

# Mining Entities from Events



**Morsheda Akter**

Matriculation Number: 2841380

Internet Technologies and Information Systems (ITIS)  
Department of Electrical Engineering and Computer Science

Leibniz Universität Hannover

This dissertation is submitted for the degree of  
*Master of Science*

February 01, 2015

The dissertation of Morsheda Akter was reviewed and approved by the following:

## **Mining Entities from Events**

### **Prof. Dr. techn. Wolfgang Nejdl**

Professor of Electrical Engineering and Computer Science  
Dissertation Supervisor  
Executive Director of L3S Research Center

### **Prof. Dr. Sven Hartmann**

Professor of Information Sciences and Technology  
Dissertation Co-Supervisor,  
Chair of Databases and Information Systems, TU-Clausthal

### **Dr. Xiaofei Zhu**

Dissertation Mentor

01 February, 2015

I would like to dedicate this thesis to my loving parents ...

## Declaration

I hereby declare that except where specific reference is made to the work of others, the contents of this dissertation are original and have not been submitted in whole or in part for consideration for any other degree or qualification in this, or any other University. This dissertation is the result of my own work and includes nothing which is the outcome of work done in collaboration, except where specifically indicated in the text. This dissertation contains less than 20,000 words including appendices, bibliography, footnotes, tables and equations and has less than 45 figures.

Morsheda Akter  
February 01, 2015

## Acknowledgements

I would never have been able to finish my dissertation within time without the guidance of my advisors, professors, help and support from my friends and family.

I would like to express my deepest gratitude to my mentor, Dr. Xiaofei Zhu, for his excellent guidance, caring, patience, motivation, enthusiasm, and friendly advice helped me all the time of research and writing of this thesis. I would like to thank Prof. Dr. techn. Wolfgang Nejd, for his patience, thoughtful advice and financial support during my research work. I also would like to thank Prof. Sven Hartmann for guiding me for the past few years and encouraging me to develop my background.

I also remember Dipl.-Ing. Mihai Georgescu, L3S Reseach Center, Hannover, who helped me a lot. And also our course coordinator Olivia Buber, I would like to thank for helping me in many ways in many problems as it's a four different universities affiliated masters program.

Besides my advisor, professors, I would like to thank the rest of my thesis committee and the countless others who have helped me along the way.

## Abstract

Now- a- day Wikipedia is becoming the main source of data for analyzing, researching and finding insights from it. Many researchers are for this reason interested about the Wikipedia data. Understand the relationship between entities according to a given set of entities which we called seed entities. Finding insights from them, it is also very important to obtain more related entities and analyze them. Entity resolution is a problem that arises in many information integration scenarios [1]. To fulfilling this purpose visualizing entities through graph has an increasing demand and interest among researchers. To analyzing this data as a source our main goal objective is mining entities from events and study how to effectively use crowdsourcing techniques to generate an automated trustable entity graph. Based on this foundation we develop a model for generating entities and inside the page based on the link entity it extends the input seed entity. We develop models and methods that find out the co-occurrences between entities based on their events and automatically generate the entity graph. Also this produces a word cloud representation according to user's given input.

# Contents

<b>Contents</b>	<b>v</b>
<b>List of Figures</b>	<b>viii</b>
<b>List of Tables</b>	<b>x</b>
<b>Nomenclature</b>	<b>x</b>
<b>1 Introduction</b>	<b>1</b>
1.1 Introduction . . . . .	1
1.2 Thesis Overview . . . . .	2
1.3 Entity Mining and Event Retrieval . . . . .	2
1.4 Extended Seeds Entity Generation . . . . .	2
1.5 Entity Relationship Identification . . . . .	3
1.6 Crowdsourcing Technique To Refine The Graph . . . . .	3
1.7 A working prototype system . . . . .	3
1.8 Thesis Outline . . . . .	3
<b>2 Related Work</b>	<b>5</b>
2.1 Entity Extraction . . . . .	5
2.2 Entity Extraction Based On Web Ontology . . . . .	5
2.3 Entity Extraction Based On Query Log . . . . .	9
2.4 Entity Extraction Based On Wikipedia . . . . .	11
2.5 Crowdsourcing . . . . .	13

---

2.6	Different Perspectives of Crowdsourcing . . . . .	14
2.6.1	Cloud Labor . . . . .	14
2.6.2	Crowd Creativity . . . . .	15
2.6.3	Crowdfunding . . . . .	15
2.6.4	Distributed Knowledge . . . . .	15
2.6.5	Open Innovation . . . . .	15
2.7	Motivational Factors For Crowdsourcing . . . . .	16
2.8	Game-Based Crowdsourcing . . . . .	17
2.8.1	Score Based Game: Image Labeling . . . . .	18
2.8.2	Point Based Game: Meta-data . . . . .	18
2.8.3	Money or Reword Based Game . . . . .	20
2.9	Micro-Task Based Crowdsourcing . . . . .	22
<b>3</b>	<b>Mining Entities from Events</b>	<b>24</b>
3.1	Entity Mining and Event Retrieval . . . . .	24
3.1.1	Natural Language Processing (NLP) . . . . .	26
3.1.2	Annotator . . . . .	26
3.2	Automated Extended Seeds Entity Generation . . . . .	28
3.2.1	Seeds Entity Generation . . . . .	29
3.2.2	Common Event Extraction . . . . .	30
3.3	Entity Relationship Identification . . . . .	31
3.3.1	The Graph Construction . . . . .	32
3.3.2	Tag Cloud Visualization . . . . .	33
3.4	Crowdsourcing Techniques to Refine the Graph . . . . .	33
3.5	Requirement Analysis . . . . .	36
3.6	Design . . . . .	36
3.7	Pipeline Outline . . . . .	36
3.8	Associated Sequence Diagram . . . . .	36
3.9	The different Perspectives of Entity Graph . . . . .	38
3.9.1	The Data Analyst and Data Mining . . . . .	38
3.9.2	Users Point of View . . . . .	38
3.10	The Prototype System . . . . .	39
3.10.1	User Login . . . . .	39
3.10.2	Description of the Main Page . . . . .	39



---

<b>4</b>	<b>Evaluation and Results</b>	<b>41</b>
4.1	Using Technologies . . . . .	41
4.2	Database Structure . . . . .	42
4.2.1	Table Description . . . . .	42
4.2.2	ERD Diagram . . . . .	43
4.3	Entity Extraction Components . . . . .	43
4.4	Category Based on NER . . . . .	44
4.5	Evaluation of the Graph . . . . .	44
4.5.1	Results of the Graph . . . . .	46
4.6	Evaluation of Crowdsourcing Process . . . . .	49
4.6.1	Results of the Crowdsourcing . . . . .	50
4.7	Discussion . . . . .	51
<b>5</b>	<b>Conclusion and Future Work</b>	<b>53</b>
5.1	Conclusion . . . . .	53
5.2	Future Work . . . . .	54
	<b>References</b>	<b>55</b>
	<b>Appendix A Extended Entity Extraction</b>	<b>61</b>
	<b>Appendix B Entity Classification Sample</b>	<b>63</b>

## List of Figures

2.1	A process for automatic ontology population . . . . .	8
2.2	Crowdsourcing Industry Revenue Growth . . . . .	13
2.3	Image Labeling ESP Game . . . . .	18
2.4	Dora’s Lost Data . . . . .	20
2.5	Cerberus Game Interface . . . . .	21
3.1	Entity Extraction Process . . . . .	25
3.2	Overall Annotator Architecture [44] . . . . .	27
3.3	Parse Tree of a Sample Sentence . . . . .	29
3.4	Wikipedia Linked Entities . . . . .	30
3.5	Overview of the Entity Extraction Process . . . . .	31
3.6	Query Search For the Entity . . . . .	33
3.7	Entity Graph Visualization . . . . .	33
3.8	Tag Cloud Visualization . . . . .	34
3.9	Adding an Entity with Event . . . . .	35
3.10	After Adding the Entity . . . . .	35
3.11	Block Diagram of the Events Extractor . . . . .	36
3.12	Class Diagram of the Graph Visualization . . . . .	37
3.13	Pipeline Outline of the Process . . . . .	37
3.14	Sequence Diagram of the extraction process . . . . .	38
3.15	User Login Page . . . . .	39
4.1	Data Model of the Events Extractor . . . . .	43

---

4.2	Event Extraction Process Algorithm . . . . .	44
4.3	Average Precision of Entities . . . . .	47
4.4	Average Precision of Edges . . . . .	47
4.5	Modification of Log Table . . . . .	49
4.6	New Reformed Entity Graph . . . . .	50
4.7	Precision before and after Crowdsourcing . . . . .	51
4.8	Precision Based On Frequencies . . . . .	52
A.1	Sample Pattern For The Extension Of Entities . . . . .	62
B.1	Sample Classification For The Extension Of Entities . . . . .	64

## List of Tables

4.1	Labeling Status of Entities . . . . .	45
4.2	Relevant and Irrelevant Entity Labeling . . . . .	46
4.3	Entity Labeling of the Graph Jimmy Carter . . . . .	47
4.4	Relevant and Irrelevant Entity Labeling of MacArthur . . . . .	50
B.1	Calculation of Entity Graph Labeling. . . . .	65

## 1.1 Introduction

As technology emerged in several ways, people are becoming dependent upon it on their daily life, such as using a hi-tech mobile or a laptop computer is now becoming a part of life. This technology also affecting peoples social life, organization, office or places in various ways. As human being involving with numerous events, this event also creates user-generated, unstructured content on the Internet which significantly increasing day by day. But those informations are particularly stored in as unstructured data, large document collection or in web pages. Event and information extraction from unstructured textual documents is an important and critical task in the realm of natural language processing and knowledge management systems.

Traditional state-of-the-art IE systems are mostly based on supervised machine learning techniques which require hand-crafted training corpus or extraction pattern as input. This manual process is also time consuming and involves substantial human effort as well as expensive. This inspired researchers to invent semi-supervised or automatic extraction system.

Information extraction for entities is a technological process based on natural language analysing in order to extract the information about events. In our paper we propose an Application which automatically extract entities from Wikipedia events and visualize entity relationship in a graphical manner. It also finds out the relevant entities related to a particular event. And finally we introduce a crowdsourcing technique to refine our entity graph based on crowds opinion which makes this tool unique for a historical contribution.

To achieve this goal we use the enormous and increasing, multilingual, free resource of online encyclopedia Wikipedia to create NE-annotated corpora. We transform links between

encyclopedia articles to mine more seed entities from them. Each new term or entity then used to extract more relevant events from those articles. For the named entity classification technique we consider person name and organization. So this historical evidence with other relevant information needs to be in a concise form for visualization. Users those who seek quick information about the entity events without reading the whole Wikipedia article, this visualization technique is perfect for them.

## **1.2 Thesis Overview**

Overall, this project aims to develop an automated web application to support identification of entities from unstructured corpus and to mine extended seeds as named entities for extract events information. And finally, visualize the relationship entity graph and events based on entity co-occurrences in the articles.

The main objectives and contributions arising from this thesis are depicted as follows:

## **1.3 Entity Mining and Event Retrieval**

We formally define extraction process which incorporates some input entities as seed to automatically generate events from unstructured text corpus like Wikipedia articles. Seeds are words, terms representing the entities. For every seed entity it generates particular entity information page from Wikipedia data.

## **1.4 Extended Seeds Entity Generation**

We propose a technique to extended the process and mining more seed entities using the machine learning techniques which make use of features like cosine-similarity between the Wikipedia articles of the target entities. Then from those entities linking we mine more seed entities and apply the CRFClassifier [59] with Stanford NER model [24] to named entity classification: organization and people's name. Those newly generated seeds then used to extract events from Wikipedia text corpus.

## 1.5 Entity Relationship Identification

Event topic related entities and relation instances are discovered based on the co-occurrences between the entities. And after go through the various filtering process the relationship between the entities along with the related associated common events are identified.

## 1.6 Crowdsourcing Technique To Refine The Graph

Based on the co-occurrences and relationship between the entities we generate the entity graph. And follow a crowdsourcing technique to refine the generated graph. Using the power of the crowd finally we display our modified refined entity graph.

## 1.7 A working prototype system

We have implemented a working prototype web application to support our proposed mining entities from events topic entity and relation extraction task. It incorporates a few of the document extraction strategies and proposed in this thesis. Several natural language processing (NLP) components like named entity recognition (NER), parsing and co-reference resolution have also been developed and incorporated into this system.

## 1.8 Thesis Outline

The rest of the thesis is organized as follows:

Chapter 2 discusses the topic related work in the area of entity extraction and crowdsourcing techniques. We discuss different evaluation techniques for entity and events relationship as well as some existing classification methods.

In Chapter 3 we present a formal definitions of our proposed mining entities from events and entity, events relation extraction process. We also discuss detailed descriptions of our crowdsourcing based refinement technique for the entity graph.

In Chapter 4 we provide a detailed explanation on the experimental setup and evaluation process of our entity extraction techniques from Wikipedia events and crowdsourcing refinement techniques for entity graph on several datasets. We explain how the dataset was selected, gathered, pre-processed and processed and then the evaluation results of the final refined entity graph. And we have also provided a thoughtful discussion at the end of this chapter.

Chapter 5 concludes this research with a summery and an overview of the possibilities of future research direction.



## 2.1 Entity Extraction

The automatic extraction of information from unstructured sources has unlocked new era for querying, analysing, and organizing data. New technology and easy online access to both structured and unstructured data has engaged a different community of researchers for various aspects of the information extraction problem such as machine learning, databases, information retrieval, web and computational linguistics etc. Depending on various forms of information extraction on the web ontology increased the extent and diversity of applications.

The classical named entity recognition systems normally classify named entities into categories, such as person, location and organization. Most of the researchers has been investigating supervised learning approaches, which generally require manually large amount of tagged texts as a training data set. It also requires lots of human effort and funding. So researchers are interested to develop semi-supervised NE recognizer systems. Some of the previous approach are shortly discussed here.

## 2.2 Entity Extraction Based On Web Ontology

Earlier entity extraction systems were mostly rule-based [7, 19, 52] whereas in recent years statistical methods [18, 21, 60] gained more popularity. These methods convert the entity extraction task to a problem of decomposing the unstructured text, and then labelling various parts of the decomposition [54]. These two models are used in parallel depending on the nature of the extraction task. Between these two models there exists another mixed model which is called hybrid models [13, 17, 23, 50] that has the both rule-based and statistical

methods benefits.

Giuliano and Gliozzo [28] represented an instance-based learning approach for fine-grained named entity categorization from a partially populated web ontology. This process is based on a method successfully employed in lexical substitution and estimated the plausibility of sentences by using Web data. To enrich an existing ontology it can be used in different domains and languages with new entities extracted from texts by a named-entity recognition system or databases. When a new instance has to be classified, they first collect snippets from the Web containing it. They substitute the new instance with each of the training instances for each snippet. To estimate the correctness of each substitution, based on the occurrences in the Web they calculate a plausibility score by ranking a given list of synonyms according to a similarity metric. In another approach Giuliano [27] presents a kernel-based process that implicitly map entities, represented by aggregating all contexts in which they occur, into a latent semantic space derived from Wikipedia. Both of these approaches, entities extracted from unstructured textual documents and collect sufficient information for each named entity in which they occur to query the search engine.

A weakly supervised approach of Tanev and Magnini [61] proposed to automatically populating ontology from text with named entities considering "two high level categories - geographical locations and person names and ten sub-classes for each category". For each sub-category, the algorithm learns a syntactic model exploiting the lexico-syntactic features and classifies a new entity. They presumed in their test data set entities are not ambiguous. For each sub-class, they automatically learn a lexico-syntactic model with weighted features from a syntactically parsed corpus and a list of training examples, i.e. co-occurring words which typically with the members of that class in certain syntactic positions. Then unknown Named Entities are used to classify with this model in the test set. In their experiment they defined Ontology Population as a given a set of terms  $T = t_1, t_2, \dots, t_n$ , a document collection  $D$ , where terms in  $T$  are supposed to appear, and a set of predefined classes  $C = c_1, c_2, \dots, c_m$ , denoting concepts in an Ontology, each term  $t_i$  has to be assigned to the proper class in  $C$ . They assume that (1) classes in  $C$  are mutually disjoint and (2) each term is assigned to just one class. With Named Entity Recognition and Classification (NERC), their defined OP shows a strong similarity [61]. In NERC each occurrences of a recognized term has to be classified separately, while in OP independently the term of the context in which it appears, that has to be classified. More statistical data such as a class appearance frequency feature could be exploited in different training terms.

Automatically ontology populating with named entities which is extracted from the unstructured text has become an interesting key issue for Semantic Web and knowledge man-

agement techniques. According to Shen [56] this process naturally consists of two subtasks: (1) for the entity mention from the unstructured text whose mapping entity does not exist in the ontology, attach it to the right category in the ontology (i.e., fine-grained named entity classification), and (2) for the entity mention whose mapping entity is contained in the ontology, link it with its mapping real world entity in the ontology (i.e., entity linking) [56]. Shen [56] proposed APOLLO, a graph based approach for populating ontology with named entities. To resolve this chore via random walks on graphs it leverages the rich semantic knowledge embedded in the Wikipedia. APOLLO is based on the extension of the distributional hypothesis [30] that if the contexts are semantically similar where two named entities appears; they are expected to belong to the same category. For an initial ontology, they used a list of labeled named entities with known categories as a training data. Therefore, first they identify all the Wikipedia concepts appearing in the context, for each given entity mention/named entity with its related document context and also consider the set of these perceived Wikipedia concepts as the semantic signature of this named entity. Then they construct a graph consisting of all the entity mentioned into the populated ontology and the Wikipedia concepts existing in the corresponding semantic signature. They weighted the edges in the graph based on the Wikipedia articles link structure. The nodes of the named entities are annotated with their category labels and applying with the Adsorption label propagation algorithm [8] other unlabeled entity mention nodes are classified based on the rich graph structure. They validate for each entity mention whether there exists a named entity in the ontology they could link with. Otherwise they attached this entity with the biggest distribution category. Also in their method they resolve the ambiguity problem of populating ontology with named entities integrally. Their conducting experimental study evaluates the performance of APOLLO and results show that it achieves significant accuracy for the ontology population.

