2017-01-01

Contextual Representation Of Documents, Entities, And Faces Of People Using A News Corpus

Md Abdul Kader
University of Texas at El Paso, abdul.kader880@gmail.com

CONTEXTUAL REPRESENTATION OF DOCUMENTS, ENTITIES, AND FACES OF PEOPLE USING A NEWS CORPUS

MD ABDUL KADER

Doctoral Program in Computer Science

APPROVED:

M. Shahriar Hossain, Ph.D., Chair

Christopher Kiekintveld, Ph.D.

Olac Fuentes, Ph.D.

Rodrigo Romero, Ph.D.

Charles H. Ambler, Ph.D.
Dean of the Graduate School

to my

PARENTS

with love

CONTEXTUAL REPRESENTATION OF DOCUMENTS, ENTITIES, AND FACES OF PEOPLE USING A NEWS CORPUS

By

MD ABDUL KADER

DISSERTATION

Presented to the Faculty of the Graduate School of

The University of Texas at El Paso

in Partial Fulfillment

of the Requirements

for the Degree of

DOCTOR OF PHILOSOPHY

Doctoral Program in Computer Science

THE UNIVERSITY OF TEXAS AT EL PASO

August 2017

Acknowledgements

This dissertation would not have been possible without the support of many people. At first, I would like to express my deep gratitude to my advisor and committee chair Dr. M. Shahriar Hossain, for his continuous support for the research and making this dissertation a reality. Throughout my Ph.D., he has been extremely supportive of whichever path I wanted to take in research. I admire his ability to think differently and see the nuances in everything. I am grateful to him for teaching me the art of technical writing. A special thanks goes to my co-advisor, Dr. Arnold P. Boedihardjo, for his valuable support and advice. His constant encouragement and feedback were vital for publishing my works.

I am thankful to Dr. Christopher Kiekintveld, Dr. Olac Fuentes and Dr. Rodrigo Romero for being in my committee and providing valuable feedback and suggestions. I also would like to thank my co-author Sheikh Motahar Naim for his friendly and supportive collaboration. I also want to thank other faculty members and staff of the UTEP Computer Science Department.

I wouldn't be where I am today without the amazing support, encouragement and love from my parents, other family members and friends. Finally, I must thank my dear wife Munni for supporting me in difficult time.

Abstract

A massive amount of unstructured data, in this information age, is composed of document collections. Examples include news articles, blog posts, scholarly publications, and reports generated by organizations as well as people. Many data mining and machine learning algorithms have been developed in the past decade to support text mining in many applications. Text mining applications cover a wide range of tasks spanning from personal information management — to organizational decision making, to disease control through epidemic prediction, and to intelligence analysis for national security. One bottleneck has been dominating data mining and machine learning theories for all these text mining applications — the quality of the outcomes of the algorithms depends on the quality of the representation of the documents.

My research exploits imagery and textual content of documents to create high quality representations for documents, document tokens (e.g., names of people), and image snippets (e.g., faces of people in news images). My argument is that the utilization of both images and textual content of documents is crucial in generating a document representation because images within a document are included by the author(s) of the document to complement the textual content. In addition, visual objects found in the images of a document sometimes provide contextual information that might be missing in the textual content of the document. As an example, consider a document published this year that briefly describes a documentary that celebrates the life of Princess Diana. The textual content of the document does not contain the name Prince Charles or Queen Elizabeth. However, there is an image in the article that contains all three faces — Princess Diana, Prince Charles, and Queen Elizabeth — in the same photo. Inclusion of the image in the document representation will provide better features in this case because the image provides additional contextual information to enrich the content. My research focuses on incorporating such contextual information in representations in absence of annotated faces.

