explained below and one we believe is more realistic when dealing with cultural entity types.

Open vs. Closed World Sense Assumptions: Traditionally, supervised approaches to identify the sense of a word have relied on the knowledge of n classes or senses that a word might belong to. However, unlike person or location type named entities, possible senses of cultural entities and the contexts surrounding them, change more often (consider movies like 'Up', 'Twilight' or 'Crash' that occur in varied contexts). The process of enumerating all distinct senses of a cultural entity does not scale, let alone labeling their occurrences in training examples.

Our approach relaxes this requirement of having a comprehensive knowledge base of senses for the entities, and uses knowledge only about the target sense, and assumes nothing about the other meanings of the entity. We note that subscribing to such an open world assumption presents a harder challenge compared with using a close world assumption. Yet, in the case of cultural entities, this is often the only feasible option.

3.3.3 Improving NER - Contributions

Specifically, we make the following contributions in this work:

1. Proposing an entity specific knowledge and corpus derived 'complexity of extraction' feature that improves existing state of the art entity extraction approaches. The measure estimates the extraction complexity of a particular sense from a given corpus for a set of target entities. This work also provides a new outlook for the role of WSD, modifying its traditional purpose and focus, and using it for entity extraction applications.

2. Proposing and developing a framework for estimating the 'complexity measure' under open-world assumptions. We discuss a two-step framework comprising of graph label propagation and graph clustering that overcomes challenges inherent to the problem definition. The idea is to propagate the influence of the known target sense in a graph built from the corpus distribution and then clustering the results of graph propagation using weights obtained in the propagation step, effectively clustering by the same dimensions of propagation. These two steps have the property of estimating the proportion of the corpus distribution that is related to the target sense, i.e. presents evidence for extracting the entity. We assert that this framework can be used with different underlying algorithmic implementations and consider this an advantage of our approach.

3. Developing algorithms for unsupervised label propagation and clustering, using *Spreading Activation Networks* (Collins and Loftus (28)) and *Chinese Whispers* (Biemann (10)) clustering algorithm. We modify and extend these approaches, with specifically tailored and detailed variations for purposes of our problem definition.

4. Extensive evaluation in the cultural entity and social media space by validating the hypothesis behind the 'complexity measure' and showing improvements in NER accuracy.

We test the usefulness of the extracted feature as a prior with three different supervised ma-

chine learning classifiers, including decision trees (C4.5) (Quinlan (92)), bagging (Breiman and Breiman (15)) and boosting decision trees (Freund and Schapire (42)) over seven different entities of varying extraction complexities, appearing in thousands of blogs. Using well-known linguistic, syntactic and lexical features, we confirm that the extraction of these named entities is in fact hard (average accuracies of 74% and F-measure score 69%, depending on the classifier used). Using a strong contextual entropy baseline and the proposed feature we observe improvements in the order of +10% in accuracy and +11% in F Measure, indicating that the conceptualized 'complexity of extraction' feature is very useful. Our feature did better than the baseline by 2%-3% consistently across classifiers, confirming our methodology for extraction of this feature.

While our evaluations focus on cultural entities in weblogs, its applicability extends to other entity types, documents and NER systems.

3.3.4 Feature Extraction

Here, we describe the two-step framework for obtaining the 'complexity of extraction' feature of target entities in a target sense from a corpus C. For sake of clarity, the algorithms underlying the framework are described for a single entity of interest.

3.3.4.1 Problem Setup

Let $D = \{d_i\}_{i=1\text{to n}}$ be the set of all documents in a corpus in which entity *e* is spotted. These mentions of *e*, denoted as $E = \{e_c\}_{c=1\text{to p}}$, are regarded as valid entity mentions if they appear within quotes or start with capitalized alphabets or are all capitalized and do not occur as part of

another candidate entity (e.g., the mention of 'Up' as a part of 'What Goes Up' is not a mention of the entity 'Up'). Additionally, nothing is unknown a-priori about the distributions surrounding *e* in the corpus, i.e., we do not know what domain the documents belong to, as would be the case in a corpus of general weblogs.

We start with providing the system some information about the sense of interest (e.g., movies) in which we are interested in extracting the entity *e*.

The sense definition of *e* is simply a list of words that sufficiently describe the meaning of *e* in the target sense. In our work, sense definitions are obtained from two sources. The first source of sense definitions, $S = \{s_j\}_{j=1\text{to m}}$, are Wikipedia Infobox entries for the entity *e* - a list of entities that are contextually associated with entity *e* and typically used in reference to e. The second smaller source is a list, S_d, of manually selected words describing the domain of the entity.

As an example, for the movie 'StarTrek', sense hints are $S = \{$ J.J. Abrams, Damon Lindelof, Chris Pine, James T. Kirk, Spock, Karl Urban.. $\}$, entities from StarTrek's movie Infobox[7] and $S_d = \{$movie, theatre, film, cinema$\}$, words indicating the domain of interest (movies).

Let G be an undirected graph built from D (documents mentioning *e*) such that vertices $X = \{x_i\}_{i=1\text{to q}}$ are co-occurring words or contexts surrounding entities *e* in D. Vertices X do not include the entity mention *e* itself, as we are only interested in the contexts *e* occurs in, to adjudicate its sense. Vertices X are either labeled or unlabeled. All x_i belonging to S and S_d are labeled as sense tag vertices or sense hints and denoted by $Y = \{y_g\}_{g=1\text{to z}}$. All other x_i are unlabeled and retained in X. Edges connecting two vertices in G indicate co-occurrence strengths of words in a same paragraph.

[7]Star Trek (film): `http://en.wikipedia.org/wiki/Star_Trek_%28film%29`

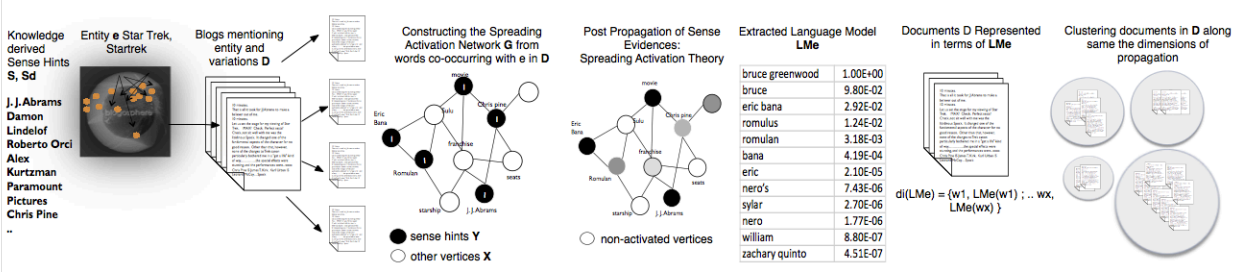


Figure 3.2: Showing steps in the 2-step framework for obtaining the Extraction Complexity of entity *e* in distribution D.

Obtaining the proposed 'complexity of extraction' of an entity *e* can now be phrased as looking for contexts in G (words co-occurring with *e* in D) that are strongly related to the target sense definition of *e* (as encoded by vertices Y in G).

One way of doing this is to propagate the sense definition in vertices Y through G to identify associated contexts in G. *Greater the contextual support for e, the easier is its extraction, and lower its 'complexity of extraction' score.*

3.3.4.2 Approach Overview

We extract our proposed measure using a two-step framework (see Figure 3.2):

1. Sense Label Propagation: To determine how much support exists for the target sense of e, we first propagate the sense definition in Y through weighted edges in G. This activates parts of G that are strongly associated with Y, i.e., extracts a language model of words that are strongly biased to the target sense of *e* as per the sense definition in Y.

However, sense hints in Y might themselves be ambiguous; i.e. they can appear in senses different from the target sense of interest. If the distribution in D supports the propagation of their alternate meanings, it can result in activation of words in G that are weakly related to the

target sense[8]. Since we do not know the other meanings of *e* (because of our open-world sense assumptions), we cannot tell if the extracted language model contains words related to other senses of e. If it does, it will mean an inaccurate estimation of support for our target sense and therefore the extraction complexity.

2. Document Clustering: In order to get the true support there is for the target sense, the extracted language model (comprising of words and their similarity to the target sense of e), is used to learn a classification plane for identifying documents that are more likely to mention entity *e* in the sense of interest and those that are not; effectively clustering by the same dimensions of propagation. Greater the number of documents indicating support for the entity in the sense of interest, lower is its 'extraction complexity' score.

3.3.5 Algorithmic Implementations

Here, we describe the algorithms underlying the two-step framework for our proposed feature extraction – a graph-based Spreading Activation algorithm and Chinese Whispers graph-clustering algorithm. We modify and extend these approaches, with tailored variations for purposes of our problem definition.

3.3.5.1 Sense Label Propagation Algorithm

We used the intuitions behind spreading activation theory to propagate the influence (label) of the sense definition of entity *e* to identify contexts in the corpus distribution that are relevant to the

[8]Character 'Kirk' that is an important sense hint for the entity 'Star Trek' in the movie sense, is also a character in the TV series. Spreading the importance of 'Kirk' in the network tends to activate words that pertain to both the movie and the TV sense (if both contexts are sufficiently present in the graph G).

target sense.

In spreading activation (e.g., (Collins and Loftus (28))(Crestani (30))), label information of vertices in a graph (a spreading activation network or SAN) is propagated to nearby vertices through weighted edges. Typically, multiple pre-selected source vertices are used as pulse nodes to propagate or spread their values in a sequence of iterations to activate other vertices in the graph. The activation process starts with an initialization step where node and edge weights are determined. Subsequent propagation of labels and termination of the spreading activation are controlled by parameters suited to the task at hand. By traversing all links in a network, spreading activation aggregates local similarity statistics across the entire word distribution graph. Our Sense Label Propagation Algorithm, modified for our problem needs proceeds as follows.

1. Pre-Adjustment phase: In this initialization step, we build the undirected graph G (see Problem Setup above), also called the spreading activation network (SAN) from words surrounding *e* in D. We initialize weights for the sense hint nodes and other vertices (those in Y and X respectively), paying particular attention to the weighting of sense hint nodes that might not truly represent the target sense of *e* (e.g. Kirk in 8). This step also initializes co-occurrence edge weights.

2. Propagation phase: Using all sense hint nodes (those in Y) as pulse nodes, we run $|Y|$ iterations of the propagation algorithm. Every iteration propagates the relevance of the pulsed sense hint node y_g through the co-occurrence weighted edges, to increment the scores of vertices in G that it touches. Each of the $|Y|$ iterations adds to the results of the older iterations, effectively propagating the cumulative relevance of all sense hint nodes through G. At the end of $|Y|$ iterations, nodes in G with the highest scores are those that are strongly associated with the initial set of sense hint nodes.

3. Termination Phase: Scores of activated nodes, those whose weights have changed because of the propagation, are normalized to obtain a language model that basically represents words and the strength of their associations with the sense definition of *e*.

Here, we describe each of these steps in more detail.

(1) Pre-Adjustment Phase: Constructing the SAN

Our spreading activation network (SAN) is the undirected graph G built from contexts or words surrounding candidate entities (see Figure 3.3). Given that our goal is to look for contextual support for the target sense of *e* in an uncharacterized distribution D, our SAN is constructed differently from typical word co-occurrence networks:

For each document d_i in D that contains an entity mention e_c

- Extract top X IDF (inverse document frequency) terms in d_i. We choose the IDF statistic since we are interested in pulling out top novel terms co-occurring with *e* in a single document. Let IDF_i denote the top X terms for document d_i.
- If a sense hint s_g in S or S_d is spotted in d_i, it is force-added to IDF_i, irrespective of its IDF score. We do this because we want to take advantage of the already known relevance of the sense hint nodes to the entity *e*, since the goal is to extract support for the target sense of *e*.
- Terms in IDF_i become the vertices in G. An edge is created between two vertices if they co-occur in the same paragraph in any document d_i. The final weight on the edge however is the total number of such contextual co-occurrences in all documents in D. The edges are undirected because all we wish to capture is that two words occurred together in a restricted context.
- Sense hints in IDF_i (those that were force-added) are the vertices $Y = \{y_g\}_{g=1toz} \in G$, labeled as sense hint nodes. All other words in IDF_i are the unlabeled vertices X in G.

Semantics of Initial Node, Edge Weights in the SAN

Weight assignments for the labeled sense hint nodes and other unlabeled vertices are derived based on their known relevance to the target sense of *e*. All sense hint vertices Y in G are initially

assigned a high weight of 1, indicating maximum relevance to the sense of entity *e*. The unlabeled vertices X in G are assigned a low weight of 0.1. The intuition behind the uniform low weight for all X is to allow the propagation mechanism to spread the relevance of the labeled sense hint nodes purely based on the distribution in the corpus. We did not wish to bias this process by using the statistical significance (IDF score) of the word, which in fact, offers no indication of relevance to the target sense of interest.

However, as we pointed out earlier, sense hints themselves are not devoid of ambiguity, i.e., they might be associated with more than one sense besides the one of interest. For example, while 'Kirk' (as a cast) is a strong sense hint for the movie sense of Star Trek, it is just as relevant in the video game and TV series sense. Depending on the underlying distribution in D, propagating the importance of 'Kirk' will activate multiple portions (words) of G, some of which will be unrelated to the target sense of interest.

Since we do not assume the existence of a comprehensive sense knowledge base, we have no way of pre-determining which of the sense hint nodes in Y have multiple senses associated with them. Moreover, an ambiguous sense node affects the purity of the extracted language model only if the underlying distribution in D facilitates the propagation of its meaning.

For example, while 'Kirk' is an ambiguous entity, if the distribution in D is completely skewed toward the movie sense, propagating the influence of 'Kirk' will not activate words in other senses. But, if the distribution in D were to cover both the movie and TV series senses of 'StarTrek', the pulsed sense hint node 'Kirk' would activate parts of the network that are relevant to both the movie and TV series sense. Not only is the underlying distribution of the corpus unknown to us, the lack of a knowledge base of senses prohibits us from evaluating what parts, if any, of the language

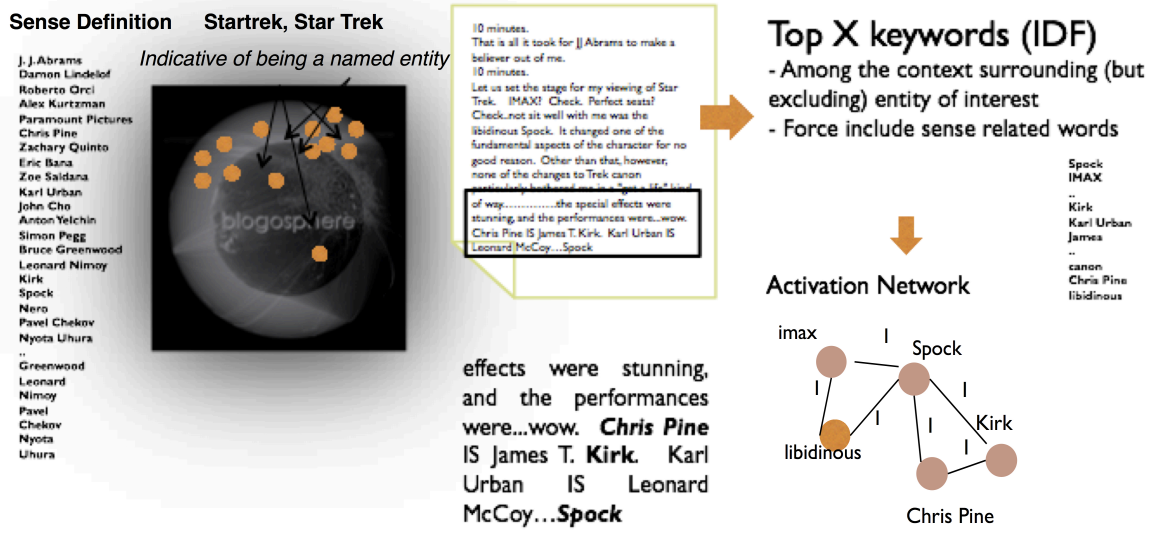


Figure 3.3: Constructing the Spreading Activation Network

model are related to senses other than the target sense of interest.

The goal is to identify which of the sense hint nodes might be relevant in senses that are different from the target sense, with respect to the distribution in D. The relevance of such nodes should then be propagated proportionately lesser, compared to those that are only related to the target sense.

Our heuristic to identify such sense hint nodes makes use of the relatively non-ambiguous sense nodes in S_d – words that are used to indicate the domain of the entity e. The idea is to measure the similarity, $Sim(y_g, S_d)$, between the sense hint nodes y_g and the non-ambiguous domain sense hints S_d, that partially define the sense of interest. Lower $Sim(y_g, S_d)$ scores indicate either insufficient context for sense hint nodes y_g in D or that contexts surrounding them are different from those surrounding the sense hints nodes in S_d.

Pre-selecting strong sense hints: For every sense hint y_g in Y, we issue an independent (non-

cumulative) pulse that propagates its importance throughout G, activating words that are related to it and eventually emerging a language model comprising of words and their relatedness only to y_g. Let us denote a vector constructed from this language model using y_g as the pulse node as $LM(y_g)$. For each y_g, we compute the total dot product similarity of its term vector $LM(y_g)$ with the vectors of all sense hints in S_d that are also in Y (as per Equation 3.1). This allows us to measure how close the sense hint y_g is to the non-ambiguous domain sense hints that partially define the sense of interest. Because the similarity is measured using the extracted language models, it reflects the underlying distribution in D.

$$Sim(y_g, S_d) = \sum_{i=1}^{|S_d|} LM(S_{d_i}) * LM(y_g) \tag{3.1}$$

Higher similarity scores indicate that the sense hint y_g is a strong target sense hint with respect to the distribution in D. If the similarity score is above a threshold γ, the initial weight of y_g (i.e. 1) is amplified by this score. Else, y_g is removed from Y but retained in G as an unlabeled vertex x_i with an initial weight of 0.1. They are not discarded because the similarity was still computed only using a few words in S_d. It is possible that a different S_d would indicate their relevance to the target sense. By retaining the node, its relevance, if any, to the target sense would turn up at the end of the propagation.

Effectively, only those sense hints that are deemed strongly relevant to the target sense are pulsed in the propagation step. After weights of nodes in Y are re-initialized to reflect $Sim(y_g, S_d)$ scores, the sense label propagation algorithm for feature extraction begins.

(2) Propagation Phase: Spreading Sense Evidence

The label propagation algorithm proceeds by propagating the weight (i.e. relevance to target sense) of each labeled vertex y_g through the weighted edges in G. Each y_g contributes to one pulse or iteration that initiates propagation, for a total of $|Y|$ iterations. Starting with every sense hint y_g (in random order) as the anchor, we initiate a BFS walk through G and propagate the weight of y_g. During any iteration, the propagation amplifies the score of any vertex (x_i or y_g) that the walk proceeds through.

For sake of explanation, assume an instance of the BFS walk from vertex i to j in G. The weight of vertex j in iteration $iter$ is amplified as per Equation 3.2:

$$w[j]_{iter} = w[j]_{iter-1} + (w[i]_{iter} * \text{co-occ}[i,j] * \alpha) \tag{3.2}$$

where co-occ$[i,j]$ is the co-occurrence strength or edge weight on the edge connecting vertices i and j and $w[i]_{iter}$ is the weight of node i during iteration $iter$. We chose not to normalize edge weights by the degree of outgoing edges. Our intent is to propagate all the relevance there is in a node. Normalizing by the number of words (degree) it connects to could bias the impact of the pulsing sense hint during its propagation in the entire graph. Note that this choice is specific to our design and in related work we discuss other approaches that could be considered.

In any iteration of the BFS walk starting at y_g, we permit the revisiting of nodes but not the edges, effectively allowing the weight of a node to be amplified by all its incoming edges, i.e. by all co-occurring words. The propagation is controlled by a dampening factor α that diminishes the effect of the propagation the farther a node is from the source sense hint node. Additionally,

a threshold β on the co-occurrence weights also controls when the propagation ceases to continue (for example, if words in vertices i and j co-occur less than 10 times, the evidence in i does not propagate through G via j).

Sensitivity of these parameters to the end result is an important consideration in the use of spreading activation theories. In this work, we experiment with different values for these tunable parameters to suit our end goal.

(3) Termination Phase - Language model

Unlike generic Spreading Activation techniques, termination of our algorithm does not depend on a convergence condition. After $|Y|$ iterations, i.e., after all sense hints in Y have been pulsed the algorithm terminates. At the end of the propagation, all vertices in G have weights larger than their initial weights if they were activated or touched in any of the iterations; and unchanged scores (not activated) otherwise. Scores of all vertices are normalized between 0 and 1 using Equation 3.3, such that nodes that were not activated end up with a score of 0 while the others are proportionately weighted based on the highest activation score received by vertices in G.

$$\text{norm-score(node)} = \frac{\text{prop-score(node)} - \text{prop-score(G)}_{min}}{\text{prop-score(G)}_{max} - \text{prop-score(G)}_{min}} \tag{3.3}$$

where $prop - score(node)$ is the activation score of a node post spreading activation; $prop - score(G)_{min}$ and $prop - score(G)_{max}$ are the minimum and maximum activation scores of nodes in G post spreading activation. The language model for entity *e*, denoted as LM_e, comprises of the words in G with normalized activation scores > 0. Given our node weighting and propagation

through a corpus distribution, the final score of a word in LM_e is proportional to its relevance to the target sense with respect to the underlying corpus.

Extracted Language Models

Figure 3.4 shows the extracted LM_e for the movie entity 'The Dark Knight', ordered top-down, left-right by normalized activation scores. The entity appeared in over 750 blogs (size of D) in our general weblog corpus. We extracted top 40 words from every blog (to construct the SAN), used a threshold γ of 30% similarity for the pre-selection of strong sense-hint nodes, a decay factor α of 0.3 for the propagation and a co-occurrence threshold β of 10. [9] A total of 12 sense hint nodes were pulsed to obtain a language model of 98 words.

teasers	theaters
christopher	comics
gotham city	animated
harvey	films
nolan	watchmen
oldman	released
aaron	knight
dent	awards
nolan's	revenge
eckhart	cameron
two-face	gigandet
gotham	thunder
christian	mtv
freeman	imdb
christopher nolan	wolverine
caine	transformers
harvey dent	terminator
morgan freeman	theatre
christian bale	cinema
aaron eckhart	pixar
michael caine	tropic
gary oldman	theater
bale	reboot
oscar	avatar
villain	samberg
ledger's	pattinson
heath	movie
ledger	mcg
superhero	academy awards
joker's	70mm
dawes	asylum
joker	film
best supporting actor	mite
posthumous	arkham
heath ledger	villains
action	superman
opening	caped
dvd	batman's
millionaire	riddler
dark	batmobile
bros	nicholson
imax	crusader
movies	bat
sequel	morrison
hbo	robin
slumdog	jhw3
begins	batman

Observations: The example clearly shows that the extracted language model is biased to the target sense and also that it generalizes well to the domain. Not only are we seeing entities related to 'The Dark Knight', we are also seeing words pertinent to the general movie sense.

The LM_e is also loosely indicative of the support there is for extracting the entity in the movie sense. If we were not able to extract a language model any larger than our initial sense hint nodes, it is an indication of poor support for the entity in the target sense. We acknowledge that depending on

[9] These parameter values were based on experiments and chosen to yield a conservative size language model.

the parameter settings, we might be extracting a conservative LM, i.e. we might be missing relevant words by not sufficiently propagating evidences. We argue that this only leads to a conservative estimate of our proposed feature and can be adjusted depending on application requirements.

3.3.6 Clustering Document Evidences

In the Sense Label Propagation algorithm, we used seed sense hints to propagate sense relevance in a graph of words to identify strong associations. Such a spreading activation lends one type of clustering by separating words strongly related to our sense from those that are not. We also used a set of heuristics to eliminate the effect of sense hints that might possibly appear in multiple senses and pollute our extracted language model, which we wish to bias to our target sense.

While the extracted language model already, albeit loosely, indicates support for the target sense of *e*, it may still be insufficiently accurate. There will inevitably be cases where different senses of an entity are so closely related in terms of overlapping vocabulary, that they will be hard to separate in the underlying distribution. Given that we do not know all senses of entity *e* a-priori, it is harder to filter such overlaps during sense label propagation.

Consider the book and movie sense of entity 'Twilight' as an example, that share many common words both in cast names, authors and plots. In such cases, it is not easy to bias the extracted language model to one target sense, especially with no prior knowledge of the other senses.

We found that a second level of clustering documents in D, along the same dimensions that were used for propagation allows a further separation; and yields a more accurate representation of how much support exists for an entity in a target sense.

Document Representation

We represent documents in D as a vector of terms with the condition that terms in the vector are only those in the document that are also present in the extracted language model of the entity, LM_e (see Equation 3.4). Weights of terms in the term vector are obtained from the extracted language model, and represent the term's relatedness to the target sense.

$$d_i(LM_e) = \{w_1, LM_e(w_1); w_2, LM_e(w_2)...; w_x, LM_e(w_x)\} \tag{3.4}$$

where w_i are words overlapping with document d_i and LM_e and $LM_e(w_i)$ is w_i's relatedness to the target sense, as obtained from the extracted language model LM_e.

A document's relatedness to the target sense is then proportional to the relatedness strengths $LM_e(w_i)$ of the words w_i in it.

Also, documents in D that have no representation in the extracted language model LM_e, will have an empty term vector and can be discarded from the analysis. The reader should note that the semantics of what a document contains is not the traditional statistical TFIDF metric, but the notion of relatedness to the target sense of entity *e* – a property that allows us to implicitly model sense relevance of an entity's distribution space. Moreover, not only has the dimensionality of a term vector vastly reduced (from all terms in a document to those overlapping with LM_e), the

number of documents under consideration also potentially drops because of no overlapping words in the document and LMe. These effects are important when dealing with volumes of data and contexts, which is often the case in social media content.

Graph-based Clustering: Chinese Whispers

In order to cluster documents along the language model dimension, we used an implementation of a graph-based clustering algorithm called Chinese Whispers (CW) (Biemann (10)). CW is a clustering algorithm that clusters undirected, weighted graphs and has a run-time complexity that is linear in the number of edges, which makes it a very efficient algorithm. Essentially, CW places every node into its own unique cluster; sorts the nodes in random order and assigns each node to the most popular cluster in its neighborhood, where popularity is defined as the sum of the node weightings in that cluster. This step is repeated until a fixed point is reached. Since CW is inherently non-deterministic, for some graphs no fixed point is reached and the algorithm oscillates between a few different graph clusters.

In our implementation of CW, every node is a document, represented by its term vector $d_i(LM_e)$. Edges represent the similarity between the documents in terms of the dot-product similarities of their term vectors. Documents are initially placed in separate clusters, and over 250 iterations of the algorithm (CW is known to plateau for less than 250 iterations for several NLP tasks), documents are grouped together in clusters based on their average maximum similarity (in terms of their term vectors) with documents in other clusters.

3.3.7 The 'complexity of extraction' score

Clustering documents along the extracted language model dimension lends interesting properties to the extracted clusters for our end task.

1. By grouping documents that have common words, we are now able to better separate evidence of multiple senses that found their way into the language model LM_e. As an example, while the extracted LM_e for say the movie 'Star Trek' could have had words both from the video game and movie senses, documents containing the video game sense words are now separated from those containing words in the movie sense (since they are associated with different words and relatedness to sense weights originating from the previous activation step).

2. A cluster's relatedness to the target sense, (by extension of a document's relatedness to the target sense), is then as high as the relatedness strengths of the words in it (see Equation 3.5). This means that if we ordered clusters by their relatedness scores, high scoring clusters have a greater chance of containing documents that mention the entity *e* in the target sense, compared to low scoring clusters (those that have words with less relevance to the target sense).

$$\text{relatedness-score}(C_k) = \sum_{i=1}^{n} count(w_i) * LM_e(w_i) \tag{3.5}$$

where w_i are all words in documents in cluster C_k, $count(w_i)$ is the number of times the word occurs in documents in cluster C_k and $LM_e(w_i)$ is the word's relatedness to the target sense, as obtained from LM_e.

Using these properties, we employ a simple heuristic for counting how many documents in D might indicate a strong support for extracting *e* in the target sense of interest. More the support,

lower is the 'complexity of extraction' of *e* in the target sense. We calculate the average score of all clusters, $avg(C)$, and pick those whose scores are greater than $avg(C)$. Let us denote a set of such clusters as C^*. The complexity of extracting an entity can then be computed as the proportion of all documents that mentioned *e*, i.e. $|D|$, and the number of documents in clusters C^*, that have a high likelihood of mentioning *e* in the target sense (see Equation 3.6).

$$\text{'complexity of extraction' of } e = \frac{1}{\frac{|C^*|}{|D|}} \tag{3.6}$$

We note that with this heuristic, we may lose some relevant support in terms of documents that might be related but score below the $avg(C)$ threshold. However, this will be common across the entities that we wish to extract and therefore will preserve the relative 'extraction complexity' ordering of entities, which is our ultimate goal.

3.3.8 Experimental Evaluations

The goal of our experiments is two fold:

1. To judge if the unsupervised algorithms described above are indeed effective in extracting the proposed feature.

2. To validate our hypothesis behind the 'complexity of extraction' measure by evaluating if it *improves* accuracy of named entity classifiers.

3.3.8.1 Efficacy of the Algorithms

For this experiment we took a list of target entities in the movie domain and used our 2-step unsupervised algorithm and calculated their 'extraction complexity' scores in a general blog corpus. The idea is to see if an ordering of their complexity measures match general intuitions about the difficulty in extracting the entities. For instance, one would expect that movies like 'Crash', 'Up' or even 'Star Trek' that have varied senses will be harder to extract than movies like 'The Time Traveler's Wife'.

We chose the movie domain since it receives a lot of attention in social media with people talking about, reviewing and commenting on movies. We leveraged a dedicated real-time blog crawling engine that covers more than $250K$ general blogs and injects millions of posts and feeds per day (Hurst and Maykov (54)). In this work we used a dump of around $2,130,000$ general blogs. We used a general blog corpus instead of a domain specialized so that there is a higher chance of multiple sense discussions of an entity that will allow an accurate characterization of an entity's 'extraction complexity'.