Amaral [6] represents a tool to build knowledge base ontology from specialized texts, which detects proper names, locations and dates from texts by using manually written linguistic rules. Their model extracts the entities as well as also interprets the information and adapt in a specific corpus in French. Since a wide variety of texts were digitized and created through Web, Information extraction is fundamental key point. To provide such information and efficiently share conceptualizations with experts and researchers, ontology learning is a good option in the area of IE. Analyse the huge quantity of textual data and continuous information evolution makes the extraction process more difficult. Therefore automatic extraction process makes it possible to handle this massive textual data. Extraction of named entities (NE) is the first and important task in ontology learning and information retrieval

domain. Their process describes a mining method of named entities with using some linguistic rules and lexicons for improving the search in annotated corpora. This ontology learning architecture transforms raw text data in a semantic network. They add lexical entries to expand the global lexicon which is created with ontology concept names and their synonyms found in a French dictionary on the Web. During the searching step it applies the lexico-syntactic rules to improve the detection of people roles and locations and learning process creates new concept names. For the natural language processing they used NooJ a syntactic parser to process and represent all types of linguistic units [57].

Unstructured texts are the source of knowledge so for the construction of knowledge-based systems it is necessary to handle and represent it automatically. Ontologies are formalism for knowledge representation capable of expressing a set of entities, their relationships, constraints and rules of a given domain [29, 47]. They are used by modern knowledge-based systems for representing and sharing knowledge about an application domain. These knowledge representation structures allow the semantic processing of information and, through more precise interpretation of data; systems have greater effectiveness and usability [26]. Ontology population is the process used to designate the techniques for extracting and classifying instances of concepts, relationships and properties of ontology [20]. Faria [20] proposed a process for semi-automatic population of ontologies to acquire and classify ontology instances from text focusing on the application of natural language processing and information extraction techniques. Figure 2.1 is showing the automatic ontology population process. According to their method it consists of two phases: "Extraction and Classification of Instances" and "Instance Representation".

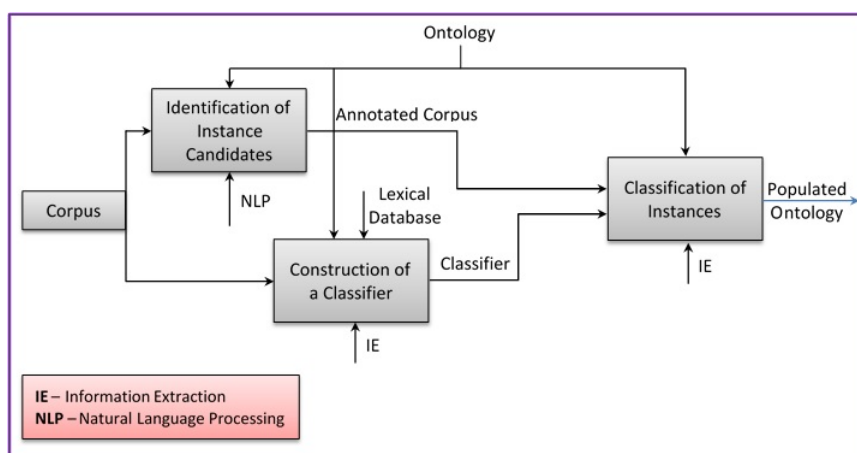


Fig. 2.1 A process for automatic ontology population

The "Extraction and Classification of Instances" phase aims at extracting a subset of all

possible relationships and class instances. In the "Instance Representation" phase, an ontology specification language like OWL is used to formally represent the ontology instances [20]. Formally, ontology can be defined as the tuple:

$$O = (C, H, I, R, P, A)$$

where,

$C = C_c \cup C_I$  is the set of entities of the ontology. They are designated by one or more terms in natural language.

$H = kind\_of(c_1, c_2) \mid c_1 \in C_c, c_2 \in C_c$  is the set of taxonomic relationships between concepts.

$I$  is the set of relationships between ontology elements and it's instances.

$R = rel_k(c_1, c_2, \dots, c_n) \mid \forall i, c_i \in C_c$  is the set of ontology relationships that are neither "kind\_of" nor "is\_a".

$P = prop_K(c_i, datatype) \mid c_i \in C_c$  is the set of properties of ontology entities and it's the basic data type.

$A = condition_x \Rightarrow conclusion_y(c_1, c_2, \dots, c_n) \mid \forall j, c_j \in C_c$  is a set of axioms.

## 2.3 Entity Extraction Based On Query Log

Jiang [36] proposed a novel solution, Comparable Entity Graph Mining (CEGM) algorithm which automatically using predefined query patterns through some heuristics and statistical measures from user search queries, firstly mine comparable seed entities. And construct an initial entity graph with vertexes (entities) and edges (degree of comparability). To group similar seeds into the same cluster a graph cut algorithm is utilized. This also helps to estimate the number of domains where entities are being frequently compared. To learn the patterns new seeds are then sent to the query log. Those discovered entity pairs are then organized into an open-domain comparable entity graph. Based on a newly proposed strategy patterns are then ranked and sent to the query log. In a bootstrapping fashion it discovers more entity pairs with a confidence classifier. In their application of the learned entity graph, the entity recommendation in Web search is empirically studied.

Valuable information is hidden inside unstructured text data. Processing each document is quite expensive and involves several steps which are not feasible for large databases. Documents can be filtered at various stages of the extraction process [42]. To extract this information with minimal time constant, Agichtein [4] proposed an automatic query-based technique to retrieve documents for extraction those are useful of a target relations with

a user-provided examples of tuples from large text databases or even the web using the DocumentSample algorithm, which can be adapted to new domains with minimal human effort. Then they run the information extraction system over the current sample documents and extract a new set of tuples. New Seed tuples are selected from these tuples as a subset to start a new sampling round. To learn queries later they match with additional useful documents applying machine learning and information retrieval techniques. Finally queries are then returned as QXtract's output and processed by the information extraction system.

Chakrabarti et al. [14] present HubRank, a new system for fast, dynamic, space efficient proximity searches in ER graphs where nodes are entities such as person, company, organization etc and edges are relations obtained by parsing unstructured text. Using query log statistics HubRank computes and indexes certain "sketchy" random walk fingerprints for a small fraction of nodes during pre-processing. At query time, a small "active" sub-graph is identified, bordered by blocker nodes with indexed cached fingerprints. To form approximate personalized Pagerank vectors (PPVs) these fingerprints are adaptively loaded to various resolutions and iteratively computed for remaining active nodes. It also saves memory and computation significantly. According to their estimation this HubRank pre-processes and indexes 52 times faster than whole-vocabulary PPV computation.

Jayaram et al. [34] proposed a system GQBE (Graph Query By Example) to query data by example entity tuples, without requiring users to form complex graph queries. To capture user's query intent the query graph discovery component of GQBE fulfills the requirement and automatically derives a hidden maximal query graph (MQG) based on input query tuples. According to several frequency and distance-based heuristics the edges of MQG are weighted. They also capture the relationship between nodes in the data graph and their neighbouring entities. Based on the matching of MQG, it models the space of query graphs by a query lattice. Its query processing algorithm efficiently finds and ranks the top-k approximate answer tuples and partially evaluates the query lattice. They also conducted extensive experiments and user study to evaluate GQBE's accuracy and efficiency on the large Freebase and DBpedia datasets. They compare their model with a graph querying framework NESS [18] and shows that GQBE is twice as accurate as NESS [37] and most of the queries it outperforms NESS on efficiency.

Managing, analysing and visualizing data from many structured and unstructured data sources is often challenging [43]. Banko [9] introduces Open IE (OIE) system, a new extraction paradigm without requiring any human input where over its corpus the system creates a single data-driven pass and extracts a large set of relational tuples. For a small sample corpus as input, Learner uses a parser [40] to automatically identify and labels candidate

extractions as "trustworthy" or not and uses to train a Naive Bayes classifier. Noise tolerant learning algorithm helps the system to recover the errors made by the parser. For all possible relations one or more candidate tuples are generated by the Extractor from each sentence and sends each candidate to the classifier. Based on a probabilistic model of redundancy in text the Assessor assigns a probability to each retained tuple [22]. They introduce a fully implemented, highly scalable OIE system TEXTRUNNER, where the tuples are allocated a probability and indexed to support proficient extraction and exploration via user queries.

## 2.4 Entity Extraction Based On Wikipedia

Our research work is also related with existing entity mining research [10, 43, 48, 56] in terms of related entity pair extraction from Wikipedia events.

Wikipedia is a large online resource of knowledge data bank about various aspects. So now a day's researchers become more interested to mine information from it. Some Wikipedia articles have a structured information block which known as infobox [10]. The key information of the entity is summarized by the infobox which is composed of a set of attribute name and value pairs. To achieve the goal of Wikipedia entity expansion and attribute extraction by mining the rich and valuable semi structured data records from the Web, Bing et al. [10] develop a new framework. In their model as seed input they used automatically collected few existing entities from a particular Wikipedia category. It also explores their attribute infoboxes to acquire clues for the discovery of more entities of the same category and as well as the attribute content of the newly discovered entities. For this a semi-supervised reliable learning model with CRF (semi-Markov CRF [55]) is developed for extracting the entities and their attributes by exploiting the unlabeled data in the semi-structured data record set and controlling the label regularization under the guidance of the proximate record graph.

In Malik's [43] "approach important entity attributes from the structured content and the entity neighbourhood in the graph are automatically summarized as the entity 'fingerprint'". A highly interactive user interface provides exploratory access to the graph and supports common business use cases. They present results of experiments performed on five years of news and broker research data, and show that Atlas is able to accurately identify important and interesting connections in real-world entities. They also demonstrate that Atlas entity fingerprints are particularly useful in entity similarity queries, with a quality that rivals existing human maintained databases. Atlas uses a directed graph where entities are represented as vertices, and "edges are generated using entity co-occurrences in unstruc-

ture documents and supervised information from structured data sources" [43]. Vertices and edges store pointers to relevant unstructured and structured content and meta-data, to facilitate efficient access to underlying content and to allow for temporal analysis. Since each entity may be connected to a large number of other entities, "Atlas computes significance scores for edges using a novel method that combines supervised, unsupervised and temporal factors into a single score" [43].

Nothman [48] proposed a technique to exploit Wikipedia's links and to create a massive corpus of named entity annotations by classifying the target articles into common entity types. Wikipedia generally performs better than comparing other cross-corpus train/test pairs like: MUC, CONLL and BBN corpora. The original sentence can be automatically annotated with facts that are related with the other page and be extracted from Wikipedia to form an enormous corpus for NER training. In their system they transform links between encyclopaedia articles into named entity annotations. Each new term or name mentioned in a Wikipedia article is often linked to an appropriate article. They proved that their Wikipedia-derived corpora are usually able to exceed the performance of non-corresponding training and test sets, by up to 8.7% F-score.

On the Internet day by day the amount of user-generated, unstructured content increases significantly. So the demand to extract information automatically from the unstructured text has increased among the researchers. Chasin [16] represents a method to extract and display temporal entities in textual documents. Using a classifier the method can identify all important events in a document along with named entities (people, places, and organizations etc.) to which they are related in terms of a time-line and a map. Event and temporal information extraction from plain text is a crucial task for natural language processing and knowledge management [16]. They used historical Wikipedia articles because of the availability of such high quality articles are huge in numbers. They also used several existing tools such as Evita, Google Maps, publicly available implementations of SVM, HMM and CRF, and the MIT SIMILE Timeline. In the pre-processing stage Evita combines linguistic and statistical knowledge for events recognition in the TimeML format. It also identifies instances of events which help to establish temporal relations and the class of an event along with its occurrences. TextRank with their own sentence similarity function automatically weight the sentences based on importance. To identify temporal relations among events they used TARSQI toolkit and regular expressions to extract occurrence times.

Ganti et al. [25] introduces a method that considers an entity's context across multiple documents containing it, and exploiting word n-grams and existing large list of related entities as features. They focus on the extraction of targeted relations based on co-occurrence



between entities. They used Wikipedia articles with list of instances to generate training and test data. An entity occurs across multiple documents within the contexts be "aggregated" and then used to categorize an entity. They showed that based on their aggregate context strategy perform better comparing it with a single context classifier using 10K n-gram features.

## 2.5 Crowdsourcing

In recent years Crowdsourcing is one of the emerging phenomena in the evaluation of information retrieval systems that has been getting increasing attention both scholars and researchers. It is a technique, that use human abilities to solve problems for computation those computers are not good at. Virtual diversity of the crowdsourcing recognized with any type of web-based collaborative activity, such as user innovation or co-creation. Jeff Howe and Mark Robinson [32], in June 2006 first introduced crowdsourcing in a Wired Magazine article. By definition, "crowdsourcing is the act of a company or institution taking a function once performed by employees and outsourcing it to an undefined network of people in the form of an open call" [33]. Following is the sample Graph 2.2 of 15 leading crowdsourcing service providers according to the massolution research where it showed that growth in the global enterprise crowdsourcing market is accelerating followed by almost 75% increase in 2011.

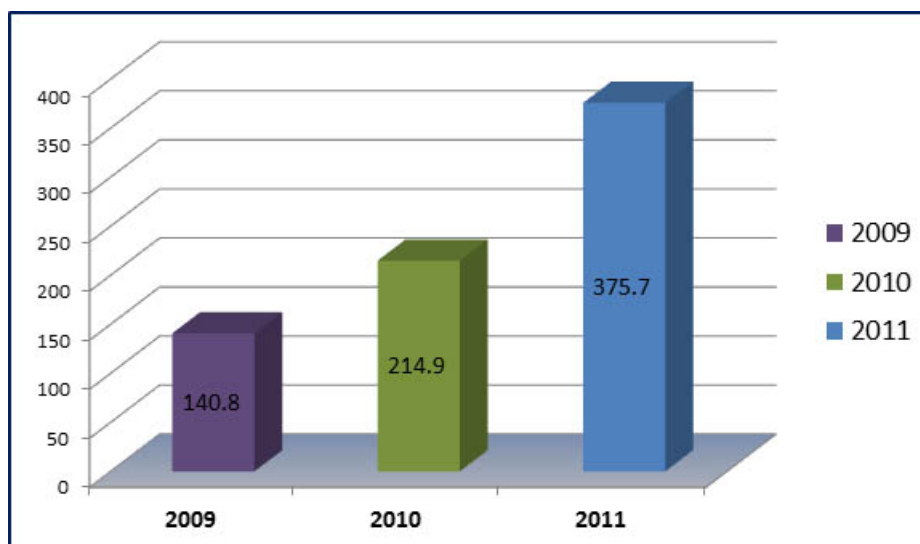


Fig. 2.2 Crowdsourcing Industry Revenue Growth

It organizes tasks in different skills and expertise with different forms and distributed

it among the crowd. A distinctive crowdsourcing process works in such a way: When an organization needs to perform some task which involves lots of human power, generally they categorize their tasks and release it in the online crowd where people are interested in performing these tasks on the organization's behalf for some specified amount of fee. After successful completion of the tasks crowd users submit their work in online platform and organization evaluates and analyse the quality of the work. Some examples of such systems are Wikipedia, Mechanical Turk, Crowdrise, Scoopshot, Cloud Flower etc.

## 2.6 Different Perspectives of Crowdsourcing

To differentiate crowdsourcing is really confusing. Researchers try to classify it in different points of view. Basically some of the points are always the same. Crowdsourcing.org and KL Communications <sup>1</sup> try to classify crowdsourcing using sentiment analysis into 5 different perspectives which are described in the following:

### 2.6.1 Cloud Labor

Cloud labor is a distributed virtual labor pool. A board range of tasks from simple to complex are available to fulfill on demand request. For the mercy of cloud computing now servers can provide huge computing power and storage facility through the internet on demand so that companies can get their workforce in online to perform their tasks. Mechanical Turk Created by Amazon in 2005, which is the low-wage virtual labor phenomenon for the digital age. Thousands of workers perform millions of tiny tasks for companies within a given day.

Some of the tasks such as extracting and processing raw data, identifying spelling errors and calculating financial figures where computer proves very worthy enough whereas some other tasks such as recognizing irony, accurately reading the text off a photograph, detecting a positive or negative bias in an article, determining ambiguous search results computers are less able to perform. Those particular jobs where a computer cannot perform well, crowdsourcing comes to play its role where individuals are tasked to do those works.

---

<sup>1</sup><http://www.crowdsourcing.org/editorial/the-five-crowdsourcing-categories-ranked-popularity-in-social-media/10176>

### **2.6.2 Crowd Creativity**

People have the capability to be working on their own with an extraordinarily innovative way. However, as a single person to draw on for inspiration each of us has a limited amount of knowledge, skills and experiences. Research shows that multiple minds are better than a single one.

To share ideas, pool resources, design and develop original art, leverage diverse skill sets, and produce fascinating, sometimes world-changing creative artifacts crowd creativity linking people through online from around the world. Studies show that certain conditions need to be fulfilled to become a highly creative group of people. One of them is diversity. Increasing diversity also increases creativity.

### **2.6.3 Crowdfunding**

Crowdfunding is the process of contributing projects by a multitude of people from online sponsors, investors or donors in order to attain a certain monetary goal for profit or non-profit initiatives or enterprises. It has three types of models: (1) Donations, Philanthropy and Sponsorship where there is no expected financial return, (2) Lending and (3) Investment in exchange for equity, profit or revenue sharing. Example of this kind of organization is Kickstarter.