I seek to answer three research questions relevant to document representations: 1) how to extract contextual information in absence of labeled data and use that context to produce better representations for text snippets, 2) how to construct contextual information for visual objects (e.g., faces) found in images of a document collection, and 3) how to combine the contextual information extracted from imagery and textual content for document representation. To address the first question, I designed an objective function that uses temporal, geographical, and topical information of documents to generate a multi-graph of relationships between text fragments. I leveraged a neural network based model that leverages the multi-graph to produce high quality representations for textual entities including names of people, location, and organization found in the documents. In response to the second question, I developed a probabilistic model that generates probability distributions over person names, locations, and countries for every human face detected in the images of a document collection. My dissertation scopes down all analyses to news articles. The visual objects in my dissertation are human faces because they are abundant in news articles and provide direct or contextual relation with the content. Finally, to answer the third question, I propose a neural language model that exploits contextual information generated for faces and textual content to represent documents in a compact continuous space. I demonstrate the effectiveness of the methods through a set of rigorous experiments and case studies. My experiments depict that the document representations generated by my proposed method improve the performance of many machine learning algorithms.

Table of Contents

Page

Acknowledgements iv

Abstract v

Table of Contents vii

List of Tables xi

List of Figures xii

Chapter

1 Introduction 1

1.1 What is context? 2

1.2 Associated research questions 4

1.2.1 How to utilize temporal, geographical, and topical information of documents in the contextual modeling of text fragments? (Chapter 3) 5

1.2.2 How to generate contextual information for the people detected in images from the documents containing text and images? (Chapter 4 and 5) 6

1.2.3 How to leverage contextual information in document representation? (Chapter 6) 7

1.3 Outline of the dissertation 8

2 Related Research 10

2.1 Document modeling 10

2.1.1 Traditional language model 10

2.1.2 Neural language model 11

2.1.3 Multimodal representation learning 11

2.1.4 Short text representation 12

2.2 Face-feature extraction 12

2.3 Context generation for face 13
2.3.1 Aligning image and text 13
2.3.2 Context extraction for image 13
2.3.3 Geographical context 15
2.4 Contextual embeddings for text fragments 15
3 Contextual Embedding for Distributed Representations of Entities 17
3.1 Problem Description 20
3.1.1 Problem Formulation 21
3.2 Methodology 22
3.2.1 Document Modeling 23
3.2.2 Expansion from a Seed Document 24
3.2.3 Construction of Entity Relationships 25
3.2.4 Vector Generation from Relationships 30
3.3 Experimental Results 31
3.3.1 Data Collection 31
3.3.2 Significance of Constraints 31
3.3.3 Contextual Relationships for a Seed Document 33
3.3.4 Evaluation through Entity Analogy 35
3.3.5 Evaluation using Clusters 37
3.3.6 Evaluation using Classification 39
3.4 Summary 40
4 Synthesizing Front Facing Views of Faces 42
4.1 Methodology 44
4.1.1 Face Detection and Feature Extraction 44
4.1.2 Facial Key Points Detection 45
4.1.3 Angle Prediction 46
4.1.4 Facial Key Points Frontalization 47
4.1.5 Face Feature Generation from Frontalized Points 48

4.2 Experimental Results . . . 49
4.2.1 Frontalization for Face Recognition . . . 50
4.2.2 Facial Key Point Selection . . . 51
4.2.3 Prediction of Frontalization Angles . . . 52
4.3 Summary . . . 53
5 F2ConText: How to Extract Holistic Contexts of Persons of Interest . . . 54
5.1 Associated Analytic Challenges . . . 56
5.1.1 Contributions . . . 58
5.2 Problem Formulation . . . 59
5.3 Methodology . . . 59
5.3.1 Features Extraction and Modeling . . . 60
5.3.2 Generation of Entity based Context . . . 62
5.3.3 Geographical Context Generation . . . 63
5.3.4 Complexity Analysis . . . 66
5.3.5 Analytical Task Extensions . . . 66
5.4 Experimental Results . . . 70
5.4.1 Quality of Entity Level Face-Contexts . . . 71
5.4.2 Context in Face Recognition . . . 74
5.4.3 Person, Location, and Geographical Contexts . . . 75
5.4.4 Comparison between Different Methods to Generate Geographical Context . . . 77
5.4.5 Tracking Geographical Contexts . . . 79
5.4.6 Context based Clustering of Faces . . . 80
5.4.7 Impact of the use of Context in Story Generation . . . 83
5.4.8 Examples of Generated Stories using Boston Marathon Bombing Data 84
5.4.9 Characteristics of Stories in Terms of User-Settable Parameters . . 86
5.4.10 Runtime Analysis . . . 88
5.5 Summary . . . 88