Experiment Setup, Results

Our test set *e* is a list of 50 movie entities released in the same time period as the blog posts were authored. For each entity, we obtained their sense definitions from Wikipedia Infoboxes. The domain sense definition S_d was a list of four words {movie, cinema, theatre, film} for all the entities. For every movie entity *e* in E, we identified all documents D, in the large blog corpus that mentioned the entity *e* (see Figure 3.5). A document mentioning *e* was included in D only if *e* appeared within quotes or started with capitalized alphabets or was all capitalized. We also placed a restriction that *e* did not occur as part of another candidate entity, i.e. was not surrounded by

other capitalized words or those in quotes.

In this process, some movie entities were eliminated because of insufficient representation in the large blog corpus. While the 'complexity of extraction' can be high if there is no representation in the corpus, we wanted to ignore this bias and focus only on cases where there was sufficient representation but possibility of multiple senses. This left us with a total of 8 movie entities that appeared at least in 500 blogs each.

For every entity *e* and its corresponding document set D, we constructed the SAN G from contexts surrounding *e* in D. Using our two-step algorithm, we propagated the sense definitions of *e* (S and S_d) in G to extract the language model LM_e and finally clustered documents in D along the language model dimension to obtain the 'extraction complexity' measure for *e*. In all experiments, we extracted top 40 words from every blog in the construction of the SAN, used a threshold γ of 30% similarity for the pre-selection of strong sense-hint nodes, a decay factor α of 0.3 for the propagation and a co-occurrence threshold β of 10.

Figure 3.5 shows the 'complexity' of extracting the entities in the movie sense as calculated by our algorithm. It also shows other 'known' senses that exist for the entity as listed on Wikipedia disambiguation pages (some might be more popular than others in social media chatter). The ordering of entities by their 'complexity measure' matches general intuition that movies like 'Twilight', 'Up' and 'Wanted' will arguably be harder to extract than a movie like 'Angels and Demons' – clearly indicating that our approach for computing 'extraction complexity' is effective. A note on the confidence of these scores is presented along with discussion of Figure 3.6.

Entity *e*	Possible senses from Wikipedia disambiguation pages for entity *e*	Computed Complexity of Extraction
Twilight	Novel, Film, Short story, Albums, Places, Comics, Poem, time of day…etc.	0.400
Up	Relative direction, Film, Series, Album, abbreviations for universities, places …etc.	0.352
Wanted	Film, Music, Literature, Video games, common verb in English.	0.161
Star Trek	TV Series, Media Franchise, Text game, Film, dialect, cuisine, ..etc.	0.114
Transformers	Electronic device, Album, Song, Toy Line, Film, Comic book series…etc.	0.085
The Hangover	Film, Album, Band, Song, unpleasant physiological effect that can follow consumption of alcohol and other drugs.	0.072
The Dark Knight	Nick name for comics superhero Batman, Film, Soundtrack, Video game, Themed roller coaster rides.	0.070

Figure 3.5: Entities, their known senses from Wikipedia, and their computed extraction complexities

3.3.8.2 NER Improvements

The underlying hypothesis behind this work is that knowing how hard it is to extract an entity will allow classifiers to use cues differently in deciding whether a mention spotted is mentioned in a valid entity in a target sense or not. In this second set of experiments, we measure the usefulness of our feature in assisting a variety of supervised classifiers.

Labeling Data

We randomly selected documents one after another from the pool of documents D collected for all entities *e* in Experiment 1 and labeled every occurrence of entity *e* and its valid variations in a document as either a movie entity or not. We labeled a total of 1500 spots. We also observed a 100% inter-annotator agreement between two annotators over a random sample of 15% of the labeled spots, indicating that labeling for movie or not-movie sense is not hard in this data.

Boolean Word-level Features: First letter capitalize, All capitalized, In quotes
Boolean, Contextual, Syntactic Feature: POS tags of words before and after *e*
Knowledge features: Presence of words from sense definition S and Sd in the same blog, same paragraph as e, title of post, URL of post, blog URL Presence of words from the extracted language model of e, $Lm_{e,}$ in the same blog, same paragraph as e, title of post, URL of post, blog URL
Priors: Baseline 'Contextual Entropy' prior Proposed 'Complexity of extraction' prior

Figure 3.7: Features used in judging NER improvements

Figure 3.6 shows statistics for the percentage of true positives found for each entity. The percentage of entity mentions that appear in the movie sense implicitly indicates how much support there is for the target movie sense for the entities.

Labeled Entity (and variations)	% of True Positives
Up	0.03
Wanted	0.15
Twilight	0.23
Star Trek	0.52
Transformers	0.57
The Dark Knight	0.95
The Hangover	0.97

Figure 3.6: Labeled Data

The reader should note that this order closely matches the extraction complexity ordering of the entities (shown in Figure 3.5) – an indication that the approach we use for extracting our feature is sound. In the process of random labeling the entity 'Angels and Demons' received only 10 labels and was discarded from this experiment.

Classifiers, Features, Experimental Setup:

We used 3 different state-of-the-art entity classifiers for learning entity extractors – decision tree classifiers, bagging and boosting (using 6 nodes and stump base learners for boosting) (Freund and Schapire (42)), (Breiman and Breiman (15)), (Quinlan (92)). The goal of using different classification models was to show that our measure is useful with different underlying prediction

approaches rather than for the purpose of finding the most suitable classifier. We trained and tested the classifiers on an extensive list of features (Figure 3.7).

We used two well-known features – word-level features that indicate whether the spotted entity is capitalized, surrounded by quotes etc., and contextual syntactic features that encode the Part-of-speech tags of words surrounding the entity.

We also used knowledge-derived features that indicate whether words already known to be relevant to the target sense of the entity (initial sense definition of *e* obtained from Wikipedia) are found in the document, surrounding paragraph, title of post or in the post or blog url. The intuition is that the presence of such words strengthens its case for being a valid mention. We also encoded a similar knowledge-level feature using the extracted language model LM_e to test the usefulness of the words we extracted as relevant to the target sense of the entity.

Baseline:

In addition to the basic word-level and syntactic features, we also measure the usefulness of our proposed feature against a *'contextual entropy'* baseline. This baseline measures how specific a word is in terms of the contexts it occurs in. A generic word will occur in varied contexts, and will have a high context entropy value (Heafield (50)). High entropy in context distribution is an indication that extracting the entity in *any sense* might be hard. This baseline is very similar in spirit to our feature, except that our proposed measure identifies how hard it is to extract an entity in a *particular* target sense.

We evaluated classifier accuracies in labeling test spots with and without our 'complexity of extraction' feature as a prior. Specifically, we used the following feature combinations (refer to Figure 3.7 for a description of the feature types):

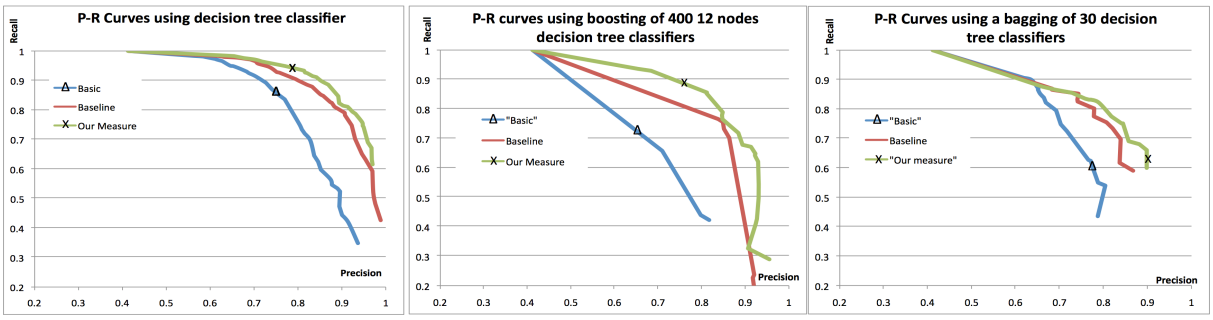


Figure 3.8: Overall P-R Curves using Decision Tree and Boosting Classifiers using 10 fold cross validation.

a. Basic features: word-level, syntactic, knowledge features obtained from the sense definitions S and S_d.

b. Baseline: Basic + 'contextual entropy' feature as a prior.

c. Our measure: Basic + knowledge features obtained from the extracted LM_e + 'complexity of extraction' feature as a prior.

Results:

Figure 3.8 shows the precision-recall curves using the basic, baseline and our proposed measures for entity classification using the decision tree and boosting classifiers. We verified stability of these results using 10 fold cross validation. We see better performance of our measure compared to both the basic setting and the strong 'contextual entropy' baseline. Notably, there is overwhelming improvement in entity extraction over traditional extractor settings, i.e., over basic features. The stability of the suggested improvement is also confirmed across both classifiers.

We see significant improvements using the proposed feature and now turn to confirm that this is indeed a consistent pattern. Here, we show the averaged performance of binary classification over 100 runs, each run using different and random samples of training and test sets (obtained from

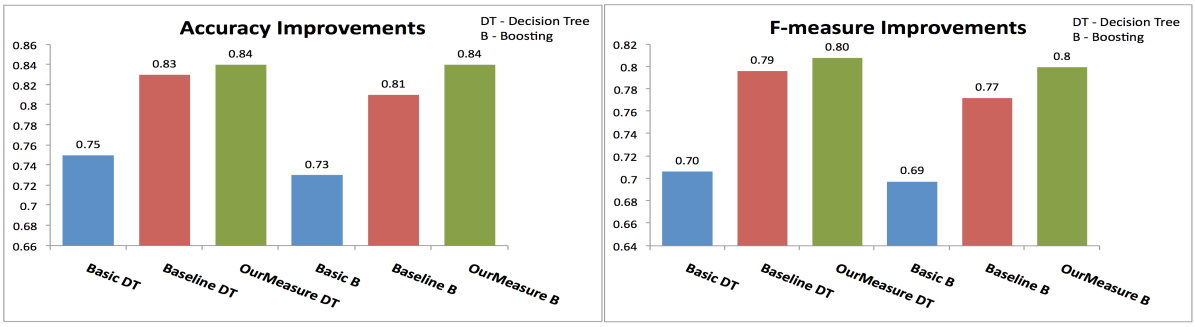


Figure 3.9: Overall F-measure and Accuracy improvements across 100 runs.

$50 - 50$ splits).

We measured the F-measure and accuracy of the classifiers using the basic, baseline and our proposed measure features. Accuracy is defined as the number of correct classifications (both true positives and negatives) divided by the total number of classifications. We use accuracy to represent general classification improvement – when we care about classifying both the correct and incorrect cases. The F-measure is the standard harmonic mean of precision and recall of classification results and we use it to represent information retrieval improvement – when we only care about our target sense. We report both of these metrics here for consistency with past literature.

In Figure 3.9 we show averaged F-measure and accuracy improvements across 100 runs of both the decision tree (DT) and boosting classifiers. These results clearly show the usefulness of our measure compared to the basic and baseline measures. The fact that these results were over 100 runs of both classifiers strengthens the consistency of our proposed feature's value.

The semantics of the strong 'contextual entropy' baseline:

While the contextual entropy computation that we use as strong baseline fits well with our overall approach it suffers from clear limitations for our needs. The 'contextual entropy' baseline only tells us *'that'* there is a varied context surrounding the entity in a corpus. It does not tell us if the

varied context is with respect to a target sense.

It is possible that the words/contexts are varied, but are all in the same target sense (something our measure will capture by effect of spreading activation, but the entropy measure will not). Also, suppose that we operated on a different corpus, one that is skewed toward the video game sense and we are interested in extracting 'Star Trek' in the movie sense. It is possible that contexts surrounding the entity in the video game corpus are not varied, thereby yielding a low entropy score for the entity. For the purpose of extracting 'Star Trek' in the movie sense, this score is meaningless since the entity's extraction in the movie sense from a video game corpus will expectedly be high.

While we are already seeing cases of our measure being better than the contextual entropy baseline, we believe this distinction will be apparent when we expand our experiments to different types of entities and corpus distributions.

The fact that the 'contextual entropy' baseline, that in some sense encodes the extraction complexity is improving NER is indicative of the usefulness of this feature for NER – confirming our hypothesis about engineering 'complexity aware' classifiers. The fact that our proposed feature does better than the baseline shows that our method for measuring complexity of extraction is sound and captures the intuition well.

Improvements at the Entity level

Having established that our proposed feature allows NER classifiers to generally make better judgments, we conducted a few more experiments drilling down to the entity level. The goal was to get some insight on how the proposed feature helps the classification of entities of varying extraction complexities (easier to harder as shown in Figure 3.5).

In this evaluation, we learn a boosting classifier (using six-node decision tree as base learner

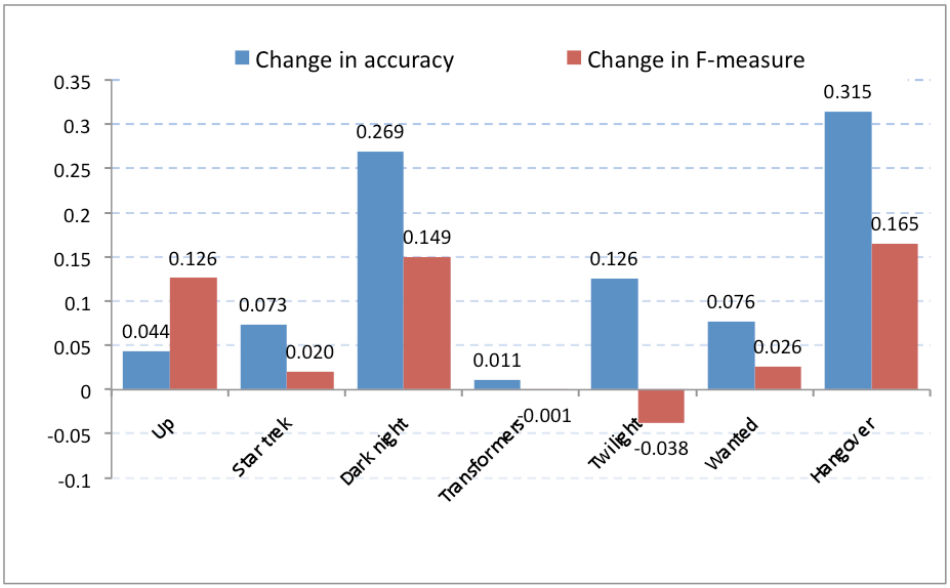


Figure 3.10: F-measure and Accuracy improvements at the entity level.

over 400 iterations) and show the change in accuracy and F-measure obtained from 10 fold cross validation of binary classification. The change in measurements is against the basic feature settings. Given the small number of labeled data points for each entity, we could not repeat evaluations over different runs. This also means that we need to interpret results with caution.

Generally, we see improvements in classification accuracy throughout and positive impact in F-measure in all entities except one – the movie, 'Twilight'. Notably, very high increase in the F-Measure is observed for the movies Up $(+12.6\%)$, The Dark Knight $(+14.9\%)$ and The Hangover $(+16.5\%)$, and significant increase in classification accuracy for The Dark Night $(+26.9\%)$ and The Hangover $(+31\%)$.

The 3% drop in the F-measure of entity 'Twilight' sparked our curiosity and we went drilling further. We found that while the extracted language model for 'Twilight' had words that were certainly related to the movie sense, the words were also related to several other uses of 'Twilight'.

Below are some examples that show words that were part of the language model extracted for

the 'movie' sense but are also common to other senses of Twilight.

> *"I spent a **romantic** evening **watching** the Twilight.."*; has the word 'romantic' that is also the genre of the movie Twilight and the word 'watching' that is strongly associated with watching a movie.
>
> *"here are **photos** of the Twilight from the bay.."* has the words 'photos' in the language model that is relevant to the target sense as in this comment, 'red carpet photos of the Twilight crew'
>
> *"I am re-reading **Edward Cullen** chapters of Twilight"* mentions Edward Cullen, a cast in the 'Twilight' movie.

Since the extracted language model is used to derive the knowledge features, i.e., used as look-up features in the content, title and url of the post and blog where the entity appears; the weak discriminatory nature of these words contributed to a poor prediction model for the target sense and therefore the lower F-measure for the classification. For entities such as 'Twilight' that are characterized by ambiguous word associations, biasing the language model to a particular target sense is rather challenging. In such cases, that are identifiable by poor performance of the proposed feature or a manual examination of clusters (step 2 of the framework), NER systems could fall back on the basic feature settings.

3.3.9 Related Work

3.3.9.1 Characterizing Extraction Difficulty

Our measure for characterizing extraction difficulty, i.e., how much support there is for an entity in a particular context and within a distribution, maps closely to several measures. Simpler ones like

frequency of the entity in a corpus to characterize extraction difficulty are less useful for our task because of the cultural nature and therefore frequent occurrence of the entities we are interested in.

A related concept, *cohesion*, has also been used in text analysis to describe topical relationships between various units of text (Halliday and Hasan (47)) and has been shown to be a useful prior in improving blog retrieval (He *et al.* (49)). Cohesion is related to our work in that word associations appearing in cohesive units of text will tend to bias to the same sense.

Popular methods of determining the coherence of an entity's distribution use information-theoretic divergence measures to compute the similarity between distributions of language models formed by the contexts the entity occurs in (e.g., using Jensen-Shannon divergence) ((71)). Since we do not assume the availability of all n senses of a cultural entity, we cannot trivially use such measures to estimate coherence in one target sense. In our work, we start with assuming some strong associations (sense definitions) and find stronger ones biased to our sense of interest to characterize what proportion of contexts surrounding our entity 'tightly stick together'.

However, a weaker estimate of coherence is *contextual entropy* that characterizes how varied the contexts surrounding an entity (Heafield (50)). More varied contexts of an entity imply lesser coherence and higher extraction complexities in any sense. While this measure is a good approximation of extraction complexity (as we found with its performance as a baseline in our work), it still does not entirely meet our needs. This measure only tells us 'that' the contexts are varied, it does not tell us if the contexts are related to our target sense. Our goal is to quantify an entity's extraction complexity in a particular target sense rather than provide an indication of its perplexity.

3.3.9.2 Estimating Extraction Difficulty

We used a combination of graph-based spreading activation and clustering algorithms in a two-step framework to gather support in terms of strong associations related to our target sense.

There has been a lot of prior work in gathering strong associations for a word. The most commonly used dependency relationships to find strong associations within a distribution are word co-occurrences (Weeds and Weir (111)). The idea is to predict strengths of word-associations from the relative frequencies of the common co-occurrences of words in large bodies of text (Lin (70)). Information theoretic measures of mutual information that encodes the mutual dependence between two words has also been used effectively for gathering word associations (Church and Hanks (25)).

Our use of Spreading Activation is grounded in the use of co-occurrence based word dependencies to find strong associations. However, our goal was to not simply extract strong associations in a distribution, but to do so while biasing it to our target sense. While 'Angels and Demons' could have been strongly associated with words like 'book' and 'reading', our goal was to find associations only pertaining to our target movie sense.

Spreading Activation allows an elegant way of doing this by allowing us to restrict the walking only to those portions of the network that are relevant to our target sense. It also allows us to consider effects of transitive dependencies between words. By traversing all dependencies starting from n words, spreading activation aggregates local similarity statistics across the entire word distribution graph.

This fits with our goal of *finding strong associations of a 'sense' and not just a 'word'*.

Another reason behind the popularity of graph based spreading techniques is the flexibility of finding associations based on a variety of dependency relationships between words – not limited to co-occurrence based relationships. In addition to encoding co-occurrence dependencies (see examples in (Crestani (30))), there has been work on encoding dependencies based on syntactic relations between words (e.g., (Padó and Lapata (88))); and using dependencies present in knowledge models like WordNet, (e.g. ((18)) , (Tsatsaronis *et al.* (103))). While we explicitly encode only co-occurrence dependencies in our Semantic Activation Network, we also use knowledge from Wikipedia to define our sense of interest that guide the activation of our network to discover strong associations.

Related to our method of spreading cumulative evidences in a network of words are association discoveries that are made using random walks in graphs (see survey in (Berkhin (8))). Random walks in graphs for finding associations are similar to spreading activation in networks, in that, they are global and iteratively propagate weights through a significant portion of a network. Particularly, work in (Ramage *et al.* (93)) uses an application of random walk Markov chain theory to measure the lexical semantic relatedness between words. A graph of words and concepts is constructed from WordNet and the random walk model starts from a word of interest and walks the graph by stochastically following local semantic relational links. They compute distinct, word-specific probability distributions over how often a source word visits all other nodes in the graph when 'starting' from a specific word. Relatedness of two words is computed as the similarity of their distributions extracted from the walk.

This is somewhat similar to what we attempted to do, but has variations given our problem definition. Our goal is to find 'sense' associations. This would translate to using random walks

from multiple starting points while restricting what links and how far to 'walk' (based on the thresholds we used). There are also differences in the semantic links they encode (from WordNet) and the co-occurrence dependencies we employ. A task worthy of investigation is to compare associations obtained from adapted random walk models to those we found via spreading activation techniques in this work.

3.3.9.3 Cultural Entity Identification and WSD

Cultural entity identification is an upcoming area of research. Evri, a discovery engine on the Web achieves accurate annotation of these entities, although no information is available on their system. It also presents an interesting use case of browsing resources associated with these entities in their cultural context.

Word Sense Disambiguation (WSD) is a closely related subtask to entity identification, that concerns itself with finding what meanings an entity's mention is associated with. Supervised techniques for WSD are very restrictive for our goals in the information about senses of a word they require (Navigli (86)), since we do not have a comprehensive list of possible senses a cultural entity might occur in. Unsupervised techniques are less sensitive to this requirement. Typically, they cluster word associations and then worry about labeling the conveyed meaning using indicators from knowledge bases (Pantel and Lin (90)). Our approach is very similar in principle to this. We use a list of entities that describe an initial sense or meaning to guide the formation and interpretation of a cluster of evidences that are strongly associated with the initial sense specification.

3.3.10 Discussion

This work addresses some pertinent and significant challenges for entity recognition systems in the social media space. At the core of our contribution is the hypothesis that improvements to entity recognition can be achieved by introducing a new kind of feature that describes the relationship between the target candidate entity and the complexity it poses for entity extraction. With such understanding, classifiers can be affected to respond differently to entities that have inherently different extraction complexities by learning different rules during training. *We believe that this work presents a radical shift toward how we learn entity-specific models for information extraction.*

Focusing on a particular class of entity types called Cultural entities, that are often fragments from everyday language, we showed that traditional approaches for extraction perform rather poorly. Using our feature across entities of varying 'extraction complexities' indicated an overwhelming improvement in entity extraction performance, both from classification accuracy and information retrieval points of view. On average, we saw more than $+10\%$ increase in performance metrics and at the top end, specific entities have shown improvements of more than 30% in accuracy and 16% in F-Measure.

These results also highlight the statement of this thesis – use of prior knowledge about a domain can inform existing statistical methods for reliable Named Entity Recognition.

The realization of our intuition, as we showed in this work was non-trivial. In particular, we observed that relaxing the closed world sense assumption was necessary for this class of entities but also opened the door to several challenges. The algorithms we presented (as part of a generic two-step framework) combined two seemingly orthogonal techniques of graph-based propagation

and clustering that come together effectively for better separation of senses. We also note that our end-to-end system design for computing the complexity measure is completely unsupervised. The initial sense definition inputs to the system were also automatically obtained from Wikipedia in this work, and could potentially come from any open resource on the Web.

Most importantly, this work provides a new outlook for the role of Word Sense Disambiguation, modifying its traditional purpose and focus, and using it for entity extraction applications (as opposed to the more traditional uses such as machine translation).

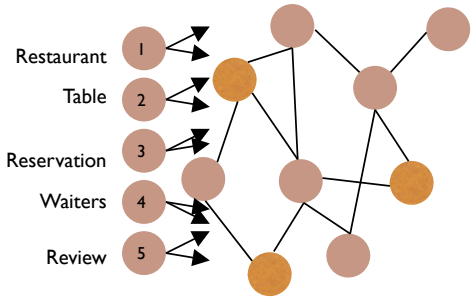
Other Applications: Automatic Lexicon Generation, Topic Classification

There are several applications of the algorithms described in this work to other text mining tasks. Here, we discuss relevance to one such task of automatic, unsupervised lexicon generation, with the clear advantage that the generated lexicons are not simply words that are related based on co-occurrence probabilities but are also 'semantically' related by a particular sense or meaning. For a in-depth discussion of the subject of lexical acquisition, see (Boguraev and Pustejovsky (12)).

Consider the task of generating a lexicon of words that are commonly used in the domain of 'restaurants'. Constructing a Spreading Activation Network from co-occurring words in a corpus and spreading the importance of domain representative words such as 'bars' and 'restaurants', essentially extracts a language model that is highly biased to a domain. Figure 3.11 shows an example of one such lexicon for the restaurants domain, that is obtained by pulsing only 5 sense related seed words in a general weblog corpus. Such topic or domain lexicons have clear applications in natural language processing, in topic classification of documents, contextual browsing, sense disambiguation etc.

Spreading Activation Network built from word co-occurrences

Sense Seed Definitions



restaurant, waiters, tasty, waiter, dish, nutrition, review, cooking, reviews, tibits, vegetarian, chef, sweet, bourdain, waitress, reservations, lunch, dishes, sushi, cuisine, burger, taste, burgers, fries, french, wines, tapas, wineries, wine, café, huang, vietnamese, espresso, anhui, coffee, shops, hotels, cafes, diners, bars, called, hefei, menus, chefs, michelin, dine, establishments, tourist, eateries, chain, meals, culinary, stores, pubs, food, retail, chains, specialty, bakeries, vendors, fuyang, restaurants, entrees, appetizers, salads, menu, assignment, shopper, shoppers, service, delicious, meal, paleo, eating, booths, tables, buffet, shrimp, chopsticks, eat, micah, tierney, dinners, dinner, mkhulu, san, tex, mexican, italian, pizza, brunch, bar, dining, steak, place, seafood, servers, salad, hostess, chinese, sandwich, patrons, bakery, eatery, local, outdoor, diner, mcdonald, greek, fancy, ate, ordering, cheese, business, thai, sandy, dined, hotel, japanese, afternoon, celebrate, birthday, cafe, table, downtown, francisco, good, seating, taco, foods, mex, night, soup, gift, chicken, banquet, anniversary, themed, pizzas, recommend, don, priced, pancakes, burrito, famous, neighborhood, drinks, potato, dessert, sausage, restaurateur, tonight, nearby, german, morton's, casual, reception, kosher, ranch, favorite, servings, crab, appetizer, steaks, toilets, veggie, grilled, baked, pho, pasta, opened, wonderful, reservation, mussels, quaint, pancake, chinatown, foodies, oasis, swanky, kitchen, enjoyed, patio, work, upscale, friend, plate, cab, corner, coworkers, cooks, valentine, celebrated, arrive, stuffed, owners, discount, bistro, vegan, …

Sense-biased lexicon generated post spreading activation of seed sense definitions

Figure 3.11: Unsupervised Lexicon Generation - Using seed sense hints to generate a list of terms related to a domain.

3.4 Cultural NER – Multiple Senses in the Same Domain

Compared to our first contribution in Cultural NER (Section 3.3) that focused on disambiguating entity names that span different domains (e.g., movies vs. video games), the focus of our second contribution detailed in this chapter is in the identification of Cultural entities that appear in *multiple senses even in the same domain*.

The occurrence of a person last names such as 'Clinton' in text, even if restricted to documents in the political domain, could refer to either Bill or Hillary or Chelsea Clinton. Figure 3.12 shows another example of the word 'Celebration' used as the name of a band, song, album and track title by multiple artists in the music domain.

'Celebration' (song), a song by Kool & The Gang, notably covered by Kylie Minogue
'Celebration' (Voices With Soul song), the debut single from girl band, Voices With Soul
'Celebration', a song by Krokus from Hardware
'Celebration' (Simple Minds album), a 1982 album by Simple Minds
'Celebration' (Julian Lloyd Webber album), a 2001 album by Julian Lloyd Webber
'Celebration' (Madonna album), a 2009 greatest hits album by Madonna
'Celebration' (Madonna song), same-titled single by Madonna
'Celebration' (band), a Baltimore-based band
'Celebration' (Celebration album), a 2006 album by 'Celebration'
'Celebration' (musical), a 1969 musical theater work by Harvey Schmidt and Tom Jones

Figure 3.12: Showing usages of the word 'Celebration' as the name of a band, hitsong, album and track title by multiple artists in the music domain.

The goal of the algorithm described in this chapter is in the fine-grained entity identification and disambiguation of such entities (in social media text) that have multiple real-world references within a same domain.

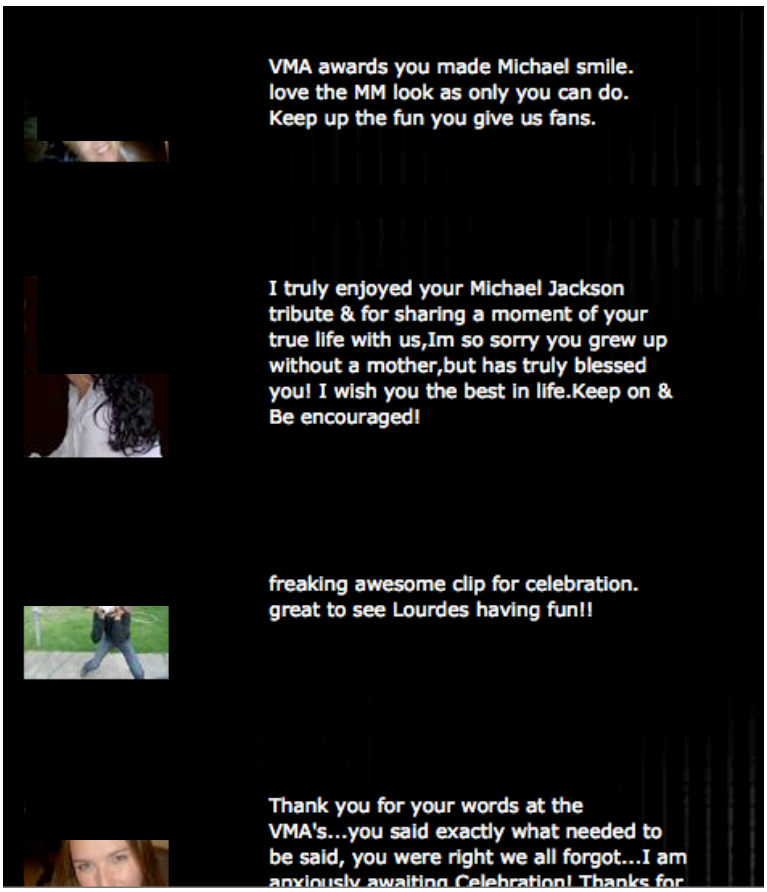


Figure 3.13: MySpace Music Forums - showing user-generated comments on Artist pages. Typically span one or two sentences making references to the artists and their work.