### **2.6.4 Distributed Knowledge**

Distributed human computation helps to collect more information resources from a distributed systems or a group of contributors. News, forecasting, journalism, user-generated knowledge systems helps to aggregate, develop and share knowledge for crowdsourcing. However a huge number of users contribute information on the open web so there has no guarantee that the provided information would be precise.

### **2.6.5 Open Innovation**

Open Innovation concept implement, develop and generate new innovative ideas outside the group. Widely used distributed knowledge decreases the boundaries between an organization and the world's open environment, companies cannot rely only their own research and ideas. To maintain a competitive market, reducing transaction cost, finding new business opportunities, building appropriate teams, or even solving a difficult problem open innovation concept is now appreciated in different companies.

## 2.7 Motivational Factors For Crowdsourcing

Crowdsourcing is one of the emerging phenomenon's that has been gained great attention and importance from both scholars and experts over the years and has been used to capture ideas from crowd. Global companies are adopting crowdsourcing idea to connect with and get feedback from the users [31]. Yet it is hard to predict what would motivate participants to innovate and to classify different motivational factors and incentives in online crowdsourcing platform.

Empirical studies have revealed that crowdsourcing system's use is driven by both intrinsic and extrinsic motivations [12, 58, 66]. In Soliman's [58] research highlights the dynamic nature of human motivation and shows that by including the aims of motivation in the analysis, they can better capture the dynamic nature of motivation across time. In intrinsic motivation people do something for their own enjoyment and do not expect anything else in return, like: fun, interest, enjoyment etc. Extrinsic motivation refers to the factors that bring something in return, like: money, benefits, recognition etc. According to their exploration, six motivational factors together have shaped the use behavior. These motivational factors are: the opportunity to gain a financial reward, the opportunity of publicity, enjoyment, curiosity, gaining non-financial rewards (e.g., skill development and future employment), and altruism.

### Financial Reward

The possibility to make some easy money always influence users intend. This is also the easy way to recruit workers in online. Using Scoopshot gives users the financial reward which makes it worthwhile. Some companies like a distributed call center, they employ people to handle online call for businesses, such as LiveOps<sup>2</sup>. Another example of intermediary for businesses is CrowdFlower<sup>3</sup>, they works with different services and compensate workers with money, gift voucher, virtual currency for games.

### Publicity

The most influential motivational factors is the gaining publicity or recognition. People like to get some attention and motivated by the public recognition. Some system motivated people for their quality answers and permitted access to desirable tasks whereas frequently submitted bad answer or work could block the account.

---

<sup>2</sup><http://www.liveops.com/>

<sup>3</sup><http://www.crowdfLOWER.com/>

### **Enjoyment**

Enjoyment is another reason for using the service. Just for Fun or games for fun also attract support. People love to pass their leisure time and having some fun which can also attract people's attention for the crowdsourcing work.

### **Curiosity**

To discover and try out a new technology always make people curious and increase interest towards the system. Some common phrases were reflecting people's initial interest such as interesting idea, give it a try and try it out.

### **Non-Monetary Personal Gains**

Various non-monetary personal gains like: skill and career-development were apparent motivations to use the system. For example people would brand to publishing their name or would refer their own work as a career perspective.

### **Altruism**

From their analyses the user's willingness to help others also emerged as a final motivational factor. Without expecting anything in return this kind of altruism reflects the users enthusiasm to contribute to the service.

## **2.8 Game-Based Crowdsourcing**

Game is a leisure activity with no explicit goal and sport of physical ability involved. In Prensky's [49] point of view it has an educational perspective; take it to be a subset of play which is structured in such a way that helps us to learn. The basic idea of the game based crowdsourcing is to involve users to play online games where users complete simple tasks, such as tagging images or correcting Optical Character Recognition (OCR) errors etc.

A well-designed crowdsourcing game also tempts to motivate more people to participate. September 2011 in US, Kabam social gaming research showed that 50 million Internet users playing casual social games and 30% people play casual games on social networks, 8% on mobile devices, and 8% on casual game portals.

### 2.8.1 Score Based Game: Image Labeling

Classification game like ESP game [5] which is an online game designed with a keyword based image labeling. Some of the images on the internet don't have their description. Image description helps to improve algorithms for image search query and inappropriate content could be filtered by this keyword. It groups players into pairs and both of them it shows the same image, after that it awards points when the same word players type on their keyboards. Points are awarded every 2.5 minutes and a new image is displayed. Figure 2.3 is an example layout of the image labeling ESP game.



Fig. 2.3 Image Labeling ESP Game

Players can't communicate each other directly. Some of the words are taboo that means they can't get any points entering for those words. If a certain number of paired players entered the same word for an image, the other new players will be notified and can't use the same word for that image. The accuracy of the ESP game indicated that 85% of the keywords associated with the images are useful [65].

### 2.8.2 Point Based Game: Meta-data

For museums, libraries and any archive related organization, it is very difficult to create a collection of archiving meta-data as it is expensive and also time consuming whereas

crowdsourcing open up the prospects to the whole world of creating and refining the fine grained content of these collections.

Usually games those are played with words are known as Meta-data Games. Something is drawing or describing in a game pictorially and we need to give a name or may be recite it. Crowdsourcing games should be enjoyable to play but as well as should create meaningful, accurate meta-data for utilization. Professional cataloguers or specialist uses lot of technical terms which is not user friendly for the general visitor. Research shows that if visitors can tag those objects that can create a link between the semantic gap and the common people.

The aim is to adding semantic meta-data and increase its discoverability Meta-data Games [51] could be useful for tagging content. Mia Ridge in his research project to help improve the records of history museum collections he designed casual browser-based games. Most of the records are practically difficult technical terms, nearly-duplicate, poorly catalogued or in either way insufficiently digitized. During a game play people usually use words to describe an object that means actually they create meta-data about that object. Facts, stories or fascinating relations contributed by players could be very handy to those people who are trying to find the same object or may be helpful to a museum to discover new information.

This tagging game 'Dora' were designed with the player's registration form or login menu in the right side panel along with a leader-board that showed the players all time achieved top scores. Also player can share this game in the popular social media sites. The tag-line "play games, make museums better" deliver a philanthropic appeal for the museums. In the Figure 2.4 is displaying the picture of the game: Dora's Lost Data.

In the game an attractive character Dora is presented herself as a young curator to the player who needs their help to replace some lost data for museum's collection. Players are asking to describe 5 random objects per round in the game and per tag they are given 5 points. After completing a level players are congratulated by Dora and presented their performance in a concise way.

Based on Mia Ridge's experiment from 46 countries there were 969 visitors with 1,438 visits and 5,512 page views in the analysis period (December 3 – March 1). Overall, 47 registered users created 6,039 tags in 196 game sessions where 2232 unique tags, and 37 facts for 36 objects.

**Dora's lost data**

"Hi, my name is Dora, and I'm a junior curator. It's my first day and I've made a big mistake - I accidentally deleted all the information we were going to add to our collections online. I need to re-label them, and quickly..."

Can you help? **Add words about the thing in the picture that would help someone find it on Google** - how it looks, what it does, who might have used it - anything you can think of."

**LA TERRA**  
In luftigen Höh'n ... Die Jungfrau und das Berner Oberland ...  
Herausgeber von der Sektion Berlin des Deutschen und O

[View object on the British Library on Flickr Commons site \(opens in new window\).](#)

Date: 1898 Place: Berli (Accession num: 1404296)

—Add words ('tags') to describe this object—  
Tags:   
Tip: separate each tag with a comma, like this: tag, label, a phrase, name, names.

Not sure about this object? [Get a different object.](#) (It won't affect your points.)

**Your score for this game:**  
No points for this game yet. Start playing to earn points  
[Login](#) or [register](#) to save your points

**Objects you've played in this game**

? ? ? ? ?

So far players like you have improved **714 records for 3 museums and libraries** through games on this site.

**Top registered players (all games)**

Han	(5620 points)
Josie	(5000 points)
lrnc	(4400 points)
mia	(3985 points)
Hapburn	(2190 points)
ad	(1785 points)
archiwicz	(1225 points)
miathepink	(1115 points)
JoK	(890 points)
jannifuchs	(870 points)

**Register or login to save your points**

Username

Password

Remember Me

[Register](#)  
[Lost Password](#)

**Help the British Library tag a**

Fig. 2.4 Dora's Lost Data

### 2.8.3 Money or Reward Based Game

Crowdsourcing uses individuals where human perception exceeds the aptitudes of machines to deal with those kind of data corpus. In their project woud [64] explores the crowd recognition power and for support of scientific research they try to find out if crowd can apply high level semantic concepts to features of the Martian surface photos. Through a serious game they were scrutinised different kinds of help function for players to identify surface features on Mars.

To motivate players they used two methods:

1. They offered small financial rewards for processing the data units.
2. The outlook and the game environment is designed attractive and as well as entertain-



ing that players will be inspired to process data for free.

Cerberus is a computer game which has developed in a way that it can allow players to tag surface features on Mars. In their experiment the player's performance is measured in terms of motivation and precision. The amount of work done by the player is epitomized by motivation and the quality of the work replicates precision. A rich gaming experience with explicit support tasks motivated players significantly. In Figure 2.5 is displaying the game interface of Cerberus.

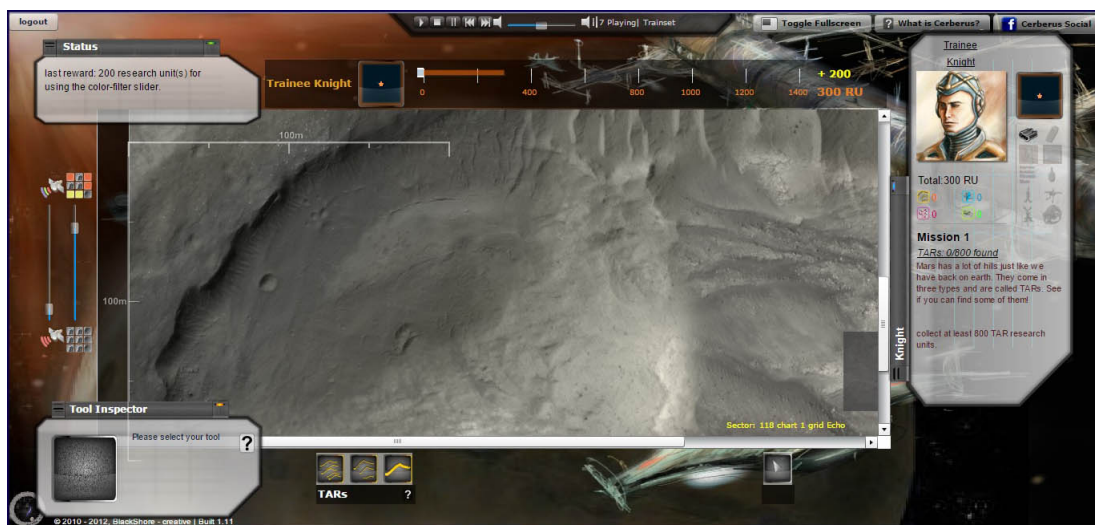


Fig. 2.5 Cerberus Game Interface

Under different game conditions the precision was not differ that much but to produce surface layering, Martian maps exposing aeolian processes, river meanders and other concepts it was sufficient enough. Based on the four possible game conditions they design their game feature. The independent variables were a poor or a rich gaming experience and implicit or explicit knowledge transfer. The dependent variables were player's achievement of precision and the motivation levels and the collective performance for the validity.

The constructed dataset is related to the 18 HiRISE [46] research themes criteria which contained photos described and pre-processed by researchers. To test the player's performance both collectively and individually they compared their annotated photos with Mars features to expert descriptions. Within those 18 themes precisely players were asked to distinguish four important types of feature on the Martian surface.

The first type of feature which covers the study of landforms formed by wind is called Aeolian Processes. Gullies and River Meanders are the second type of feature, places on Mars where there has a possibility to have water in the past and may be caused by water

erosion. Vertically ordered ground which is Layers, the third type of feature often created by sediments and laid down by dust storms, water, volcanic eruptions or crater impacts. When a player identifies anything strange and does not fit into any particular category but still it would be interesting, they defined it as a fourth feature, anomalies. Example of some anomalies are strange shapes, strangely colored mountains or even Mars landers like the Phoenix lander.

Based on the player's annotation of a feature he gains a point and previously annotated total number that other players made also added with that point. When an error occurred players receive low score. Each new annotations then formed the dataset and was transferred into a database.

## 2.9 Micro-Task Based Crowdsourcing

Amazon's Mechanical Turk, One of the earliest and best known crowdsourcing micro-task marketplaces that exploits "human intelligence" of independent freelancers who usually complete small online tasks, also known as 'MTurk'. It provides online flexible work force in the world cloud on requester's or client's demand.

The general work element of Mechanical Turk is called a Human Intelligence Task (HIT). To perform each HITs this web service pay small amounts of money to those online workers often referred to as "Turkers". As requesters and workers derive from all over the world, the payment always vary significantly and usually 0.01 or 0.02 cents per HIT. "In March 2007, Amazon claimed the user base of Mechanical Turk (who commonly refer to themselves as "Turkers") consisted of over 100,000 users from over 100 countries".

Studies that could be applied by Mechanical Turk such as image classification, human linguistic annotation, information retrieval and other data mining research groups. Tasks that Mechanical Turk typically offers are contains keyword searches, labeling, surveys, editing and writing jobs, blog comments, photo captioning, tagging, human powered translation services, data verification etc. <sup>4</sup>.

Kittur [38] proposed an experiment to test the efficacy of Amazon's Mechanical Turk as a user study platform. In their experiment they checked the quality of Wikipedia articles by collecting quantitative user ratings and qualitative feedback. Using the Mechanical Turk they conduct their experiment and asked users to rate 14 Wikipedia articles to compare their rating with the group of expert Wikipedia administrators [39]. They used old versions of the articles from 7/2/2006 for different purpose but quality rated by experts. Their goal was

---

<sup>4</sup><http://www.nytimes.com/2007/03/25/business/yourmoney/25Stream.html>

to improve the user's answers to match with the expert user responses and to reduce the invalid numbers. They designed the questionnaire in such a way that the user feel familiar and comfortable with the content and provide quality ratings.

In their rating procedure before rating the quality of the article, four questions with the verifiable and quantitative answers were required to complete by the users. Users were needed to input the number of references, sections and images of the article and 4-6 keywords to express the article contents summery were also required to provide as an answer. After that they were asked to rate the overall quality of the article as a 7 point scale. Also they were asked to give a feedback insight about their decision.

For 14 articles 124 users delivered 277 ratings which is per article 19-20 ratings. As a result between Mechanical Turk and Wikipedia expert ratings, the positive correlation was higher and better. Among them only 7 responses were meaningless or copy paste. And the median completion time was 1:30 which is also higher.

All the above presented approaches has the similarity with our works however, it also differs from these systems. As human involvement is costly and error prone we can't rely on supervised technique for the entity extraction process of an entity graph. Instead, we use an entity extraction technique that automatically finds more relevant entities in unstructured documents and again extract necessary information to establish entity connections based on co-occurrences in these documents.

The evaluation of information retrieval systems with crowd-sourcing services is a recent line of research that has been getting increasing attention in recent years [63]. According to Vallet [63] for search system evaluation or creation of test collections crowd-sourcing services have been proven to be a valuable resource. Considering user factors still there are no clear protocols to perform a user-centered evaluation of approaches, such as personalization or diversification of results.

## Mining Entities from Events

In this chapter, first we describe entities and events and then the definitions of Named Entity Recognition (NER) system, domain entity and relation classes with some notation that is used rest of the thesis. Then we define the process to extract entities from Wikipedia events and the technique to mine more related seed entities from relevant events based on the named entity recognition technique. We also provide an overview of the extraction algorithm and a profound explanation of relation extraction task with co-occurrences.

### 3.1 Entity Mining and Event Retrieval

In our entity extraction task, we represent event by a set of entity and relation instances where entity instances symbolize the people, organization, location, temporal information and other information related to the specific event. Events that allocate participants and occur for a period of time or some point of time usually called an event. Events usually define as sequences of changes activities or it define among the arguments or their relations involved with the transitional state of changes.

The relation instances signify the links between these entities and their inter relationship. In the extraction procedure, we input some term and our event extractor function automatically generate entity centric index with the name of each term from Wikipedia knowledge corpus.

Let  $E$  be a set of entity classes, i.e.  $E = E_1, E_2, \dots, E_n$ , and  $V$  be a set of event classes,  $Ev = Ev_1, Ev_2, \dots, Ev_m$ , where  $Ev_1, Ev_2$  these are term index pages.

So,

$$E \subseteq Ev$$

And let  $R$  be a relation between two entities, and co-occurrences between two entities is denoted as  $f$ .

Then,

$$R = E_1, E_2, f$$

For the extraction process we used Stanford Core NLP<sup>1</sup> API and Named Entity Recognition (NER) process. The Overview of our Entity Extraction Process is depicted in the following Figure 3.1.