6 An Unsupervised Framework for Representing Documents Containing Images and Text 90
6.1 Problem Formulation 93
6.2 Methodology 93
6.2.1 Context Extraction for Persons 94
6.2.2 Embedding Generation for Entities 96
6.2.3 Contextual Vector Representation of Documents 98
6.3 Experimental Results 102
6.3.1 Separability of Document Vectors 103
6.3.2 Person Context for Document Classification 104
6.3.3 Evaluation using Clusters 105
6.3.4 Impact of Complementary Document Representation 106
6.4 Summary 108
7 Conclusions 109
7.1 Event prediction 110
7.2 Contextual representation of videos 111
Curriculum Vitae 128

List of Tables

3.1 List of symbols 23

3.2 Selected set of documents and corresponding relationships for a seed document that describes cholera outbreak. 34

3.3 Top 10 contextually similar entities for *Qaddafi*. 36

3.4 Top 10 contextually analogous entities for *Burma*. 38

5.1 List of frequently used symbols 60

5.2 Sample stories generated with and without context. (URLs of the original news documents are provided with the article IDs in blue and can be reached by clicking on the IDs.) 83

5.3 Stories using Boston Marathon Bombing data. (URLs of the original news documents are provided with the article IDs in blue and can be reached by clicking on the IDs.) 85

List of Figures

1.1 Overview of the research: (a) contextual information construction for persons, (b) contextual modeling of text fragments and (c) document representation by exploiting contextual information. 3

3.1 Contextual pieces of information around entity *Obama*. Three entity relationship graphs show three different geographical contexts (Afghanistan, Russia, and Middle East) of three different years (2010, 2012, and 2014). . 18

3.2 Variation of context of the entity *Obama* from November 2008 to September 2014. 20

3.3 *(left)* Candidate generation process from a seed document d_0. *(right)* Cholera outbreak and its preceding related events. 26

3.4 *(left)* The relationship *Cholera:Storm* is found in both documents of each of the k columns indicating a strong contextual relevance. *(right)* None of the relationships is present in all k pairs of documents indicating that the relationships are not very evidential. The objective function will favor the left set of selected documents because it reveals coherent relationships. . . 26

3.5 Two layers neural network for entity vectorization. 30

3.6 (top) Addition of constraints increases the likelihood of having topically similar documents. (middle and bottom) The effect of time constraints on topical evolution. 32

3.7 Experimental results for analogous pairs of entities. 35

3.8 *(left)* Our approaches exhibit positive and higher average Silhouette coefficient than Word2Vec. *(right)* Vectors generated by our neural network based method provides the best Dunn index. 39

3.9 Accuracy, F1 score and ROC curve for classifying documents from 20newsgroups dataset based on the entity vectors produced by the methods. . . . 41

4.1 An embedded tweet in a news snippet. 42
4.2 Three faces with three different orientations. 43
4.3 Detected eyes, nose and mouth corner points using Cascaded Deep Convolutional Neural Network for few faces. 46
4.4 Facial key-points frontalization. 47
4.5 Twenty angles generated from frontalized five facial key points. 49
4.6 Comparison of face recognition accuracies with and without Frontalization technique. 50
4.7 Comparison of two facial key point detection techniques. 52
4.8 Mean square errors of (a) azimuth angle predictors and (b) elevation angle predictors. 53

5.1 Generated contexts of a face. From left to right: the face image for which contexts are generated, person context, location context, geographical context using a bar chart, and geographical context laid on a map. The geographical context shows that the United Kingdom has the highest probability for the British Prime Minister, David Cameron. 54
5.2 Location ambiguities in dataset. 64
5.3 The story contains five news documents associating Boston bombers' involvement in the Waltham triple murder. The story includes trial phase. . 67
5.4 Quality of context in terms of the appearance of the person name in the context of a face. 72
5.5 Resemblance between a face context and a relevant Google image search. . 73
5.6 Comparison of context generation methods for human annotated test data. Adding frontalized features improve quality of context. 74
5.7 Context in face recognition. 75