We focus on user-generated content from a particular class of social media platforms that are characterized by limited user-generated content and therefore fewer contextual cues in the text. User forums as on MySpace social networking site and micro-blogging platforms such as Twitter are examples of such platforms where users either by preference or by protocol limit content to one or two sentences. Figure 3.13 shows examples of user posts from MySpace Music forums where users are making references to the music artist Madonna and her work.

Sense disambiguation of such Cultural entities becomes even more challenging when there is insufficient context surrounding the discourse; and the language used is in the Informal English domain – a blend of abbreviations, slang and context dependent terms delivered with an indifferent

approach to grammar and spelling. However, such social media platforms are also characterized by rich contextual cues outside the content, such as the structure of the page, poster metadata, time and location of a post etc.

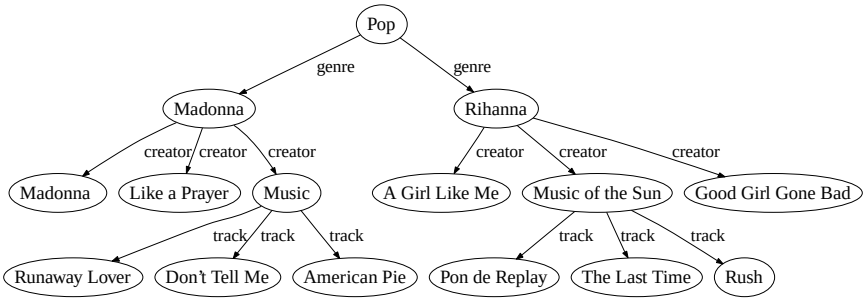
In this work, we make use of a wide variety of contextual cues, obtained from the content and the medium, in addition to prior knowledge about a domain in order to achieve fine-grained identification and disambiguation of entity mentions. We focus our experiments and evaluations in the Music domain that is rife with Cultural entities with multiple sense references.

3.4.1 Use of Domain Knowledge for Cultural NER

Knowledge bases such as dictionaries, taxonomies and ontologies encode a wealth of information as to how entities in a domain might relate. However, in the absence of a training corpus with in-line references to the entities (a "pre-annotated corpus"), it becomes difficult to identify and disambiguate named entities in text to leverage these relationships in more complex tasks (Minkov *et al.* (81)).

In this work we explore how the application of domain models (represented as a relationship graph) can complement traditional statistical NLP techniques to increase accuracy of entity identification in informal content from the music domain. Semantic annotation of track and album name mentions are performed with respect to MusicBrainz RDF[10] - a knowledge base of instances, metadata and relationships in the music domain. An example snapshot of the MusicBrainz RDF[11] is shown in Figure 3.14.

[10] http://wiki.musicbrainz.org/RDF
[11] RDF =




```
I went to <artist id=89>Madge's</artist> concert last night.
<artist id=262731>Rihanna</artist> is the greatest!
I love <artist id=357688>Lily's</artist> song <track id=8513722>smile</track>.
```

Figure 3.14: RDF Snapshot of MusicBrainz and example of in-line annotations. These annotations illustrate how messages in our corpus can be tagged with universally unique identifiers (in this case the MusicBrainz id number) to facilitate searches both for individual mentions as well as aggregate statistics on mentions in the corpus.

Challenges in the Music Domain

Availability of domain models is increasingly common with today's many Semantic Web initiatives. However, employing them for annotating Informal English content is non-trivial, more so in the music domain.

Song titles are often short and ambiguous. Songs such as "The" (four songs), "A" (74 songs), "If" (413 songs), and "Why" (794 songs) give some idea of the challenges in spotting these entities (also see Table 3.4). In annotating occurrence of these elements in text, for example, 'Yesterday' in "loved your song Yesterday!", we need to identify which entity 'Yesterday', among the many in the ontology, this one refers to.

The mapping of regions of text to entries in an ontology also becomes harder when the regions are words used commonly in everyday language, such as "Yesterday," which could refer to the previous day, a Beatles song (one of 897 songs with that title), or a movie (there are three

Bands with a song "Merry Christmas"	60
Songs with "Yesterday" in the title	3,600
Releases of "American Pie"	195
Artists covering "American Pie"	31

Table 3.4: Challenging features of the music domain.

productions so named).

3.4.1.1 Our Approach and Contributions

We present an approach that systematically expands and constrains the scope of domain knowledge from MusicBrainz used by the entity spotter to accurately annotate such challenging entity mentions in text from user comments. The snapshot of MusicBrainz used in this work contains 281,890 artists who have published at least one track and 4,503,559 distinct artist/track pairs.

We begin with a light weight, edit distance based entity spotter that works off a constrained set of potential entities from MusicBrainz. The entities we are interested in spotting in this work are track, album and song mentions. We constrain the size of the set of potential entities by manually examining some of the restrictions that can be applied on the MusicBrainz ontology. Restrictions are obtained using additional information from the context of the entity spot. For example, when considering a spot in a comment from a discussion group on country music, we may only consider artists and songs from that genre.

Further improvement is needed to disambiguate the usage of song titles. For example, while Lilly Allen has a song titled 'Smile,' not all mentions of this word on her MySpace page refer to the song, for example, "your face lights up when you smile". We disambiguate the output of our naive spotter with more advanced NLP techniques using an SVM classifier that takes into account

the characteristics of word usages.

We find that constraining the domain of possible entity matches before spotting can improve precision by several orders of magnitude over an admittedly poor baseline of the light weight spotter. We note that these improvements follow a Zipf distribution, where a reduction of possible entity matches by 50% equals a doubling of precision. We also find that use of our NLP system can improve accuracy by more than another 50%. These two steps, presented in the rest of this paper, can form the beginning of a processing pipeline to allow higher precision spot candidates to flow to upstream applications.

3.4.2 Related Work

Named Entity Recognition and use of Domain Knowledge

Nadeau and Sekine present a comprehensive survey of the Named Entity Recognition space since 1991 (Nadeau and Sekine (84)). The KnowItAll Information Extraction system (Etzioni *et al.* (35)) makes use of entity recognition techniques, in a domain-independent fashion. Related work by Chieu and Ng has shown high performance in entity extraction with a single classifier and information from the whole document to classify each word (Chieu and Ng (23)).

Closely related to our work, domain dictionaries have been widely used in NER, including Wikipedia (Bunescu and Pasca (19)) and Wiktionary (Muller and Gurevych (82)), DBLP (J Hassell and Arpinar (57)), KAON (Bozsak *et al.* (14)), and MusicBrainz (Alba *et al.* (2)). They have also been used for the task of disambiguating entity senses, an important step in accurately extracting entities. Work in (Bunescu and Pasca (19)) exploited the link and textual features of Wikipedia

to perform named entity disambiguation. Entity disambiguation by gathering context from the document and comparing it with context in the knowledge base was also explored in (J Hassell and Arpinar (57)).

These provide inspiration for our work, demonstrating that it is possible to do efficient and accurate NER on a document-by-document basis using domain knowledge supplemented with natural language processing techniques. Our work differs in how we constrain a domain knowledge base in order to annotate a set of known named entities in Informal English content.

Named Entity Recognition in Informal English

The challenge of NER in noisy and informal text corpora has been explored from several angles. Minkov et al. were the first to address NER in "informal text" such as bulletin board and newsgroup postings, and email (Minkov *et al.* (81)). Their work on recognizing personal names in such corpora is particularly relevant, as it uses dictionaries and constraining dictionary entries. They use a TF/IDF based approach for constraining the domain space, an approach we considered in early versions of our music miner. However, we found this approach to be problematic in the music domain, as song titles often have very low novelty in the TF/IDF sense (e.g. the Beatles song, "Yesterday"). Work by Ananthanarayanan et al. has also shown how existing domain knowledge can be encoded as rules to identify synonyms and improve NER in noisy text (Ananthanarayanan *et al.* (4)).

Our approach to NER in informal text differs in that it is a two step process. Given a set of known named entities from the MusicBrainz RDF, we first eliminate extraneous possibilities by constraining the domain model using available metadata and further use the natural language

Madonna	an artist with a extensive discography, a recent album and concert tour
Rihanna	a pop singer with recent accolades including a Grammy Award and a very active MySpace presence
Lilly Allen	an independent artist with song titles that include "Smile," "Allright, Still", "Naive", and "Friday Night" who also generates a fair amount of buzz around her personal life not related to music

Table 3.5: Artists in the Ground Truth Data Set

context of entity word-usages to disambiguate entities that appear as entities of interest and those that do not. Some word-usage features we employ are similar to those used in the past (Minkov *et al.* (81)), while others are derived from our domain of discourse.

3.4.3 Restricted Entity Extraction

We begin our exploration of restricted RDF graphs or Ontologies to improve entity spotting by investigating the relationship between the number of entities (artists, songs and albums) considered for spotting and the precision of the entity spotter. The result is a calibration curve that shows the increase in precision as the entity set is constrained. This can be used to gauge the benefit of implementing particular real world constraints in annotator systems. For example, if detecting that a post is about an artist's recent album requires three weeks of work, but only provides a minor increase in precision, it might be deferred in favor of an "artist gender detector" that is expected to provide greater restriction in most cases.

3.4.3.1 Ground Truth Data Set

Our experimental evaluation focuses on user comments from the MySpace pages of three artists: Madonna, Rihanna and Lily Allen (see Table 3.5). The artists were selected to be popular enough

Artist (Spots scored)	Good spots Agreement		Bad spots Agreement	
	100%	75 %	100%	75%
Rihanna (615)	165	18	351	8
Lily (523)	268	42	10	100
Madonna (720)	138	24	503	20

Table 3.6: Manual scoring agreements on naive entity spotter results.

to draw comment but different enough to provide variety. The entity definitions were taken from the MusicBrainz RDF (see Figure 3.14), which also includes some but not all common aliases and misspellings.

We establish a ground truth data set of 1858 entity spots for these artists (breakdown in Table 3.6). The data was obtained by crawling the artist's MySpace page comments and identifying all exact string matches of the artist's song titles. Only comments with at least one spot were retained. These spots were then hand scored by four of the authors as "good spot," "bad spot," or "inconclusive."

The human taggers were instructed to tag a spot as "good" if it clearly was a reference to a song and not a spurious use of the phrase. An agreement between at least three of the hand-spotters with no disagreement was considered agreement. As can be seen in Table 3.6, the taggers agreed 4-way (100% agreement) on Rihanna (84%) and Madonna (90%) spots. However ambiguities in Lily Allen songs (most notably the song "Smile"), resulted in only 53% 4-way agreement.

We note that this approach results in a recall of 1.0, because we use the naive spotter, restricted to the individual artist, to generate the ground truth candidate set. The precision of the naive spotter after hand-scoring these 1858 spots was 73%, 33% and 23% for Lilly Allen, Rihanna and Madonna respectively (see Table 3.6). This represents the best case for the naive spotter and accuracy drops quickly as the entity candidate set becomes less restricted. Next, we take a closer look at the

relationship between entity candidate set size and spotting accuracy.

3.4.3.2 Impact of Domain Restrictions

One of the main contributions of this work is the insight that it is often possible to restrict the set of entity candidates, and that such a restriction increases spotting precision. Here, we explore the effect of domain restrictions on spotting precision by considering random entity subsets.

We begin with the whole MusicBrainz RDF of 281,890 publishing artists and 6,220,519 tracks, which would be appropriate if we had no information about which artists may be contained in the corpus. We then select random subsets of artists that are factors of 10 smaller (10%, 1%, etc). These subsets always contain our three actual artists (Madonna, Rihanna and Lily Allen), because we are interested in simulating restrictions that remove invalid artists. The most restricted entity set contains just the songs of one artist (≈0.0001% of the MusicBrainz taxonomy).

In order to rule out selection bias, we perform 200 random draws of sets of artists for each set size - a total of 1200 experiments. Figure 3.15 shows that the precision increases as the set of possible entities shrinks. For each set size, all 200 results are plotted and a best fit line has been added to indicate the average precision. Note that the figure is in log-log scale.

We observe that the curves in Figure 3.15 conform to a power law formula, specifically a Zipf distribution ($\frac{1}{n^{R^2}}$). Zipf's law was originally applied to demonstrate the Zipf distribution in frequency of words in natural language corpora (Zipf (120)), and has since been demonstrated in other corpora including web searches (Cunha *et al.* (32)). Figure 3.15 shows that song titles in Informal English exhibit the same frequency characteristics as plain English. Furthermore, we

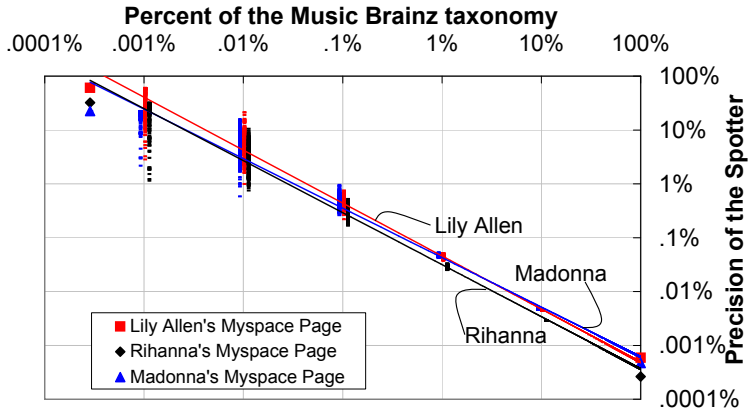


Figure 3.15: Precision of a naive spotter using differently sized portions of the MusicBrainz Taxonomy to spot song titles on artist's MySpace pages

can see that in the average case, a domain restrictions of 10% of the MusicBrainz RDF will result approximately in a 9.8 times improvement in precision of a naive spotter.

This result is remarkably consistent across all three artists. The R^2 values for the power lines on the three artists are 0.9776, 0.979, 0.9836, which gives a deviation of 0.61% in R^2 value between spots on the three MySpace pages.

3.4.4 Real World Constraints

The calibration results from the previous Section show the importance of "ruling out" as many artists as possible. We observe that simple restrictions such as gender that might rule out half the corpus could potentially increase precision by a factor of two. One way to impose these restrictions is to look for real world constraints that can be identified using the metadata about entities as they appear in a particular post. Examples of such real world constraints could be that an artist has released only one album, or has a career spanning more than two decades.

We are interested in two questions. First, *do real world constraints reduce the size of the entity*

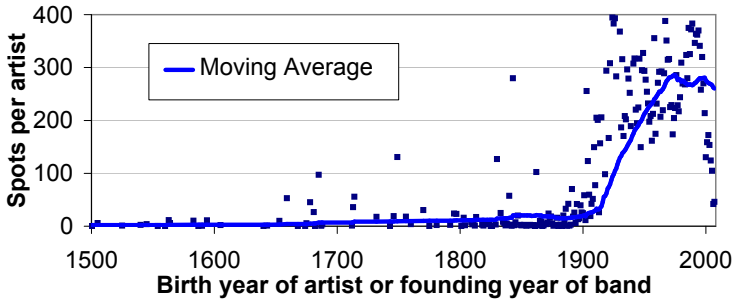


Figure 3.16: Songs from all artists in our MySpace corpus, normalized to artists per year.

spot set in a meaningful way? Second, *by how much does the trivial spotter improve with these real world constraints* and does this match with our predicted improvements from Figure 3.15?

The effect of restricting the RDF by artist's age can be seen in Figure 3.16, which shows spots per artist by birth date. Interestingly, we can see a spike in the graph beginning around 1920 with the emergence of Jazz and then Rock and Roll, reflecting the use of common words as song titles, (e.g. "Blues" and "South" by Louis Armstrong). For all artists since this date (94% of the MusicBrainz Ontology, and 95.5% of the naive spots on our corpus), the increased use of natural language utterances as song titles is evidence that we should expect the Zipf distribution to apply to any domain restriction over the corpus.

Having established that domain restrictions do reduce spot size, we look for further constraints that can be inferred from the user-generated text. As an example, we observe that comments such as "Saw you last night in Denver!!!" indicate the artist is still alive. A more informational post such as "Happy 25th B-DAY!" would allow us to further narrow the RDF graph to 0.081% of artists in the Ontology, and 0.221% of the naive spots on Lily Allen's MySpace Page.

Our constraints are tabulated in Table 3.7, and are derived *manually* from comments such as, "I've been a fan for 25 years now," "send me updates about your new album," and "release your

new album already! i'm getting tired of playing your first one on repeat!" Since we have chosen our corpus to represent three specific artists, the name of the artist is a further narrowing constraint.

Key	Count	Restriction
Artist Career Length Restrictions- Applied to Madonna		
B	22	80's artists with recent (within 1 year) album
C	154	First album 1983
D	1,193	20-30 year career
Recent Album Restrictions- Applied to Madonna		
E	6,491	Artists who released an album in the past year
F	10,501	Artists who released an album in the past 5 years
Artist Age Restrictions- Applied to Lily Allen		
H	112	Artist born 1985, album in past 2 years
J	284	Artists born in 1985 (or bands founded in 1985)
L	4,780	Artists or bands under 25 with album in past 2 years
M	10,187	Artists or bands under 25 years old
Number of Album Restrictions- Applied to Lily Allen		
K	1,530	Only one album, released in the past 2 years
N	19,809	Artists with only one album
Recent Album Restrictions- Applied to Rihanna		
Q	83	3 albums exactly, first album last year
R	196	3+ albums, first album last year
S	1,398	First album last year
T	2,653	Artists with 3+ albums, one in the past year
U	6,491	Artists who released an album in the past year
Specific Artist Restrictions- Applied to each Artist		
A	1	Madonna only
G	1	Lily Allen only
P	1	Rihanna only
Z	281,890	All artists in MusicBrainz

Table 3.7: The efficacy of various sample restrictions.

We consider three classes of restrictions - career, age and album based restrictions, apply these to the MusicBrainz RDF to reduce the size of the entity spot set in a meaningful way and finally run the trivial spotter. For the sake of clarity, we apply different classes of constraints to different artists.

We begin with restrictions based on length of career, using Madonna's MySpace page as our

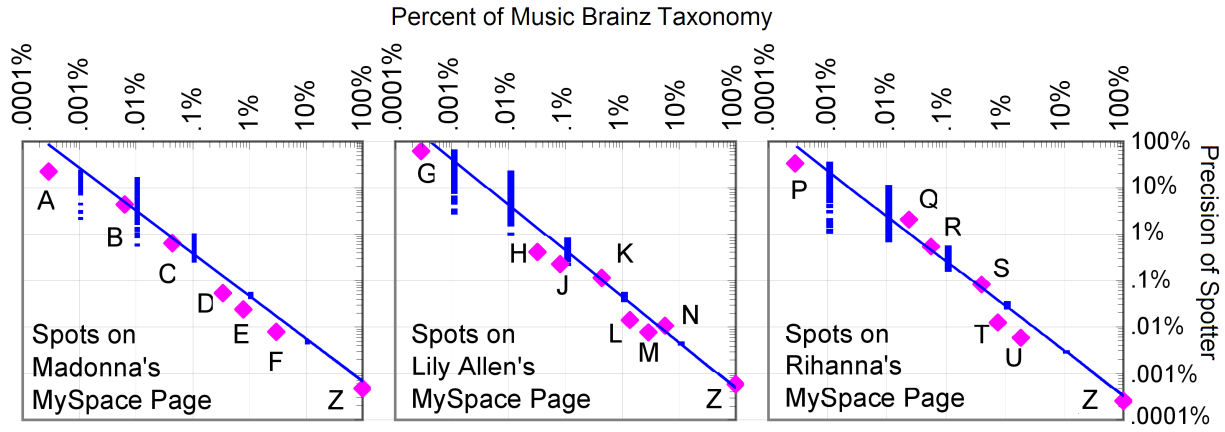


Figure 3.17: Naive spotter using selected portions of the MusicBrainz RDF based on descriptive characteristics of Madonna, Lily Allen and Rihanna, respectively. The Key to the data points is provided in Table 3.7

corpus. We can restrict the RDF graph based on total length of career, date of earliest album (for Madonna this is 1983, which falls in the early 80's), and recent albums (within the past year or 5 years). All of these restrictions are plotted in Figure 3.17, along with the Zipf distribution for Madonna from Figure 3.15. We can see clearly that restricting the RDF graph based on career characteristics conforms to the predicted Zipf distribution.

For our next experiment we consider restrictions based on age of artist, using Lily Allen's MySpace page as our corpus. Our restrictions include Lily Allen's age of 25 years, but overlap with bands founded 25 years ago because of how dates are recorded in the MusicBrainz Ontology. We can further restrict using album information, noting that Lily Allen has only a single album, released in the past two years. These restrictions are plotted in Figure 3.17, showing that these restrictions on the RDF graph conform to the same Zipf distribution.

Finally, we consider restrictions based on absolute number of albums, using Rihanna's MySpace page as our corpus. We restrict to artists with three albums, or at least three albums, and

can further refine by the release dates of these albums. These restrictions fit with Rihanna's short career and disproportionately large number of album releases (3 releases in one year). As can be seen in Figure 3.17, these restrictions also conform to the predicted Zipf distribution.

The agreement of the three types of restrictions from above with the random restrictions from the previous Section are clear from comparing the plots in Figure 3.17. This confirms the general effectiveness of limiting domain size to improve precision of the spotter, regardless of the type of restriction, as long as the restriction only removes off-target artists. A reduction in the size of the RDF graph results in an approximately proportionate increase in precision.

This is a particularly useful finding, because it means that any restriction we can apply will improve precision, and furthermore we can estimate the improvement in precision.

3.4.5 NLP Assist

While reducing extraneous possibilities improved precision of the naive spotter significantly, false positives resulting from spots that appear in different senses still need attention (see Table 3.8). The widely accepted *'one sense per discourse'* notion (Yarowsky (118)) that the sense or meaning of a word is consistent within a discourse does not hold for this data given the use of common words as names for songs and albums.

The task is to assess whether a spot found is indeed a valid track or album. This is similar to the word sense disambiguation problem where the task is to resolve which one of many pre-defined senses is applicable to a word (Ide and Véronis (56)). We use a learning algorithm over local and surrounding word contexts, an approach similar in principle to several past efforts but adapted to

Valid: Got your new album *Smile*. Loved it!
Invalid: Keep your *SMILE* on. You'll do great!
Valid: *Thriller* was my most fav MJ album
Invalid: this is *bad* news, ill miss you MJ.

Table 3.8: Spots in multiple senses, some of which are non-music mentions.

our domain of discourse (Tatar (101)).

Formally, our task can be regarded as a binary classification problem. Consider the set of all spots found by the naive spotter. Each spot in this set can be labeled 1 if it is a track; and -1 if it is not, where the label is associated with a set of input features that characterize a spot *s*. This is implemented as a Support Vector Machine (SVM), a machine learning approach known to be effective for solving binary pattern recognition, named entity recognition and document classification problems (Joachims (59)).

3.4.5.1 Feature Space for NER

We trained and tested the SVM learner on two sets of features collectively observed in the tagged data (see Section 3.4.3.1); *basic features*, that characterize a spot and *advanced features* that are based on the context surrounding the spot.

Basic features: We encode a set of spot-level boolean features (see Table 3.9) that include whether the spot is all capitalized, starts with capital letters or is enclosed in quotes. If the entire comment including the spot is capitalized, we do not record a 1 for *s.allCaps* or *s.firstCaps*.

We also encode features derived from the part-of-speech (POS) tags and NP-chunking of comments (see syntactic features in Table 3.9)[12]. To encode syntactic features, we created a list of

[12]Obtained using the Stanford NL Parser http://nlp.stanford.edu/software/lex-parser.shtml

Syntactic features	**Notation-S**
+POS tag of *s*	s.POS
POS tag of one token before *s*	s.POS$_b$
POS tag of one token after *s*	s.POS$_a$
Typed dependency between *s* and sentiment word	s.POS-TD$_{sent^*}$
Typed dependency between *s* and domain-specific term	s.POS-TD$_{dom^*}$
Boolean Typed dependency between *s* and sentiment	s.B-TD$_{sent^*}$
Boolean Typed dependency between s and domain-specific term	s.B-TD$_{dom^*}$
Word-level features	**Notation-W**
+Capitalization of spot *s*	s.allCaps
+Capitalization of first letter of *s*	s.firstCaps
+*s* in Quotes	s.inQuotes
Domain-specific features	**Notation-D**
Sentiment expression in the same sentence as *s*	s.S$_{sent}$
Sentiment expression elsewhere in the comment	s.C$_{sent}$
Domain-related term in the same sentence as *s*	s.S$_{dom}$
Domain-related term elsewhere in the comment	s.C$_{dom}$

+Refers to basic features, others are advanced features

*These features apply only to one-word-long spots.

Table 3.9: Features used by the SVM learner

the Penn Treebank tag set[13] also used by the Stanford parser. If the parser returns a tag for the spot, we obtain the tag's index position in the list to encode this feature. If the sentence is not parsed this feature is not encoded.

Advanced features: We encode the following advanced features intended to exploit the local context surrounding every spot. We encode the POS tags of word tokens appearing before and after a spot in a sentence.

Sentiment expressions and domain-specific terms: We found that entity spots that co-occurred with sentiment expressions and domain-specific words such as 'music', 'album', 'song', 'concert', etc. were more likely to be valid spots (see Figure 3.18). We encode these boolean features in the

[13]http://www.cis.upenn.edu/~treebank/

Your music in *celebration* is really *bangin*!
You're a *genius* in *Alright still*! Keep droppin bombs!
I really *love* the album *A girl like me.*

Figure 3.18: Examples of comments where domain and sentiment expressions co-occur with entity names.

following manner.

First, we curated a sentiment dictionary of 300 positive and negative expressions from Urban-Dictionary [14] (UD), given the use of slang by this poster demographic. Starting with expressions such as 'good', and 'bad', we obtained the top 10 related sentiment expressions for these words. We continued this process for the newly obtained words until we found no new words. Note that we are not concerned with the polarity, but mere co-occurrence of sentiment expressions with spots. A dictionary of 25 domain-specific terms, such as 'music', 'album', 'track', 'song' etc. was created manually by consulting MusicBrainz.

If one or more sentiment expressions, domain-specific terms or their word forms were spotted in the same sentence as the spot, values for $s.S_{sent}$ and $s.S_{dom}$ are recorded as 1. Corresponding $s.C_{sent}$ and $s.C_{dom}$ features were also used to record similar values when these terms were found elsewhere in the comment. Encoding the actual number of co-occurring sentiment or domain expressions did not significantly change the classification result.

Typed Dependencies: We also captured the typed dependency paths (grammatical relations) via the $s.POS\text{-}TD_{sent}$ and $s.POS\text{-}TD_{dom}$ boolean features. These were obtained between a spot and co-occurring sentiment and domain-specific words by the Stanford parser (Marneffe *et al.* (74)) (see example in 3.10). We also encode a boolean value indicating whether a relation was found at

[14] www.urbandictionary.com

all using the *s.B-TD*$_{sent}$ and *s.B-TD*$_{dom}$ features. This allows us to accommodate parse errors given the informal and often non-grammatical English in this corpus.

Valid spot: Got your new album **Smile**. Simply *loved* it! **Encoding:** nsubj(loved-8, Smile-5) : **Smile** is the nominal subject of the expression *loved*.
Invalid spot: Keep your **smile** on. You'll do *great*! **Encoding:** No typed dependency between **smile** and *great*

Table 3.10: Typed Dependencies between entity spots and sentiment expressions.

3.4.6 Data and Experiments

Our training and test data sets were obtained from the hand-tagged data (see Table 3.6). Positive and negative *training examples* were all spots that all four annotators had confirmed as valid or invalid respectively, for a total of 571 positive and 864 negative examples. Of these, we used 550 positive and 550 negative examples for training. The remaining spots were used for test purposes.

Our positive and negative *test sets* comprised of all spots that three annotators had confirmed as valid or invalid spots, i.e. had a 75% agreement. We also included spots where 50% of the annotators had agreement on the validity of the spot and the other two were *not sure*. We further divided our negative test set into two disjoint equal sets that allowed us to confirm generality of the effect of our features. Finally, our test set of *valid spots*, Set 1, contained 120 spots and the two test sets for *invalid spots*, Set 2 and Set 3, comprised of 229 spots each.

We evaluated the efficacy of features shown in Table 3.9 in *predicting the labels assigned by the annotators*. All our experiments were carried out using the SVM classifier from (Chang and Lin (22)) using 5-fold cross-validation. As one way of measuring the relative contribution of advanced contextual and basic spot-level features, we removed them one after another, trying

	Features	**Valid**	**Invalid Spots**			**Acc.**
		Set1	Set2	Set3	Avg.	Split
(1)	W	45	88	84	**86**	45 - 86
(2)	W+S	74	43	37	40	74 - 40
(3)	W+D	62	85	83	**84**	62 - 84
(4)	D	70	50	62	56	70 - 56
(5)	D+S	72	34	36	35	72 - 35
(6)	W+D+s.POS	61	66	74	70	61 - 70
(7)	W+D+s.POS$_{b,a}$+s.POS-TDs	**78**	47	53	50	78 - 50
(8)	W+D+s.POS$_{b,a}$+s.B-TDs	**90**	33	37	35	90 - 35
(9)	W+D+only s.POS$_{b,a}$	62	81	87	**84**	62 - 84
(10)	W+D+only s.POS-TDs	60	79	91	**85**	60 - 85
(11)	W+D+only s.B-TDs	71	68	72	70	71 - 70
(12)	All features	42	89	93	**91**	42 - 91

Table 3.11: Classifier accuracy in percentages for different feature combinations. Best performers are shown in bold.

several combinations. Table 3.11 reports those combinations for which the accuracy in labeling either the valid or invalid datasets was at least 50% (random labeling baseline). Accuracy in labeling valid and invalid spots refer to the percentage of true and false positives that were labeled correctly by the classifier. In the following discussion, we refer to the average performance of the classifier on the false positives, Sets 2 and 3 and its performance on the true positives, Set 1.