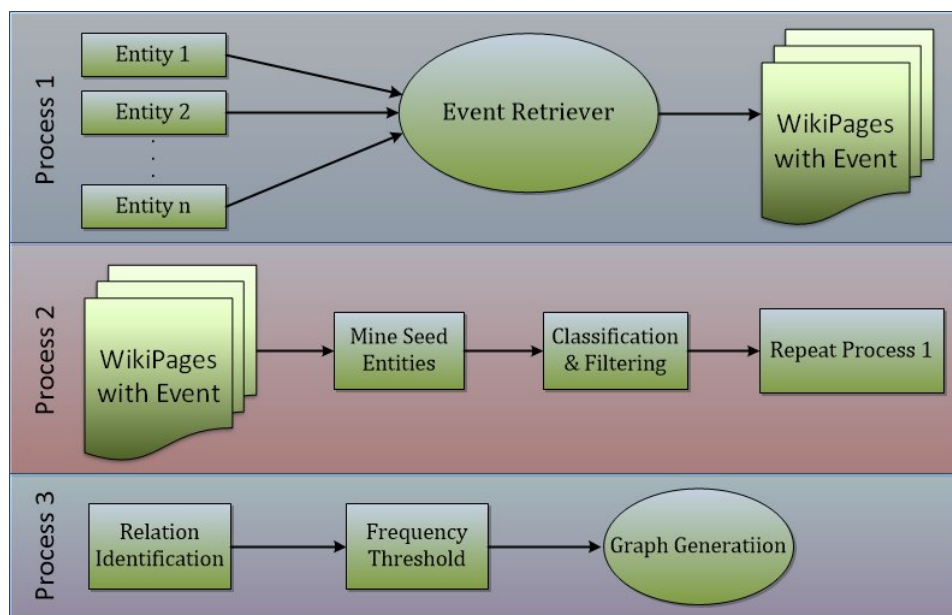


Fig. 3.1 Entity Extraction Process

For each input entity contextualizer class retrieve the corresponding Wikipedia page and extract its content splitted in sentences. It also checks for the duplicate entity. After processing the current entity it extract the text of the Wikipedia page referring to the current entity. Temporal and geographic expressions are extracted from the plain text and entities are extracted from the Wiki-formatted text. It Rank every sentence extracted from the Wikipedia pages. The ranking is computed by considering how many entities, temporal expressions and geographic expressions in the sentence match an entity, temporal expression or geographic expression in the original document. Inside the current text, iterating over the named entities to extract just entities such as people, organizations, and misc.

<sup>1</sup><http://nlp.stanford.edu/software/corenlp.shtml>

In general when we extract to generate some entities from different document text corpus we need to apply several techniques from different research fields like Information Extraction (IE), Natural Language Processing (NLP) and other parsing approaches where specific tasks have been developed according to the process needs. These tools give us the opportunity to combine them with a new way of accessing information. For better understanding the Named Entity Recognition (NER) system is described in section 3.1.1.

### 3.1.1 Natural Language Processing (NLP)

There are several toolkits or APIs are available now-a-days for extraction system that enrich it with linguistic or layout information. We used Stanford NLP<sup>2</sup> toolkit, an extensible java based pipeline that provides core natural language analysis. This open source API is widely used among the research community as well as commercial purposes. Some of the details algorithm procedure is described in this section.

### 3.1.2 Annotator

An Annotator has the uniform interface that adds some analysis information to text by taking it in an Annotation object. This basic architecture is illustrated in Figure 3.2.

The annotators can work with any character encoding but default is UTF-8 encoding and it also supports various human languages processing. Most of the models are trained from annotated corpora using supervised machine learning and others are rule-based.

#### Tokenization

Tokenization is the process that identifies the boundaries of sentences or word in a document and converts a character stream into tokens. The basic form of this whenever a space is found it would split the characters in an input character stream. The other predefined sets of delimiters are like commas, dots, hyphens etc.

#### POS Tagger

In corpus linguistics, part-of-speech tagging or POS tagging is the process of taking a sequence of words as input, and labels each word as corresponding to a specific part of speech, e.g. noun, verb, adjective etc. based on both its definition and its context like, relationship

---

<sup>2</sup><http://nlp.stanford.edu/software/corenlp.shtml>

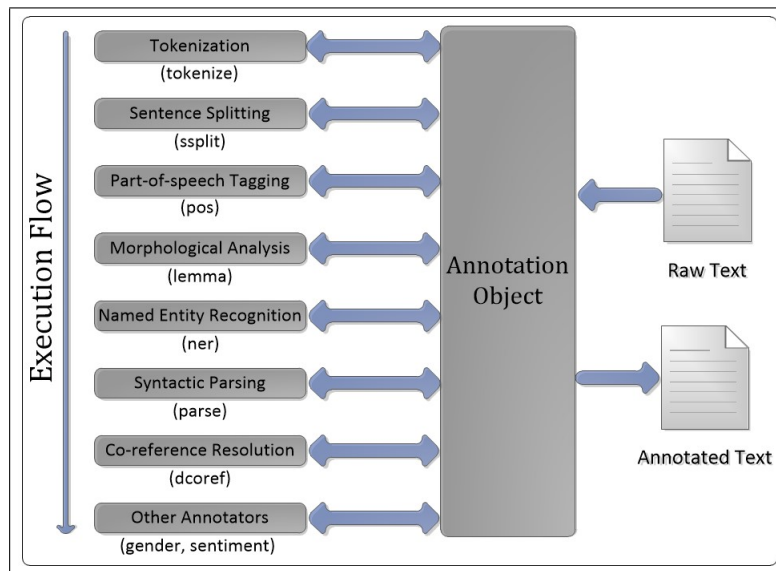


Fig. 3.2 Overall Annotator Architecture [44]

with adjacent and related words in a phrase, sentence, or paragraph. Data-driven taggers for English have been shown to achieve a precision of up to 97% [53, 62].

An example of POS tags attached to a sentence look like below:

*This/DTis/VBZa/DTsample/NNsentence/NNfrom/INthe/DTbook/NNof/INRA/NNP*

### Lemmatization

Another way of translating words into their normalized forms is Lemmatization which is usually based around dictionary lookups. When determining the correct normalized form it also takes the context of the word in account. Lemmatization gives valid words which is its advantage.

### NER Recognizer

Recognizes and classify elements in text into named, like: PERSON, LOCATION, ORGANIZATION, MISC and numerical, like: MONEY, NUMBER, DATE, TIME, DURATION, SET entities. Named entities are recognized using a combination of CRF sequence taggers trained on various corpora, with the default annotators [24], while numerical entities are recognized using rule-based systems [15]. For English State-of-the-art NER systems produce near-human performance. For example, the best system entering MUC-7 scored 93.39% of F-measure while human annotators scored 97.60% and 96.95% [11, 45]. For example of an un-annotated block of text:

*IBM headquarters in Armonk, New York, United States*

And producing an annotated block of text of the names of entities are:

*< ORGANIZATION > IBM < /ORGANIZATION > headquarters in < LOCATION > Armonk < /LOCATION > , < LOCATION > NewYork < /LOCATION > , < LOCATION > UnitedStates < /LOCATION >*

### **Parser**

In natural language processing a significant work is the development of parsers, which main goal is to capture structure and meaning of a single sentence in terms of its constituent phrase types. Various levels of linguistic information, including part-of-speech of each word, the presence of prominent phrases, semantic roles and grammatical structures are captured by the parser and it returns a parse tree or directed graph for an input sentence.

Query Sentence:

"Babylon is the most famous city from ancient Mesopotamia whose ruins lie in modern-day Iraq 59 miles south-west of Baghdad".

The structure and annotations provided by parsers are useful in entity extraction and also for identifying relationships between entities within a single sentence because they provide valuable linkages between verbs and their arguments. An example of a parse tree is provided in Figure 3.3.

### **Regexner**

Building on Java regular expressions regexner implements a simple, rule-based NER over token sequences. It provides a simple framework to allow a user to integrate NE labels that are not annotated in traditional Natural Language corpora. For example, the default regular expressions in the models that identifies nationalities (NATIONALITY), religions (RELIGION), and titles (TITLE) etc.

## **3.2 Automated Extended Seeds Entity Generation**

For the extension of our entity list we generate more seed entity from the Wikipedia Events page, where each entity or term is the index of Wikipedia document corpus. Each Wikipedia term page there has more entities which are related to that entity and other relevant Wikipedia



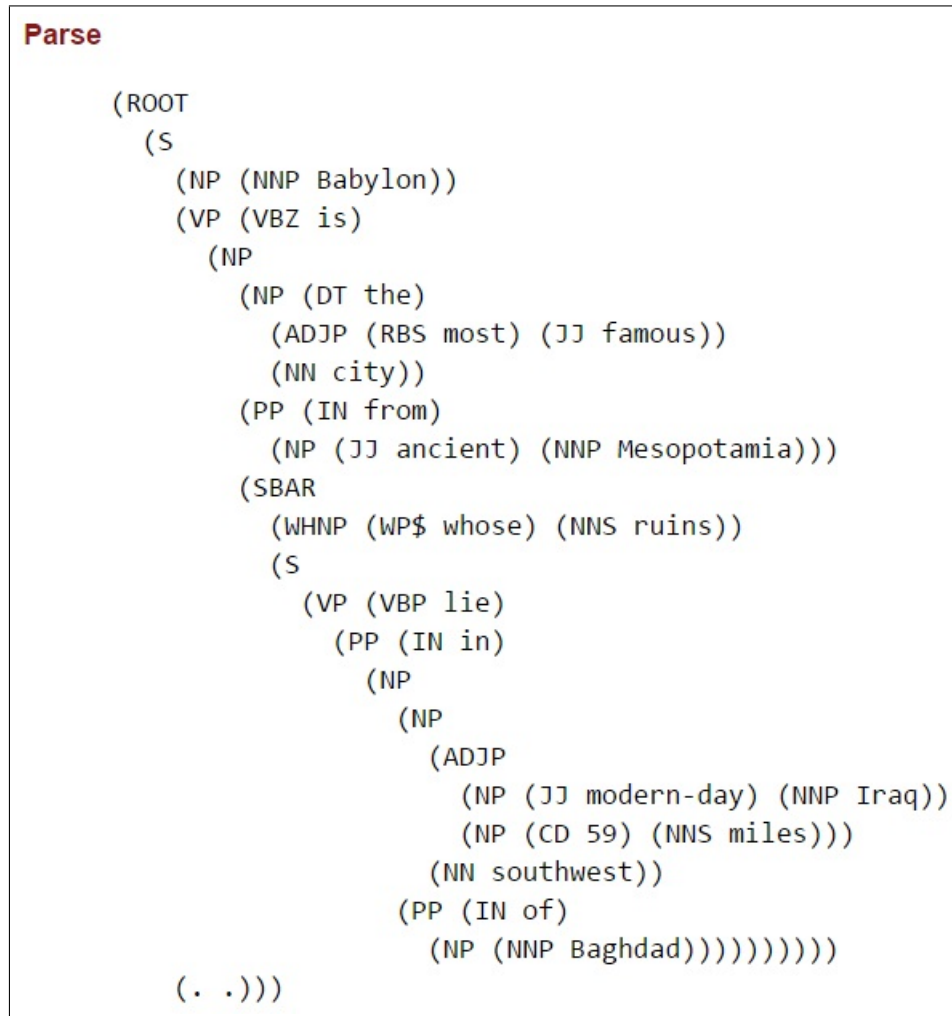


Fig. 3.3 Parse Tree of a Sample Sentence

Event page. These entities are stated in Wikipedia page as hyperlink. In the following picture, Figure 3.4 states the entity George H. W. Bush is connected with several linked (blue color text) entities.

### 3.2.1 Seeds Entity Generation

In our algorithm it captures those linked entities to mine more seed entities. Capturing entities are then refined, structured by eliminating special characters and remove the duplicate entity names. The named entity recognition technique then classify entities according to Organization and Person names. Those newly mined seeds are then put into the input list to extract more relevant events from Wikipedia term page. The set of all entities contained in  $E$  we use  $\xi$ , i.e.  $\xi = \cup_{i=1}^n E_i$  and for all set of relation instances in  $R$  we denote it as  $\mathfrak{R}$ , i.e.



Fig. 3.4 Wikipedia Linked Entities

$\mathfrak{R} = \cup_{i=1}^m R_i$ . Suppose we have a set of seed entity instances  $X$  and a collection of Wikipedia document we denote it  $D$ . So  $X$  is the small subset of  $\xi$ . The seed entity  $X$  is given to identify the relevant document from the Wikipedia. Based on the co-occurrences of entities in the relevant events it extract the events along with entities. The following Algorithm 1 is our approach:

### 3.2.2 Common Event Extraction

The sequence of the extracted data are entity1, entity2, event which is mathematically:

$$E_1, E_2, E_v$$

As for the event extraction we also used to identify common events, our class identify common event first map the entity1 with relevant entity2 and then map with the event with those relevant entities. Figure 3.5 is showing the overview of the event extraction process.

**Algorithm 1** Seeds Entity, Events and Relation Extraction Algorithm**Input:**  $X, D$ 

- 1: **for** each document  $d_j$  in  $D$  **do**
- 2:     Apply  $X$  on  $d_j$  to obtain  $\xi'_j, \mathfrak{R}'_j$
- 3: **end for**  
       //select hyperlink entity,  $X'$  on  $d_j$
- 4: Move  $X'$  from  $D$  to  $X$
- 5: **for** each document  $d_j$  in  $D$  **do**
- 6:     Apply  $X$  on  $d_j$  to obtain  $\xi'_j, \mathfrak{R}'_j$
- 7: **end for**
- 8: **for** each document  $d_i$  in  $D'$  **do**
- 9:     Apply  $X$  on  $d_i$  to obtain  $\xi, Ev, \mathfrak{R}$
- 10: **end for**

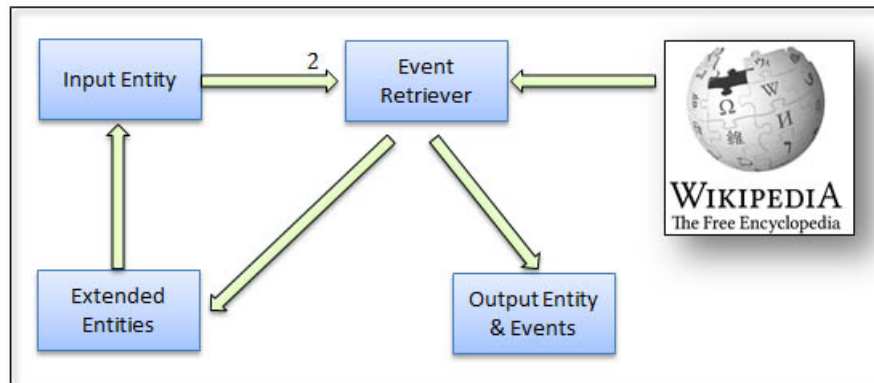
**Output:**  $E_1, E_2, F, Ev$ 

Fig. 3.5 Overview of the Entity Extraction Process

### 3.3 Entity Relationship Identification

In our experiment the relationship is measured by the co-occurrences between two entities. If the co-occurrences value is higher, then the entities are strongly related to each other and if not then it is not so strongly related or connected. We filter this frequency value and a given threshold 2 is accepted in this case. Those entities frequency value is 2 or greater than 2, only those are accepted.

There are two methods for entity extraction: Rule-based extraction and statistical extraction. Rule-based extraction methods are determined by hand coded or learnt from examples. Statistical methods are determined by the based on a decomposition of the unstructured text and labeling. For our entity and relationship extraction as we used Wikipedia events corpus so statistical methods is more appropriate. As we already discussed the tokenization

in section 3.1.2, and to understanding the extraction process and labeling technique in section 3.2.1 we discussed statistical methods along with the model Conditional Random Field (CRF) [41] for given the features of a token sequence, predicting the label sequence. Then each token is allocated to an entity label or an entity sub-part label and with the same entity label other entities are marked as sequential tokens.

For a training set  $S$  of labeled independent graphs in where each graph is a distributed sample that has an internal dependent structure. In a text document corpus if we presume a document as a graph then adjacent terms have strong dependence and the node will be each term and the dependency of terms symbolize the edges of the graph. So the CRF probability model for each graph sample, each node has a label and an observation as well as each edge  $e = v, v' \in E$  represents the mutual dependence of a pair of labels. Then, the conditional probability is  $p(y | x, \lambda)$ , where  $x$  is the observation sequence of all vertices in  $G$ ,  $y$  is the label sequence. The probability of this model will change if the labels of a pair of vertices or adjacent to it have changed. For each graph in CRF a feature function is applied to model the conditional probability.

$$p(y | x, \lambda) = \frac{1}{Z(x)} \exp(\sum_j \lambda_j f_j(y, x))$$

Here,  $Z(x)$  is a normalization factor and  $f_j(y, x)$  is a feature function.

A chain structure is adequate for capturing label dependencies in typical extraction tasks. The labels of tokens or vertices which are adjacent to the label  $y_i$  of the  $i$ -th token are also influence. A scoring function  $\psi(y_{i-1}, y_i, x, i)$  capture the dependency between the labels of adjacent tokens or vertices. In terms of weighted functions the score is defined as follows:

$$\psi(y_{i-1}, y_i, x, i) = e^{\sum_{k=1}^K \lambda_k f_k(y_i, x, i, y_{i-1})} = e^{\lambda \cdot f(y_i, x, i, y_{i-1})}$$

So the conditional distribution of a label sequence  $y$  given a token  $x$  is as follows:

$$p(y | x, \lambda) = \frac{1}{Z(x)} \prod_{i=1}^n \psi(y_{i-1}, y_i, x, i) = \frac{1}{Z(x)} e^{\sum_{i=1}^n \lambda \cdot f(y_i, y_{i-1}, x, i)}$$

These are state features and denoted by  $f(y_i, x, i)$ . The remaining features are transition features which are dependent of the previous label.

### 3.3.1 The Graph Construction

In our search query it takes entity name as input and construct the entity graph according to co-occurrence of the entities. After login to the page Figure 3.6 shows the input window for the entity. And Entity Graph: 'Ayub Khan' is illustrated as below in Figure 3.7.