5.8 An example of geographical context: (left) Probabilities of top twenty countries with highest values, and (right) probabilities are highlighted in a map. 76
5.9 (left) The combination of country distributions $D^1(f)$ and $D^3(f)$ performs best for various values of λ. (middle) Performance of $D(f)$ against baseline $D^B(f)$: lower average KL-divergence using our method indicates better results. (right) $D(f)$ performs better than $D^B(f)$ more than 87% of the times even during the worst choice of λ. 78
5.10 Geo-contextual trends of leaders over time. 79
5.11 Two clusters of faces generated by DBSCAN using name context elements as the features of the faces. 81
5.12 An example of a geo-context based clustering. 82
5.13 Person context clustering has higher quality (small values with large number of topics) than the baseline. 82
5.14 Impact of search parameters on characteristics of stories. 87
5.15 Runtime to generate entity contexts for faces. 88

6.1 Histograms of the entropies of documents. 91
6.2 Overview of the proposed framework. 94
6.3 Context of David Cameron, the former Prime Minister of United Kingdom. 95
6.4 Entity relationships across documents. 96
6.5 A shallow neural network for entity vectorization. 97
6.6 Graphical representation illustrating the model in Equation 6.5. 100
6.7 Document features modeling based on similarity of entity vectors. 101
6.8 First two principal components of the document vectors and the overlap of the components between categories. 103
6.9 Comparison of three document representation techniques for document classification task. 104

6.10 Clustering based evaluation of document vectors using two internal evaluation schemes. Both of the Dunn index and Silhoutte coefficient indicate *DocVecPC* contains useful features for better clustering of documents. . . . 105
6.11 It compares the performance of *DocVecES* and *TFIDF* on three datasets of short documents for classification tasks. 106
6.12 Comparison of the combinations of document representations. The name of the classifiers is given in a parenthesis with legend. 108

Chapter 1

Introduction

In recent times, huge amount of data is being generated from a plethora of heterogeneous sources, such as social media, news organizations, Internet services, scientific communities, biological studies and many others, at an unprecedented rate. A large portion of the data is publicly available and is comprised of unstructured documents, some of which are news articles, blog posts, tweets and Facebook posts. Due to the ease of access and the rise of multimedia, social media and the Internet of things, most of the documents nowadays contain a mixture of different types of content. For example, a news article may contain images, videos, tweets, Instagram posts, audio clips and graphic charts besides textual content. Although the heterogeneous content of a document and its contexts portray a single coherent story of the document, different types of content bring different kinds of contextual information. In the case of imagery content, the context can be a set of persons, a set of locations and a set of countries related to a person depicted in an image. For example, Figure 1.1(a) shows three different contexts – person context, location context, and geographical context – constructed for an image of David Cameron, the former Prime Minister of the United Kingdom. Furthermore, the contextual information might not be present in the associated textual content of the images. The textual context, on the other hand, can be a multi-graph of text fragments, where two text fragments are directly connected if they are contextually related. For example, *Russia* and *Moscow* are contextually connected by the country-capital relationship between them. Moreover, two literally dissimilar text fragments may refer to the same thing, such as *The World Health Organization* and *WHO* referring to the same organization.

Documents containing different kinds of content are a great source of contextual in-

formation and are used by a variety of applications, such as information retrieval [79], document classification [104], document categorization [80] and document indexing [13]. However, most of these applications cannot utilize contextual information and cannot use the content of documents directly, therefore, greatly rely on a representation of documents. Moreover, the performance of the applications is directly correlated with the quality and versatility of the document representation. Various document representation techniques have been proposed in the past, but most of them emphasize the textual content and utilize the semantics and frequencies of text fragments [47, 102, 71, 11, 74]. In spite of numerous research efforts on text-based document representation, very few attempts have been made to avail the opportunities of exploiting non-textual content in document representation. A few researchers have approached this problem by utilizing user-tagged images and text together to represent documents in a continuous space [109]. However, the exploitation of the contextual information of textual and imagery content in a document representation remains undone.