3.4.6.1 Usefulness of Feature Combinations

Our experiments revealed some expected and some surprising findings about the usefulness of feature combinations for this data. For valid spots, we found that the best feature combination was the word-level, domain-specific and contextual syntactic tags (POS tags of tokens before and after the spot) when used with the boolean typed dependency features. This feature combination labeled 90% of good spots accurately. The next best and similar combination of word-level, domain-specific and contextual tags when used with the POS tags for the typed dependency features yielded

Feature Combination	Mean Accuracy	Std.Dev Accuracy
All Features	99.4%	0.87%
W	91.3%	2.58%
W+D	83%	2.66%
W+D+only s.POS-TDs	80.8%	2.4%
W+D+only s.POS$_{b,a}$	77.33%	3.38%

Table 3.12: Average performance of best feature combinations on 6 sets of 500 invalid spots each

an accuracy of 78%. This suggests that *local word descriptors along with contextual features* are good predictors of valid spots in this domain. For the invalid spots (see column listing average accuracy), the use of all features labeled 91% of the spots correctly. Other effective combinations included the word-level; word-level and domain-specific; word-level, domain-specific and POS tags of words before and after the spot; word-level, domain-specific and the typed dependency POS tags, all yielding accuracies around 85%. It is interesting to note that the POS tags of the spot itself were not good predictors for either the valid or invalid spots. However, the POS typed dependencies were more useful than the boolean typed dependencies for the invalid spots.

This suggests that not all syntactic features are useless, contrary to the general belief that syntactic features tend to be too noisy to be beneficial in informal text. Our current investigations to improve performance include the use of other contextual features like commonly used bi-grams and tri-grams and syntactic features of more tokens surrounding a spot.

Accuracy in Labeling Invalid Spots:

As further confirmation of the generality of effect of the features for identifying incorrect spots made by the naive spotter, we picked the best performing feature combinations from Table 3.11 and tested them on a dataset of 3000 known invalid spots for artist Rihanna's comments from her

MySpace page. This dataset of invalid spots was obtained using the entire MusicBrainz taxonomy excluding Rihanna's song/track entries - effectively allowing the naive spotter to mark all invalid spots. We further split the 3000 spots into 6 sets of 500 spots each. The best feature combinations were tested on the model learned from the same training set as our last experiment. Table 3.12 shows the average and standard deviation performance of the feature combinations across the 6 sets. The feature combinations performed remarkably consistently for this larger test set. The combination of all features was the most useful, labeling 99.4% of the invalid spots correctly.

3.4.7 Improving Spotter Accuracy Using NLP Analysis

The last set of experiments confirmed the usefulness of the features in classifying whether a spot was indeed a track or not. In this next experiment, we sought to measure the improvement in the overall spotting accuracy - first annotating comments using the naive spotter, followed by the NLP analytics. This approach of boosting allows the more time-intensive NLP analytics to run on less than the full set of input data, as well as giving us a certain amount of control over the precision and recall of the final result.

Figure 3.19 shows the improvement in precision for spots in the three artists after boosting the naive spotter with the NLP component. Ordered by decreasing recall, we see an increase in precision for the different feature combinations. For example, the precision of the naive spotter for artist Madonna's spots was 23% and almost 60% after boosting with the NLP component and using the feature combinations that resulted in a $42 - 91$ split in accurately labeling the valid and invalid spots.

Although our classifier was built over the results of the naive spotter, i.e. it already knew that

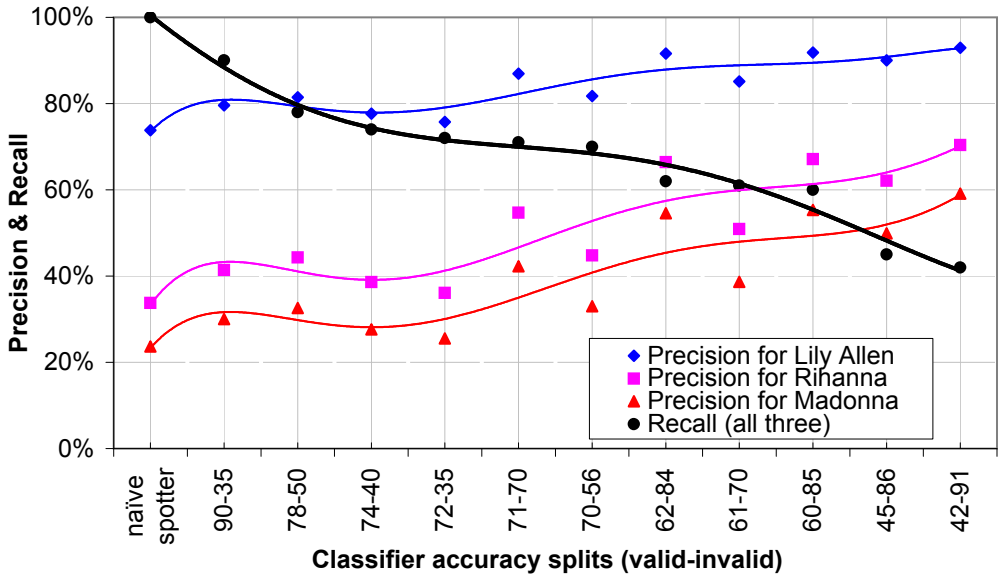


Figure 3.19: NLP Precision-Recall curves for three artists and feature combinations

the spot was a potential entity, our experiments suggest that the features employed might also be useful for the traditional named entity recognition problem of labeling word sequences as entities.

Our experiments also suggest that although informal text has different characteristics than formal text such as news or scientific articles, simple and inexpensive learners built over a dictionary-based naive spotter can yield reasonable performance in accurately extracting entity mentions.

3.4.8 Discussion

Spotting music tracks in Informal English is a critical enabling technology for applications that allow real-time tracking of user activity and listening preferences in on-line forums. Rapid detection of events (e.g. the artist "OK Go" ramping up the chart within hours of being featured on the popular TV show Big Brother) illustrate the possibilities of these systems. There are several challenges in constructing these systems. Discussions of music produce millions of posts a day, which need

to be processed in real-time, prohibiting more computational intensive NLP techniques. Moreover, since 1920, song titles based on common words or phrases have become very popular (see Figure 3.16), making it difficult to spot and disambiguate song titles.

In this paper, we presented a two stage approach - entity spotting based on scoping a domain model followed by SVM based NLP system to facilitate higher quality entity extraction. We studied the impact of restricting the size of the entity set being matched and noted that the spot frequency follows a Zipf distribution. We found that R^2 for this distribution is fairly consistent among a sample of artists. This allows a reasonable a priori evaluation of the efficacy of various restriction techniques using the calibration curve shown in Figure 3.15. We found that in many cases such restrictions can come from the language of the spotted text itself.

Given these potential spots, we show that simple classifiers trained on generic lexical, word and domain specific characteristics of a spot can effectively eliminate false positives in a manner that can improve accuracy up to a further 50%. Our experiments suggest that although informal text has different characteristics than formal text, learners that improve a dictionary-based naive spotter can yield reasonable performance in accurately extracting entity mentions.

The ability to achieve reasonable performance in this problem suggests that this approach will work well in other, less challenging domains where the entities are less overlapping (e.g. company name extraction) or the English is less informal (e.g. news releases).

3.5 Summary of NER Contributions

In this chapter, we described two algorithms for identifying cultural entities in social media text. Both algorithms take a binary classification and disambiguation intensive approach where we assume that the algorithm knows what entities and types it wishes to recognize in text. In both cases, we find that the combination of word and syntactic features gleaned around an entity spotted in a document corpus and domain specific features obtained a domain knowledge base yield reliable judgements for cultural NER. While we focused on cultural named entities, the presented algorithms are not by any means restricted to this class of entities.

3.5.1 Applications of NER in Social Media Content

Annotation of named entities in social media text will play an increasingly important role as business and companies seek to better understand their customers and keep up with the volume of user-generated content. In Chapter 5 of this thesis, we present one such use-case of NER in the music domain that contributes to the goal of mining popular artists and tracks from use-generated content on social media platforms.

We present challenges and lessons learned in building a near real-time Social Intelligence system in collaboration with researchers at IBM Almaden for the BBC 'SoundIndex' (Gruhl *et al.* (44)). This real-time dashboard system draws over 40 million datum a day from a variety of sources, identifies artist, track and album entities, eliminates spam, identifies associated sentiments (like or dislike an artist and their work) and generates Top-N lists of artists that continuously captures the buzz around popular music.

4. Summarizing User-generated Content

4.1 Key Phrase Extraction - 'Aboutness' of Content

Key phrase extraction is the process of identifying or assigning key phrases or words to a document that indicate what an article is 'about'. Key phrase extraction has several applications, both in the human and machine consumption of content; in summarization, indexing, labeling, categorizing, clustering, browsing, and searching.

While Named Entity Recognition aims to extract specific types of information, such as the name of a company, people etc. key phrase extraction is not limited to specific information nuggets and is tasked with producing topical phrases for any document.

Techniques for key phrase extraction can be broadly classified as key phrase assignment or key phrase extraction (for a survey, see (Turney (106)). In key phrase assignment, the goal is to assign one or more keywords from a pre-defined list of phrases to a document, either because they appear in the document or because the document categorizes under the topic represented by the phrase. Key phrase extraction on the other hand focuses on identifying phrases explicitly mentioned in the document that are descriptive of its contents.

The contributions made in this thesis fall under the second category of *extracting key phrases* that are explicitly present in the content and are also indicative of what the document is 'about'.

The focus of previous approaches to key phrase extraction have been on extracting phrases that summarize *a document*, e.g. a news article, a web page, a journal article or a book. In contrast, the focus of this thesis is not in summarizing a document generated by users on social media platforms but to extract key phrases that *are descriptive of information present in multiple observations (or documents) made by users about an entity, event or topic of interest*.

The primary motivation is to obtain an abstraction of a social phenomenon that makes volumes of unstructured user-generated content easily consumable by humans and agents alike. As an example of the goals of our work, Table 4.1 shows key phrases extracted from online discussions around the 2009 Health Care Reform debate and the 2008 Mumbai terror attack, summarizing hundreds of user comments to give a sense of what the population cared about on a particular day.

2009 Health Care Reform	**2008 Mumbai Terror Attack**
Health care debate	Foreign relations perspective
Healthcare staffing problem	Indian prime minister speech
Obamacare facts	UK indicating support
Healthcare protestors	Country of India
Party ratings plummet	Rejected evidence provided
Public option	Photographers capture images of Mumbai

Table 4.1: Showing summary key phrases extracted from more than 500 online posts on Twitter around two news-worthy events on a single day.

Solutions to key phrase extraction have ranged from both unsupervised techniques that are based on heuristics to identify phrases and supervised learning approaches that learn from human labeled key phrase examples in documents. The goals of prior key phrase extraction algorithms

have also spanned several applications, from document classification, index creation to metadata creation for documents and web pages.

In both cases however, features exploited by different algorithms can be broadly classified as follows:

- Syntactic Cues: Phrases present in quotes, italics, bold; phrases present in document headers; phrases collocated with acronyms etc. For example, see Krulwich (66).
- Document and Structural Cues: Two word phrases, phrases appearing in the beginning of a document, sentence or paragraph indicating the first mention of an important topic, number of words in a phrase, frequency of the phrase, phrases present in multiple similar documents etc. For example, see Muñoz (83), Steier and Belew (99), Turney (105), Mihalcea and Tarau (79).
- Linguistic Cues: Stemmed form of a phrase, phrases that are simple and compound nouns in sentences etc. For example, see Nakagawa (85).

In this thesis, we make use of similar features in an unsupervised algorithm that extracts and scores patterns from user-generated content toward the end goals of facilitating human browsing and indexing for 'aboutness' understanding of volumes of social media content.

4.1.1 Thesis Focus - Summarizing Social Perceptions

Extracting key phrases for summarizing several social media data points is a different problem compared to summarizing a document or article for the following reasons:

- **The 'social' behind social data:** The key phrases that are extracted to summarize user-generated content should not only summarize the topic in focus but also need to do so in a manner that preserves the social perceptions behind the data. For example, the cultural logic of a politically liberal versus a conservative state is different and tends to be reflected in the data that is generated by the people from the two states. In extracting summaries of what people are saying, it is important to preserve this 'social' aspect behind the data that facilitates the process of understanding 'why' a society does 'what' it does.

- **Redundancy, Variability and Off-topic Nature of Content:** Redundancy and variability are common features in any form of self-expression and communication. Multiple users tend to share the same information around a trending topic or event – there were more than 500 messages on Twitter urging people to 'support an up-or-down vote on the health care public option'. Multiple users also share the same information in a variety of ways – users used the phrases 'big three' and the name of the three companies 'General Motors', 'Ford' and 'Chrysler' together, interchangeably to convey similar messages. On social media platforms, both redundancy and variability are predominant features given the variety of topics that generate interest and the diversity of participants involved. Normalizing for redundant and variable expressions is an important task in summarizing social media content.

  Additionally, since communication on social media is inter-personal in nature, off-topic discussions are fairly common. If we are interested in public perceptions of a new product (e.g., Nike's Gladiator shoes), the off-topic discussion of dinner options, as in the following user comment, must be eliminated.

  "Met up with mom for lunch, she looks lovely as ever, good genes .. Thanks Nike, I

love my new Gladiators ..smooth as a feather. I burnt all the calories of Italian joy in one run.. if you are looking for good Italian food on Main, Buca is the place to go."

Social components of User-generated Content

The primary goal of our key phrase extraction algorithm is to extract key phrases that summarize online discussions around a topic of interest while also preserving the 'social' behind the social data. The 'social' components of user-generated data can be derived from several features.

Similarities in thematic aspects of the content are themselves a reflection of the social and cultural logic of a society. Clustering or extracting key phrases based on co-occurrence patterns (for example, Liu *et al.* (72)) tap into the idea that it is possible to extract latent characteristics of a population by looking at the content it generates.

Poster demographic properties (age, gender, education, economic status etc.), the location and time information associated with content are additional social signals that help us understand 'why' the data was generated. In particular, community and geography have shown to play a key role in communication. Local contexts affect user perceptions and play an important role in the information that a society produces, consumes and shares. Studies have shown that social ties are not only more likely between geographically proximate individuals but that people living in the same geographic area also tend to have similar interests and information needs (McPherson *et al.* (77)).

On social media platforms several of these 'social' attributes of user-generated content are readily mineable and provide an opportunity to observe social phenomena from the microscopic

to the macroscopic level, in a range of scales from the local to the global, over the last few days to months and years etc.

Spatio-Temporal-Thematic Key Phrase Extraction

Unlike past approaches that have focused on the content and page or link structures to extract key descriptors in documents, in our work, we investigate the role of the three dimensions of *space, time and theme* to reflect social components of user-generated content and extract key phrases from multiple user observations.

As will be shown in the rest of this chapter, the use of spatio, temporal and thematic contextual cues in unison yield more informative key phrases than the use of baseline TFIDF type novelty indicators alone.

Spatio-temporal attributes of content also serve as contextual cues in normalizing variations in user expressions. Consider the user comment "This is sad news about the King", that is in fact a reference to Michael Jackson, the 'King of Pop'. Knowing that the content was generated around the time of his death is valuable in associating the word 'King' with Michael Jackson and not Martin Luther King!

We also find that taking these three dimensions into account while processing, aggregating, connecting and visualizing social data provide useful organization and consumption principles for real-time data on the social Web.

Thesis Contributions:

The key phrase extraction algorithm presented in this chapter makes the following contributions:

- The algorithm summarizes *multiple* observations or documents on *social media platforms* that describe any topic of social significance.

- In addition to using textual context indicators for extracting key phrases (as has been the focus of past work), our algorithm investigates the role of contextual cues that are external to text and highly indicative of social signals, such as *when* or *where* the content was generated. We show that utilizing 'social' components of content not only extracts more meaningful key phrases but also preserves the cultural logic behind the multiple viewpoints expressed by users.

- The presented algorithm compensates for the informal nature of user-generated content, particularly in normalizing for the creativity, inter-personal off-topic chatter and redundancy in user expressions.

4.1.2 Key Phrase Extraction - Approach Overview

We approach the key phrase extraction task as a two step process – selection of candidate key phrases and subsequent elimination of off-topic key phrases.

Step 1: A key phrase is defined a sequence of words or *n-grams* that is present in a user comment. Table 4.2 shows an example of n-gram phrases for $n = 3$ extracted from a user comment.

Every phrase is weighted according to its *thematic, spatial and temporal* importance in a corpus of user-generated comments relevant to the topic. This process of extracting and scoring phrases is repeated for every user comment in the corpus. Phrases that have higher 'spatio-temporal-thematic' scores across the entire corpus are supported by multiple user comments and

User Post:
President Obama in trying to regain control of the health-care debate will likely shift his pitch in September
Extracted n-gram phrases:
1-grams: President, Obama, in, trying, to, regain, ...
2-grams: 'President Obama', 'Obama in', 'in trying', 'trying to'...
3-grams: 'President Obama in', 'Obama in trying'; 'in trying to'...

Table 4.2: Sample n-gram phrases extracted from a single user post, for $n = 3$

are considered more descriptive of the topic under investigation.

Step 2: Weighted phrases are ranked according to their 'spatio-temporal-thematic' scores and the top X phrases are chosen as summaries if a threshold based approach is preferred. For applications that require high precision phrases, off-topic n-gram phrases are eliminated using a mutual-information based approach to identify phrases that are strongly related in contexts relevant to the topic and eliminate phrases that are weakly relevant. While key phrase contextual relevance is computed for the topic under focus, the approach for elimination is domain-independent.

4.1.3 Key Phrase Extraction - Selection

The first step in the proposed key phrase extraction algorithm is to extract and score phrases that are descriptive of the topic. Fundamental to extracting key phrases that *preserve the social perceptions* is a simple intuition that different events and topics have different spatial and temporal biases that need to be taken into account while processing observations pertaining to them.

For example, when summarizing the 2008 Mumbai terror attack event, there might be interest in looking at *country level* activity on a *daily* basis. For longer running events like the financial crisis, applications might be interested in looking at *country level* activity on a *weekly* basis. User

Figure 4.1: Showing steps in the extraction of topical key phrases from user-generated content around a topic or event of interest.

comments around a topic that are grouped by these spatial and temporal preferences (e.g., separating comments from India and Pakistan on the same day) will tend to reflect current and local social perceptions.

The algorithm for key phrase selection and weighting assumes a corpus of relevant user-generated content made available to it. This can be achieved using a number of ways from simple key word selection methods (e.g., all documents mentioning the phrase 'health care reform' are related to the topic) or using a corpus of pre-categorized documents.

Given a topical corpus of user-generated content, the key phrase extraction algorithm proceeds as follows (also see Figure 4.1):

1. Creating spatio-temporal sets of user comments: We first partition the volume of user content into spatio-temporal sets based on two tuneable parameters - the spatial parameter δ_s and the temporal parameter δ_t. Together these two define the granularity at which we are interested in extracting summaries. δ_s for example is defined to cover a spatial region - a continent, a country, city etc. Similarly, δ_t is defined along the time axis of hours, days or weeks. Depending on the spatial and temporal bias that an event has, the user picks values for δ_s and δ_t.

Using these two parameters, we slice our data into *Spatio-Temporal Sets* $S = \{S_1, S_2...S_n\}$ where n is the number of sets generated by first partitioning using δ_s and next using δ_t. If δ_s = 'country' and δ_t = '24 hour', observations are grouped into separate spatial (country) clusters.

Every spatial set is then divided further into sets that group observations per day, generating n spatio-temporal sets.

User-generated content is grouped in a spatio-temporal set depending on the values they have for their timestamps and location attributes. A spatio-temporal set can be represented as $S_i = \{T_i, \delta_{si}, \delta_{ti}\}$ where $T_i = \{t_1, t_2, ..\}$ is a set of tuples where $t_i = \{t_{id}, t_c, t_t, t_g\}$ such that $\forall t_i \in T_i$; $t_g \in \delta_{si}$ and $t_t \in \delta_{ti}$.

By processing sets in isolation for key phrases or event descriptors, we ensure that the signals present in one do not amplify or discount the effect of signals in the other sets – thereby extracting summaries reflective of the local social perceptions.

2. Extracting key phrases: Given a spatio-temporal set definition, we proceed to extract strong descriptors that are descriptive of the topic and are also local to this set. Here, we formalize the interplay between the three dimensions of space, time and theme and define functions that extract key phrases.

Considering each user-generated content or document t_i as a sequence of words, we define a descriptor in our work as a vector of n-grams (see Table 4.2). Each t_i can then be represented as a vector of word tokens $ngrams_i = \{w_1, w_2, ...\}$ where w_i is the weight of the i^{th} n-gram. w_i is quantified as a function of three scores:

- the n-gram's **thematic score** that reflects how important the n-gram is in the document and the corpus while also taking into account its variations (e.g. 'Obama' and 'president Obama' or 'big three' and the three companies 'Ford', 'GM' and 'Chrysler')

- the n-grams's **spatial score** that reflects the global vs. local significance of a n-gram key phrase

- and the n-gram's **temporal score** that reflects its most recent to past popularity in user discus-

sions.

Here we discuss each of these scores in more detail.

2A. Thematic Importance of a key phrase: We start by calculating the thematic score of an n-gram descriptor, $ngram_i(tfidf)$, as a function of its TFIDF score in addition to using the following heuristics that compensate for the informal nature of user-generated content and extract meaningful descriptors from volumes of data.

1. The n-gram's baseline TFIDF score is calculated as per the standard definition of TFIDF (Salton and Buckley (96))

The term frequency (TF) of a key phrase t_i normalized for long documents is defined as $\mathrm{tf_{i,j}} = \frac{n_{i,j}}{\sum_k n_{k,j}}$ where $n_{i,j}$ is the number of occurrences of the key phrase (t_i) in document d_j, and the denominator is the sum of number of occurrences of all n-grams (n restricted depending on application requirements) in document d_j. The inverse document frequency (IDF) is a measure of the general importance of the phrase in a corpus $\mathrm{idf_i} = \log \frac{|D|}{|\{d:t_i \in d\}|}$ where $|D|$ is the total number of documents in the spatio-temporal set and $|\{d : t_i \in d\}|$ is the number of documents where the key phrase t_i appears.

The TFIDF score of a key phrase is computed as $\mathrm{(tf\text{-}idf)_{i,j}} = \mathrm{tf_{i,j}} \times \mathrm{idf_i}$ and reflects how important a key phrase is in a collection of documents in a spatio-temporal set.

2. Following the intuition that descriptors with nouns in them are stronger indicators of meaningful entities, we parse a tweet using the Stanford Natural Language Parser (Klein and Manning (63)) and amplify the n-gram's TFIDF score by the fraction of words in the key phrase that are tagged as nouns.

3. The TFIDF score is also amplified based on the fraction of words in the key phrase that are

not stop words in order to discount phrases such as 'trying to get' that are less informative as a summary or descriptor.

4. Lower and higher-order n-grams that have overlapping segments (e.g., 'general' and 'general motors') and the same TFIDF scores are filtered by picking the higher-order n-gram.

Accounting for variability in user expression : Owing to the varied vocabulary used by posters to refer to the same descriptor, region specific dictions and evolving popularity of words, the above thematic score is not entirely representative of a n-gram's importance. Consider this scenario where the phrase 'Big Three' meant to refer to the three car giants 'GM', 'Ford' and 'Chrysler' was not used as frequently as the three words together or vice versa.

Ideally words occurring in similar contexts should receive the same importance irrespective of their frequency based significance that is reflected by their TFIDF scores. In other words, the importance of the phrase 'Big Three' should be as high as the phrases 'GM', 'Ford' and 'Chrysler', when used together.

While the presence of contextually relevant words should strengthen the score of a key phrase, we also need to pay attention to changing viewpoints in user perceptions and observations that may result in descriptors occurring in completely different contexts. If the usage of 'Ford' is not in the context of the 'Big Three', but surround its new 'Ford Focus' model, its presence should not affect 'Big Three's' importance.

Contextually Enhanced Thematic Score: Here, we describe how the thematic score of an extracted descriptor, 'Big Three' in the above example, is amplified as a function of the importance of its strong associations - 'General Motors', 'Ford' and 'Chrysler' and the association strengths between the descriptor and the associations.

The goal is to amplify the score of every key phrase with the score of other strongly associated key phrases. For sake of brevity, let us call the $ngram_i$ descriptor whose thematic score we are interested in affecting as the focus word fw and its strong associations as $C_{fw} = \{aw_i, aw_2...\}$. The thematic score of the focus word is then enhanced as:

$$fw(th) = fw(tfidf) + \sum assoc_{str}(fw, aw_i) * aw_i(tfidf) \tag{4.1}$$

where $fw(tfidf)$ and $aw_i(tfidf)$ are the TFIDF scores of the focus and associated word as per Step 3 in the previous section; $assoc_{str}(fw, aw_i)$ is the association strength between the focus word and the associated word. Here we describe how we find strong associations for a focus word and compute $assoc_{str}$ scores.

Our algorithm begins by first gathering all possible associations for fw and places it in C_{fw}. We define *associations* or the *context of a word* as thematically strong descriptors (trivially the top X n-grams in a document) that co-occur with the focus word in the given spatio-temporal corpus.

The goal is to amplify the score of the focus word only with the strongly associated words in C_{fw}. One way to measure strength of associations is to use word co-occurrence frequencies in language (25). Borrowing from past success in this area, we measure the association strength between the focus word and the associated words $assoc_{str}(fw, aw_i)$ using the notion of point-wise mutual information in terms of co-occurrence statistics. We measure $assoc_{str}$ scores as a function of the point-wise mutual information (PMI) between the focus word and the *context of* aw_i. This is done to ensure that the association strengths are determined *in the contexts* that the descriptors occur in.

Let us call the contexts for aw_i as $Caw_i = \{caw_1, caw_2..\}$, where caw_k's are thematically strong descriptors that collocate with aw_i. $assoc_{str}(fw, aw_i)$ is computed as:

$$assoc_{str}(fw, aw_i) = \frac{\sum_k (pmi(fw, caw_k))}{|Caw_i|}, \forall caw_k \in Caw_i$$

where the PMI between fw and caw_k (the context of aw_i), is calculated as:

$$pmi(fw, caw_k) = log\frac{p(fw, caw_k)}{p(fw)p(caw_k)} = log\frac{p(caw_k|fw)}{p(caw_k)} \tag{4.2}$$

where $p(fw) = \frac{n(fw)}{N}; p(caw_k|fw) = \frac{n(caw_k,fw)}{n(fw)}$; $n(fw)$ is the frequency of the focus word; $n(caw_k, fw)$ is the co-occurrence count of words caw_k and fw; and N is the number of tokens. All statistics are computed with respect to the corpus defined by the spatio-temporal setting. As we can see, this score is not symmetric. Lower the relatedness between the context of aw_i and fw, lower the $assoc_{str}(fw, aw_i)$ score.

At the end of evaluating all associations in C_{fw}, we pick those descriptors whose association scores are greater than the average association scores of the focus word and all associations in C_{fw}.

The thematic weights of these associations along with their strengths are plugged into Eqn 4.1 to compute the enhanced thematic score $ngram_i(th)$, of the n-gram descriptor. This process is repeated for all top X key phrases in a spatio-temporal corpus such that the TFIDF score of contextually similar phrases is increased proportionately.

B. Temporal Importance of a key phrase: While the thematic scores are good indicators of what is important in a spatio-temporal set of documents, certain descriptors always tend to dominate discussions. In order to allow for less popular, possibly interesting descriptors to surface, we

discount the thematic score of a descriptor depending on how popular it has been in the recent past. The temporal discount score for a n-gram, a tunable factor depending on the nature of the event, is calculated over a period of time as:

$$ngram_i(te) = temporal_{bias} * \sum_{d=1}^{D} \frac{ngram_i(th)^d}{d}$$

where $ngram_i(th)^d$ is the enhanced thematic score of the descriptor on day d, D is the duration for which we wish to apply the dampening factor, for example, the recent week. However, this temporal discount might not be relevant for all applications. For this reason, we also apply a $temporal_{bias}$ weight ranging from 0 to 1 - a weight closer to 1 gives more importance, while a weight closer to 0 gives lesser importance to past activity.

C. Spatial Importance of a key phrase: We also discount the importance of a descriptor based on its occurrence in other spatio-temporal sets. The intuition is that key phrases that appear in user content from all spatial regions (e.g., every country) are not as interesting compared to those that occur only in the spatio-temporal set of interest (e.g., local region). We define the spatial discount score for an n-gram as a fraction of spatio-temporal sets that mention a key phrase.

$$ngram_i(sp) = \frac{k}{|spatio - temporalsets|} * (1 - spatial_{bias})$$

where k = number of spatio-temporal sets the n-gram occured in. Similar to the temporal bias, we also introduce a $spatial_{bias}$ that gives importance to local vs. global activity for the descriptor on a scale of 0 to 1. A weight closer to 1 does not give importance to the global spatial discount while a weight closer to 0 gives a lot of importance to the global presence of the descriptor.

Depending on the topic or event of interest, both these discounting factors can also vary for different spatio-temporal sets. For example, when processing tweets from India for the Mumbai attack setting the $spatial_{bias}$ to 1 eliminates the influence of global social signals. While processing tweets from the US, one might want a stronger global bias given that the event did not originate there. Both these parameters are set before we begin the processing of observations.