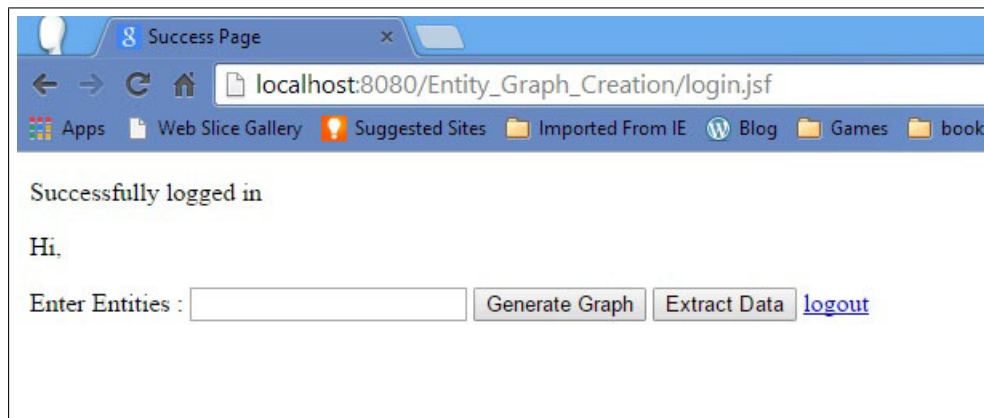


Fig. 3.6 Query Search For the Entity

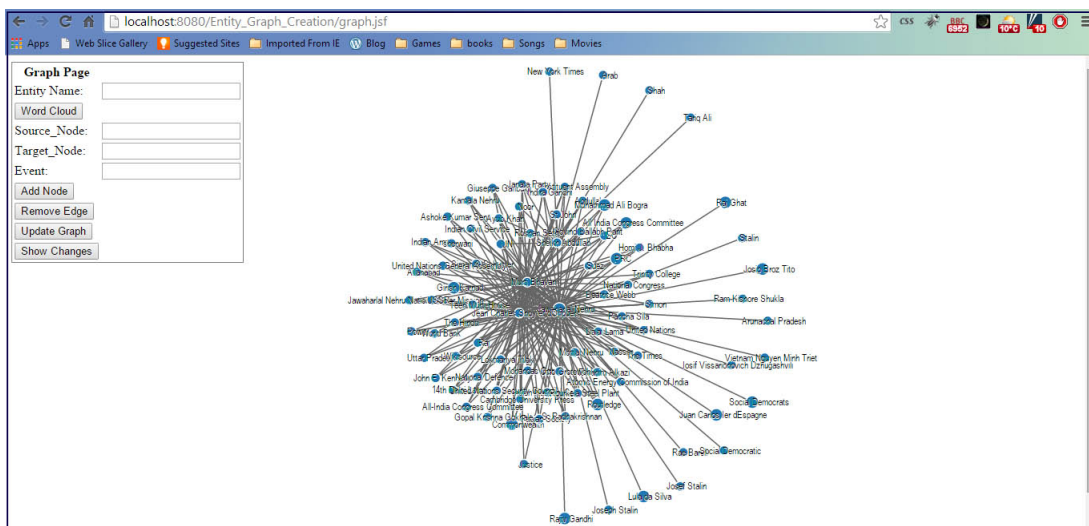


Fig. 3.7 Entity Graph Visualization

### 3.3.2 Tag Cloud Visualization

A tag cloud is a visual depiction of the tags (topics) on a Web site. The tags are usually listed alphabetically and the size or the color defines its relative importance according to their frequency. For the Word Cloud visualization in our model it takes the entity name from the user and shows the relevant Tag Cloud of that entity Figure 3.8.

## 3.4 Crowdsourcing Techniques to Refine the Graph

The rapid growth of crowdsourcing within research and industry has shaped many innovative ideas and accomplished tasks on a global scale with organizing web users. Following

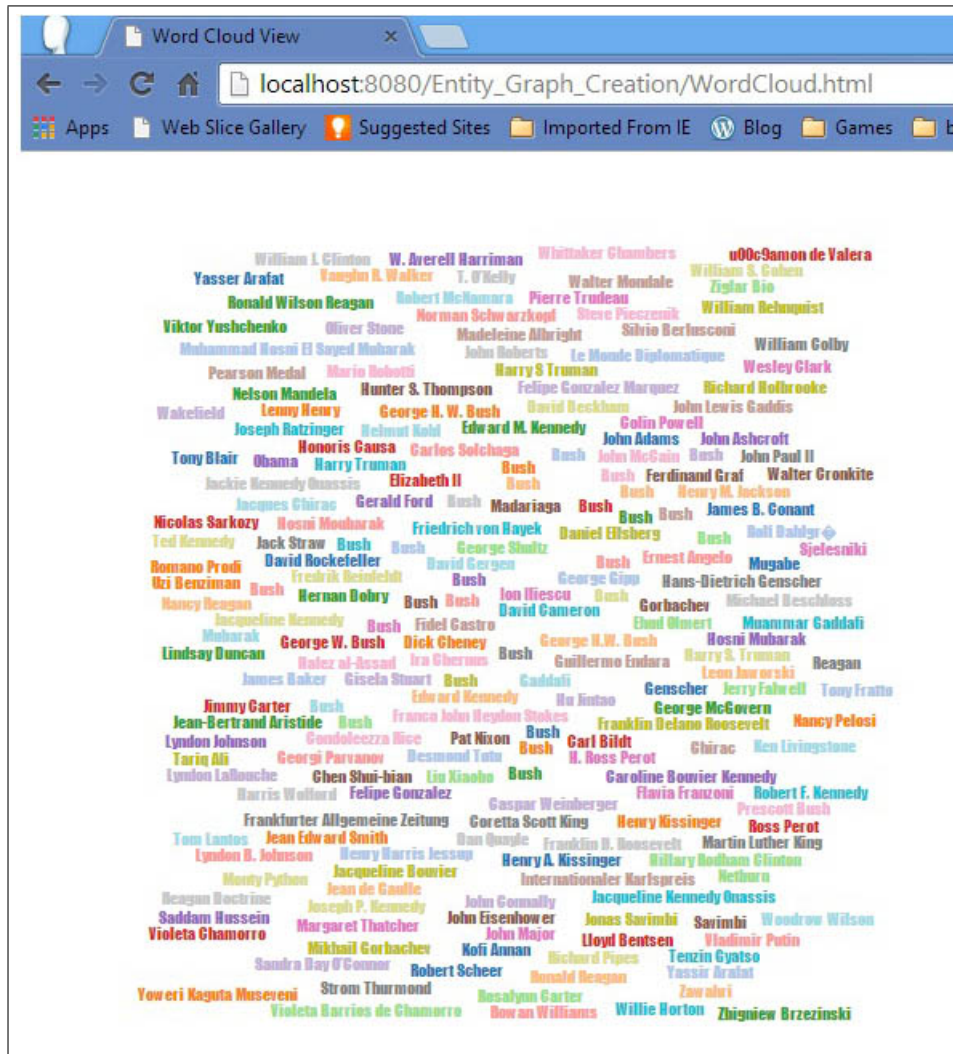


Fig. 3.8 Tag Cloud Visualization

pictures in Figure 3.9 and 3.10 are showing the sequence of data added in the graph using a crowd user.

In our model based on our graphical interface system, users can refine the produced graph with their opinion. User factors, such as personalization or diversification of the produced graph will reflect the newly generated graph. Adding or deleting an edge or the connectivity of the related entities and a particular node or the specific entity along with the related events can easily change by the user opinion. This contribution will then save into our database. And our interface's 'Show Log' button can display the entire changes that occurred in a specific node or edge according to user's login data. This could also help to observe the ongoing changes over the graph.

Based on the crowds opinion refining the associated nodes and the events also reflect

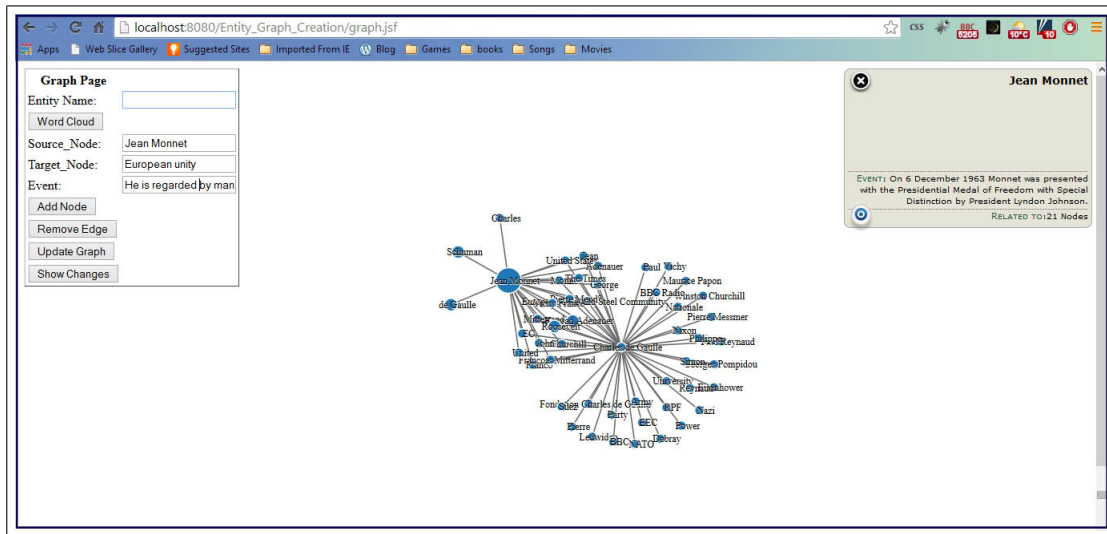


Fig. 3.9 Adding an Entity with Event

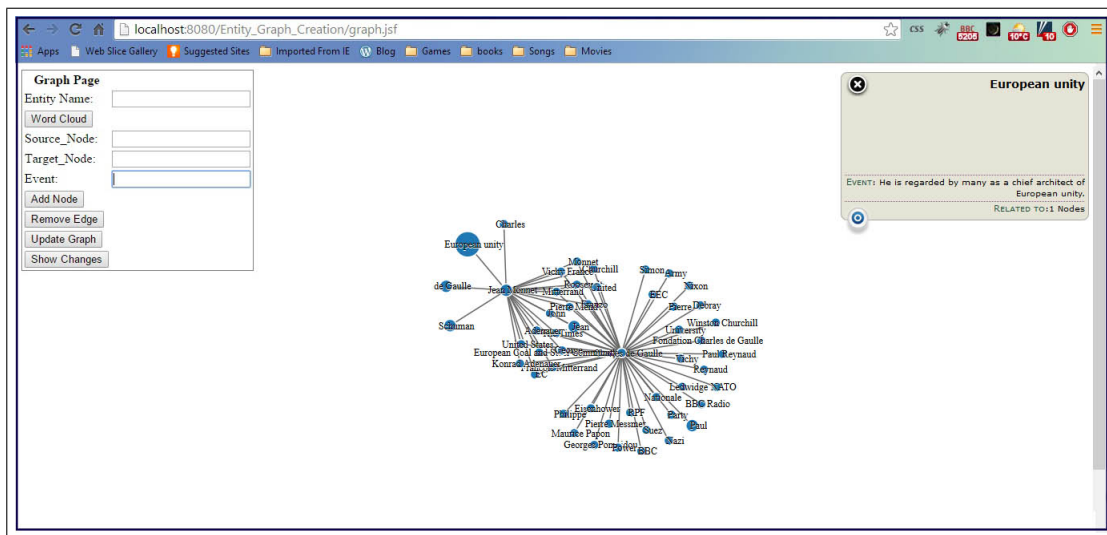


Fig. 3.10 After Adding the Entity

the entire graph. If the crowds opinion comes to this point for a node and the majority of the crowds opinion about the node is deleting or adding for a specific related node with its' events, then the changes also reflect into the main database and database would be saved accordingly. After reviewing the entire changes the system visualized the final refined graph through an 'Update Graph' button. This way the crowds could engaged to refine the historical changes or the database corpus and provide their knowledge to enrich the entire data corpus.

## 3.5 Requirement Analysis

Different types of files contain the output entities. The file types that we used mainly txt and json file format. Users' are providing input entities it extract relevant events and output some text files as term index. After identifying the relationship, it calculates the frequency of the co-occurrences and events and preserve the data as a text in a json file format. Finally it uploaded the json data into the MySQL database server and generates the entity graph. When user modifies an edge or node it also uploaded into the database server.

Figure 3.11 represent the block diagram of the events extraction process using Event Extractor to generate the output in json format.



Fig. 3.11 Block Diagram of the Events Extractor

## 3.6 Design

The overall class diagram design is depicted in the following Figure 3.12. In the main interface class we also used tu-darmstadt<sup>3</sup> API for the parsing Wikipedia Data dump and stanford CoreNLP<sup>4</sup> API for classification.

## 3.7 Pipeline Outline

The entire application pipeline outlined in the Figure 3.13

## 3.8 Associated Sequence Diagram

In this interaction Sequence diagram Figure 3.14 is showing how the extraction processes operate with one another and in which order. It is also showing the object interactions

<sup>3</sup><https://www.ukp.tu-darmstadt.de/software/jwpl/>

<sup>4</sup><http://nlp.stanford.edu/software/corenlp.shtml>



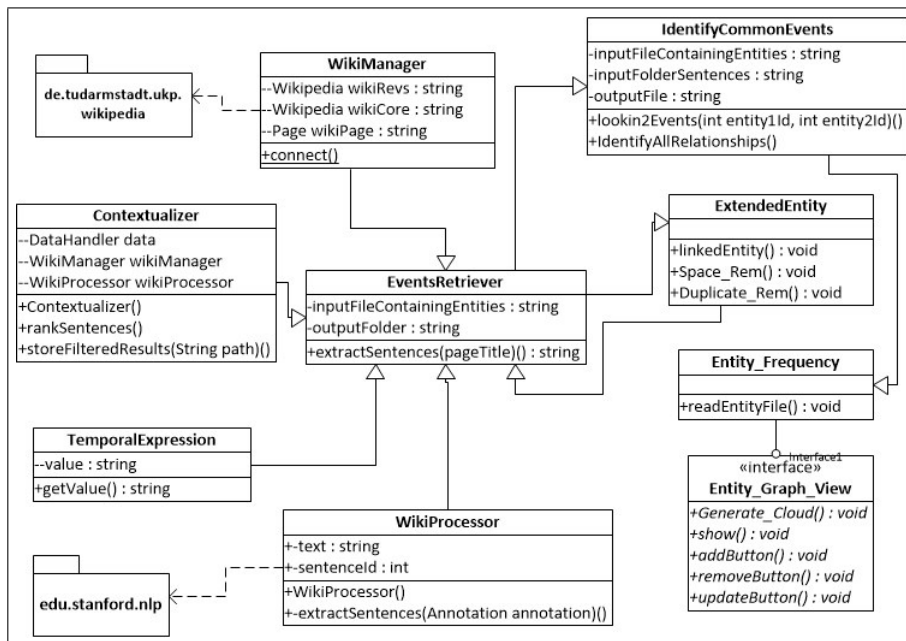


Fig. 3.12 Class Diagram of the Graph Visualization

Purpose	Entity Mining from Events and visualise entity graph in automatic way.
Description	The purpose of the Graph Visualization system is to provide all the related entities relationship in an automated trustable way. It provides a web based interface for the experts to understand the relationship and look a quick view of it. And if necessary then modify the data. The demo implements 4 approaches in order to find out related entities from Wikipedia events. First it extracted entities from Wikipedia events for a given list entities and then fine tunes the entity list and pick relevant linked entities or mine seed entities from those events. Finally it produces all related event list along with the relevant entity. In second approach it collects those entities from events and uploads it into the database. In third approach it visualizes the Entity Graph and Tag Cloud in the web. And in final step it provides the facility for Crowdsourcing Techniques to refine the graph.
Query formulation	The query can be formulated as supported text based entities as input both extraction and visualization for graph and tag cloud.
Data collection	The entire extracted Wikipedia dump data stored into the PC as text format and pre-processed then uploaded it into the server database and every time enriches it.
Result	Depending on the query type, "The Graph Visualization" System visualizes the most relevant relationships between the entities and Tag Cloud.

Fig. 3.13 Pipeline Outline of the Process

which is arranged in time sequence order. It depicts our system's objects and classes and the functionality of the scenario between the objects that needed to carry out. Under the development of the system's logical view of the sequence diagram which is connected with

use case realizations.

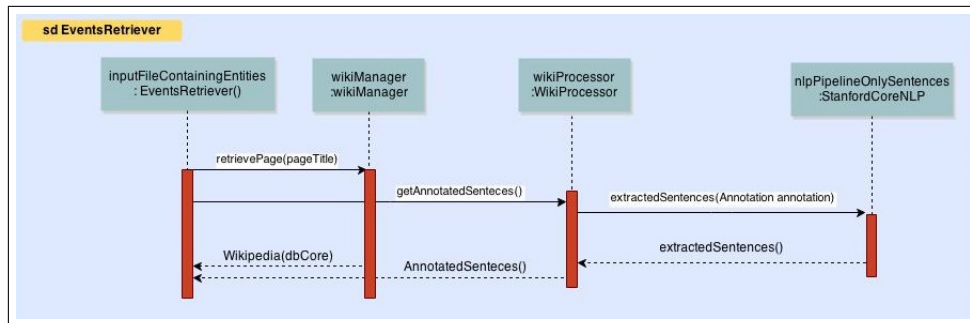


Fig. 3.14 Sequence Diagram of the extraction process

## 3.9 The different Perspectives of Entity Graph

Basically two main perspectives of our application, which we try to elaborate in the following.

### 3.9.1 The Data Analyst and Data Mining

It is an extension of data models used in the database community, in particular graph database models. As we extract data based on Wikipedia events there are lots of insights which experts can use their analysis purpose. An event will show the specific entities with the incident that occurred in that particular time frame. They can also view the connected entities which are closely related to each other. Another approach is word cloud or tag cloud. It represents visual text analysis. The objective of tag clouds is presenting meta-information in a visually appealing way [35]. Day by day it has become a quite familiar technique.