1.1 What is context?

According to Gallagher, *"Context is broadly defined as information relevant to something under consideration"* [43]. In this dissertation, context is represented as a list of entities (such as person name, organization, and location) associated with an object. For example, the context of the face of person will contain a list of person names, organization names, or even locations that are associated with the face. For example, Figure 1.1(a) shows the context constructed for the former Prime Minister of United Kingdom (UK), David Cameron. The related names of David Cameron include his name, his wife's name, the name of the Queen of UK and other relevant names. The associated countries are the United Kingdom and France. To be more precise, the context related to a person-face in this dissertation consists of three probability distributions: 1) a probability distribution over all the person names found in a corpus, 2) a probability distribution over all the

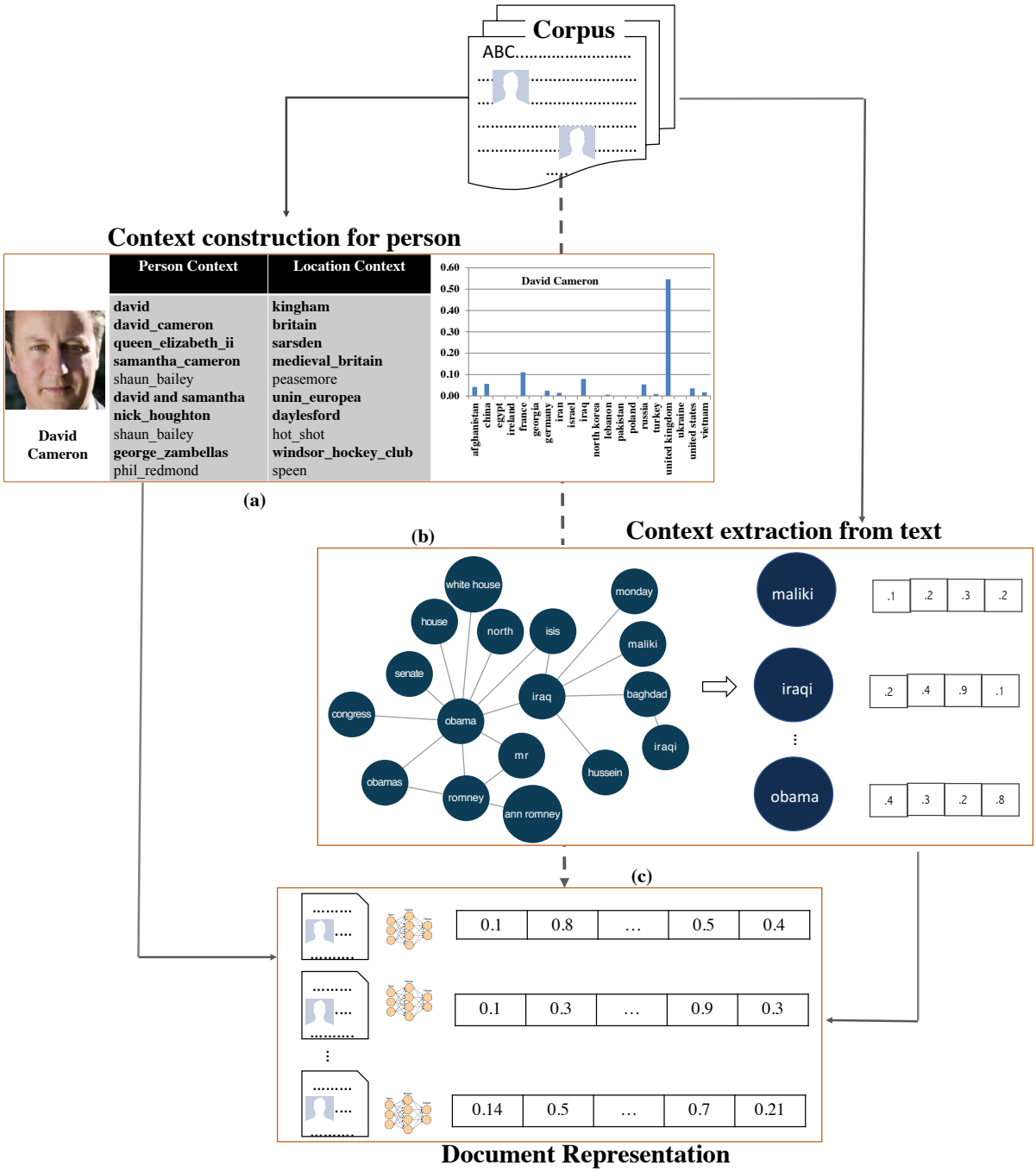


Figure 1.1: Overview of the research: (a) contextual information construction for persons, (b) contextual modeling of text fragments and (c) document representation by exploiting contextual information.