Spatio-Temporal-Thematic score of a key phrase: Finally, the spatial and temporal effects are discounted from the final score, making the final spatio-temporal-thematic (STT) weight of the n-gram as

$$w_i = ngram_i(th) - ngram_i(te) - ngram_i(sp) \tag{4.3}$$

Figure 4.2 illustrates the effect of our enhanced STT weights for extracted key phrases or event descriptors pertaining to the 2008 Mumbai terror attack event, in the US on a particular day. User content around the topic were curated from `www.twitter.com`; we used a temporal bias of 1 suggesting that past activity was important and a spatial bias of 0 giving importance to the global presence of the descriptor. As we see, descriptors generic to other spatial and temporal settings (e.g., mumbai and mumbai attacks) get weighted lower, allowing the more interesting ones to surface higher. Figure 4.3 shows top 15 extracted descriptors in the US across five days (days that had at least three citizen observations) giving us a sense of how key user perceptions around the same topic evolved over a period of 5 days.

mumbai	1.4553	pakistan pres promised	1.0065
photographers capture images	1.3998	mumbai attacks	0.9594
images of mumbai	1.2792	foreign relations	0.9490
foreign relations perspective	1.2165	rejected evidence	0.8741
attacks in mumbai	1.1261	evidence provided	0.8741
photographers capture	1.0986	uk indicating	0.8741
capture images	1.0986	mumbai attacks in	0.7927
india prime minister	1.0839	rejected evidence provided	0.7916
country of india	1.0280		

Event descriptors sorted by their TFIDF scores

foreign relations perspective	1.7185	photographers capture images	1.3028
india prime minister	1.5853	rejected evidence provided	1.2933
country of india	1.5295	mumbai attacks	1.2048
pakistan pres promised	1.5080	images of mumbai	1.1822
foreign relations	1.4510	mumbai	1.1083
rejected evidence	1.3758	mumbai attacks in	1.0797
evidence provided	1.3758	photographers capture	1.0017
uk indicating	1.3758	capture images	1.0017
attacks in mumbai	1.3293		

Event descriptors sorted by their enhanced spatio-temporal-thematic scores

Figure 4.2: Key phrases extracted from tweets on `www.twitter.com` around the 2008 Mumbai Terror Attack event and sorted by their TFIDF vs. spatio-temporal-thematic scores.

Day1	Day2	Day3	Day4	Day5
world india blasts	november 2008 donation	liking	mumbai outfit bjp	foreign relations perspective
blasts pakistani denial	manmohan singh	terror outfit	terror backlash	india prime minister
india blasts pakistani	solution india	work	india network	country of india
world india	2008 donation page	teachings	paki gov	pakistan pres promised
attack mahal	donation page provided	assisting terror outfits	punjabi taliban	foreign relations
denial on terrorism	newspakistan seeks	mafia assisting terror	obama aide	rejected evidence
hotels scarred	closure	statesman terror mafia	backlash	evidence provided
month attack mahal	rtt news voice	india expert	cripple	uk indicating
terror hotels	long term damage	terrorist attacks awaken	earthquake dia plane	attacks in mumbai
nri family	economy manmohan singh	teachings exposed	strife china earthquake	photographers capture images
alleged terror attacker	attacks in mumbai	attacks awaken bollywood	terrorism pakistan	rejected evidence provided
a month attack	quickfix solution india	film stars dictate	thing terrorism	mumbai attacks
defiant terror inches	seeks closure	terror mafia assisting	war on terror	images of mumbai
terror attacker seeks	terror attack	revision of history	a mumbai outfit	mumbai
newsnri family drawn	manmohan singh rtt	efforts asks china	attack punjabi taliban	mumbai attacks in

Figure 4.3: Top 15 key phrases extracted from tweets on `www.twitter.com` from the US for the 2008 Mumbai Terror Attack event, across 5 consecutive days

4.1.4 Key Phrase Extraction - Elimination of Off-topic Phrases

Several social media platforms such as message boards and questioning answering forums encourage user discussions making off-topic chatter common-place. While the use of frequency based measures as heuristics for selecting discriminatory features surfaces popular descriptive phrases, it does not eliminate off-topic discussions that are also popular.

For example, consider comments curated from MySpace forums for activity surrounding the music artist Madonna. Discussions about Madonna's personal life are as common and sustained as

discussions around her music. Figure 4.4 shows the top-5 most common terms used in association with Madonna during the same time period, showing the volume of tweets. This time period includes the release of Madonna's latest single, titled 'Celebration', discussions of her upcoming shows and concerts, and the release of several photographs of Madonna with her boyfriend, Jesus. Applications interested in mining discussion only surrounding an artist's 'music' popularity will not be interested in extracting other such popular but irrelevant key phrases.

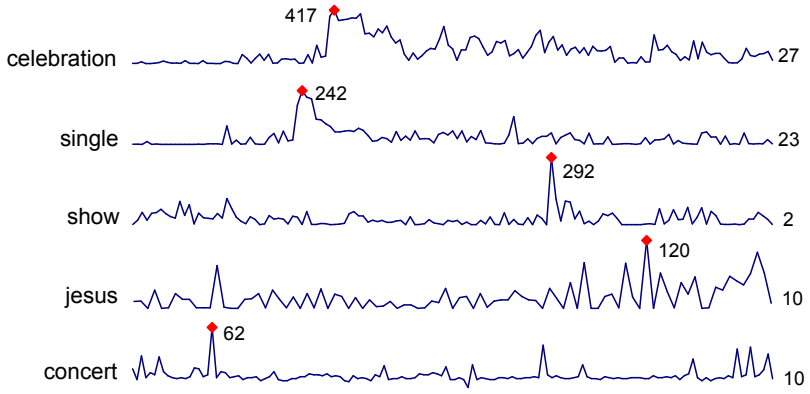


Figure 4.4: Showing volume of tweets per hour for the five most common terms used in association with Madonna. For applications interested in mining her music popularity, several of these terms are off-topic.

In this section, we present our algorithm that starts with key terms or phrases extracted from a document or corpus (for e.g., using the method described above) and eliminates key phrases that are off-topic for the application or topic under focus.

One possible approach to identifying topical key phrases from a group of phrases is to use clustering techniques to group phrases that have strong semantic associations with one another, namely words that are called to mind in response to a given stimulus, thereby separating strongly related and unrelated phrases. Word clusters created using co-occurrence based association strengths have been used in the past for assigning words to syntactic and semantic categories, learning lan-

guage models etc (for example, see Brown *et al.* (17)). Similar techniques exist for measuring lexical cohesion in text that exploit the idea that phrases that often occur together tend to be semantically related (Halliday and Hasan (47)).

However, generating semantically cohesive key phrase clusters alone do not indicate which clusters are relevant to the topic under focus for our end task of eliminating off-topic phrases.

Approach Overview: Our algorithm for eliminating off-topic phrases starts with a simple heuristic of assuming one or more seed keywords or phrases that are unambiguously representative of the topic. These could be provided manually, for example, 'tsunami alarm system' and 'earthquake' are fairly representative of the 2009 Tsunami event. Using such keywords as stimulus, our algorithm expands the relevant context by including only those extracted key phrases that are strongly associated with the seed keywords.

Our clustering algorithm starts by placing all seed keywords in cluster $C1$ and extracted key phrases in cluster $C2$. The idea is to gradually expand $C1$ by adding phrases from $C2$ that are strongly associated with $C1$.

At every iteration, the algorithm measures the change in Information Content (IC) of $C1$, $IC(C1, k_i)_\delta$, before and after adding a key phrase k_i from $C2$ to $C1$. The key phrase that results in a positive and minimum $IC(C1, k_i)_\delta$ score is added to $C1$ and removed from $C2$. Additionally, phrases resulting in negative $IC(C1, k_i)_\delta$ scores are discarded as off-topic. The algorithm terminates when all phrases in $C2$ have been evaluated or when no more phrases in $C2$ have positive $IC(C1, k_i)_\delta$ scores (no strong associations with $C1$).

Word association strengths are measured using the information theoretic notion of mutual information. Word co-occurrence counts are obtained from the spatio-temporal set of user com-

1. Seed keyword/phrase: **C1**: ['camcorder']
2. Main Post: yeah i know this a bit off topic but the other electronics forum is dead right now. im looking for a good camcorder, somethin not to large that can record in full HD only ones so far that ive seen are sonys
Reply: Canon HV20. Great little cameras under $1000.
C2: ['electronics forum', 'hd', 'camcorder', 'somethin', 'canon', 'little camera', 'canon hv20', 'cameras', 'off topic']
3. $\mathbf{IC}(\mathbf{C1}, k)_\delta$ scores of$C1$and C2 keywords:

['camcorder', 'canon']	:0.00015
['camcorder', 'canon hv20']	:0.000011
['camcorder', 'cameras']	:0.00009
['camcorder', 'hd']	:0.000079
['camcorder', 'little camera']	:0.000029
['camcorder', 'electronics forum']	:-0.00000006
['camcorder', 'somethin']	:-0.0000000012
['camcorder', 'off topic']	:-0.000000019

4. Eliminated Keywords: ['somethin', 'off topic', 'electronics forum']
5. Final C1 using maximally constrained contexts: ['camcorder', 'canon hv20', 'little camera', 'hd', 'cameras', 'canon']
6. Final C1 using minimally constrained contexts: ['camcorder', 'canon', 'cameras']

Table 4.3: Eliminating off-topic noise and reaching contextual keywords

ments. Alternate means of obtaining co-occurrence counts are tapping into the Web (Keller and Lapata (61)) with the risk of biasing counts by co-occurrences in all and possibly varied contexts.

First, we describe preliminaries and then detail the clustering algorithm using an example shown in Table 4.3.

Preliminaries: The algorithm starts by adding every keyword from $C2$ to $C1$ and measuring the change in Information Content (IC) of $C1$. $IC(C1)$ is the strength of the semantic associations between words in the cluster and is defined as the average pairwise Mutual Information (MI) of the words.

$$IC(C1) = MI(C1) / \begin{pmatrix} |C1| \\ 2 \end{pmatrix} \tag{4.4}$$

where $|C1|$ denotes the cardinality of the cluster $C1$ and $\begin{pmatrix} |C1| \\ 2 \end{pmatrix}$ is the number of word pairs in the cluster $C1$, normalizing for clusters of different sizes. $MI(C1)$ is the Mutual Information of cluster $C1$, defined as the sum of pairwise Mutual Information of words within the cluster.

$$MI(C1) = \sum_{w_i, w_j \in C1, i \neq j} MI(w_i, w_j) \tag{4.5}$$

Recall that w_i or w_j can be a single word or a phrase. The MI of words $w_i, w_j \in C1$ measures their association strength in terms of their co-occurence statistics. It is defined as the point-wise realization of the MI between two random variables W_i and $W_j \in V$, a vocabulary of words (see Church and Hanks (25)).

$$\begin{aligned} MI(w_i, w_j) &= p(w_i, w_j) \log \frac{p(w_i, w_j)}{p(w_i)p(w_j)} \\ &= p(w_i)p(w_j|w_i) \log \frac{p(w_j|w_i)}{p(w_j)} \end{aligned} \tag{4.6}$$

Standard definition for point-wise mutual information ignores the joint probabilty term, $p(w_i, w_j)$ in (3). We keep this term to ensure the consistency of (2). Here, $p(w_j|w_i)$ is the probability of w_j co-located with word w_i (preceeding or following) within a window. Unlike standard bi-gram models in language modeling that require words to occur in a sequence, we do not care about word order. Maximum likelihood estimates of the parameters are calculated as

$$p(w_i) = \frac{n(w_i)}{\mathcal{N}}; p(w_j|w_i) = \frac{n(w_i, w_j)}{n(w_i)} \tag{4.7}$$

where $n(w_i)$ is the frequency of word w_i in the spatio-temporal set; $n(w_i, w_j)$ is the co-occurrence

count of words w_i and w_j; $\mathcal{N}$ is the number of tokens in the spatio-temporal set.

Plugging (4) into (3), we have the MI between two words as shown in (5). This measure is symmetric, i.e., $MI(w_i, w_j) = MI(w_j, w_i)$. When $n(w_i, w_j) = 0$, we define $MI(w_i, w_j) = 0$.

$$MI(w_i, w_j) = \frac{n(w_i, w_j)}{\mathcal{N}} \log\left(\frac{n(w_i, w_j)\mathcal{N}}{n(w_i)n(w_j)}\right) \tag{4.8}$$

As phrase k_i is added from $C2$ to $C1$, the change in Information Content of $C1$ is measured as

$$IC(C1, k_i)_\delta = IC(C1, k_i) - IC(C1) \tag{4.9}$$

where $IC(C1, k_i)$ is the information content of $C1$ after adding k_i from $C2$. $IC(C1, k_i)_\delta$ is *positive* when k_i is *strongly associated* with words in $C1$ and *negative* when k_i is *unrelated* to words in $C1$. Bullet 3, Table 4.3 shows the computed $IC(C1, k_i)_\delta$ scores for words in $C2$ at the end of the first iteration.

At this time, the algorithm eliminates keywords that result in negative $IC(C1, k_i)_\delta$ scores (Bullet 4). This is done only at the first iteration when $C1$ has only seed keywords. The intuition is that if phrases are unrelated to the context-indicating seed keywords, they will not contribute to subsequent steps that build the seed keyword cluster.

Next, the phrase that results in a positive and minimum $IC(C1, k_i)_\delta$ score, 'canon hv20' in this example, is greedily added to $C1$. The reasoning behind the pick is as follows. A phrase k_i occuring in specific contexts with words in $C1$ will increase the Information Content of the $C1$ relatively less than a phrase that occurs in generic contexts. For example, if $C1$ has the words 'speakers', keyword 'beep' that occurs in maximally constrained or specific contexts of malfunc-

tioning speakers will have lower association strengths with $C1$ compared to a word 'logitech' that occurs in minimally constrained or broader contexts with 'speakers'.

As the algorithm continues, the keyword occurring in a *maximally constrained context* with $C1$ is removed from $C2$ and added to $C1$ at every iteration. This strategy has the tendency of adding specific to general keywords from $C2$ to $C1$ (see Bullet 5). The alternate strategy is to greedily add the keyword that occurs in minimally constrained or generic contexts with $C1$. This tends to pick generic keywords first and runs out of keywords that add to the Information Content of $C1$ (see Bullet 6). An application interested in extracting several related and specific key phrases will opt the first strategy while those interested in fewer general can opt for the second strategy.

Algorithm Complexity: Using seed keywords as starting points reduces the context space from all keywords to a few seed keywords. The best case running time of our algorithm is $O(MN)$ where $M = |C1|$, size of the title cluster and $N = |C2|$, size of the content cluster. Best case scenario occurs when all keywords in $C2$ are off-topic or only one $C2$ keyword is contextually relevant. One iteration of the algorithm after computing MN association strengths suffices to partition relevant and noisy keywords. Worst case complexity is $O(MN^2)$ when there are no off-topic keywords and the algorithm has to evaluate all N keywords in $C2$ one after another, computing MN association strengths at every step, for N iterations. It is possible that multiple words resulting in similar Information Content change scores in the same iteration can be added to $C1$ to reduce the time complexity of the algorithm.

4.1.5 Experiments and Evaluation

The process of selecting from a document what the topical and non-relevant phrases are, is inherently a subjective process that can get even challenging when domain expertise is required. Unlike datasets such as the computer science papers from the Computer Science Technical Reports collection of the New Zealand Digital Library Project [1] that have been used to evaluate key phrase extraction algorithms, there are no benchmark datasets for social media content for us to evaluate our contributions in topical key phrase extraction over informal text.

Evaluating the key phrases extracted by our algorithm will require us to have a bench-marked dataset where multiple annotators have marked phrases as relevant or not to the topic of the document. With such a dataset, we can evaluate the coverage or recall (how many of the phrases marked as relevant did our algorithm extract) and the precision (how many of the phrases extracted by our algorithm were indeed marked as relevant by the annotators).

In the absence of such a dataset, we measure the effectiveness of the extracted key phrases via a qualitative social web application and a content delivery task that uses the extracted key phrases as search/index terms.

4.1.5.1 Evaluating Extracted Key Phrases for Browsing Real-time Data on the Web

The first qualitative evaluation of our key phrase extraction algorithm is a deployed social Web application that uses spatial, temporal and thematic contexts implicit in the user-generated tweets (posts on Twitter) to extract summaries of social perceptions behind real-time, news-worthy events. Since Twitter restricts user content to 140 bytes, the chances of off-topic content are rare. We

[1] See `http://www.nzdl.org/`.

Figure 4.5: Showing a snapshot of Twitris components. Pane 1 shows the temporal navigation; Pane 2 shows n-gram key phrase summaries extracted from tweets originating in Florida around the Health Care Reform debate; Pane 3,4 and 5 show semantically related content pulled from Twitter, Google news and Wikipedia for the phrase 'Health Care Reform'.

employ only the key phrase extraction (without the elimination) algorithm to generate thematic summaries or n-gram tag clouds of tweets and also separating perceptions across time and space. Extracted social summaries are further used to pull semantically related content from the Web and provide richer contexts to the end user. Figure 4.5 shows a snapshot of the Twitris application where a user is presented with a tag cloud summary of activity around the health care debate on a particular day from the state of Florida in the US.

Twitris currently contains more than two million processed tweets surrounding world events in 2009 (see Table 4.4). Twitris also acts as a test bed for browsing of real-time data and was recently featured at the Semantic Web Challenge at the International Semantic Web Conference 2009 (see `http://challenge.semanticweb.org/`).

Event Name:	**Iran Election**
Number and type of spatial clusters:	23 countries
Date range	06/12/2009 - 06/30/2009
Event Name:	**Health Care Reform**
Number and type of spatial clusters:	44 states in the USA
Date range:	08/19/2009 - 09/28/2009
Event Name:	**Copenhagen Climate Conference**
Number and type of spatial clusters:	70 countries
Date range:	12/09/2009 - 01/09/2010
Event Name:	**2009 South-east Asian Tsunami**
Number and type of spatial clusters:	17 countries
Date range:	10/01/2009 - 10/09/2009
Event Name:	**Swine flu**
Number and type of spatial clusters:	26 states in the USA
Date range:	10/08/2009 - 10/21/2009

Table 4.4: Current version of Twitris provides browsing of spatio-temporal-thematic summaries extracted from more than 2 million tweets surrounding news-worthy events in 2009

4.1.5.2 Evaluating Extracted Key Phrases for Targeted Content Delivery

As a second evaluation, we quantitatively evaluate the effectiveness of the extracted topical key phrases for a content delivery task that uses the key phrases as search/index terms. We measure if the phrases extracted by our work are representative of the 'aboutness' of the content, and therefore are effective as index terms or search queries.

Experiment Set Up: For these experiments we focus on 12000 user-generated posts for the topic of Computers, Electronics and Gadgets on MySpace forums. These forums are characterized by verbose and off-topic discussions that make it an ideal dataset for evaluating our contributions in topical key phrase extraction.

We begin with extracting key phrases descriptive of the user posts. Since the spatial and temporal parameters are not relevant to the end task, we only focus on the thematic components of the

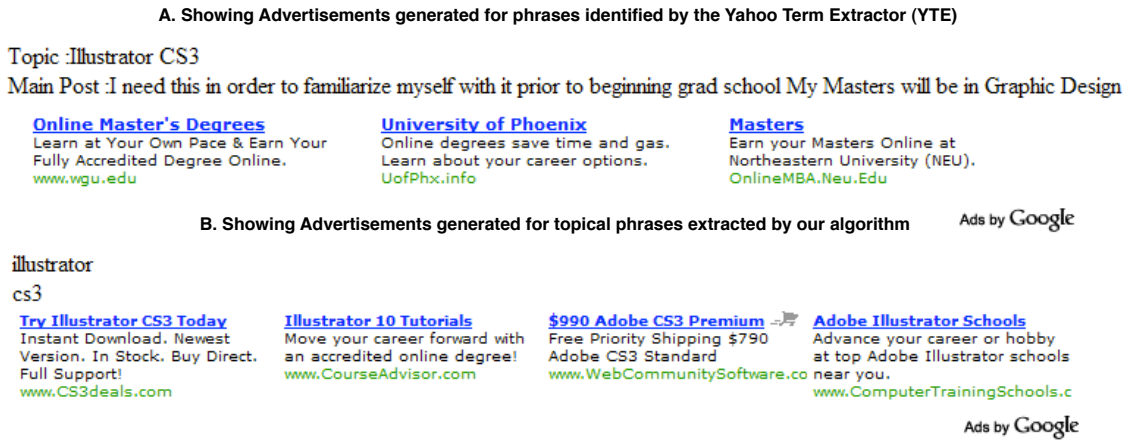


Figure 4.6: Showing a snapshot of advertisements generated against key phrases extracted using YTE over the content and against key phrases extracted using our algorithm.

key phrase extraction (refer to Section 4.1.1). This is followed by the off-topic phrase elimination algorithm that eliminates phrases that are not related to the topic of Computers, Electronics and Gadgets.

The extracted phrases are used as index terms for the user posts and supplied to a content delivery application to evaluate its effectiveness. In this experiment, we use Google AdSense ([2]) as the targeted content delivery application. The script takes text (full text, key words or phrases) as input and returns a list of sponsored advertisements that match the input text. See Figure 4.6 for an example of advertisements generated against content.

We use a random set of 60 posts for our user studies. The Yahoo Term Extractor (YTE) [3], an off-the-shelf keyword extraction service built over Yahoo's search API is used as the baseline for key phrases. YTE takes as input a text snippet and returns key words and phrases in text. Table 4.5 shows an example from YTE's online web interface demo.

[2]see `https://www.google.com/adsense/`
[3]see `http://developer.yahoo.com/search/content/V1/termExtraction.html`

Content: Italian sculptors and painters of the renaissance favored the Virgin Mary for inspiration.
Phrases extracted by the Yahoo Term Extractor: italian sculptors, virgin mary, painters, renaissance, inspiration

Table 4.5: Sample request and response to Yahoo Term Extractor's service.

The phrases generated by the YTE engine are sent to the Google AdSense script to return a set of targeted ads, denoted as Ads_c. A similar experiment is carried out using the key phrases extracted by our algorithm and used as index terms for the user posts. The set of advertisements generated using the extracted key phrases is denoted as Ads_k. Each user post has a maximum of 8 ads, 4 in each set. The effectiveness of the two sets of advertisements are compared via user studies where 3 users are asked to judge the relevance of the advertisements to the post.

The intuition is that index terms or key phrases that are more representative of the content will generate advertisements more targeted to the content of the post.

Results: Users responded by picking advertisements that they thought were relevant to the post. We aggregated responses for the 60 posts by counting the number of advertisements that users picked from each set. We counted only advertisements that two or more evaluators picked to ensure at least a 50% inter-evaluator agreement. Table 4.6 shows statistics for the total number of advertisements displayed for the key phrases extracted using YTE and our algorithm and the number of advertisements users picked as relevant from the two sets.

Users thought that 52% of the advertisements shown using keywords returned by our algorithm were relevant, compared to the 30% of relevant advertisements generated using the baseline key phrases generated by YTE. For several posts, Ads_c and Ads_k had advertisements in common. A more accurate measure of user feedback is the number of advertisements that were deemed relevant

Using key phrases extracted by the Yahoo Term Extractor (YTE)	
Number of ad impressions	144
Number and % of advertisements picked as relevant	43, 29.8%
Number and % of Unique advertisements picked as relevant	25, 17.36%
Using keywords returned by our algorithm	
Number of ad impressions	162
Number and % of advertisements picked as relevant	85, 52.47%
Number and % of Unique advertisements picked as relevant	64, 39.5%

Table 4.6: Showing effectiveness of extracted topical key phrases as index terms for content delivery

and were unique to each set. Table 4.6 also shows these statistics. According to evaluator picks, the key phrases extracted using our algorithm led to 22% more targeted unique advertisements.

These results highlight the importance of the off-topic phrase elimination step for social media content. They are also indicative of the effectiveness of the extracted key phrases in describing the 'aboutness' of content and its use as indexing terms.

4.1.6 Related Work, Applications of Key Phrase Extraction

There are over 100 million active users of MySpace[4]. Facebook has over 400 million users[5] and Twitter is growing at over 1300% a year[6]. Social networks produce more data every day than most companies see in a year. It is not surprising that with roughly the same number of users as the population of the United States, a wide variety of topics get discussed. With the ability to rapidly disseminate information, new topics can generate tremendous buzz in a matter of hours.

A substantial research and engineering effort is required to get a handle on this very large and

[4] http://www.myspace.com/pressroom?url=/fact+sheet/
[5] http://www.facebook.com/press/info.php?statistics
[6] http://blog.nielsen.com/nielsenwire/onlinemobile/twitters-tweet-smell-of-success/

rapidly evolving data set. This challenge is worth tackling as it enables us to tap into this wisdom of the crowds in near real-time.

In this chapter, we explored one approach to gauging the pulse of a populace around a significant event or topic under focus by analyzing multiple observations relayed from the same or different locations. Our goal is to facilitate summaries for situation awareness applications that concerns itself with 'knowing what is going on so you can figure out what to do'(Adam (1)).

Abstracting the multiple, often complementary and naturally appearing viewpoints requires attention to several social dimensions of content. The three-dimensional spatio-temporal-thematic extraction of summaries presented in this work is analogous to past efforts in processing of social stream, newswire or blog data where thematic, temporal, spatial and poster aspects of the data have been taken into account. In (Zhao *et al.* (119)), the goal was to extract events from social text streams taking content, social, and temporal aspects into account. An event in their work is a set of text pieces (topically clustered) conditioned by social actors that talk about the same topic over a certain time interval with similar information flow patterns. Work in (Yang *et al.* (117)) attempts to identify spatiotemporal thematic patterns in blog data. They extract common themes (semantically coherent topics defined over a text collection) from weblogs and subsequently generate theme life cycles for a given location and theme snapshots for a given time period. Work in (Kumar *et al.* (67)) used a graph-theoretic approach to discover story lines or latent themes among the top search results for a query.

In our work, we did not attempt to identify available latent themes, story lines or events in a given corpus of text. We start with a corpus of observations pertinent to an event and attempt to extract meaningful units that are good descriptors of the underlying event. We also took an

entity-driven approach to summarize social perceptions in citizen observations, as opposed to a document collection approach in past efforts.

There are several applications of this work in social awareness monitoring, tracking the pulse of a populace around a topic, in delivering local information systems etc. Chapter 5 of this thesis presents our efforts in using the results of this algorithm in browsing real-time data around news-worthy events. We present a deployed social intelligence Web application that gathers user-generated content from Twitter and processes the data along the theme, space and time axes to extract summaries of *what* people are saying from *when* and *where*.

5. Applications of Understanding User-generated Content

The tasks of Named Entity Recognition and Key Phrase Extraction that we presented in this thesis can be leveraged for many subgoals of Natural Language Understanding like sentiment analysis (Pang and Lee (89)), anaphora resolution (School *et al.* (97)) etc. The ability to quickly and easily examine the selections, comments and preferences of millions of users also provides a powerful environment for engineering crowd-sourced 'Social Intelligence' applications for sense-making, anthropological studies and studying the evolution of communities, topics and even color preferences (Locke (73)).

It is important to remember, however, that not all crowds might be the right source to tap for a social intelligence application. While 'crowds' may provide important insight into what music is popular, they may not be the right source for an opinion on how to perform a tricky bypass surgery (again, depending on the crowd). Making sure you are asking the right crowd the right question is very important, and there are still some situations where even crowds of experts do poorly, e.g., stock market bubbles. This highlights the need to test the results of a Social Intelligence application against real world proof points before putting too much faith in them.

Thesis Contributions

In this thesis we present two Social Intelligence applications of 'popularity charting' and 'social perception monitoring' that draw on different aspects of crowd intelligence from a variety of modalities and data sources.

The first application we describe is the BBC SoundIndex, a dashboard system that provides near real-time information about the popularity of various artists, tracks and genres of music by analyzing the comments of music listeners on social networking online communities such as MySpace. The second Web application, Twitris, uses spatial, temporal and thematic contexts associated with user-generated content on Twitter to extract summaries of social perceptions behind real-time events. Twitris provides an alternative to browsing scores of tweets generated around a popular trending event by presenting extracted social summaries of citizen reports from Twitter and overlaying it with Wikipedia and news articles to facilitate contextual browsing.

In both these applications, we primarily focus on *user generated textual content* and explore the opportunities that Social Intelligence systems can provide in building socially aware and personalized applications by mining textual social data. We discuss the challenges we faced in gathering relevant social data and in processing informal user-generated content that is laden with broken English sentences, spam, and domain-specific slang; and describe the experiments performed to evaluate the success of the applications.

5.1 Mining Online Music Popularity

The first Social Intelligence application we present in this thesis measures music popularity by mining music-enthusiasts' comments on artist pages on MySpace – a popular online music community [1]. The goal is to create a Top-N list of popular artists that reflects the preferences of online listeners as opposed to the Top-N list from `Billboard.com` where popularity charts are based on airplay and sales [2].

5.1.1 Vision and Motivation

The BBC had noticed an increasing drift in the artists and songs reported in popular music charts and what their subject matter experts (i.e., DJs) thought. This is not surprising as the methodology for generating these charts dates back over half a century. In the 1950's and 1960's, record sales and radio plays were good predictors of music popularity. Since both recording music onto phonograph records and broadcasting music over radio waves required specialized machinery, it was safe to presume that any recorded music being listened to came from one of these sources. Simply counting the number of records sold and songs played acted as a reasonable proxy for what people listened to (e.g., Billboard.com).

While counting is the goal, in reality these numbers are derived from polling a relatively small number of record stores and radio stations. Challenges in polling are well known (Pliny the

[1] www.myspace.com

[2] From Wikipedia: The Billboard Hot 100 is the United States music industry standard singles popularity chart issued weekly. Chart rankings are based on airplay and sales; the tracking-week for sales begins on Monday and ends on Sunday; while the airplay tracking-week runs from Wednesday to Tuesday. A new chart is compiled and officially released to the public by Billboard on Thursday. Each chart is dated with the "week-ending" date of the following Saturday (Wikipedia (112))

Younger wrote about them in 105 A.D.(6)). This raises the obvious concern — are the sample record stores really representative of the way most people get their music? Is the radio still the most popular medium for music? In 2007 less than half of all teenagers purchased a CD[3] and with the advent of portable MP3 players, fewer people are listening to the radio.