### 3.9.2 Users Point of View

Users can get the visual impression of the entity graph as well as the tag cloud. In the graphical representation users can add or delete an edge or node if they think that the graph needs to be corrected. And tag cloud allows users to get an overview of a specific node connection.

## 3.10 The Prototype System

### 3.10.1 User Login

Users can login to the application by registering it. After successfully login users are redirected to the query page where they need to put the desired name of the entity and pressing the button Generate Graph. This will open a new window with the entity graph. Figure 3.15 is the outlook of the login page.



Fig. 3.15 User Login Page

### 3.10.2 Description of the Main Page

After successfully login, users can use this application. There are following contents display on the page:

- Input entities: Users can input their desired entity in the input text box to view the specific query graph.
- Add entity: Users can add Edge or node by entering the source, target and event mentioning in the field.
- Delete Entity: Users can also delete their desired node or edge mentioning it in the text box by pressing the delete edge button.
- View Changed Entity Graph: Finally to view the modified graph users need to press the update graph button, which will display the new modified entity graph.
- Crow Sourcing: For the crowds or users, by clicking a node will show the events list related to that specific node or entity in the right side small panel view.

- View Tag Cloud: To view the Tag Cloud or Word Cloud, users can input the desired entity name into the cloud textbox and by pressing the Generate Cloud button will display the Tag Cloud.

## Evaluation and Results

In this chapter we describe the system we implemented for extracting entities from Wikipedia Events and how we used the producing graph in a modified way along with the Tag Cloud. This chapter organizes in the following way: First we provide a detailed description of the experiment setup, including the using technologies and structure of datasets, information extraction procedure, graph generation and crowdsourcing refinement technique and tag cloud visualization. The process followed by the evaluation, experimental results and discussion.

### 4.1 Using Technologies

This system is a web application. The following technologies are used to develop the application-

- Java [2]
- Web Server: Apache Tomcat 7.0 [3]
- Database: MySQL
- J2EE [1]
- JSF framework
- XML, JavaScript, XHTML
- IDE: Eclipse
- Browser: Mozilla Firefox
- API: D3.js, Stanford CoreNLP

It is a web based small module implemented in Java. We have developed it with JavaServer Faces (JSF) component-based user interface framework. JsF is at the middle and front layer. We have used MySQL server as a back-end layer to store all the entities and related event information. For the graph and tag cloud implementation we have used d3.js, JScript and XHTML.

For the entity and event extraction, 'EventsRetriver' is the main class which mine events from the Wikipedia and identifies the entities. The 'ExtendedEntity' class mine more entities and add it into the input list and IdentifyCommonEvents class finds out the common events for those entities. For the graphical visualization 'EntityGraphView' is the main bean class.

## 4.2 Database Structure

Entity Graph visualization application used a database to store all the required nodes, links and events meta-data. And for the changing information we used log table. We used MySQL database. For predefined entity list we input 468 entities and crawl the data. After extending the extraction process we mined more than 4000 new entity list as input. And extract more than 4000 document from the Wikipedia data corpus. More than 102025 events are generated from our extracted data with the classification entity: people's name and organization. The table description and the ERD diagram 4.1 are depicted in the following.

Here are the parameters that were used in conjunction with the Wikipedia Article page Search API in the gathering of this dataset:

- Date, time, year of a specific event history.
- Co-occurred entities within the same event define the relational status.

### 4.2.1 Table Description

- user: All the registered users information like users name, id and password are stored in this table for future references.
- logs: All the modified informations are stored in this table. When user changes a nodes or edges, it automatically updated into the logs table.
- nodes: All the entity id, name and group are stored in this table.

- links: All the link or edge relation are stored in this table, such as source, target and frequency of the entities.
- events: All the related events are stored in this table along with the corresponding nodes.

### 4.2.2 ERD Diagram

The data model of the output is look like Figure 4.1.

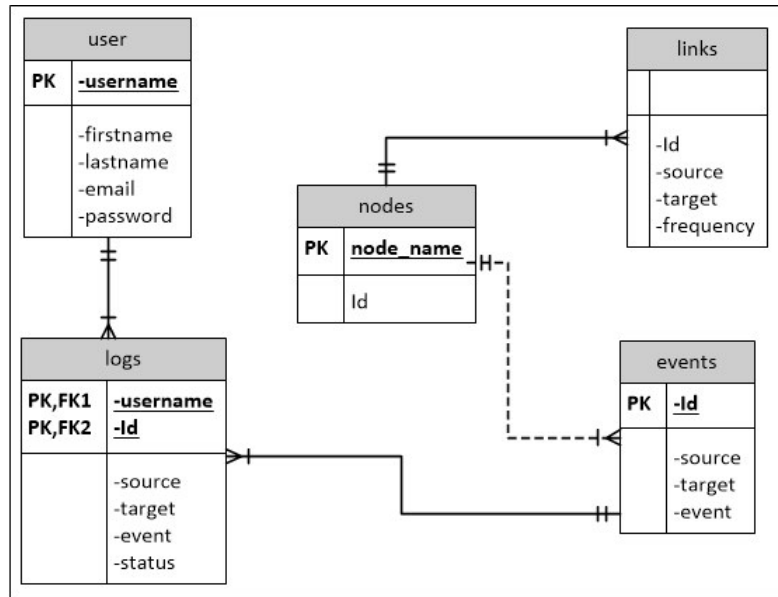


Fig. 4.1 Data Model of the Events Extractor

## 4.3 Entity Extraction Components

In the process algorithm our experiment requires extraction of entities applied by some information Extraction system components. The information extraction system can perform extraction of entity and relation instances as well as events from the text document. In the Figure 4.2 the whole event extraction process algorithm is illustrated. In our work we choose Stanford CoreNLP<sup>1</sup> components. It is an open source API which is a well known named entity extraction system and classification technique, free to use for research purpose. We classify our entities mostly used classification label as organization and people's name based.

<sup>1</sup><http://nlp.stanford.edu/software/corenlp.shtml>

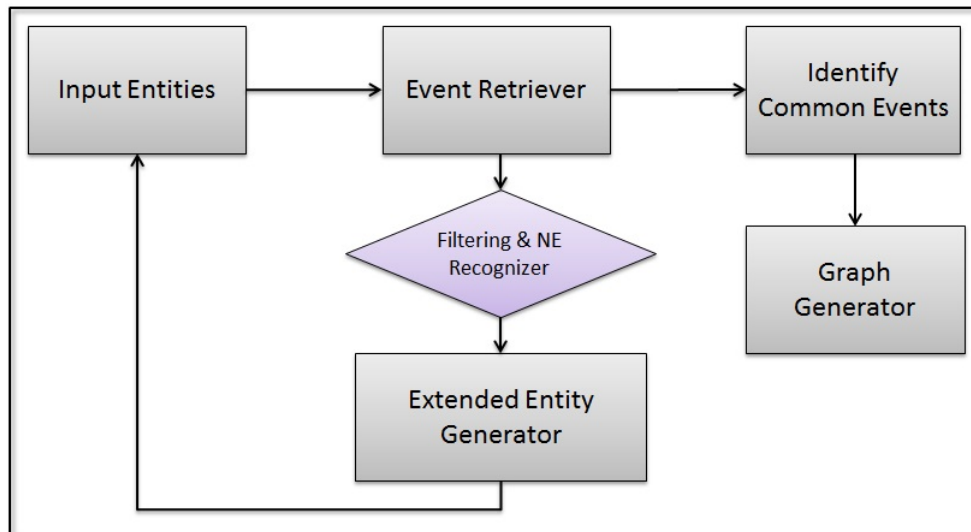


Fig. 4.2 Event Extraction Process Algorithm

The objective of the extraction system components is to define the specific entities and their inter-relationships along with name entity recognition as well as the related events determination and graphical representation.

## 4.4 Category Based on NER

For the mining more seed entities we used classification technique to extract more entity as seeds. In our experiment we only consider person and organization instances. As we consider the Wikipedia hyperlink to extract entity seeds that also produce lots of noisy entities and unnecessary text. To filter this we go through in a process with the classification technique which successfully identified 95% of the named entities. Our process only take Wikipedia hyperlink entities from the extracted data corpus. Sample entity hyperlink is referred in the Appendix A.1. To remove duplicate entities and unwanted symbols and words we filter our entity list. And also we categorized our entities to get extended entity list. Sample figure is given in Appendix B.1.

## 4.5 Evaluation of the Graph

An event usually defined as an incident or a historical circumstances or a thing that happened in a specific time or in a place. So there has a relationship between the entities and the events. Measuring the co-occurrences between those entities which we called frequency that will



give us the strength or weaknesses of that entity. As an example of labeling nodes and events are given in the following Table 4.1.

<b>Entity1</b>	<b>Entity2</b>	<b>Label</b>
Bill Clinton	Aristide	0
Charles de Gaulle	EEC	2
Hillary Rodham Clinton	Eleanor Roosevelt	2
Charles de Gaulle	Churchill	1
Bill Clinton	Jimmy Carter	2
European Community	Robert Schuman	2
Bill Clinton	David	1
Charles de Gaulle	Franco	0
Hillary Rodham Clinton	Flavia Franzoni	0
Euratom	European Community	2

Table 4.1 Labeling Status of Entities

Here, 0 means it labelled incorrectly, 1 means it labeled partially-correct and 2 means it labeled correctly. The link between the each node measured by frequencies and the entity graph generated from it. Here, node means each entity and link between the nodes are edges.

For the entities, events and edges of the entity graph, we evaluate our results measuring the precision. For each entity graph there has some nodes or entities. The size of the entities (correctly labeled or relevant and incorrectly labeled or irrelevant) are denoted as  $N$  and the number of entities have been correctly labeled denoted as  $F_c$  and partially correct as  $F_{pc}$ .

So,

$$\text{Precision, } P = \frac{F_c + F_{pc}}{N}.$$

For the simplification of the process in our calculation we consider the correctly labeling and partially correct labeling together as an addition ( $F_c + F_{pc}$ ). As the actual number of the entities and edges are too many, so in this case measuring the recall is not feasible.

To evaluate our automated system as we extracted small subset of entities from Wikipedia entries to calculate the precision. These subsets were randomly chosen. Then we have

manually labeled and checked the events and the entities along with the frequency co-occurrences.

### 4.5.1 Results of the Graph

We also noticed that higher frequency gives us better performance and higher precision for the entity graph and with higher frequency the irrelevant labeling number decrease significantly but the relevant entity numbers fall down. Table 4.2 shows the relevant and irrelevant entity labeling numbers along with it's frequency for the entity graph of '**Charles de Gaulle**'. The table is evidently showing that at level 5, the relevant entity number is 14, whereas the irrelevant entity number is 0, so the precision is 100% whereas normally the precision is 93.33%. This performance measure focus on how much relevant entities the events contain in a specific graph.

<b>F. Range</b>	<b>Relevant</b>	<b>Partial Relevant</b>	<b>Irrelevant</b>	<b>Precision</b>
1	93	5	7	93.33
2	57	5	4	93.93
3	37	4	2	95.35
4	22	4	1	96.3
5	14	4	0	100.0

Table 4.2 Relevant and Irrelevant Entity Labeling

We have manually checked 12 entity graphs based on the top 100 nodes for each graph, approximate  $12 * 100 = 1200$  entities and more than 1200 events and labeling them to calculate the average precision of entities in our Entity Graph System. The following Figure 4.3 is showing the average precision of nodes for the 12 entity graphs. As in our system the threshold value is frequency 2, so here the precision is also calculated based on level 2. From our graph we have observed that the average precision of node is approximate 91.87% for 12 entity graphs.

We have also manually checked the edges of 12 entity graphs and calculate the average precision of edges. We have observed that the average precision of edge is approximately 94.35% which is depicted in the following Figure 4.4.

To better understanding the evaluation process of our entity graph calculation and labeling we have choose the entity graph of '**Jimmy Carter**' as an example.

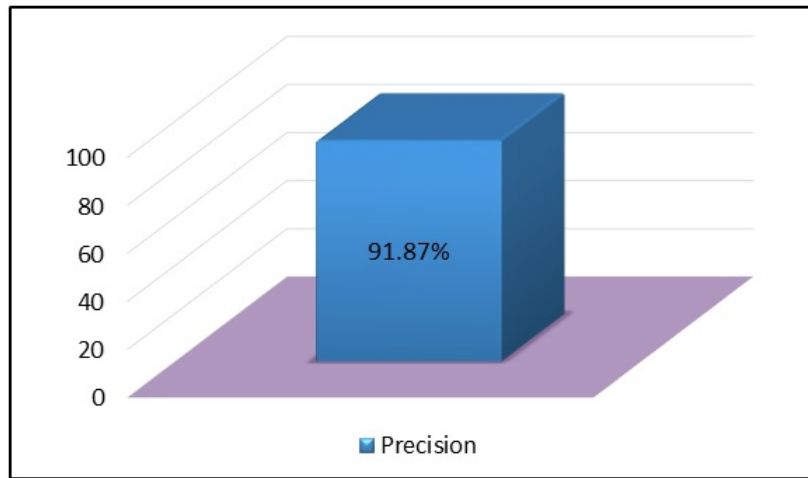


Fig. 4.3 Average Precision of Entities

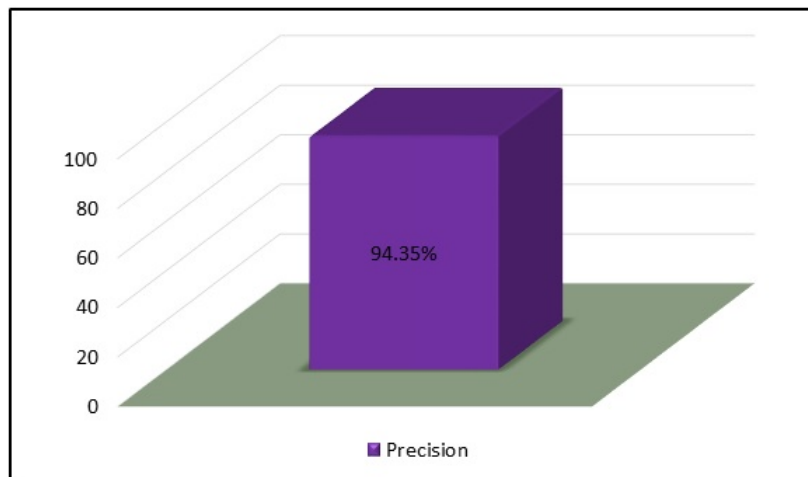


Fig. 4.4 Average Precision of Edges

In the following Table 4.3 is showing the entity labeling of the graph Jimmy Carter.

<b>F. Range</b>	<b>Relevant</b>	<b>Partial Relevant</b>	<b>Irrelevant</b>	<b>Precision</b>
1	78	13	10	90.1
2	36	10	4	92.0
3	15	6	2	91.3

Table 4.3 Entity Labeling of the Graph Jimmy Carter

In the table we have observed that total irrelevant entities are 10 in number. The reasons

those entities are labeled irrelevant described in the following:

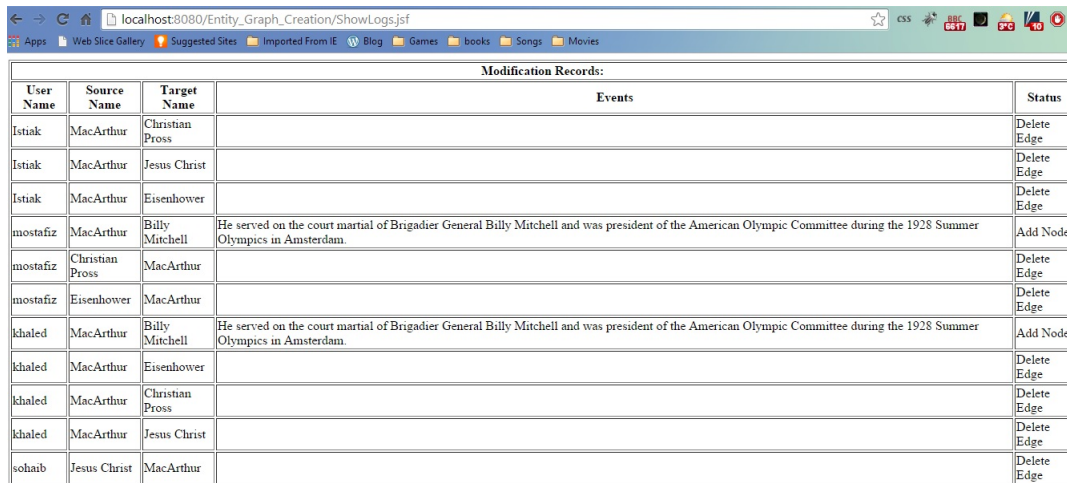
- Alex: In the classification process Alex is identified as a person name. But it's not the full name of the person, only the half part of the name. So the name could be Alex Baldwin, Alexander etc. anything starting with Alex. In this case it is Alexej. As Alexej is not even name of a person but it is mislabeled by our application software because the portion of the word contains Alex.
- Arab: Arab is not a person name or an organization. In this case Arabic is also mislabeled because of the first portion of the word contains Arab.
- Arthur: Arthur is one part of the full person's name, such as: Arthur Milnes, Arthur Schlesinger etc. In our case the application identified only Arthur and mislabeled it. In the events it contains various Arthur. So the name of the person is not specific which Arthur.
- EC: The abbreviation of the organization name EC are European Community, European Commission, Electoral Council etc. In here EC is the short form and in Wikipedia it does not contain any single page. So our application software could not find any Wikipedia page to extract. In the containing event its' finds out Electoral Council which is also mislabeled.
- Giulio Andreotti: There are 2 events identified as related with Jimmy Carter and Giulio Andreotti. But because of the picture in the page some sentences are not consistent with the entities. In this case Giulio Andreotti is directly connected with Michele Sindona but not with Jimmy Carter. So it is also mislabeled.
- Hernan Dobry: This is a name of the reporter and does not have any direct connection with Jimmy Carter.
- Jean de Gaulle: This name is also not directly connected with Jimmy Carter but is mentioned in the book Higgins, M. (2004). So it is inconsistent.
- Merry Hermanus: This name is in the French Wikipedia but most probably the event data is old. So it is inconsistent.
- Michail Gorbachev: The name is not consistent and most probably old data.
- Uzi Benziman: This is also the name of the article writer and does not have any direct connection with Jimmy Carter. So it is mislabeled in this case.