locations found in that corpus and 3) a probability distribution over all countries in the world, where the probability exhibits the degree of associativity with that person.

While context can be generated for image objects, context for text fragments may appear in the form of relationships between entities. Some examples are as follows, (1) flood—cholera are contextually related because cholera outbreaks are seen after severe flood damages in many parts of the world, (2) Myanmar—Burma are related in the sense that Burma is the former name of the country Myanmar, and (3) plant—flora are related because they are synonyms. Analyses of syntactic relationships and use of dictionaries may cover synonym based context but evidential relationships like flood—cholera and Myanmar—Burma require methods that are less dependent on knowledge-bases and more dependent on the holistic appearance of entities in the documents published at different times in a large dataset.

Through this dissertation, I discover that the document representations that include contextual information with the content rather than the content alone provide superior results in many machine learning applications.

1.2 Associated research questions

The broad research theme of this dissertation is to analyze and invent a new representation of documents of a big unstructured dataset considering the contextual aspects of each document-content. The theme can be divided into three broad research questions: how to perform contextual modeling of text fragments, how to extract context for persons in images, and how to represent multimodal documents. A brief overview of the research demonstrated in this dissertation and the relations between the associated research questions are presented in Figure 1.1. The following sections illustrate the research questions in detail.

1.2.1 How to utilize temporal, geographical, and topical information of documents in the contextual modeling of text fragments? (Chapter 3)

Analyzing all-inclusive meaning of a text involves interpreting the meaning of the constituent text fragments, i.e., words, phrases and entities, and the context in which they are embedded. In recent years, there has been much work in studying the semantic and contextual meaning of text fragments. The widely used WordNet synsets [92] is a prime example of a database of semantic relations. For the contextual analysis of text fragments, many research works utilize the co-occurrence of words in a sentence, in a paragraph or in a certain proximity. Two recently developed notable works are GloVe [89] and word2vec [82]. Both of them produce distributed vector representation for words such that analogous words have similar vector representation. GloVe uses co-occurrence statistics of the corpus, whereas word2vec considers a context window for every word.

Rather than only utilizing co-occurrence or local context of words, this dissertation focuses on exploiting the temporal, thematic, and geographical aspects of a document to extract relationships between text fragments. A text fragment can be the context of and related to another text fragment while both reside in different documents. For example, usually a massive *flood* precedes a deadly *cholera* outbreak in the same region. A time and place dependent causal relationship might be present between *cholera* and *flood*, although they might appear in two different documents. In this dissertation, relationships between text fragments are extracted by optimizing an objective function. The extracted relationships are finally used to produce embeddings for text fragments. The associated tasks are described below:

Task 1: *Finding relations between text fragments:* Two text fragments can reside in a sentence, in a paragraph or even in two different documents while retaining some kinds of relationships. There are many ways to extract pairs of related text fragments from sentences or paragraphs, but there exist hardly any mechanism that finds relationships

between text fragments across documents. This dissertation attempts to extract pairs of related text fragments by exploiting temporal, thematic, and geographical information associated with documents in which two text fragments may or may not coexist in a document.

Task 2: *Vector representation for text fragments:* A multi-graph of text fragments can be constructed by considering each relationship as an edge to the multi-graph. To utilize the information present in the multi-graph for document representation this dissertation produces embedding for each text fragment from the multi-graph.

The details of the proposed methods are explained in Chapter 3.