With the rise of new ways in which communities are exposed to music, the BBC saw a need to rethink how popularity is measured. Could the wealth of information in online communities be leveraged by monitoring online public discussions, examining the volume and content of messages left for artists on their pages by fans and looking at what music is being requested, traded and sold in digital environments? Furthermore, could the notion of "one chart for everyone" be replaced with a notion that each group might wish to generate their own charts reflecting the popularity of people like themselves (as the Guardian[4] put it — "What do fortysomething female electronica fans in the US rate?"). Providing the data in a way that users can explore the data of interest to them was a critical goal of the project.

5.1.2 Top N Lists

Top-N lists have been a fascination of people since at least the fifth century BC when Herodotus published his "Seven Wonders of the World" (113). From the superlatives in a high school yearbook to political polling, a community defines itself in part by ranking interests and preferences. In areas such as music, ranking also serves as a means of providing recommendations. For instance, a new artist appearing on a "Top Artists" Techno chart may be popular with fans of other Techno artists that appear on the list. Of course, this has non-trivial sales implications, so using

[3] NPD Group: Consumers Acquired More Music in 2007, But Spent Less

[4] http://www.guardian.co.uk/music/2008/apr/18/popandrock.netmusic

and manipulating chart position has long been a controversial part of marketing (McIntyre).

The challenge of determining the Top-N list has lead to a host of innovative approaches to popularity rankings. Some of the hardest domains are those where tastes change quickly, such as popular music. Music often suffers from "over play" fatigue, where popular songs are played so frequently that they cease being as popular. As a result, attempts have been made to identify reasonable objective, observable proxies for interest.

5.1.3 Proxies for Popularity

The advent of records and radio as the primary means of distribution of popular music presented a reasonable "choke point" for such measurement. Since both recording music onto gramophone records and broadcasting music over radio waves required specialized machinery, it was safe to presume that any recorded music being listened to came from one of these sources. Simply counting the number of records sold and songs played acted as a reasonable proxy for what people listened to (e.g. `Billboard.com`).

While this may have been true in the 50's and 60's, record sales and radio plays have become increasingly poor predictors of what is popular in the face of rapidly increasing on-line music trading and downloads (both legal and illegal), device to device music sharing, on-line discussion forums targeting music (e.g., MySpace), Internet radio stations, etc. With the rise of new ways in which communities are exposed to music, comes the need to rethink how popularity is measured.

Another approach to measure popularity is conducting polls, i.e., asking people for their opinion. Challenges in polling are well known (Pliny the Younger wrote about them in 105 A.D.(6)).

For this domain, one of the largest problems is that polling large samples is problematic and expensive. Rather than directly asking people what they think, however, we can make use of the wealth of information in online communities. Popularity can now be determined by monitoring on-line public discussions, examining the volume and content of messages left for artists on their pages by fans and looking at what music is being requested, traded and sold in digital environments.

5.1.4 Thesis Contributions

This work measures music popularity by mining music-enthusiasts' comments on artist pages on MySpace. To comprehend the 'aboutness' of the user comments on MySpace and gauge poster sentiment toward a music entity, we had to first overcome challenges due to broken sentences, unconventional writing and spam. We mined comments for music related information and aggregated the results to create a Top-N list of popular artists.

To test the effectiveness of our ranking system, we also compared our top artists ranking to the Top-N list from `Billboard.com` and found that our system creates a closer connection between the popular lists and listener preferences. The net effect is more accurate and specific data, predicting today's top and upcoming artists, rather than reporting on last week's sales and airplay. In this chapter, we describe the motivation behind this work, the challenges involved in engineering this social intelligence applications and show how crowd-sourced information can reliably challenge traditional polling methods.

5.1.5 Corpus Details

Music popularity or opinions on music are the subject of many heated discussions in online communities. Our choice of online data corpora is motivated by two main factors: a target audience of teenagers, and a desire for music-centric content.

We choose teenagers because of their considerable effect on the overall popularity of music. A trio of industry reports around the effect of communities on music consumption identifies the growing population of online teenagers as the biggest music influencers; 53% of which is also spreading word about trends and acting as primary decision-makers for music sales (MediamarkResearch). We exploit this majority's appreciation of music in online music communities to complement traditional metrics for ranking popular music. While our extended work in this area has used six online communities, we limited ourselves to MySpace for generating data in this paper to avoid the issues around multi-site ranking fusion.

MySpace

MySpace is a popular social networking site that has a section dedicated to music artists and fans. Both major, independent, and unsigned music artists have taken advantage of this popular and free-for-all social networking site to manage online relationships with fans. Members of this community, over half of whom are our target demographic, learn about latest artists, albums, and events, and express their opinions on the comments section of an artist's page. We crawl and mine comments from such artist pages to determine popularity numbers. In addition, we also gather user demographic information like comment poster's age, gender and location to derive demographic trends. Table 5.1 shows the crawled structured and unstructured data.

Type	Crawled Data
Structured	Artist Name, Albums, Tracks, Genre, Country, User, Age, Sex, Location
Unstructured	Posted comments mentioning Artist, Album, Tracks, Sentiments and Spam

Table 5.1: Description of Crawled Data

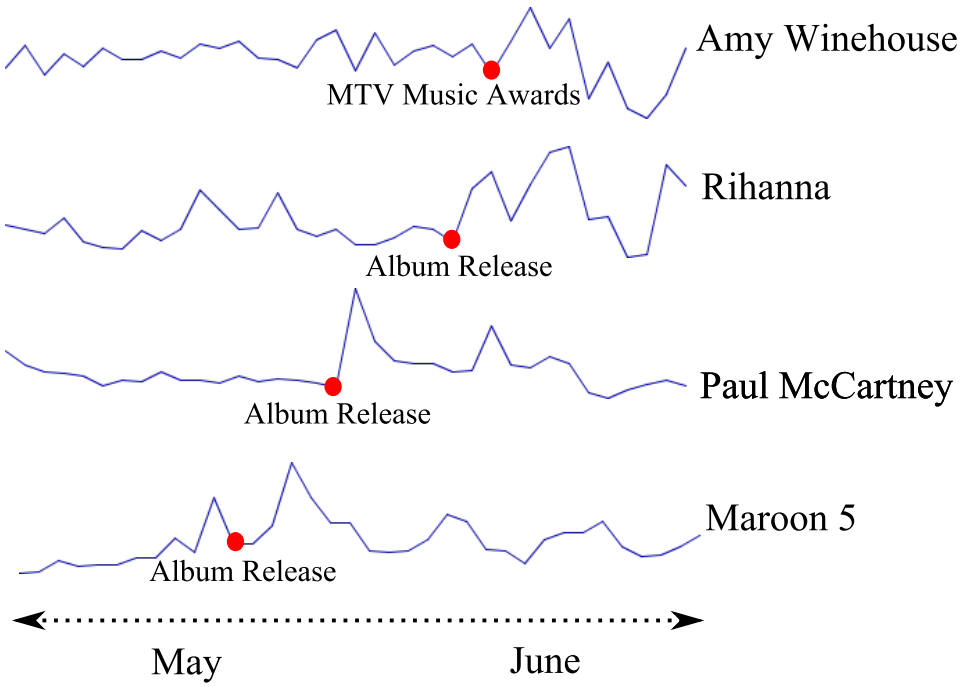


Figure 5.1: Spikes in comment volumes and rises in popularity occur after newsworthy events.

Corpus Characteristics

One of the salient features of such corpora is the availability of near realtime data. We show that it is possible to assess popularity trends, correlate chatter with external events (like artists winning awards) and identify the beginning and persistence of trends to enable marketing focus on early-adopter segments without the lag from sales data (which may take weeks to collect and aggregate). Over a period of 26 weeks (Jan through Jun 2007) 788,384 unique comments were observed for

the top 100 artists in this time frame. The volume of comments highlights the importance of a scalable crawling and mining system. Figure 5.1 illustrates spikes in comment volumes on artist pages that coincide with real events. The ability to gauge buzz and popularity the day after an artist releases an album or appears on television is invaluable to record labels as they attempt to sway the buying decisions and loyalties of fans.

We conducted some experiments on a random sample of 600,000 of these comments and observed the following characteristics of the unstructured component of the corpus:

- More than 60% of terms used to indicate sentiment contained slang that required special treatment.
- Less than 4% of the comments expressed negative sentiments about artists; comparative or sarcastic comments were rare occurrences. Detection of sentiments proved to be an important step in the process of spam detection.
- Almost 40% of comments on an artist's page were self-promotional or advertisement related spam. Spam comments were often less than 300 words long, and appreciative comments less than 100 words long.
- The natural language construction of over 75% of the non-spam comments was non-conventional, often resulting in inaccurate or failed linguistic parses.

Our annotator system, which is responsible for gleaning structure out of this unstructured content, effectively deals with these limitations by using a combination of statistical and linguistic techniques.

5.1.6 System Design

The basic design of the system, which we will describe in more detail in the remainder of this chapter, is as follows (also see Figure 5.2).

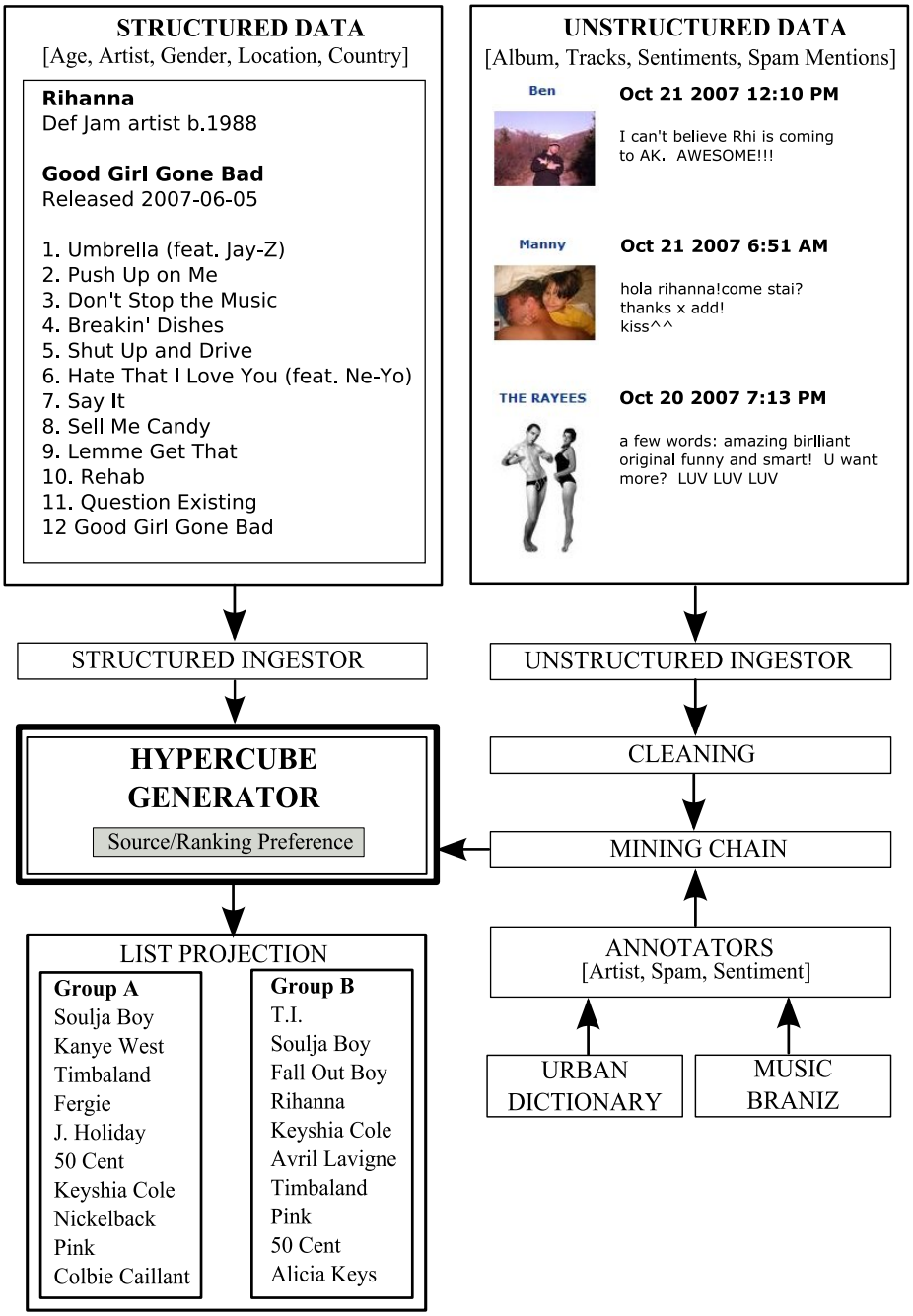


Figure 5.2: Basic design of the SoundIndex system

- Crawling: Fetching the data from the source site, transforming the pages and comments into common formats, and ingesting the data into the database.

- Annotating: Ingested comments are passed through a UIMA(38) chain of annotators to adjudicate if the comment is related to the artist or music, if it has any associated positive or negative sentiments, and if it is a spam comment.

- Hypercube construction: The data is rolled up by a variety of dimensions (e.g., age, gender, locale) and a summary hypercube of comment and sentiment volume is constructed.

- Projection to a list: Ultimately we want a Top-N list, so we need to project this hypercube to a single value which is used to order the list of artists, tracks, albums, etc.

We will explore each of these steps in turn, with particular attention to approach and lessons learned.

5.1.7 Crawling and Ingesting User Comments

The crawling and ingesting component gathers data from a potentially diverse set of sources and maps the data to a normalized format for further processing. It must do so in a way that is scalable to millions of comments and extensible to changes in the data sources and annotation schemes.

Given constrained data acquisition bandwidth, we need to prioritize how to examine the subparts of the site and assign a frequency with which to revisit each artist page. As an example, we might seed our set of artists to consider by looking at a "top artists" list from social networking sites such as MySpace, or from published top charts such as Billboard's "Top Singles" charts.

Given this seed list, we can then identify the artist pages for these candidates and begin to pull semantic information on fan preferences (i.e., comments) from these pages. For the sake of politeness we need to wait a few seconds between fetches to reduce the load on any given server and achieve sustainable crawling, but given a multiplicity of sites this does not impact overall crawl rate. Comment data, as described in Section 5.1.5, consists of a structured component such as artist name, a time stamp, the user demographics of the poster, plus an unstructured component (i.e., the comment text).

The list of artists can be quite extensive. There are nearly 50,000 artists in an initial set. With a politeness wait between requests, this means that one could only check a few hundred artists an hour and exhaustive rescans could take days. In the fast changing environment of a social network music community, rapidly emerging artists could be missed for extended periods. With a goal of obtaining a near real time pulse of the community the desiderata is a Top-N list once every 4 hours. Without allocating more bandwidth, this means new data representing only a couple thousand artists is possible. Fortunately, not all artists are commented on with the same frequency – thus crawling with a prioritization scheme is possible.

We use two data gathering schedules that arbitrarily split the available crawl bandwidth:

- Priority crawl: A process that scans roughly one thousand artist pages in 4 hour cycles. These are the artists have the highest variance/uncertainty in their comment incidence rate[5].

- Exhaustive crawl: A process that scans all the artists at the rate of about one thousand per

[5] Since only a small subset of artists is examined in our Priority crawl we create a simple estimator of the number of comments an artist would have at any time. Over several scans we can then create an estimate of error. We can then look at how long it has been since we last obtained firm data on a source to generate expected error, and then sort the priority crawl list to maximally reduce uncertainty. This equation can also be back solved to define requisite crawl rate for a given error bound.

hour. In each scan we collect all the comments generated since the last scan and generate new estimates for the comment rate and its uncertainty.

These techniques allow us to bring in the maximally useful comment stream, which is transformed via a site-specific remapping function into a normalized data format. This one step is by far the most brittle of the entire system, as it needs to deal with the site format and access pattern changes of the crawled sites.

Once the data is normalized it is stored in a relational database (DB2) using a data model that is easily extensible for future additions of data sources. Each comment is uniquely identified by a combination of user, data source, artist and time-stamp (best estimate or exact) values. We track artists across data sources, but we do not at this time link posters across data sources. Comment annotations are stored in an extensible schema of two tables: one storing the list of annotations and the other storing a comment, annotation pair per record. When a new annotation is generated for the existing comments only the new information can be added to the set of tables.

5.1.8 Annotation Components

The annotation component automatically processes comments to compute the total number of positive comments for each artist. We use the scalable, UIMA(38) based framework to host a short chain of three annotators:

- Artist and Music Annotator: Spotting artist, album, track, and other music related (e.g. labels, tours, shows, concerts) mentions.

- Sentiment Annotator: Spotting and transliterating sentiments in comments.

- Spam Annotator: Identifying comments that are spam or do not directly contribute to artist/music popularity figures (e.g. comments about an artist's DUI charge).

Each annotator is an "analysis component" that processes one comment at a time to find the entity of interest independently. However, the output of each of the three annotators is made available to the other annotators to allow observations to be made incrementally. Such composibility helps deal with short comments or those containing spam and non-spam content in the same "sentence". Annotation results are then aggregated over time periods to characterize the volume of positive, negative and spam comments. Additionally, counts of tracks and album mentions on an artist's page are also tallied.

All of these annotators are driven off of basic entity spotting. We look to simple arbitrary window-based entity spotting techniques backed by domain dictionaries which have been used in the past with fairly reasonable success (40), (98). While publicly available artist and track dictionaries provide the necessary dictionary support, the possible variations of the entity (misspelling, nick names, abbreviations, etc.) occurring in this often teen-authored corpus approaches the infinite.

Considering other techniques, there is good work on using natural language (NL) parses to spot nouns (for example) and/or a statistical strength to indicate an entity's importance in the corpus (41). Unfortunately, the "broken English" and possible variations of entities in this corpus make simple NLP problematic.

As a result, we have gone with a hybrid of these two methods: a dictionary and window-based spotter complemented with a part-of-speech tag analysis and the corpus-wide distribution strength of an entity. The natural language parsing of sentences is obtained using the Stanford NL Parser(62) and the distribution strength of an entity in the corpus is found based on an implementation of the Bayardo pruning method(95).

To evaluate our annotators, we processed a corpus of 600,000 comments gathered over a period of 26 weeks. All precision and recall figures presented in this section are calculated over a random sample of 300 comments from 9 artists (the restricted set due to the need to hand tag the entire test corpus for recall numbers). Tunable cut-off thresholds for annotators were determined based on experiment.

5.1.8.1 Music related / Artist-Track Annotator

The goal of this annotator is to spot artist and track mentions in a comment. Empirical evaluation suggests that the number of occurrences of comments on an artist's page that mention some other artist or tracks of other artists is insignificant (and thus ignored at this point). This annotator is backed by an artist's tracks and albums list from MusicBrainz and a short dictionary of music related words like tour, concert, album, etc.

Using our named entity spotter described in Section 3.4.3, we spot and disambiguate by first annotating comments using the naive spotter, followed by the NLP analytics using all the features we experimented with (see Table 3.9).

Table 5.2 shows the results of the annotator and when excluding the NL parse technique, i.e. only using the naive spotter. We contend that a combination of NLP and statistical techniques

yields good results in such casual broken English corpora.

Annotator Type	Precision	Recall
Artist	1.0	0.86
Track	0.67	1.0
Artist excluding NLP component	1.0	0.64

Table 5.2: Artist-Track Annotator

Analysis of results indicated two main reasons for lowered precision of the track annotator. First, false positives such as one word track names such as 'Smile','Dare' etc. were used in free-speech in combination with poorly structured sentences. Secondly, common heuristics like capitalized first letter or tagged as a noun/noun phrase often failed due to misspellings and non-standard writing conventions.

We observed that the recall suffers due to arbitrary variations of names (e.g. 'Rihanna' is sometimes referred to in the corpus as 'Riri'), odd sentence constructions and incomplete artist dictionaries (often missing names of members of a band).

5.1.8.2 Sentiment Annotator

Another metadatum used for gauging popularity is the sentiment associated with a comment post. In a classic setting with a paragraph or multiple paragraphs of text, the task would typically be to extract the sentiment and the polarity as directed toward an artist or their work mentioned in the comment. Given the nature of our data, where a comment is one or two sentences long and there is typically only one artist and/or their work mentioned, we make a simplifying assumption that the spotted sentiment expressions are always associated with the spotted entities in the comment. We do not explicitly verify if there is an attachment or syntactic dependency. Empirical evaluation

also suggests that for sources such as MySpace where each artist has a page, few other artists are mentioned on pages other than their own. If no entity was spotted, for example in comments such as "I loved seeing you yesterday", the sentiment is assumed to attach to the artist whose page the comment was found on.

Our approach for quantifying crowd preferences is related to work in opinion mining (OM) from public boards such as blogs, reviews and forums (Pang and Lee (89)). Our mining of sentiments about artists or their music differs from past work in OM because of the nature of our corpus and our goal of popularity ranking. Specifically, we limit OM to coarse assignments of positive and negative comments on an artist's page. In this respect, the goal of our work is similar to (Hatzivassiloglou and McKeown (48), Turney and Littman (108) Esuli and Sebastiani (33) and Kamps *et al.* (60)).

One of the unique challenges that we faced, compared to previous efforts in this area, was the varied number of ways that users, typically in the teen demographic, tend to express sentiment. Slang sentiment expressions such as "wicked" to mean "good" or "tight" to mean "awesome" are commonplace.

Slang expressions of opinions are harder to detect because their usage has changed over time – for example, the usage of the word 'sick' has changed from bearing a negative to a positive connotation.. While we borrow from past work in using linguistic cues that identify tokens of words that might indicate sentiment expressions (traditional or slang) and Turney's (Turney and Littman (108)) work in identifying polarities, we also rely on an external domain resource to assist in this process given the informal nature of user-generated content. Our system first mines a dictionary of traditional and slang sentiments from UrbanDictionary.com (UD) and uses this to

assist in the identification of sentiment expressions and their polarities.

Building a Sentiment Dictionary

Our sentiment dictionary maps frequently used sentiment expressions to their orientations, i.e., positive or negative. The dictionary is built off a popular slang dictionary, UrbanDictionary.com (UD), that provides a set of related tags and user-defined and voted definitions for a term. Since glossary definitions are not necessarily accurate and automating the process of reducing them to a single sentiment is problematic, our system uses the related tags. For example, the slang expression 'wicked' has the following tags associated with it - 'cool, awesome, sweet, sick, amazing, great' etc. It is worthy to note that related tags are only indicators of possible transliterations. The tag 'sick' for example, appears as a related tag of both words 'good' and 'bad'.

Our algorithm for mining a dictionary of sentiment expressions and their orientations taps into crowd agreements around the most current usage of slangs and proceeds as follows. Starting with a seed of five positive and five negative sentiment expressions (good, awesome, bad, terrible, etc.), UD is queried to obtain the top five related tags for each seed word. For each obtained related tag, we calculate its Semantic Orientation score with respect to the known positive and negatively oriented seed words. If the orientation of the related tag is toward the positive seed words, we pick the top positive seed word appearing in the list of words associated with the related tag as its transliteration. This process of obtaining new top five tags and determining their transliterations and orientations continues until no new tags are found.

We borrow Turney's work for calculating the Semantic Orientation scores of words (Turney and Littman (108)), with one modification. Instead of using the entire Web for co-occurrence statis-

tics, we limit our calculations to UrbanDictionary where slang usage is unbiased by co-occurrences outside the slang context.

Sentiment Annotator: Experiments and Findings

Gauging the sentiment associated with a user comment, i.e., identifying word tokens that might express a sentiment and finding its polarity proceeds as follows:

1. Perform a shallow NL parse of a sentence to identify adjectives or verbs that suggest the presence of a sentiment (48).

2. Look for the spotted sentiment in the mined dictionary. If the word is not found, then compute the word's possible transliteration using support from the corpus. To illustrate, transliterate the slang- sentiment "tight" to "awesome" because of the following co-occurrence strengths of "tight" with expressions in the mined dictionary sentiment words in the corpus: "tight-awesome" has a co-occurrence count of 456, "tight-sweet" 136, "tight-hot" 429, etc. Since "tight" co-occurs the most with "awesome", the polarity of the slang "tight" is recorded as positive. The suggested transliteration is picked only if statistically significant. The former method of transliteration via the mined dictionary associates a larger confidence with the spot compared to the latter that relies on weak corpus indicators of meaning.

3. Increase the confidence in the spotted sentiment if there is also an artist/music related entity spotted by the named entity spotter.

4. If the confidence is greater than a tunable threshold, record the sentiment and its polarity as an annotation value.

This annotator was evaluated over the 600,000 comments gathered over a period of 26 weeks. Precision and recall figures were calculated over a random sample of 300 comments for nine artists. The comments were hand labeled for the presence and orientation of sentiment expressions. Tunable cut-off thresholds for annotators were determined based on experiments. Table 5.3 shows the accuracy of the annotator and illustrates the importance of using transliterations in such corpora.

Annotator Type	Precision	Recall
Positive Sentiment	0.81	0.9
Negative Sentiment	0.5	1.0
PS excluding transliterations	0.84	0.67

Table 5.3: Transliteration accuracy impact

Results indicate that the syntax and semantics of sentiment expression in informal text is difficult to determine. Words that were incorrectly identified as sentiment bearing, because of incorrect parses due to sentence structure, resulted in inaccurate transliterations that in turn lowered precision (especially in the case of the Negative Sentiment annotator). We also experimented with the dependency relationships between entities and sentiments expressions but found them to be both expensive and minimally effective, most likely due to poor sentence constructions.

Low recall was consistently traced back to one of the following reasons: A failure to catch sentiment indicators in a sentence (inaccurate NL parses) or a word that was not included in our mined dictionary, either because it did not appear in UD or because it had insufficient support in the corpus for the transliteration (e.g., 'funnn'). Evaluating the annotator without the mined dictionary, i.e., only with the corpus support, significantly reduced the recall (owing to sparse co-occurrences in the corpus). However, it also slightly improved the precision, indicating the need for more selective transliterations. This speaks to our method for mining a dictionary of transliterations that does not take into account the context of the sentiment expression in the comment – an important

near-term investigation for this work. Empirical analysis also revealed that the use of negation words such as 'not', 'never' was rare in this data and therefore ignored at this point. However, as this may not be true for other data sources, this is an important area of future work.

5.1.8.3 Spam Annotator

Like many online data sets today, this corpus suffers from a fair amount of spam — off-topic comments that are often a kind of advertising. A preliminary analysis shows that for some artists more than half of the comment postings are spam (see Table 5.4). This level of noise could significantly impact the data analysis and ordering of artists if it is not accounted for.

Gorillaz	54%	Placebo	39%
Coldplay	42%	Amy Winehouse	38%
Lily Allen	40%	Lady Sovereign	37%
Keane	40%	Joss Stone	36%

Table 5.4: Percentage of total comments that are spam for several popular artists.

A particular type of spam that is irrelevant to our popularity gauging exercise are comments unrelated to the artist's work; such as those making references to an artist's personal life. Eliminating such comments and only counting those relevant to an artist or their work is an important goal in generating ranked artist lists. Since many of the features in such irrelevant content overlap with the ones of interest to us, classifying spam comments only using spam-phrases or features was not very effective. However, a manual examination of a random set of comments yielded useful cues to use in eliminating spam.

1. The majority of spam comments were related to the domain, had the same buzz words as many non-spam comments and were often less than 300 words long.

SPAM: 80% have 0 sentiments

CHECK US OUT!!! ADD US!!!
PLZ ADD ME!
IF YOU LIKE THESE GUYS ADD US!!!

NON-SPAM: 50% have *at least* 3 sentiments

Your music is really bangin!
You're a genius! Keep droppin bombs!
u doin it up 4 real. i really love the album. keep doin wat u do best. u r so bad!
hey just hittin you up showin love to one of chi-town's own. MADD LOVE.

Figure 5.3: Examples of sentiment in spam and non-spam comments.

2. Like any auto-generated content, there were several patterns in the corpus indicative of spam. Some examples include "Buy our cd", "Come check us out..", etc.

3. Comments often had spam and appreciative content in the same sentence which implied that a spam annotator would benefit from being aware of the previous annotation results.

4. Empirical observations also suggested that the presence of sentiment is pivotal in distinguishing spam content. Figure 5.3 illustrates the difference in distribution of sentiments in spam and non-spam content.

These observations also speak to the order in which our annotators are applied. We first perform named entity spotting, followed by the sentiment annotator and finally the spam filter.

Spotting Spam: Approach and Findings

Our approach to identifying spam (including those irrelevant to an artist's work) differs from past work, because the small size of individual comments and the use of slang posed new challenges. Typical content-based techniques work by testing content on patterns or regular expressions (75) that are indicative of spam, or by training Bayesian models over spam and non-spam content

(11). Recent investigations on removing spam in blogs use similar statistical techniques with good results (102). These techniques were largely ineffective on our corpus because comments are rather short (1 or 2 sentences), share similar buzz words with non-spam content, are poorly formed, and contain frequent variations of word/slang usage. Our approach of filtering spam is an aggregate function that uses a finite set of mined spam patterns from the corpus and other non-spam content such as artist names, and sentiments that are spotted in a comment.

Our spam annotator builds off a manually assembled seed of 45 phrases found in the corpus that are indicative of spam content. This seed was assembled by empirical analysis of frequent 4-grams in comments that are indicative of spam content. First, the algorithm spots these possible spam phrases and their variations in text using a window of words over the user comment and computing the string similarity between the spam phrase and the window of words in the user comment. These spots along with a set of rules over the results of the previous entity and sentiment annotators are used to decide if a comment is spam. Aa an example, if a spam phrase, artist/track name and a positive sentiment were spotted, the comment is probably not spam. By looking for previously spotted meaningful entities we ensure that we only discard spam comments that make no mention of an artist or their work.

Table 5.5 shows the accuracy of the spam and non-spam annotators for the hand-tagged comments. Our analysis indicates that lowered precision or recall in the spam annotator was a direct consequence of deficiencies in the preceding named entity and sentiment annotators. For example, in cases where the comment did not have a spam pattern from our manually assembled list, and the first annotator spotted incorrect tracks, the spam annotator interpreted the comment to be related to music and classified it as non-spam. Other cases included more clever promotional comments

that included the actual artist tracks, genuine sentiments and very limited spam content. (e.g., "like umbrella ull love this song…"). As is evident, the amount of information available in a comment (one or two sentences) in addition to poor grammar necessitates more sophisticated techniques for spam identification. This is an important focus of our ongoing research.