For the calculation materials of manually labeling and data of 12 entity graphs are available in the Appendix B.1

## 4.6 Evaluation of Crowdsourcing Process

In order to obtain the real scenario we have conducted two experiment using 10 serious crowd users to refine our graph. As a crowd platform both 'Amazon Mechanical Turk' and 'CrowdFlower' are suitable. Each user registered in the web application and login to the site. For the test set we have used one particular entity '**MacArthur**' as an query entity sub-graph. For user's given query the program generate the graph. Each users are then requested to modify the entities and events according to their knowledge, if the crowd user thinks that specific entities are wrongly connected with each other or may be the described event is wrong.

The experiment was conducted carefully and the instructions were clear to the crowd. For their better understanding and knowledge they are also allowed to use the search engine for improved results. All the crowd user's modifications are then stored into the log table. In the Figure 4.5 is showing the described log table.



Modification Records:				
User Name	Source Name	Target Name	Events	Status
Istiaik	MacArthur	Christian Pross		Delete Edge
Istiaik	MacArthur	Jesus Christ		Delete Edge
Istiaik	MacArthur	Eisenhower		Delete Edge
mostafiz	MacArthur	Billy Mitchell	He served on the court martial of Brigadier General Billy Mitchell and was president of the American Olympic Committee during the 1928 Summer Olympics in Amsterdam.	Add Node
mostafiz	Christian Pross	MacArthur		Delete Edge
mostafiz	Eisenhower	MacArthur		Delete Edge
khaleed	MacArthur	Billy Mitchell	He served on the court martial of Brigadier General Billy Mitchell and was president of the American Olympic Committee during the 1928 Summer Olympics in Amsterdam.	Add Node
khaleed	MacArthur	Eisenhower		Delete Edge
khaleed	MacArthur	Christian Pross		Delete Edge
khaleed	MacArthur	Jesus Christ		Delete Edge
sohaib	Jesus Christ	MacArthur		Delete Edge

Fig. 4.5 Modification of Log Table

Based on the modification if specific node or entity and the related events addition or deletion is higher than a given threshold, the system change the main database corpus and also the main graph visualisation is affected according to the changes. In our experiment for addition and deletion the threshold is greater than 7, which is more than half of the crowds (two third). At this point we have given the priority of the majority of the crowd's opinion.

We review the crowd’s opinion and find out that among the 10 users, 7 of them are deleted the node 'Christian Pross', 6 of them are deleted the node 'Jesus Christ', 8 of them are deleted the node 'Eisenhower', and 4 of them added a node 'Billy Mitchell'. We also asked about their opinion for deletion and addition of a node. Most of them think the 'Dwight D. Eisenhower' and 'Eisenhower' are the same and the events are also the same but 'Dwight D. Eisenhower' is the full name of 'Eisenhower', so they think that 'Eisenhower' should be deleted. Thus a new reformed graph is generated from the new improved data. The newly generated graph is illustrated in Figure 4.6.

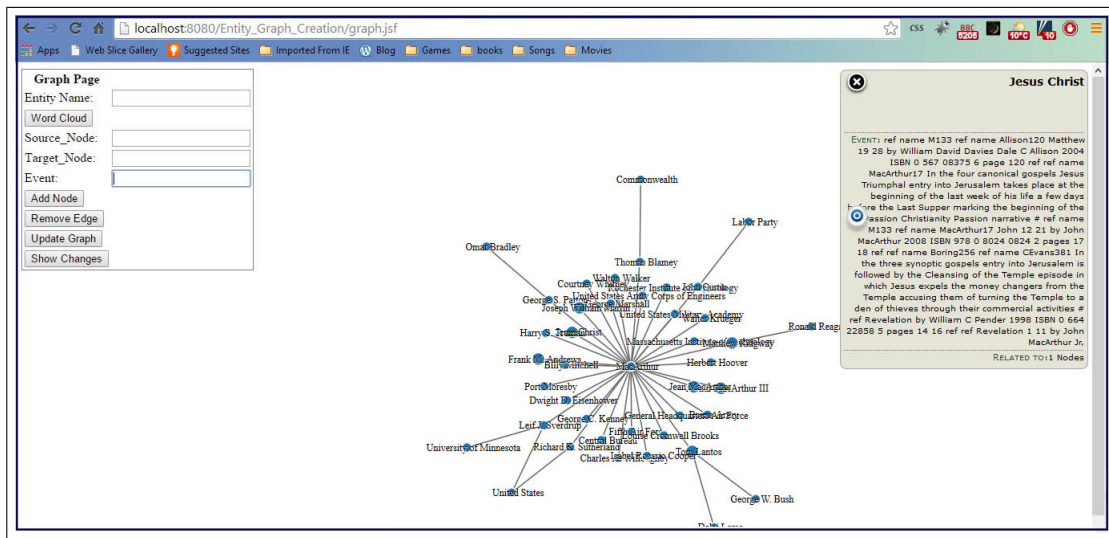


Fig. 4.6 New Reformed Entity Graph

### 4.6.1 Results of the Crowdsourcing

To evaluate the results of our graph 'MacArthur' we manually labeled and checked the nodes and associated events and calculated the precision. Following Table 4.4 is the result of manually labeled entity graph 'MacArthur'.

Relevant	Partial Relevant	Irrelevant
36	3	2

Table 4.4 Relevant and Irrelevant Entity Labeling of MacArthur

In this case the precision is 95% before the crowdsourcing. In the new graph we have observed that the node 'Christian Pross' are completely deleted based on the crowd opinion

which were wrongly connected before but 'Jesus Christ' is still there. Therefore, the new precision is 97.44%. Following Figure 4.7 is showing the calculated precision of the graph before and after applying the crowdsourcing technique respectively.

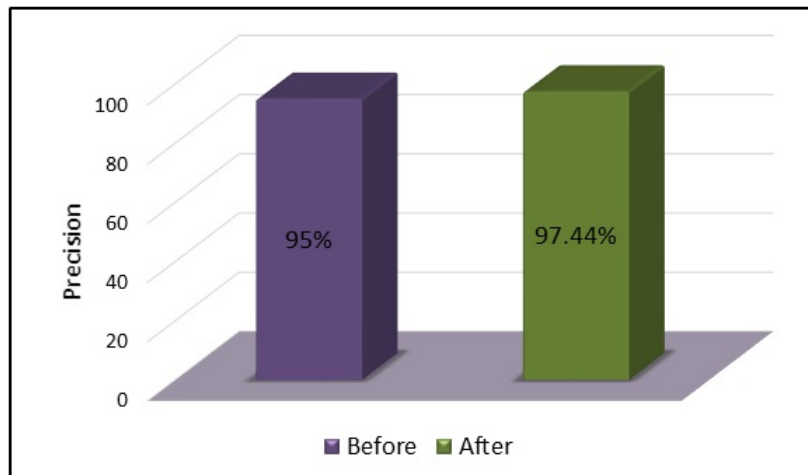


Fig. 4.7 Precision before and after Crowdsourcing

The depicted picture definitely stated that after applying the crowdsourcing technique the entity graph's precision has been improved significantly.

## 4.7 Discussion

The success of our system depends on the identification of all the entities properly. We observed some of the entities has synonym, popular entity list has different variations of entities with variations of spelling. Specially the peoples name has a large number of variations, even if it is a small difference of the name of a person our system defined it as a single entity for each of them. For example, Bush has a different name and title version like George W. Bush, George Walker Bush, whereas we need to minimize those variations for the better performance. Hence, our system has a limitation which can be improved in future.

To identify all the unique names and convert the other different names as a single unique entity is really very difficult. We also observed that if we ignore these various names then it would also affect the performance and some of the important events may not be mined or overlooked and the relationship between the events would also be effected.

To calculate the precision and recall our observation is that under different frequency level it changes and reflect the calculation of precision and recall percentage. In Figure 4.8 is displaying the average precision based on the frequency ranges for 12 entity graph. Most

of the entities are recognized in the level of frequency 1 but it also contains lot of noisy entities. From the given figure with three different precision we can also predict that level 3 frequency measure has the highest precision 92.13%.

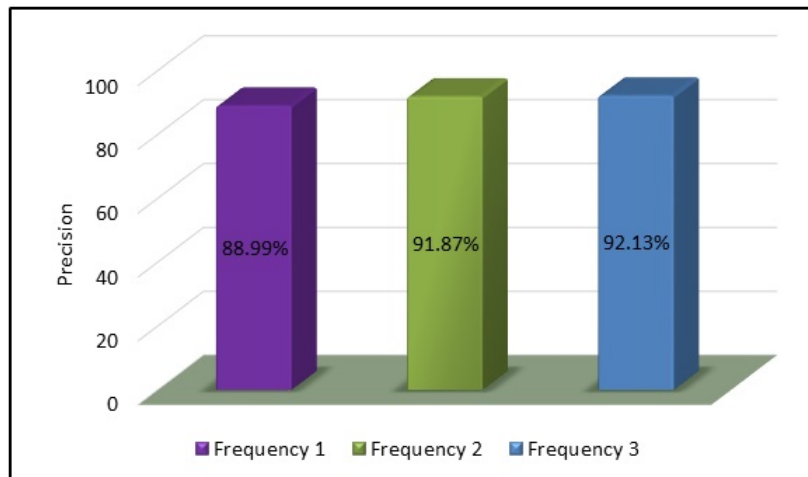


Fig. 4.8 Precision Based On Frequencies

In some cases, we observed that when we consider the irrelevant entities it may possible that some graphs has irrelevant entities on higher frequency. When we are measuring the precision for the frequency level 1 or 2 we have more entities and considering all the irrelevant numbers but in the frequency level 3 or higher we have less entities but the counting of irrelevant entities would not change in this case which would affect the precision and as a result, in higher frequency we would get less precision.

For the crowdsourcing we have used small sub-graph for the experiment to observe the human behaviour about the crowdsourcing process. From our observation under different databases it would act differently. Also the precision would affected accordingly or changed. The success of the crowdsourcing process sometimes depends on users' prize or achievement gain. For a bigger entity graph we may not be able to achieve the 100% precision.



## Conclusion and Future Work

### 5.1 Conclusion

In this thesis we have been focusing to automatically mining entities from Wikipedia events and based on the co-occurrences to generate an entity graph. Our objective is to refine the generated graph by the power of the crowd. This refinement also change the visualization of the graph and depicted the fine grained entity graph.

To represent the gist of the entire history, relation between people, organization, company, events or even an important analysis can easily performed through the entity graph. The evaluation shows that Wikipedia is a better suited data corpus for entity mining and relation extraction tasks.

In addition we extracted entities in a two way form that ensure more refined data and more entities to predict the relationship between entities as well as relevant events. This two way extraction procedure ensures the extended relationship and the co-occurrences of all entities and events.

We have also designed an entity graph visualization technique incorporated with the frequency based on the co-occurrences for this task. The objective of this visualization task is to show the relevant events that generate from the Wikipedia and related to those entities. The impact of this events depiction has a highly historical appeal that could prove interesting influential phenomena.

We have defined the precision and recall of our entity graph to measuring the performance and its accuracy. We also constructed the log data for the changes that user made for improvement and stored the reformed data corpus. Crowdsourcing refinement process also impact the whole graphical view for the entities and events of our fine grain refined graph.

A working prototype system was developed to support the proposed work of mining

entities from events and relation extraction task along with the entity graph visualization, display events, tag cloud creation and crowdsourcing entity graph refinement technique.

Although the proposed systematic strategy works fine and have shown a satisfied result outcome, still there are some limitations in our system. One of them is entity synonym disambiguation where as we consider every entity synonym as a separate entity in the list. Still there has a plenty of space left to improve the entire system for future work.

## 5.2 Future Work

Some of the important future works that would open the different possibilities in the research direction are mentioned here:

There has a plenty of work needs to be done to improve the performance of the Information Extraction Process System and investigate the impact on different IE system components and analyse it. The performance of our system highly depends on the performance of the information extraction process. The main extraction process takes a time to process all the necessary information and the related common entity events. As we adapted several entity extraction process and parsing techniques, instead some of the instances were still unpredictable and unable to be extracted. So a better extraction process component will definitely improve its performance and generate more accurate entities from events and define the relationship.

Another improvement is possible with the synonym problem. We observed that same entity has different synonyms and relations. In our experiment we consider all of them but it is possible to identify those synonym as patterns and define as a unique synonym for each entity and its relationship. Therefore, it requires further refinement technique for the synonym problem.

We provide entities as a text file input. For the flexibility and advancement it is also possible to design the system in such a way that it would take the input entities inside the system and users have more flexibility and control.

It is possible that this work could be extended in another direction except Wikipedia. In the climatology related terms or even environmental changes and historical science invention this system would perform brightly in the context of visualization and refinement direction.

## References

- [1] Java platform, enterprise edition (java ee), version: 7. <http://www.oracle.com/technetwork/java/javaee/overview/index.html>.
- [2] Java, version: 7. <http://www.java.com/en/>.
- [3] (2014-11-07). Apache tomcat, version: Tomcat 8.0.14 Released. <http://tomcat.apache.org/>.
- [4] Agichtein, E. and Gravano, L. (2003). Querying text databases for efficient information extraction. In *Proceedings of the 19th International Conference on Data Engineering, March 5-8, 2003, Bangalore, India*, pages 113–124.
- [5] Ahn, L. v. (2006). Games with a purpose. *Computer*, 39(6):92–94.
- [6] Amaral, A. D. (2013). Rule-based named entity extraction for ontology population. In *Recent Advances in Natural Language Processing, RANLP 2013, 9-11 September, 2013, Hissar, Bulgaria*, pages 58–62.
- [7] Appelt, D. E., Hobbs, J. R., Bear, J., Israel, D. J., and Tyson, M. (1993). FASTUS: A finite-state processor for information extraction from real-world text. In *Proceedings of the 13th International Joint Conference on Artificial Intelligence. Chambéry, France, August 28 - September 3, 1993*, pages 1172–1178.
- [8] Baluja, S., Seth, R., Sivakumar, D., Jing, Y., Yagnik, J., Kumar, S., Ravichandran, D., and Aly, M. (2008). Video suggestion and discovery for youtube: taking random walks through the view graph. In *Proceedings of the 17th International Conference on World Wide Web, WWW 2008, Beijing, China, April 21-25, 2008*, pages 895–904.
- [9] Banko, M., Cafarella, M. J., Soderland, S., Broadhead, M., and Etzioni, O. (2007). Open information extraction from the web. In *IJCAI 2007, Proceedings of the 20th International Joint Conference on Artificial Intelligence, Hyderabad, India, January 6-12, 2007*, pages 2670–2676.

- [10] Bing, L., Lam, W., and Wong, T. (2013). Wikipedia entity expansion and attribute extraction from the web using semi-supervised learning. In *Sixth ACM International Conference on Web Search and Data Mining, WSDM 2013, Rome, Italy, February 4-8, 2013*, pages 567–576.
- [11] Borthwick, A., Sterling, J., Agichtein, E., and Grishman, R. (1998). Nyu: Description of the mene named entity system as used in muc-7. In *In Proceedings of the Seventh Message Understanding Conference (MUC-7)*.
- [12] Brabham, D. C. (2010). Moving the crowd at threadless. *Information, Communication and amp; Society*, 13(8):1122–1145.
- [13] Califf, M. E. and Mooney, R. J. (2003). Bottom-up relational learning of pattern matching rules for information extraction. *Journal of Machine Learning Research*, 4:177–210.
- [14] Chakrabarti, S. (2007). Dynamic personalized pagerank in entity-relation graphs. In *Proceedings of the 16th International Conference on World Wide Web, WWW '07*, pages 571–580, New York, NY, USA. ACM.
- [15] Chang, A. X. and Manning, C. D. (2012). SUTIME: A library for recognizing and normalizing time expressions. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC-2012), Istanbul, Turkey, May 23-25, 2012*, pages 3735–3740.
- [16] Chasin, R., Woodward, D., Witmer, J., and Kalita, J. (2014). Extracting and displaying temporal and geospatial entities from articles on historical events. *Comput. J.*, 57(3):403–426.
- [17] Choi, Y., Cardie, C., Riloff, E., and Patwardhan, S. (2005). Identifying sources of opinions with conditional random fields and extraction patterns. In *HLT/EMNLP 2005, Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing, Proceedings of the Conference, 6-8 October 2005, Vancouver, British Columbia, Canada*.
- [18] Culotta, A., Kristjansson, T., McCallum, A., and Viola, P. (2006). Corrective feedback and persistent learning for information extraction. *Artif. Intell.*, 170(14-15):1101–1122.
- [19] Cunningham, H., Maynard, D., Bontcheva, K., and Tablan, V. (2002). A framework and graphical development environment for robust NLP tools and applications. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics, July 6-12, 2002, Philadelphia, PA, USA.*, pages 168–175.
- [20] de Faria, C. G. and Girardi, R. (2011). An information extraction process for semi-automatic ontology population. In *Soft Computing Models in Industrial and Environmental Applications, 6th International Conference SOCO 2011, 6-8 April, 2011, Salamanca, Spain*, pages 319–328.
- [21] Dietterich, T. G. (2002). Machine learning for sequential data: A review. In *Structural, Syntactic, and Statistical Pattern Recognition, Joint IAPR International Workshops SSPR 2002 and SPR 2002, Windsor, Ontario, Canada, August 6-9, 2002, Proceedings*, pages 15–30.