1.2.2 How to generate contextual information for the people detected in images from the documents containing text and images? (Chapter 4 and 5)

Due to the widespread presence of images in documents, analyzing images and visual objects becomes a prerequisite for document representation. Images and the constituent visual objects of images bring additional contextual knowledge in describing documents. Recently, some researchers have made significant progress towards describing images using natural language descriptions [64, 125]. As an example, let us imagine an image where a person is standing in front of a microphone. The algorithm will describe this image as *"A person standing in front of a microphone"*. Instead of describing it in this fashion, providing some related names and locations, such as his/her name, spouse name, city and country, of the person pictured in the image will bring broader context into consideration. To extract such broader contextual information of persons from unstructured documents this dissertation focuses on facial objects found in images.

The contextual information of a person consists of related names of persons, locations and countries. The majority of the approaches related to this work focus on face annotation

and attempt to annotate faces by the name of the persons [46, 38, 39]. These works assume that a face and the name of that person co-appear in a paragraph. In other words, these methods are ineffective when the name of a person does not appear in the same document with the face. My methods, on the other hand, are capable of generating contextual information for every face even when the name of the person is not present in a document collection. Moreover, my methods do not require any labeled information and work on unstructured documents. Two associated tasks are explained as follows:

Task 1: *Mappings between faces and text fragments:* The textual content may contain entities, such as name, location and organization, which may not be related in any way to the persons in the images of a document. On the other hand, images may contain faces of people who are not described in the text of a document. For example, it is unlikely that all the family members of a president appearing in an image will be described in the associated text. All of these together bring a tremendous challenge to construct contextual information for persons in images.

Task 2: *Connecting content-driven context to country:* The location entities found in a document might not always be countries, rather the locations can be region names, towns, cities, or even some organizations that are representative of some areas. Additionally, the same region name can be present in multiple countries. Because of the ambiguities in locations and the lack of explicit presence of countries in documents, identifying related countries of a person is challenging.

The details of the methods are described in Chapter 4 and 5.

1.2.3 How to leverage contextual information in document representation? (Chapter 6)

Many applications such as document retrieval [79], document classification [104], and document clustering [80] primarily rely on a high-quality representation of documents. Although

research on document representation has achieved great progress in the past decade, a majority of approaches rely on textual content alone. Some examples of document modeling techniques are doc2vec [74], bag-of-words [47], vector space modeling [102].

Since images and text together make the description of a document, the utilization of both imagery and textual content in a document representation becomes a necessity. Researchers have attempted to address this problem by combining two separate representations of text and images [37, 129, 59]. Although there are few attempts to model and learn two disparate representations jointly [96], there hardly exists any work that provides a unified representation. Srivastava et al. [109] proposed a method that learns a joint density model over multimodal inputs for a unified representation, but it requires user-tagged images for the training. Moreover, none of these methods utilize contextual information available in each individual content. Therefore, exploiting two disparate content in a unified representation is as important as utilizing each of them individually. In this dissertation, I present a neural language model that exploits two different modalities (imagery and textual) of the context extracted from documents in a vector representation of documents. The related task is described below:

Task: *Exploiting contextual information in a language model for document representation:* Although imagery and textual content of a document together describe the document, the types of the content are dissimilar in nature. Moreover, the multiple modalities of the context extracted separately from two different types of content have distinct statistical properties. Therefore, exploiting these two disparate type of contexts together for a document representation is not a trivial task.

Chapter 6 explains the methods in more detail.

1.3 Outline of the dissertation

The remainder of the dissertation is organized as follows. In Chapter 2, I present related literature on document representation techniques, face feature extraction methods, con-

text generation models for faces, and embedding generation techniques for text fragments. Chapter 3 describes a framework for extracting contextual information from the textual content and generating embeddings for text fragments. Parts of Chapter 3 have been published in the Proceedings of Machine Learning Research (PMLR) [60]. Chapter 4 presents a facial key points frontalization technique that is used to generate complementary face features. In Chapter 5, a probabilistic framework is presented that constructs context for image of people from unstructured documents. A major portion of Chapter 4 and a part of Chapter 5 have been published as a student abstract in the Conference on Artificial Intelligence (AAAI) [61]. Majority of Chapter 5 has been submitted to the Knowledge and Information Systems (KAIS) journal. The submission is currently under review after a second revision. Chapter 6 describes a neural language model that exploits multiple modalities of context extracted from imagery and textual content of documents. A paper with parts of Chapter 6 is planned to be submitted in the IEEE International Conference on Big Data. Finally, Chapter 7 presents future research directions and concludes this dissertation.