Annotator Type	Precision	Recall
Spam	0.76	0.8
Non-Spam	0.83	0.88

Table 5.5: Spam annotator performance

5.1.9 Generation of the Hypercube

We use a data hypercube (also known as an OLAP cube(26)) stored in a DB2 database to explore the relative importance of various dimensions to the popularity of musical topics. The dimensions of the cube are generated in two ways: from the structured data in the posting (e.g., age, gender, location of the user commenting, timestamp of the comment, artist), and from the measurements generated by the above annotator methods. This annotates each comment with a series of tags from unstructured and structured data. The resulting tuple is then placed into a star schema in which the primary measure is a relevance with regards to musical topics. This is equivalent of defining a function.

$$M : (Age, Gender, Location, Time, Artist, ...) \rightarrow M \tag{5.1}$$

In our case, we have stored the aggregation of occurrences of non-spam comment at the intersecting dimension values of the hypercube. Storing the data this way makes it easy to examine rankings over various time intervals, weight various dimensions differently, etc. Once (and if) a total ordering approach is fixed this intermediate data staging step might be eliminated.

5.1.9.1 Projecting to a list

Ultimately we are seeking to generate a "one dimensional" ordered list from the various contributing dimensions of the cube. In general we project to a one dimensional ranking which is then used to sort the artists, tracks, etc. We can aggregate and analyze the hypercube using a variety of multi-dimensional data operations on it to derive what are essentially custom popular lists for particular musical topics. In addition to the traditional billboard "Top Artist" lists, we can slice and project (marginalize dimensions) the cube for lists such as "What is hot in New York City for 19 year old males?" and "Who are the most popular artists from San Francisco?" They translate to following mathematical operations:

$$L_1(X) : \sum_{T,...} M(A = 19, G = M, L = \text{“}NewYorkCity\text{”}, T, X, ...) \tag{5.2}$$

$$L_2(X) : \sum_{T,A,G,...} M(A, G, L = \text{“}SanFrancisco\text{”}, X, ...) \tag{5.3}$$

where

X = Name of the artist

T = Timestamp

A = Age of the commenter

G = Gender

L = Location

Note that for the remainder of this paper we aggregate tracks and albums to artists as we wanted as many comments as possible for our experiments. Clearly track based projections are

equally possible and the rule for our ongoing work. This tremendous flexibility is one advantage of the cube approach.

5.1.10 Experiments - Testing and Validation

Is the social network you are looking at the right one for the questions you are asking? Is the data timely and relevant? Is there enough signal? All of these questions need to be asked about any data source, and even more so of social networks. It is critical when developing an SI application such as the SoundIndex to validate the results. We used a combination of point polling with groups in the target audience along with ongoing verification with subject matter experts to help identify problems with the system and raise confidence that the numbers being generated are credible.

To test the effectiveness of our popularity ranking system we conducted a series of experiments. We prepared a new top-N list of popular music to contrast with the most recent Billboard list. To validate the accuracy of our lists, we then conducted a study.

5.1.10.1 Generating our Top-N list

We started with the top-50 artists in Billboard's singles chart during the week of September 22nd through 28th, 2007. If an artist had multiple singles in the chart and appeared multiple times, we only kept the highest ranked single to ensure a unique list of artists. MySpace pages of the 45 unique artists were crawled, and all comments posted in the corresponding week were collected.

We loaded the comments into DB2 as described in Section 5.1.7. The crawled comments were passed through the three annotators to remove spam and identify sentiments. The tables

below show statistics on the crawling and annotation processes.

Number of unique artists	45
Total number of comments collected	50489
Total number of unique posters	33414

Table 5.6: Crawl Data

38%	of total comments were spam
61%	of total comments had positive sentiments
4%	of total comments had negative sentiments
35%	of total comments had no identifiable sentiments

Table 5.7: Annotation Statistics

As described in Section 5.1.9, the structured metadata (artist name, timestamp, etc.) and annotation results (spam/non-spam, sentiment, etc.) were loaded in the hypercube.

The data represented by each cell of the cube is the number of comments for a given artist. The dimensionality of the cube is dependent on what variables we are examining in our experiments. Timestamp, age and gender of the poster, geography, and other factors can all be dimensions in hypercube, in addition to the measures derived from the annotators (spam, non-spam, number of positive sentiments, etc.).

For the purposes of creating a top-N list, all dimensions except for artist name are collapsed. The cube is then sliced along the spam axis (to project only non-spam comments) and the comment counts are projected onto artist name axis. Since the percentage of negative comments was very small (4%), the top-N list was prepared by sorting artists on the number of non-spam comments they had received independent of the sentiment scoring.

In Table 5.9 we show the top 10 most popular Billboard artists and the list generated by our analysis of MySpace for the week of the survey.

System Details

All experiments were run on Xen virtual machines hosted on an IBM 3650 with quad-core processor running at 2.66GHz. The VMs run Redhat Enterprise Linux WS release 4 update 5. Each is allocated 1-2 GB of physical RAM. Data storage is done on a 4 drive SATA Raid5 array on which each VM has an image file served through QEMU-DM. Data Management was done via DB2 v9.1 EE for Linux.

5.1.10.2 The Word on the Street

Having fetched more that 50,000 comments, gone to great lengths to remove the spam and parse the informal English found within, tallied and scored and ultimately derived an alternative Top-N list for popular music, the obvious question raised is – does it work? Do people actually post online about music they prefer? Could a list generated from casual comments on a social networking site be a more accurate representation than that offered up by the record industry itself? Fully answering this question would (and will) require numerous studies beyond the scope of this paper, but we were able to perform a casual preference poll of 74 people in the target demographic.

At the conclusion of the data sampling week, we conducted a survey among students of an after-school program (Group 1), Carnegie Mellon (Group 2), and Wright State (Group 3). Of the three different groups, Group 1 comprised of respondents between ages 8 and 15; while Group 2 and 3 comprised primarily of college students in the 17-22 age group. Table 5.8 shows statistics pertaining to the three survey groups.

The survey was conducted as follows: the randomly chosen 74 respondents were asked to

Groups and Age Range	No. of male respondents	No. of female respondents
Group 1 (8-15)	8	9
Group 2 (17-22)	21	26
Group 3 (17-22)	7	3

Table 5.8: Survey Group Statistics

study the two lists shown in Table 5.9. One was generated by Billboard and the other through the crawl of MySpace. They were then asked the following question: *'Which list more accurately reflects the artists that were more popular last week'?* Their response along with their age, gender and the reason for preferring a list was recorded.

The sources used to prepare the lists were not shown to the respondents, so they would not be influenced by the popularity of MySpace or Billboard. In addition, we periodically switched the lists while conducting the study to avoid any bias based on which list was presented first.

Billboard.com	MySpace Analysis
Soulja Boy	T.I.
Kanye West	Soulja Boy
Timbaland	Fall Out Boy
Fergie	Rihanna
J. Holiday	Keyshia Cole
50 Cent	Avril Lavigne
Keyshia Cole	Timbaland
Nickelback	Pink
Pink	50 Cent
Colbie Caillat	Alicia Keys

Table 5.9: Billboard's Top Artists vs our generated list

5.1.11 Results

The raw results of our study immediately suggest the validity of our hypothesis, as can be seen in Table 5.10. The MySpace data generated list is prefered over 2 to 1 to the Billboard list by our 74 test subjects, and the preference is consistantly in favor of our list across all three survey groups.

	Group 1	Group 2	Group 3
MySpace-Generated List	15	30	6
Billboard List	2	17	4

Table 5.10: Experiment Results: number of people who prefered each list

A more fine grained statistical analysis of the data only improves upon the initial suggestion of the data. 68.9% of all subjects prefered our list. Calculating the standard error for these 74 responses, we have:

$$\frac{s}{\sqrt{n}} = \frac{1}{\sqrt{n}}\sqrt{\frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{x})} = 0.054$$

computed using $s = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{x})}$, the estimated standard deviation, and $\bar{x}$, the mean of the data points $\{x_1, x_2, x_i, ..x_n\}$.

This estimated standard deviation of the sample mean provides the result that 68.9$\pm$ 5.4% of subjects prefer our list to the Billboard list. Looking specifically at Group 1, the youngest survey group whose ages range from 8-15, we can see that our list is even more successful. Even with a smaller sample group (resulting in a higher standard error), 88.2$\pm$ 8.1% of subjects prefer our list to Billboard. This striking result shows a 6 to 1 preference for our list from younger listeners.

We can further calculate a confidence level for our data using the common statistical method, the t-distribution. This method is generally accepted to be usable for sample sizes of more than 30 observations without the need to establish that the data is normally distributed. (Cameron)

We employ the standard t-test to determine a critical value, denoted $T(\frac{\alpha}{2}, n-1)$ for n samples and a confidence interval of $k = 1 - 2\alpha$. The confidence interval for the confidence level k is than given by the critical value times the standard error:

$$T(\frac{\alpha}{2}, n-1)\frac{s}{\sqrt{n}}$$

Solving this for the confidence level which shows a preference of MySpace list versus the Billboard list (i.e., a more then 50% preference for the MySpace list) gives:

$$\bar{x} - T(\frac{\alpha}{2}, n-1)\frac{s}{\sqrt{n}} \geq 0.5$$

$$0.69 - T(\frac{\alpha}{2}, 73)0.055 \geq 0.5$$

which solves using t-test tables to $\alpha = 0.001$.

This gives a 99.9% confidence interval that a randomly polled group of similar individuals will show an overall preference for the MySpace data generated list over the Billboard list. Thus, we can say with a high degree of confidence that on-line sources are a *better* indicator than the traditional record charts for people in our sample group.

We can also tentatively conclude that our list is preferred equally by men and women. Groups 2 and 3 had equal preference for the myspace list (approximately 64% and 60%, respectively), however group 2 was mostly female (55%), whereas group 3 was mostly male (70%).

Another interesting observation: after concluding the survey, we asked some of the subjects which they thought was the most popular set (as opposed to the one they preferred). That is,

the correlation between perceived popularity and preference. 83% of subjects believed that their prefered list was also the most popular list, similarly distributed across those who preferred the MySpace generated list and those that preferred Billboard.

We conclude that new opportunities for self expression on the web provide a *more* accurate place to gather data on what people are really interested in than traditional methods. The even stronger results from the younger audience suggests that this trend is, if anything, accelerating.

5.1.12 Lessons Learned, Broader Impact

Deploying this system brought to light many areas which warranted further research and development. They include the gathering of data from low reliability sources and the need for enhanced natural language processing capabilities for application to the Informal English often found in these social networks.

Online communities are a virtual gold mine of GenX/iGen music opinions. Regardless of a musician's genre, label, or age, one is hard pressed to find a band without a MySpace profile, and most popular bands have a fairly active fan community. Even more traditional bands such as the Beatles and the Rolling Stones have active presences. Providing fans with the ability to deliver personal messages and feel as though they have spoken directly to the band has proved to be very appealing, and, as we have shown, very valuable for us to gauge popularity and buzz within these communities.

Conventional wisdom around market intelligence suggests statistical surveys of sample populations as the preferred method to determine prevailing opinions. While statistically valid, these

surveys require active participation and hopes that the sample population will not bias the survey. We believe that a new methodology (like ours) for market intelligence that gathers opinions from a large population (ideally, an entire population) would more accurately determine prevailing opinions. Previously, this was considered infeasible due to the difficulty of reaching 100% of the population. With the penetration of online social networking sites (e.g., MySpace, Orkut) and acceptance of blogging by GenX/iGen populations, including topic-specific blogs such as Slashdot and Blogger, data mining the online opinions of large portions of this population can be quickly implemented.

Topical Popularity

There are many other topics where we could employ our methodology to gauge popularity and sentiment. Sports teams, movies, and video games are just a few – but in order to accurately assess popularity and sentiment, an active corpus with many user-generated comments must be available. Online forums are a starting point, but they are dependent on the online "footprint" that these topics have in the online forums, blogs, and larger social networking sites. For example, trying to track popular topics for the San Francisco Symphony would mean we would have to crawl many smaller data sources where the postings may contain many topics unrelated directly to the symphony itself. Using message boards with lesser information about participants (such as Usenet) would not give us the ability to easily determine age, gender and geographic preference correlations.

Broken English

Broken English is not limited to social networking sites. Fragmented grammar appears in call center transcripts, chat logs from instant messaging clients, email messages, text messages, voice-to-text transcripts with poor precision, etc. Our analysis framework can be used to create metadata, rewrite acronyms, etc. in all of these domains. As electronic communication methods becomes less formal, our annotators become increasingly valuable. We plan to continue research with other media such as these to extend our work into new domains.

Enterprise Applications

We chose the music domain and built content annotators to create our own Top-N list due to the impact that music has on popular culture. This approach is by no way limited to teenage opinion surveys as the same text mining techniques can also be applied to call center transcripts, instant messaging chat logs, and emails.

One commercial example of how our work can be applied to an enterprise is by performing similar analysis on electronic communication (emails and corporate IMs). Many applications exist that monitor email and instant messaging behavior between employee accounts, but these tend to monitor message flows and not the actual content. By adding in the content analysis, we can use similar annotators to monitor employee sentiment, behavior trends, topics of interest, and compliance with legislative regulations.

5.2 Social Signals from Experiential Data on Twitter

The second Web application that we present in this chapter uses spatial, temporal and thematic contexts associated with user-generated content on Twitter to extract summaries of social perceptions behind real-time events.

The emergence of mirco-blogging platforms like Twitter, Friendfeed etc. have revolutionized how unfiltered, real-time information is disseminated and consumed by citizens. A side effect of this has been the rise of citizen journalism, where humans as sensors are "playing an active role in the process of collecting, reporting, analyzing and disseminating news and information"[6].

A significant portion of information generated and consumed by this interconnected network of participatory citizens is *experiential* in nature (58), i.e., contains first-hand observations, experiences, opinions made in the form of texts, images, audio or video about *real-world events*.

In the recent past, such experiential attributes of an event have proved valuable for crowd-sourced situational awareness applications. The text messages, pictures and videos that originated from Mumbai during the 2008 terrorist attacks, or during the civil unrest after the 2009 Iranian elections are examples of experiential data surrounding an event that formed a rich backdrop against traditional reports from the news media.

Perhaps, the most interesting phenomenon about such citizen generated data is that it acts as a lens into the social perception of an event in any region, at any point in time. Citizen observations about the same event relayed from the same or different location offer multiple, and often complementary viewpoints or story lines about an event. What is more, these viewpoints evolve over

[6]http://en.wikipedia.org/wiki/Citizen_journalism

time and with the occurrence of other events, with some perceptions gaining momentum in certain regions after being popular in some others.

Consequently, in addition to what is being said about an event (theme), where (spatial) and when (temporal) it is being said are integral components to the analysis of such data. The central thesis behind this work is that citizen sensor observations are inherently multi-dimensional in nature and taking these dimensions into account while processing, aggregating, connecting and visualizing data will provide useful organization and consumption principles.

5.2.1 Thesis Contributions - Twitris

The goal of the social intelligence application Twitris (109) (a portmanteau of Twitter and Tetris, for arranging activity in space, time and theme) is to analyze citizen generated data to extract insightful summaries of social perceptions behind events.

This work is motivated by the need to easily assess local and global social perceptions or signals that underlie events that evolve with time. Data pertaining to real-world events have unique characteristics because of the event they represent. Certain real-world events naturally have a spatial and temporal bias while some others do not. For example, when observing what India is saying about the Mumbai attack, one might wish to not be biased by global and possibly contrasting perceptions from Pakistan.

In this chapter, we present our experiences and challenges in building Twitris when using Twitter as the medium for obtaining citizen observations.

The Social Media / Twitter Phenomenon:

Twitter has nearly six million members worldwide who gather, consume, produce and share multi-modal information surrounding topics of interest and participate in dialogues that enable large-scale information diffusion. Consider the following post or tweet made on Twitter.com to understand the anatomy of a tweet.

> `RT @liberalmom Another holier-than-thou fundie maroon...is exposed as a fraud. http://bit.ly/1XHbX /but hey, she's not a commie! @accountant`
>
> Posted at 3:36 PM Sep 1st from Twitter.com

Tweets are restricted by the medium to no more than 140 bytes. The above tweet sent by a professor to a user 'liberalmom', includes his comment, a pointer to an unflattering article concerning an appointee in the South Carolina government, and a reference to his ongoing exchange with the accountant, concerning the political appointee Van Jones. The professor also copies another user 'accountant' on the exchange.

Not included in this tweet, but common practice nevertheless, is the inclusion of hash-tags, e.g., #democrats. Hash-tags indicate explicit topic categorization on the part of users and also serves to route the message to groups of participants with a common interest or cause.

Twitter participants may also choose to follow tweets that someone posts. A user 'follows' another in order to receive their tweets. Each Twitter participant has a Twitter page, often including a wealth of demographic variables and quantitative measures of influence including the number of followers and tweets issued. Given the follower connections on Twitter, every posted message has

a hidden social graph attached to it - as does a 'retweet' (echo or forwarding of a tweet) and a reply to a tweet. This connectivity of the follower graph largely dictates how a message travels the twitter sphere.

In addition to the textual mode of expression, tweets may also contain hyperlinks that point to other media content. Time metadata is trivially attached to a tweet when it is posted. Location of a post can be obtained directly from the metadata if the tweet is made from a GPS enabled device. In all other cases, the location information from the poster's profile page is an approximation for where the tweet came from.

The fundamental question we ask of the Twitter data surrounding any topic or event for extracting social perceptions is the following: *What is a region paying attention to today?* Our goal is to extract meaningful descriptors or entities, i.e. *key words and phrases* that summarize hundreds and thousands of citizen observations pertaining to an event for any spatial and temporal setting.

Selecting discriminatory keywords has been a problem of historical importance with probability distribution methods like TFIDF being the most popular (96). In our work, cues for a descriptor's importance are not only found in a corpus of tweets (theme), but also in the space and time co-ordinates associated with it.

Consider this scenario where two descriptors 'mumbai attacks' and 'hawala funding' pertaining to the Mumbai Terror Attack occurred in the tweets originating from the US on the same day. The phrase 'mumbai attacks' occurred every day the last week while 'hawala funding' is a new descriptor for today. Users are more likely to be interested in novel perspectives and experiences. Looking at spatial contexts, we also find that 'hawala funding' did not appear in any other country on the same day, while 'mumbai attacks' occurred in almost all countries that day. This suggests

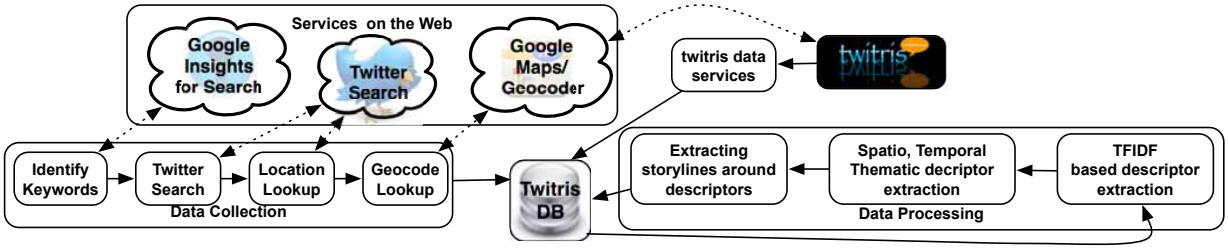


Figure 5.4: Twitris System Architecture – Data Collection, Analysis and Visualization Components

that the discussion around 'hawala funding' is a perspective shared by citizens local to this spatial setting while 'mumbai attacks' is a weaker descriptor in terms of uniqueness to the local region.

We exploit this three-dimensional interplay between the *space, theme and time* attributes of user-generated content in order to cull out words and phrases that best summarize the observations while also preserving the social perceptions underlying the data.

5.2.2 Twitris System Overview

We present our approach for gathering relevant social signals (tweets), and extracting and visualizing event descriptors as an implemented system. In its current version, Twitris contains more than two million processed tweets surrounding world events in 2009 (see Table 4.4). Twitris is designed to:

- **Collect user posted tweets from Twitter:** Given an event or topic of interest, as indicated by one or more keywords and hashtags, the system employs rules to collect highly relevant observations from Twitter. Additional constraints for the crawl could include the location and time period during which the posts are made.

- **Process obtained tweets to extract key descriptors:** Re-using our algorithm for key phrase extraction as described in Section 4.1, we extract n-gram phrases that summarize Twitter activity around the event.

- **Present extracted summaries to users:** Our approach to presenting extracted descriptors adopts the interface design paradigm of *experience design*[7]. One of the goals of experience design is to consider the multiple contexts surrounding the use of an application and create unified user interaction models across all contexts. Our goal is to is to create a visualization model that allows users to browse thematic descriptors of events in their spatio-temporal contexts.

Figure 5.4 illustrates the various steps and services involved in the data collection, analysis and visualization process. Here we describe the three main components in more detail.

5.2.2.1 Gathering Topically Relevant Data

The process of obtaining citizen observations from Twitter deserves some explanation since Twitter does not explicitly categorize user messages into topics. However, there is a search API[8] to extract tweets. A popular trend in Twitter has been the community-driven convention of adding additional context and metadata to tweets via *hashtags*, that can also be used to retrieve relevant tweets. Hashtags are similar to tags or index terms added to a document, except they are added inline to a tweet. They are created simply by prefixing a word with a hash symbol, for example, users would tag a tweet about Madonna using the hashtag #madonna.

[7] http://en.wikipedia.org/wiki/Experience_design
[8] http://search.twitter.com/search.json

Our strategy for obtaining posts relevant to an event uses a set of seed keywords, their corresponding hashtags and the Twitter search API. Seed keywords are obtained via a semi-automatic process using Google Insights for Search[9], a free service from Google that provides top searched and trending keywords across specific regions, categories, time frames and properties. The intuition is that keywords with high search volumes indicate a greater level of social interest and therefore more likely to be used by posters on Twitter.

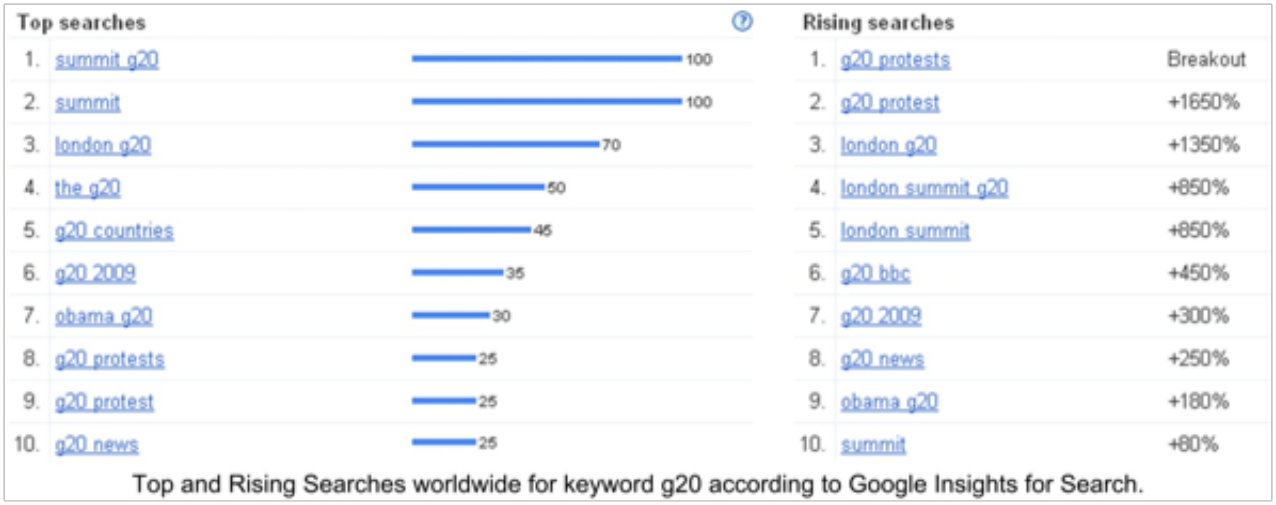
We start with a search term that is highly pertinent to an event and get top X keywords during a time period from Google Insights. For the g20 summit event for example, one could use the keyword g20 to obtain seed keywords. These keywords are manually verified for sufficient coverage for posts using the Twitter Search API, placed in set $\hat{\mathrm{K}}$, and used to kick-start the data collection process. Past this step, the system can be configured to automatically collect data. The list of keywords $\hat{\mathrm{K}}$ is also continually updated using two heuristics:

1. The first uses Google Insights to periodically obtain new keywords using those in $\hat{\mathrm{K}}$ as the starting query. Figure 5.5 shows popular search terms on Google that are related to the keyword 'g20'.

2. The second uses the corpus of tweets collected so far to detect popular keywords that were not previously used for crawling. A keyword is considered to be a good data extractor if it has a high TFIDF score (96) and high collocation scores with the keywords in $\hat{\mathrm{K}}$. The keyword with the highest score is periodically added to the set $\hat{\mathrm{K}}$.

The nature of an event determines the strategy for data collection. For long-running events, data is collected on a regular basis but in longer intervals. Shorter events demand more frequent

[9] http://www.google.com/insights/search/



Top searches			Rising searches	
1.	summit g20	100	1. g20 protests	Breakout
2.	summit	100	2. g20 protest	+1650%
3.	london g20	70	3. london g20	+1350%
4.	the g20	50	4. london summit g20	+850%
5.	g20 countries	45	5. london summit	+850%
6.	g20 2009	35	6. g20 bbc	+450%
7.	obama g20	30	7. g20 2009	+300%
8.	g20 protests	25	8. g20 news	+250%
9.	g20 protest	25	9. obama g20	+180%
10.	g20 news	25	10. summit	+80%

Top and Rising Searches worldwide for keyword g20 according to Google Insights for Search.

Figure 5.5: Showing an example of using Google Insights to obtain new keywords relevant to a seed keyword 'g20'.

data collection and continuous update of keywords.

Spatial, Temporal and Thematic Attributes of Twitter Posts:

The content of a Twitter post is the thematic component of a citizen observation. In this work, we ignore urls and links posted by users in a tweet and only use the textual component.

Spatial attributes for Twitter data can be of two types - location where the data originated from, and the location mentioned *in* the content. We do not concern ourselves with the latter since our goal is to study the social signals *originating* from a location in response to an event. There are two ways to obtain the spatial information associated with a tweet. The first method is to provide a location as a parameter to the search API. The other is to use the poster's location as an approximation for the origination of the tweet. We adopt the second alternative, as our crawl needs to be location independent.

The location information for an author either has geocoordinates (in cases where GPS enabled

devices were used in accessing Twitter) or has a location descriptor (city, state or country) free-text information provided by posters in their profiles. In case of the former, we use the coordinate information as is, while in the latter, we make use of Google GeoCoding API [10] to identify the coordinates. We realize the limitations of this approach (for example, an author might have posted a tweet from Boston, but updated his location later), but given the lack of geocoding information in the tweets, we consider this approach as a sufficient approximation.

The temporal information for each tweet is obtained from the time the tweet was posted (available via the API). Since we are interested in social signals over time, we do not concern ourselves with identifying temporal information that might be available *in* the content of a tweet.

We model a tweet t as a 4-tuple; $t = \{t_{id}, t_c, t_t, t_g\}$ where t_{id} is a unique alpha-numeric identifier, t_c is the textual content, t_t and t_g are the time and geographical coordinates obtained for the tweet. $t_g = \{lat, lng\}$ where lat is the latitude and lng is the longitude of the geographical coordinates of t_g.

5.2.2.2 Processing Citizen Observations

Fundamental to the processing of citizen observations is a simple intuition - "depending on what the event is, social perceptions and experiences reported by citizen sensors might not be the same across spatial and temporal boundaries". One of the goals in the formulation of our algorithm was to preserve these different story-lines that naturally occur in data. The questions we wish to answer via this application is – for any given spatial location and temporal condition, can we get an idea of what entities or event descriptors are dominating the discussion in citizen observations?

[10] http://code.google.com/apis/maps/documentation/geocoding/index.html

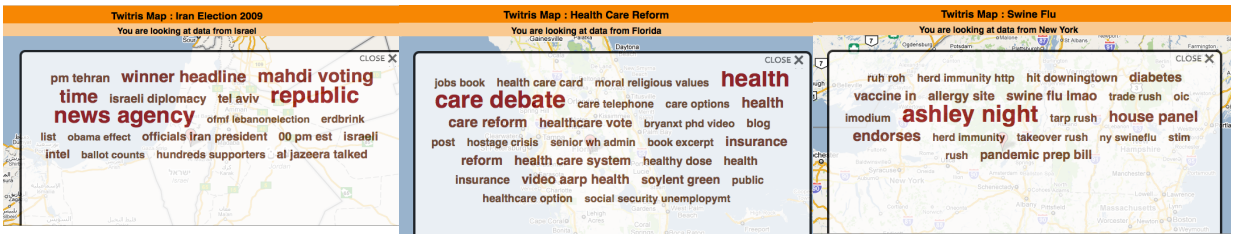


Figure 5.6: Key Phrases extracted for three different events from different locations and time periods – the 2009 Iran Election, the Health Care Reform debate and the 2009 Swine Flu epidemic

We use our approach described in Section 4.1 that uses contexts from the textual components of a user post and the spatial and temporal attributes of the posts for extracting key phrases that summarize multiple observations around an event.

To reiterate the key phrase extraction process, the user observations are first separated into spatio-temporal clusters (for example, all tweets from country X on day Y are grouped in one cluster) and key phrases are extracted from these clusters to reflect social perceptions from a location and time period.

At the end of the key phrase extraction, top X phrases that are novel and descriptive of the event in a spatio-temporal-thematic slice are used for display. Figure 5.6 shows three different events for which top X key phrases were extracted from specific locations and time periods.

5.2.2.3 User Interface and Visualization

The primary objective of the Twitris user interface is to integrate the results of the data analysis (extracted key descriptors) with emerging visualization paradigms to facilitate *sensemaking*. Sensemaking, defined in (64), is the understanding of connections between people, places and events. Awareness of *who, what, when and where* is a critical component in sensemaking.