- [22] Downey, D., Etzioni, O., and Soderland, S. (2005). A probabilistic model of redundancy in information extraction. In *IJCAI-05, Proceedings of the Nineteenth International Joint Conference on Artificial Intelligence, Edinburgh, Scotland, UK, July 30-August 5, 2005*, pages 1034–1041.
- [23] Feldman, R., Rosenfeld, B., and Fresko, M. (2006). Teg-a hybrid approach to information extraction. *Knowl. Inf. Syst.*, 9(1):1–18.
- [24] Finkel, J. R., Grenager, T., and Manning, C. D. (2005). Incorporating non-local information into information extraction systems by gibbs sampling. In *ACL 2005, 43rd Annual Meeting of the Association for Computational Linguistics, Proceedings of the Conference, 25-30 June 2005, University of Michigan, USA*.
- [25] Ganti, V., König, A. C., and Vernica, R. (2008). Entity categorization over large document collections. In *Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Las Vegas, Nevada, USA, August 24-27, 2008*, pages 274–282.
- [26] Girardi, R. (2010). Guiding ontology learning and population by knowledge system goals. In *KEOD 2010 - Proceedings of the International Conference on Knowledge Engineering and Ontology Development, Valencia, Spain, October 25-28, 2010*, pages 480–484.
- [27] Giuliano, C. (2009). Fine-grained classification of named entities exploiting latent semantic kernels. In *Proceedings of the Thirteenth Conference on Computational Natural Language Learning, CoNLL '09*, pages 201–209, Stroudsburg, PA, USA. Association for Computational Linguistics.
- [28] Giuliano, C. and Gliozzo, A. M. (2008). Instance-based ontology population exploiting named-entity substitution. In *COLING 2008, 22nd International Conference on Computational Linguistics, Proceedings of the Conference, 18-22 August 2008, Manchester, UK*, pages 265–272.
- [29] Guarino, N., Masolo, C., and Vetere, G. (1999). Ontoseek: content-based access to the web. *Intelligent Systems and their Applications, IEEE*, 14(3):70–80.
- [30] Harris, Z. (1981). Distributional structure. In Hiž, H., editor, *Papers on Syntax*, volume 14 of *Synthese Language Library*, pages 3–22. Springer Netherlands.
- [31] Hossain, M. (2012). Users’ motivation to participate in online crowdsourcing platforms. In *Innovation Management and Technology Research (ICIMTR), 2012 International Conference on*, pages 310–315.
- [32] Howe, J. (2006a). Crowdsourcing: A definition. *Crowdsourcing: Tracking the rise of the amateur*.
- [33] Howe, J. (2006b). The rise of crowdsourcing.
- [34] Jayaram, N., Khan, A., Li, C., Yan, X., and Elmasri, R. (2013). Querying knowledge graphs by example entity tuples. *CoRR*, abs/1311.2100.

- [35] Jeanquartier, F., Kroll, M., and Strohmaier, M. (2009). Intent Tag Clouds: An Intentional Approach To Visual Text Analysis.
- [36] Jiang, Z., Ji, L., Zhang, J., Yan, J., Guo, P., and Liu, N. (2013). Learning open-domain comparable entity graphs from user search queries. In *22nd ACM International Conference on Information and Knowledge Management, CIKM'13, San Francisco, CA, USA, October 27 - November 1, 2013*, pages 2339–2344.
- [37] Khan, A., Li, N., Yan, X., Guan, Z., Chakraborty, S., and Tao, S. (2011). Neighborhood based fast graph search in large networks. In *Proceedings of the ACM SIGMOD International Conference on Management of Data, SIGMOD 2011, Athens, Greece, June 12-16, 2011*, pages 901–912.
- [38] Kittur, A., Chi, E. H., and Suh, B. (2008). Crowdsourcing user studies with mechanical turk. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI '08*, pages 453–456, New York, NY, USA. ACM.
- [39] Kittur, A., Suh, B., Pendleton, B. A., and Chi, E. H. (2007). He says, she says: Conflict and coordination in wikipedia. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI '07*, pages 453–462, New York, NY, USA. ACM.
- [40] Klein, D. and Manning, C. D. (2003). Accurate unlexicalized parsing. In *Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics, 7-12 July 2003, Sapporo Convention Center, Sapporo, Japan.*, pages 423–430.
- [41] Lafferty, J. D., McCallum, A., and Pereira, F. C. N. (2001). Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of the Eighteenth International Conference on Machine Learning, ICML '01*, pages 282–289, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.
- [42] Lewis, D. D. and Tong, R. M. (1992). Text filtering in muc-3 and muc-4. In *Proceedings of the 4th Conference on Message Understanding, MUC4 '92*, pages 51–66, Stroudsburg, PA, USA. Association for Computational Linguistics.
- [43] Malik, H. H., MacGillivray, I., Olof-Ors, M., Sun, S., and Saroha, S. (2011). Exploring the corporate ecosystem with a semi-supervised entity graph. In *Proceedings of the 20th ACM Conference on Information and Knowledge Management, CIKM 2011, Glasgow, United Kingdom, October 24-28, 2011*, pages 1857–1866.
- [44] Manning, C. D., Surdeanu, M., Bauer, J., Finkel, J. R., Bethard, S., and McClosky, D. (2014). The stanford corenlp natural language processing toolkit. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics, ACL 2014, June 22-27, 2014, Baltimore, MD, USA, System Demonstrations*, pages 55–60.
- [45] Marsh, E. and Perzanowski, D. (1998). Muc-7 evaluation of ie technology: Overview of results. In *Proceedings of the Seventh Message Understanding Conference (MUC-7)*. [http://www.itl.nist.gov/iaui/894.02/related\\_projects/muc/index.html](http://www.itl.nist.gov/iaui/894.02/related_projects/muc/index.html).
- [46] McEwen, A. S., Eliason, E. M., Bergstrom, J. W., Bridges, N. T., Hansen, C. J., Delamere, W. A., Grant, J. A., Gulick, V. C., Herkenhoff, K. E., Keszthelyi, L., Kirk, R. L.,

- Mellon, M. T., Squyres, S. W., Thomas, N., and Weitz, C. M. (2007). Mars Reconnaissance Orbiter's High Resolution Imaging Science Experiment (HiRISE). *Journal of Geophysical Research (Planets)*, 112:5.
- [47] Nirenburg, S. and Raskin, V. (2004). *Ontological semantics*. MIT Press.
- [48] Nothman, J., Curran, J. R., and Murphy, T. (2008). Transforming wikipedia into named entity training data. In *In Proceedings of the Australasian Language Technology Association Workshop 2008*, pages 124–132.
- [49] Prensky, M. (2003). Digital game-based learning. *Comput. Entertain.*, 1(1):21–21.
- [50] Ramakrishnan, G., Joshi, S., Balakrishnan, S., and Srinivasan, A. (2007). Using ILP to construct features for information extraction from semi-structured text. In *Inductive Logic Programming, 17th International Conference, ILP 2007, Corvallis, OR, USA, June 19-21, 2007, Revised Selected Papers*, pages 211–224.
- [51] Ridge, M. (2011). *Playing with difficult objects: game designs for crowdsourcing museum metadata*. PhD thesis.
- [52] Riloff, E. (1993). Automatically constructing a dictionary for information extraction tasks. In *Proceedings of the 11th National Conference on Artificial Intelligence. Washington, DC, USA, July 11-15, 1993.*, pages 811–816.
- [53] Sang, E. F. T. K. and Buchholz, S. (2000). Introduction to the conll-2000 shared task: Chunking. *CoRR*, cs.CL/0009008.
- [54] Sarawagi, S. (2008). Information extraction. *Foundations and Trends in Databases*, 1(3):261–377.
- [55] Sarawagi, S. and Cohen, W. W. (2004). Semi-markov conditional random fields for information extraction. In *Advances in Neural Information Processing Systems 17 [Neural Information Processing Systems, NIPS 2004, December 13-18, 2004, Vancouver, British Columbia, Canada]*.
- [56] Shen, W., Wang, J., Luo, P., and Wang, M. (2012). A graph-based approach for ontology population with named entities. In *21st ACM International Conference on Information and Knowledge Management, CIKM'12, Maui, HI, USA, October 29 - November 02, 2012*, pages 345–354.
- [57] Silberztein, M., Váradi, T., and Tadic, M. (2012). Open source multi-platform nooj for NLP. In *COLING 2012, 24th International Conference on Computational Linguistics, Proceedings of the Conference: Demonstration Papers, 8-15 December 2012, Mumbai, India*, pages 401–408.
- [58] Soliman, W. and Tuunainen, V. K. (2015). Understanding continued use of crowdsourcing systems: An interpretive study. *JTAER*, 10(1):1–18.
- [59] Sutton, C. and McCallum, A. (2006). *Introduction to Conditional Random Fields for Relational Learning*. MIT Press.

- [60] Takeuchi, K. and Collier, N. (2002). Use of support vector machines in extended named entity recognition. In *Proceedings of the 6th Conference on Natural Language Learning - Volume 20*, COLING-02, pages 1–7, Stroudsburg, PA, USA. Association for Computational Linguistics.
- [61] Tanev, H. and Magnini, B. (2006). Weakly supervised approaches for ontology population. In *EACL 2006, 11st Conference of the European Chapter of the Association for Computational Linguistics, Proceedings of the Conference, April 3-7, 2006, Trento, Italy*.
- [62] Toutanova, K., Klein, D., Manning, C. D., and Singer, Y. (2003). Feature-rich part-of-speech tagging with a cyclic dependency network. In *HLT-NAACL*.
- [63] Vallet, D. and Castells, P. (2011). On diversifying and personalizing web search. In *Proceedings of the 34th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR '11*, pages 1157–1158, New York, NY, USA. ACM.
- [64] van 't Woud, J., Sandberg, J., and Wielinga, B. (2011). The mars crowdsourcing experiment: Is crowdsourcing in the form of a serious game applicable for annotation in a semantically-rich research domain? In *Computer Games (CGAMES), 2011 16th International Conference on*, pages 201–208.
- [65] von Ahn, L. and Dabbish, L. (2004). Labeling images with a computer game. In *Proceedings of the 2004 Conference on Human Factors in Computing Systems, CHI 2004, Vienna, Austria, April 24 - 29, 2004*, pages 319–326.
- [66] Zheng, H., Li, D., and Hou, W. (2011). Task design, motivation, and participation in crowdsourcing contests. *Int. J. Electron. Commerce*, 15(4):57–88.



A large, stylized, grey letter 'A' is positioned in the top right corner of the page, partially overlapping the grey header bar.

## Extended Entity Extraction

In the sample picture shows that only hyperlink pattern entities are generated separately as an extension of entity list. But there has lot of noisy text also mined with these entity list which we filter in our process work.

1	European Parliament election, 1979, Minister of Foreign Affairs (France) French foreign minister, Robert Schuman
2	Treaty of Paris (1951), signed not only by France and West Germany, but also by Italy and the three Benelux
3	EEC and EURATOM, with whom it shared its membership and some Institutions of the European Union institutions
4	European Economic Community
5	EEC and EURATOM, with whom it shared its membership and some [[Institutions of the European Union institutions
6	Roy Jenkins President Jenkins, in June 1979, the elections were held in all the then-members (see European Parliament election, 1979
7	Carlo Schmid (German politician) Carlo Schmid
8	Raymond Aron
9	European Parliament, held its first elections in 1979, slowly gaining more influence over Community decision making.
10	Bernard Montgomery, Dwight D. Eisenhower, Georgy Zhukov and Jean de Lattre de Tassigny.
11	James Dillon (Fine Gael politician) James Dillon
12	Franklin D. Roosevelt {{sfn Gromyko 1989 pp=48&#x2D;9}} even though he believed him to a representative of the bourgeoisie class
13	Russian President Dmitry Medvedev whilst on a state visit to Russia on 18 December 2008.
14	Grover Cleveland and then from President William McKinley
15	Woodrow Wilson ordered the U.S. occupation of Veracruz occupation of Veracruz
16	Charles Pelot Summerall, commander of the 1st Infantry Division (United States) First Infantry Division and V Corps (United States) V Corps
17	Alain Souchon
18	Jacques Chirac against Jean-Marie Le Pen in the french presidential election, 2002 2002 presidential election
19	List of Belgian monarchs King of the Belgians
20	Leopold III of Belgium King Leopold III (1901&#x2D;83) and his first wife, Princess Astrid of Sweden
21	Mary Lillian Baeis
22	Wilfried Martens
23	Jean-Marie Guyau

Fig. A.1 Sample Pattern For The Extension Of Entities



## Entity Classification Sample

In the sample picture shows that after generating the hyperlink pattern entities classification technique has been applied to distinguish the person and organization class for the entity input list, which is also a filtering process in our system.

```

1 <PERSON>Dmitry Medvedev</PERSON> in the <LOCATION>United States</LOCATION> 25 September 2009
2 <PERSON>Jacques Chirac</PERSON>,<PERSON>Chirac</PERSON> of <LOCATION>France</LOCATION> on a state visit to <LOCATION>Algiers</LOCATION>
3 President of <LOCATION>Brazil</LOCATION>,<PERSON>Lula da Silva</PERSON>,<PERSON>Lula da Silva</PERSON>, on a state visit to Brasília, in 2005.
4 Mayor of <LOCATION>Ibiza</LOCATION> in 1970 and 1971 and became Senator for <LOCATION>Ibiza</LOCATION> and <LOCATION>Formentera</LOCATION>
5 <ORGANIZATION>DeLoz Commission</ORGANIZATION>
6 <ORGANIZATION>European Parliament</ORGANIZATION> and spokesman for the <ORGANIZATION>People's Party</ORGANIZATION> (<LOCATION>Spain</LOCATION>)
7 <PERSON>Leopold III</PERSON> of <LOCATION>Belgium</LOCATION>,<PERSON>Leopold III</PERSON>
8 <PERSON>Johann Nepomuk Hiedler</PERSON>
9 <PERSON>Johann Georg Hiedler</PERSON>
10 <ORGANIZATION>Nuremberg Trials</ORGANIZATION>,<LOCATION>Nuremberg</LOCATION> in 1945, <ORGANIZATION>Nazi</ORGANIZATION>
11 <LOCATION>Braunau</LOCATION> am Inn, <LOCATION>Austria-Hungary</LOCATION>
12 <PERSON>Edmund Hitler</PERSON>,<PERSON>Edmund</PERSON>
13 <PERSON>Alfons Heck</PERSON>, a former member of the Hitler Youth
14 <PERSON>Benito Mussolini</PERSON>'s "March on <LOCATION>Rome</LOCATION>"
15 <PERSON>Wilhelm Keitel</PERSON>,<PERSON>Keitel</PERSON>,<PERSON>Friedrich Paulus</PERSON>,<PERSON>Paulus</PERSON>, and <PERSON>Walther von Brauchitsch</PERSON>
16 <ORGANIZATION>Eastern Front</ORGANIZATION> (World War II),Eastern Front
17 <ORGANIZATION>National Socialist German Workers Party</ORGANIZATION>
18 Free State of <LOCATION>Brunswick</LOCATION>,<LOCATION>Brunswick</LOCATION>, who was a member of the <ORGANIZATION>NSDAP</ORGANIZATION>
19 <ORGANIZATION>KGB</ORGANIZATION> team with detailed burial charts secretly exhumed five wooden boxes which had been buried at the SMERSH
20 <ORGANIZATION>German Army</ORGANIZATION> (German Empire), 'Reichsheer', unit = 16th Bavarian <ORGANIZATION>Reserve Regiment</ORGANIZATION>
21 <ORGANIZATION>Harriman Brothers & Company</ORGANIZATION>
22 <ORGANIZATION>Friedrich Neumann Foundation</ORGANIZATION>
23 <PERSON>Schuman</PERSON> Plan to create the European Coal and <ORGANIZATION>Steel Community</ORGANIZATION> (<ORGANIZATION>ECSC</ORGANIZATION>)
24 diplomacy in 1911 when he entered the <ORGANIZATION>United States Foreign Service</ORGANIZATION>,<ORGANIZATION>Foreign Service</ORGANIZATION>
25 Treaties of <LOCATION>Rome</LOCATION> establishing the <ORGANIZATION>EEC</ORGANIZATION> and <ORGANIZATION>Euratom</ORGANIZATION>
26 Treaties of <LOCATION>Rome</LOCATION> establishing the <ORGANIZATION>EEC</ORGANIZATION> and <ORGANIZATION>Euratom</ORGANIZATION>
27 <ORGANIZATION>NATO</ORGANIZATION>

```

Fig. B.1 Sample Classification For The Extension Of Entities

Following Table B.1 is the Calculation of manually labeled 12 entity graphs.

<b>Entity Name</b>	<b>Entity Precision</b>			<b>Edge Precision</b>
Winston Churchill	88	90.2	91.2	89.94
Richard Nixon	88.35	92.73	93.94	95.41
Edward Heath	92.55	92.68	95.65	97.89
Benito Musolini	90.54	92.45	93.55	97.65
Bertrand Russell	84.29	91.67	91.67	92.68
Jacques Chirac	89.16	92.11	87	86.27
Jimmy Carter	90.1	92	91.3	91.36
Ronald Reagan	90	90.91	91	89.49
JeanMonnet	84.5	88	90.91	96.62
Hillary	86.3	93.54	93.33	97.51
Charles de Gaulle	93.33	93.93	95.35	98.41
Bill Clinton	90.76	92.31	90.7	98.98

Table B.1 Calculation of Entity Graph Labeling.

For details of the entity graphs' event and calculation will be found in the following link:  
<http://1drv.ms/15v8GG4>