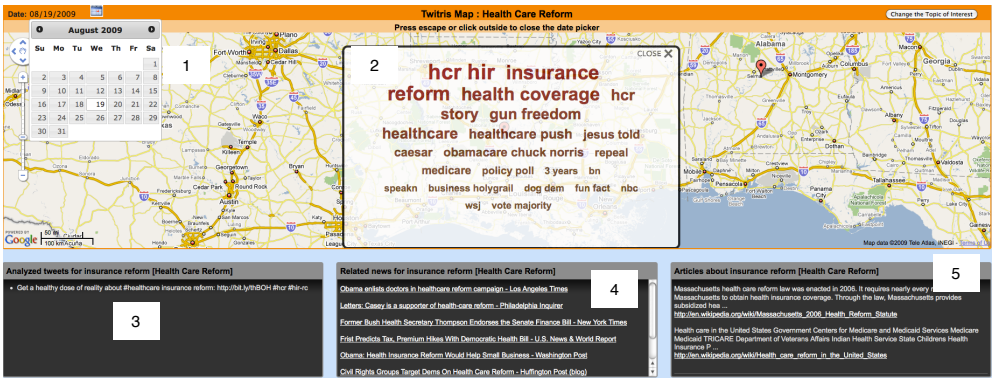


Figure 5.7: Showing parts of the Twitris interface for browsing real-time data. Pane 1 shows the temporal navigator for the event under focus, Pane 2 shows the thematic slice or the extracted key phrases, Panes 3, 4 and 5 assist browsing key phrases in context by showing tweets and news, Wikipedia articles related to a key phrase.

The Twitris user interface facilitates effective browsing of the *when*, *where*, and *what* slices of social perceptions behind an event. Figure 5.7 illustrates the theme, time and space components of the interface. To start browsing, users first select an event to restrict the theme under focus. Once a theme is chosen, the date is set to the earliest date of recorded observations for an event and the map is overlaid with markers indicating the spatial locations from where observations were made on that date. Users can further explore thematic activity or extracted key phrases in a particular space by clicking on the overlay marker.

Top X event descriptors extracted from observations in this spatio-temporal setting are displayed as a tag cloud. The spatio-temporal-thematic scores (calculated as shown in Section 4.1) determine the size of the descriptor in the tag cloud (see Pane 2 in Figure 5.7).

Browsing Social Data in Context:

When abstracting multiple user observations to a set of key phrases, an important requirement is being able to still interpret the data in the right context. Social media content that is already relatively poor in context can be rather ambiguous if not presented in sufficient context.

Twitris facilitates the process of consuming social data in context by supplementing extracted key phrases with contextual information from the Web.

Consider the key phrase 'Soylent Green' that appeared in several discussions around the 2009 Health Care Reform debate. Only in context of the news articles, tweets and Wikipedia articles it becomes clear that users are making a connection between the lifestyle mandated in the 1973 movie and policies suggested by the 2009 Health Care Reform.

On Twitris, clicking on an extracted key phrase, displays all tweets that mention this phrase in addition to news and Wikipedia articles that describe the key phrase in context of the event in focus (see panes 3,4 and 5 in Figure 5.7).

5.2.3 Broader Impact

Social media has played a key role in attracting real-time traffic, enabling large-scale information diffusion and creating tangible effects on participating economies and societies. Social media's influence was powerful following the terrorist attacks in Mumbai and in the civil reaction to the Iranian elections. Online digital footprints that users leave behind provide an excellent opportunity to observe social phenomena from the microscopic to the macroscopic level, in a range of scales from the local to the global, over the last few days to months and years, all of which could previously only be imagined.

While a significant challenge in making sense of social content (e.g., Twitter posts, blogs, votes on articles) is the deluge of information and the informal nature of the communication medium, there are equally important challenges in interpreting the results of computational anal-

ysis of this data within their cultural contexts. The harder questions often are, what insights do the behavioral data points give us about the cultural logic or perceptions that generated this data. What can learn from the activity of people on social media about the people and society at large? Can we truly get to the heart of what a society thinks by studying what they do (behavioral data) online?

Understanding the cultural perceptions behind social data also serves as a feedback loop in building systems that promote certain social dynamics. For example, extracting and presenting diverse opinions allows tools to showcase an information landscape where all popular and less-popular perspectives get a fair representation. Distilling such cultural viewpoints from volumes of data is at the heart of understanding 'why' a society does 'what' it does.

The core of the analysis infrastructure behind Twitris is aimed at facilitating such studies – in coding, analyzing and interpreting the cultural perceptions and opinions of a society by looking at the behavioral data it generates.

6. Conclusions and Future Directions

This thesis was motivated by the need to understand the characteristics of user-generated textual content on social media. We have shown how text on social media is qualitatively different, less formal and lacking in context compared to news, Wikipedia or scientific articles that have been the predominant focus of recent text mining applications. We have also suggested that as a consequence, text on social media platforms require new approaches to information analysis.

As a first step, we showed that the variability and lack of sufficient context in user-generated text affects the performance and reliability of state-of-the-art algorithms for 'aboutness' understanding tasks. Using multiple sources of contextual information from sources external to a document corpus, we showed that it is possible to improve statistical NLP and machine learning algorithms for two subgoals of named entity recognition and key phrase extraction.

Specifically, we made the following contributions in this thesis:

- We described frameworks and implementations that showed how contextual knowledge from three rich sources – the document corpus (both labeled and unlabeled documents), metadata from the social medium (time, location, poster demographics) and rich models of facts about a domain (entities and how they are related) can be integrated with features used by existing

natural language processing and machine learning algorithms.

- We demonstrated the usefulness of these frameworks in combining different types of contextual cues for two end tasks of named entity recognition and key phrase extraction. The results over the two tasks support the thesis conjecture that using multiple contextual cues together lends to reliable inter-dependent decisions, better than using the cues in isolation. The results are also promising in terms of the frameworks' applicability in combining contextual evidences for other language understanding tasks.

- The thesis also presented two Social Intelligence applications that were built over the results of the two tasks of named entity recognition and key phrase extraction. We showed that an in-depth analysis of social textual content can offer various insights into the popularity preferences and cultural perceptions of users behind topics of interest.

6.1 Future Directions

There are several directions that this thesis should be expanded along. First, we would like to validate the techniques presented in this thesis on other types of content. A straight-forward extension of our work is identifying and disambiguating named entities in the bio-medical domain using contexts in scientific corpora and from domain models such as the Unified Medical Language System (UMLS) [1].

Secondly, to associate greater confidence with results over crowd-sourced data, we would like to validate what we find from a medium by contrasting it with effects observed in other proxies

[1] See `www.nlm.nih.gov/research/umls/`

(e.g., contrasting what we find on Twitter with the news media or traditional polling methods). This will not only shed light on whether the medium used was a sufficient proxy but will also tell us if we are measuring the right signals in the data.

We believe that such studies are essential for interpreting what we find in almost natural settings of a social media platform with what we find in more controlled, structured settings such as a publisher oriented news media.

6.1.1 Computational Social Science

The volume and variety of user-generated content and the user participation network behind it are creating new opportunities for understanding web-based practices and contrasting it with what we find in offline user interactions. Parallel developments in processing large amounts of data, and the interest from multiple disciplines to create and provide models of a domain [2] are also facilitating new models for computational linguistic analysis.

A computational analysis of user behavioral data is however only a first step in understanding the dynamics of online user interactions. The harder and more important challenges are in the interpretation of results in their cultural contexts. For example, while our analysis may indicate that tweets from Florida around the healthcare debate on a particular day focus on topics X, Y and Z while those from Washington focus on A, B; the important questions to ask are 'why' a society is sharing what it is sharing.

My interest is in building tools that expose social data and results of analyses and provide methods of interaction in a fashion that will allow scientists to slice and dice the data to observe

[2] See `http://semanticweb.org/wiki/Ontology`

interactions across different scales – local to global, dated to most recent, posted by a certain population segment, involving static friend networks to dynamic goal-driven communities etc. Such systems complemented with user studies will 'close the loop' so to speak, for social/information scientists in terms of identifying patterns and outliers that signify cultural importance.

6.1.2 Poster, Content and Network Interactions and a Social System

In understanding user-generated content, this thesis focused only on the text and associated contextual information. There is however additional valuable information in the network and poster components of a social system that can also enrich the analysis of data and user behavior.

Our near-term research interests are in understanding the interaction effects of micro-level variables of user-generated content, the people and network connections on web-based practices. We see the work presented in this dissertation as a building block toward studying this synergy.

While we understand dynamics of large networks very well today, not a lot is known about how the semantics or style of content fits into the observations made about the network. The same can be said of our understanding of a 'corpus' of textual content, which is far sophisticated compared to our thesis of the network effects on conversations. These dynamics are also not devoid of the participant component – are the observed changes in a network shaped by the attributes of individuals (passionate advocate or an objective observer)? Often times, this three-dimensional dynamic of people, content and link structure is what shapes the social dynamic. As an example, we would like to understand how the interplay of the topic of discussion, emotional charge of a conversation, the presence of an expert and connections between participants, together affect emerging social order in an online conversation or help explain information propagation in a social

network.

Our interest is in analyzing behavioral data in a social system in the context of Lasswell's maxim of Communication theory – *'who (people) said what (content) to whom (network structures) in what channel with what effect.'*

We would like to approach the problem of understanding a social process by studying the coupling of behavior at micro-scales of interaction at the content, network and people level, with qualitative changes at macro-levels (e.g., the formation of groups and networks, information diffusion) and the resulting consequences (e.g., political decisions or consumer behaviors).

At a theoretical level, we would like to study this synergy in a principled manner by building and observing models tailored toward these interactions. Such models would have clear benefits in explaining the combined effects and dominant factors among the three players (content, network and people interactions) on any social phenomena.

The long-term outlook of this dissertation is to research online social user interactions and build and design systems that help us understand and impact the way society produces, consumes and shares data. The near-term goals are naturally more modest, and aim at transformative and robust ways of coding, analyzing and interpreting user observations on social media.

These goals present several opportunities to leverage properties of online behavioral data, allow us to learn something new about the dynamics of social media and develop methods that support collaborative analysis. They capture our ideas for this evolving, multi-disciplinary field of computational social science that will allow us to bring lessons from multiple disciplines – computer science, computational linguistics, social sciences, AI, large-scale and distributed computing, HCI etc. to tackle challenges in a unique way.

Bibliography

[1] Adam, E., 1993. Fighter cockpits of the future. pages 318–323.

[2] Alba, A., V. Bhagwan, J. Grace, D. Gruhl, K. Haas, M. Nagarajan, J. Pieper, C. Robson, and N. Sahoo, 2008. Applications of voting theory to information mashups. In *ICSC*, pages 10–17. IEEE Computer Society.

[3] Alfonseca, E. and S. Manandhar, 2002. An unsupervised method for general named entity recognition and automated concept discovery. In *Poceedings of the First International Conference on General WordNet*, Mysore, India.

[4] Ananthanarayanan, R., V. Chenthamarakshan, P. M. Deshpande, and R. Krishnapuram, 2008. Rule based synonyms for entity extraction from noisy text. In *ACM Workshop on Analytics for noisy unstructured text data*, pages 31–38.

[5] Back, M. D., J. M. Stopfer, S. Vazire, S. Gaddis, S. C. Schmukle, B. Egloff, and S. D. Gosling, 2010. Facebook profiles reflect actual personality, not self-idealization. *Psychological Science*.

[6] Balinski, M. and R. Laraki, 2007. A theory of measuring, electing, and ranking. *PNAS*, 104(21):8720–8725.

[7] Barzilay, R. and M. Elhadad, 1997. Using lexical chains for text summarization. In *In Proceedings of the ACL Workshop on Intelligent Scalable Text Summarization*, pages 10–17.

[8] Berkhin, P., 2005. A survey on pagerank computing. *Internet Mathematics*, 2:73–120.

[9] Berry, M. W., 2003. *Survey of Text Mining*. Springer-Verlag New York, Inc., Secaucus, NJ, USA.

[10] Biemann, C., 2006. Chinese whispers: an efficient graph clustering algorithm and its application to natural language processing problems. In *TextGraphs '06: Proceedings of TextGraphs: the First Workshop on Graph Based Methods for Natural Language Processing on the First Workshop on Graph Based Methods for Natural Language Processing*, pages 73–80, Morristown, NJ, USA. Association for Computational Linguistics.

[11] Blosser, J. and D. Josephsen, 2004. Awarded best paper! - scalable centralized bayesian spam mitigation with bogofilter. In *LISA '04: Proceedings of the 18th USENIX conference on System administration*, pages 1–20, Berkeley, CA, USA. USENIX Association.

[12] Boguraev, B. and J. Pustejovsky, editors, 1996. *Corpus processing for lexical acquisition*. MIT Press, Cambridge, MA, USA.

[13] boyd, d., 2007. *Why Youth (Heart) Social Network Sites: The Role of Networked Publics in Teenage Social Life*, pages 119–142. MacArthur Foundation Series on Digital Learning. MIT Press, Cambridge, MA.

[14] Bozsak, E., M. Ehrig, S. Handschuh, S. H, A. Maedche, A. Hotho, E. Maedche, B. Motik, D. Oberle, C. Schmitz, N. Stojanovic, Rudi, S. Staab, L. Stojanovic, and V. Zacharias, 2002. Kaon - towards a large scale semantic web. In *Proc. of EC-Web 2002, LNCS*, pages 304–313. Springer.

[15] Breiman, L. and L. Breiman, 1996. Bagging predictors. In *Machine Learning*, pages 123–140.

[16] Brin, S., 1998. Extracting patterns and relations from the world wide web. In *WebDB Workshop at 6th International Conference on Extending Database Technology, EDBT'98*.

[17] Brown, P. F., P. V. deSouza, R. L. Mercer, V. J. D. Pietra, and J. C. Lai, 1992. Class-based n-gram models of natural language. *Comput. Linguist.*, 18(4):467–479.

[18] Budanitsky, A. and G. Hirst, 2006. Evaluating wordnet-based measures of lexical semantic relatedness. *Computational Linguistics*, 32(1):13–47.

[19] Bunescu, R. C. and M. Pasca, 2006. Using encyclopedic knowledge for named entity disambiguation. In *EACL*. The Association for Computer Linguistics.

[Cameron] Cameron, A. C. Statistical inference for univariate data.

[21] Chakrabarti, S., 2002. *Mining the Web: Discovering Knowledge from Hypertext Data*. Morgan Kaufmann, 1st edition.

[22] Chang, C.-C. and C.-J. Lin, 2001. *LIBSVM: a library for support vector machines*. Software available at `http://www.csie.ntu.edu.tw/~cjlin/libsvm`.

[23] Chieu, H. L. and H. T. Ng, 2002. Named entity recognition: A maximum entropy approach using global information. In *COLING*.

[24] Chinchor, N. A., 1998. Muc-7 named entity task definition (version 3.5). Technical report, MUC.

[25] Church, K. W. and P. Hanks, 1990. Word association norms, mutual information, and lexicography. *Comput. Linguist.*, 16(1):22–29.

[26] Codd, E., S. Codd, and C. Salley, 1993. *Providing OLAP (on-line Analytical Processing) to User-analysts: An IT Mandate*. Codd & Date, Inc.

[27] Collins, A. and M. Quillian, 1969. Retrieval time from semantic memory. *Journal of Verbal Learning and Verbal Behavior*, 8(2):240–247.

[28] Collins, A. M. and E. F. Loftus, 1975. A spreading-activation theory of semantic processing. *Psychological Review*, 82(6):407–428.

[29] Collins, M. and Y. Singer, 1999. Unsupervised models for named entity classification.

[30] Crestani, F., 1997. Application of spreading activation techniques in information retrieval. *Artif. Intell. Rev.*, 11(6):453–482.

[31] Cucchiarelli, A. and P. Velardi, 2001. Unsupervised named entity recognition using syntactic and semantic contextual evidence. *Computational Linguistics*, 27(1):123–131.

[32] Cunha, C., A. Bestavros, and M. Crovella, 1995. Characteristics of www client-based traces. Technical report, Boston University, Boston, MA, USA.

[33] Esuli, A. and F. Sebastiani, 2005. Determining the semantic orientation of terms through gloss classification. In *CIKM '05: Proceedings of the 14th ACM international conference on Information and knowledge management*, pages 617–624, New York, NY, USA. ACM Press.

[34] Etzioni, O., M. Cafarella, D. Downey, A.-M. Popescu, T. Shaked, S. Soderland, D. S. Weld, and A. Yates, 2005. Unsupervised named-entity extraction from the web: an experimental study. *Artif. Intell.*, 165(1):91–134.

[35] Etzioni, O., M. Cafarella, and et al., 2004. Web-scale information extraction in knowitall: (preliminary results). In *WWW '04*, pages 100–110. ACM.

[36] Fairthorne, R. A., 1969. Content analysis, specification and control. *Annual Review of Information Science and Technology*, 4:73–109.

[37] Fano, R. M. and D. Hawkins, 1961. Transmission of information: A statistical theory of communications. *American Journal of Physics*, 29(11):793–794.

[38] Ferrucci, D. and A. Lally, 2004. Uima: an architectural approach to unstructured information processing in the corporate research environment. *Nat. Lang. Eng.*, 10(3-4):327–348.

[39] Firth, J. R., 1957. A synopsis of linguistic theory, 1930-1955. *Studies in Linguistic Analysis*, pages 1–32.

[40] Freitag, D., 1998. Information extraction from html: application of a general machine learning approach. In *AAAI '98/IAAI '98: Proceedings of the fifteenth national/tenth conference on Artificial intelligence/Innovative applications of artificial intelligence*, pages 517–523, Menlo Park, CA, USA. American Association for Artificial Intelligence.

[41] Freitag, D. and N. Kushmerick, 2000. Boosted wrapper induction. In *Proceedings of the Seventeenth National Conference on Artificial Intelligence and Twelfth Conference on Innovative Applications of Artificial Intelligence*, pages 577–583. AAAI Press / The MIT Press.

[42] Freund, Y. and R. E. Schapire, 1995. A decision-theoretic generalization of on-line learning and an application to boosting. In *European Conference on Computational Learning Theory*, pages 23–37.

[43] Gene Ontology Consortium, 2010. The gene ontology in 2010: extensions and refinements. *Nucleic acids research*, 38(Database issue):D331–335.

[44] Gruhl, D., M. Nagarajan, J. Pieper, C. Robson, and A. Sheth, 2010. Multimodal social intelligence in a real-time dashboard system.

[45] Hall, E. T., 1977. *Beyond Culture*. Anchor.

[46] Halliday, M. A. and C. M. Matthiessen, 2004. *An Introduction to Functional Grammar*. Arnold Publishers.

[47] Halliday, M. A. K. and R. Hasan, 1976. *Cohesion in English (English Language)*. Longman Pub Group.

[48] Hatzivassiloglou, V. and K. R. McKeown, 1997. Predicting the semantic orientation of adjectives. In *Proceedings of the eighth conference on European chapter of the Association for Computational Linguistics*, pages 174–181, Morristown, NJ, USA. Association for Computational Linguistics.

[49] He, J., W. Weerkamp, M. Larson, and de M. Rijke, 2009. An effective coherence measure to determine topical consistency in user-generated content. *Int. J. Doc. Anal. Recognit.*, 12(3):185–203.

[50] Heafield, K., 2008. Word context entropy. Technical report, Google Inc.

[51] Hearst, M. A., 1992. Automatic acquisition of hyponyms from large text corpora. In *Proceedings of the 14th conference on Computational linguistics*, pages 539–545, Morristown, NJ, USA. Association for Computational Linguistics.

[52] Heylighen, F., 1999. Advantages and limitations of formal expression. *Foundations of Science*, 4:25–56.

[53] Heylighen, F. and J.-M. Dewaele, 1998. Variation in the contextuality of language: An empirical measure. In *Context in Context, Special issue of Foundations of Science*, pages 293–340.

[54] Hurst, M. and A. Maykov, 2009. Social streams blog crawler. In *ICDE '09: Proceedings of the 2009 IEEE International Conference on Data Engineering*, pages 1615–1618, Washington, DC, USA. IEEE Computer Society.

[55] Hutchins, W. J., 1997. The concept of "aboutness" in subject indexing. pages 93–97.

[56] Ide, N. and J. Véronis, 1998. Word sense disambiguation: The state of the art. *Computational Linguistics*, 24:1–40.

[57] J Hassell, B. A.-M. and I. Arpinar, 2006. Ontology-driven automatic entity disambiguation in unstructured text. In *Proceedings of the International Semantic Web Conference, ISWC*.

[58] Jain, R., 2003. Experiential computing. *Commun. ACM*, 46(7):48–55.

[59] Joachims, T., 1998. Text categorization with support vector machines: Learning with many relevant features. In *Lecture Notes in Computer Science: Machine Learning*, pages 137–142. Springer Verlag.

[60] Kamps, J., M. Marx, R. Mokken, and de M. Rijke, 2004. Using wordnet to measure semantic orientation of adjectives.

[61] Keller, F. and M. Lapata, 2003. Using the web to obtain frequencies for unseen bigrams. *Comput. Linguist.*, 29(3):459–484.

[62] Klein, D. and C. D. Manning, 2002. Fast exact inference with a factored model for natural language parsing. In *NIPS*, pages 3–10.

[63] Klein, D. and C. D. Manning, 2003. Fast exact inference with a factored model for natural language parsing. In *Advances in Neural Information Processing Systems*, volume 15. MIT Press.

[64] Klein, G., B. Moon, and R. Hoffman, 2006. Making sense of sensemaking 1: alternative perspectives. *IEEE Intelligent Systems*, 21(4):70–73.

[65] Kripke, S. A., 1980. *Naming and Necessity (Library of Philosophy & Logic)*. Blackwell Publishers.

[66] Krulwich, B., 1995. Learning document category descriptions through the extraction of semantically significant phrases. In *In Proceedings of the IJCAIÕ95 Workshop on Data Engineering for Inductive Learning*.

[67] Kumar, R., U. Mahadevan, and D. Sivakumar, 2004. A graph-theoretic approach to extract storylines from search results. In *KDD*, pages 216–225.

[68] Lee, B. T., J. Hendler, and O. Lassila, 2001. The semantic web. *Scientific American*.

[69] Leskovec, J., 2008. *Dynamics of large networks*. PhD thesis, Pittsburgh, PA, USA. Adviser-Faloutsos, Christos.

[70] Lin, D., 1998. Automatic retrieval and clustering of similar words.

[71] Lin, J., 2002. Divergence measures based on the shannon entropy. *Information Theory, IEEE Transactions on*, 37(1):145–151.

[72] Liu, Z., P. Li, Y. Zheng, and M. Sun, 2009. Clustering to find exemplar terms for keyphrase extraction. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing*, pages 257–266, Singapore. Association for Computational Linguistics.

[73] Locke, L. A., 2004. Super searches. *Time Magazine*. Oct. 31.

[74] Marneffe, M., B. Maccartney, and C. Manning, 2006. Generating typed dependency parses from phrase structure parses. In *Proceedings of LREC-06*, pages 449–454.

[75] Mason, J., 2002. Filtering spam with spamassassin. In *Proceedings of HEANet Annual Conference*.

[McIntyre] McIntyre, M. Hubbard hot-author status called illusion. `http://www.scientology-lies.com/press/san-diego-union/1990-04-15/hubbard-hot-author-status-illusion.html`.

[77] McPherson, M., L. S. Lovin, and J. M. Cook, 2001. Birds of a feather: Homophily in social networks. *Annual Review of Sociology*, 27(1):415–444.

[MediamarkResearch] MediamarkResearch. Teen market profile. `http://www.magazine.org/content/files/teenprofile04.pdf`.

[79] Mihalcea, R. and P. Tarau, 2004. Textrank: Bringing order into texts. In *Conference on Empirical Methods in Natural Language Processing*, Barcelona, Spain.

[80] Miller, G. A., R. Beckwith, C. Fellbaum, D. Gross, and K. Miller, 1990. Wordnet: An on-line lexical database. *International Journal of Lexicography*, 3:235–244.

[81] Minkov, E., R. C. Wang, and W. W. Cohen, 2005. Extracting personal names from email: Applying named entity recognition to informal text. In *HLT/EMNLP*. The Association for Computational Linguistics.

[82] Muller, C. and I. Gurevych, 2008. Using wikipedia and wiktionary in domain-specific information retrieval. In *Working Notes for the CLEF 2008 Workshop*, Aarhus, Denmark.

[83] Muñoz, A., 1997. Compound key word generation from document databases using a hierarchical clustering art model. *Intell. Data Anal.*, 1(1-4):25–48.

[84] Nadeau, D. and S. Sekine, 2007. *A survey of named entity recognition and classification*. Linguisticae Investigationes.

[85] Nakagawa, H., 1997. Extraction of index words from manuals. In *RIAO*, pages 598–615.

[86] Navigli, R., 2009. Word sense disambiguation: A survey. *ACM Comput. Surv.*, 41(2):1–69.

[87] Nowson, S., J. Oberlander, and A. J. Gill, 2005. Weblogs, genres and individual differences. In *Proceedings of the 27th Annual Conference of the Cognitive Science Society*, pages 1666–1671.

[88] Padó, S. and M. Lapata, 2007. Dependency-based construction of semantic space models. *Comput. Linguist.*, 33(2):161–199.

[89] Pang, B. and L. Lee, 2008. Opinion mining and sentiment analysis. *Found. Trends Inf. Retr.*, 2(1-2):1–135.

[90] Pantel, P. and D. Lin, 2002. Discovering word senses from text. In *KDD '02: Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 613–619, New York, NY, USA. ACM.

[91] Pennebaker, J. W. and M. E. Francis, 1999. *Linguistic Inquiry and Word Count*. Lawrence Erlbaum, 1 edition.

[92] Quinlan, J. R., 1993. *C4.5: programs for machine learning*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA.

[93] Ramage, D., A. N. Rafferty, and C. D. Manning, 2009. Random walks for text semantic similarity. In *TextGraphs-4: Proceedings of the 2009 Workshop on Graph-based Methods for Natural Language Processing*, pages 23–31, Morristown, NJ, USA. Association for Computational Linguistics.

[94] Riloff, E. and R. Jones, 1999. Learning dictionaries for information extraction by multi-level bootstrapping. In *In Proceedings of the Sixteenth National Conference on Artificial Intelligence*, pages 474–479.

[95] Roberto J. Bayardo, J., 1998. Efficiently mining long patterns from databases. In *SIGMOD '98*, pages 85–93, New York, NY, USA. ACM Press.

[96] Salton, G. and C. Buckley, 1988. Term-weighting approaches in automatic text retrieval. *Information Processing and Management*, 24(5):513–523.

[97] School, R. M., R. Mitkov, and W. W. Sb, 1999. Anaphora resolution: The state of the art. Technical report.

[98] Soderland, S., 1997. Learning to extract text-based information from the world wide web. *Proceedings of Third International Conference on Knowledge Discovery and Data Mining (KDD-97)*.

[99] Steier, A. M. and R. K. Belew, 1993. Exporting phrases: A statistical analysis of topical language. In *Second Symposium on Document Analysis and Information Retrieval*, pages 179–190.

[100] Swartz, A., 2002. Musicbrainz: a semantic web service. *Intelligent Systems, IEEE [see also IEEE Intelligent Systems and Their Applications]*, 17(1):76–77.

[101] Tatar, D., 2004. Word sense disambiguation by machine learning approach: A short survey. *Fundam. Inf.*, 64(1-4):433–442.

[102] Thomason, A., 2007. Blog spam: A review. In *Fourth Conference on Email and Anti-Spam CEAS 2007*.

[103] Tsatsaronis, G., M. Vazirgiannis, and I. Androutsopoulos, 2007. Word sense disambiguation with spreading activation networks generated from thesauri. In *IJCAI'07: Proceedings of the 20th international joint conference on Artifical intelligence*, pages 1725–1730, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

[104] Turney, P., 2001. Mining the web for synonyms: Pmi-ir versus lsa on toefl. pages 491–502.

[105] Turney, P., 2003. Coherent keyphrase extraction via web mining. In *In Proceedings of IJCAI*, pages 434–439.

[106] Turney, P. D., 1999. Learning to extract keyphrases from text.

[107] Turney, P. D., 2002. Mining the web for lexical knowledge to improve keyphrase extraction: Learning from labeled and unlabeled data.

[108] Turney, P. D. and M. L. Littman, 2003. Measuring praise and criticism: Inference of semantic orientation from association. *ACM Trans. Inf. Syst.*, 21(4):315–346.

[109] Twitris, 2009. Twitter through space, time and theme. `http://twitris.knoesis.org`.

[110] Watts, D. J., 2004. *Six Degrees: The Science of a Connected Age (Open Market Edition)*. W.W. Norton & Co.

[111] Weeds, J. and D. Weir, 2005. Co-occurrence retrieval: A flexible framework for lexical distributional similarity. *Comput. Linguist.*, 31(4):439–475.

[112] Wikipedia. Billboard charts. `http://en.wikipedia.org/wiki/Billboard_charts#The_Billboard_Hot_100`.

[113] Wikipedia. Wonders of the world. `http://en.wikipedia.org/wiki/Seven_Wonders_of_the_World`.

[114] Wilson, P., 1968. *Two Kinds of Power: An Essay on Bibliographical Control*. Univ of California Press, Berkeley, CA.

[115] Witten, I. H., 1999. Text mining. In *in a Digital Library. International Journal on Digital Libraries archive*.

[116] Wu, F., B. A. Huberman, L. A. Adamic, and J. Tyler, 2003. Information flow in social groups.

[117] Yang, Y., T. Pierce, and J. Carbonell, 1998. A study of retrospective and on-line event detection. In *SIGIR '98*, pages 28–36, New York, NY, USA. ACM.

[118] Yarowsky, D., 1999. *Hierarchical Decision Lists for WSD*. Kluwer Acadmic Publishers.

[119] Zhao, Q., P. Mitra, and B. Chen, 2007. Temporal and information flow based event detection from social text streams. In *AAAI*, pages 1501–1506.

[120] Zipf, G. K., 1949. *Human Behavior and the Principle of Least Effort: An Introduction to Human Ecology*. Addison-Wesley, Cambridge, Mass.