Disclosure of the authors' contributions:[1] The papers produced from the content of Chapter 3, 4, and 6 have four authors, namely, Md Abdul Kader, Arnold P. Boedihardjo, Sheikh Motahar Naim, and M. Shahriar Hossain. The journal submitted to KAIS has the following three authors: Md Abdul Kader, Arnold P. Boedihardjo, and M. Shahriar Hossain. In all of these works, I am the first author.

[1] I use 'we' in the later chapters to indicate the contributions of my co-authors in publishing parts of the chapters as papers.

Chapter 2

Related Research

This chapter describes related literature of this dissertation. It can be largely separated into four areas: document modeling, face features extraction, context generation for face, and contextual embeddings for text fragments. In the following sections, the details of these areas are described.

2.1 Document modeling

The related research works for document modeling are divided into four groups: traditional language model, neural language model, multimodal representation learning and short text representation.

2.1.1 Traditional language model

Bag-of-words model [101] is the widely used representation technique for unstructured documents in natural language processing and information retrieval (IR). It considers words found in a corpus as features and disregards the sequence in which the words occur in the documents. The vector space model [102] employs features of bag-of-words and creates feature vectors for each document. The feature value can be a binary value, frequency count or TF-IDF [78] weight of a word in a document. Other feature representation techniques use n-grams [20], phrases [42], concept categories [95], named entities [69] and bag-of-concepts [100]. The features could be reduced by Latent Semantic Indexing (LSI) [30] that transforms the original document vectors to a low-dimensional space such that similar documents are placed in the same topic even if they do not share terms.

2.1.2 Neural language model

One of the major issues of traditional document representations is the large dimensionality. Neural language model [11] addresses this issue by representing words and documents in a low-dimensional continuous space. It started when Bengio et al. [11] introduced a full neural network based model that learns a distributed representation for words from unstructured text. Recently, Mikolov et al. [82] proposed two models – skip-gram and continuous bag-of-words – that use a shallow neural network architecture to learn word embeddings. By the inspiration of these works, Le et al. [74] proposed a model that learns continuous distributed vector representations for pieces of texts or documents.

2.1.3 Multimodal representation learning

The widespread existence of multimodal data encourages researchers to model documents across modalities. A variety of task specific models are present in the literature. To support approximate nearest neighbor search in data with text and images, cross-modal hashing [134, 135] techniques attracted considerable attention to researchers. For the purpose of information retrieval, multimedia documents are represented in terms of visual and textual tokens [59]. Instead of tokens the authors in [96] correlated two unimodal features, concepts using LDA [15] for text and bag-of-visual-keypoints using SIFT histogram [29] for images. In [37, 129] two discrete unimodal representations are linearly coupled to correlate modalities for the cross-modal retrieval (e.g., text query to retrieve image and vice versa).

Few works have been proposed for representing multimodal data in continuous space. Bruni et al. [18] presents a multimodal distributional semantic model that represents words by their distributions over latent dimensions through a probabilistic process from the pattern of co-occurrence of both textual and visual words in documents. In [67] a neural language model is proposed that learns word representations and image features together. A closely related work to our proposed work is presented by Srivastava et al. [109] that introduces a unified representation for documents using deep Boltzmann machine as a joint

model of images and text. It exploits bag-of-words features, real valued image features and user tagged images, whereas our proposed model uses only unlabeled information, such as local context for words and contextual information of faces, to represent documents.

2.1.4 Short text representation

Due to lack of word repetition and context, short texts require special attention for its representation. Most existing approaches include extra contextual information (e.g., search engine result [114], topic derived from Wikipedia [90] and WordNet synsets [53]) from external sources. Many researchers enrich features by obtaining internal semantics such as bag-of-concepts [128] and latent topics [24]. Our approach, on the other hand, uses contextual similarity of named entities and does not rely on external sources.

2.2 Face-feature extraction

This dissertation leverages existing techniques for face-feature extraction and uses the deep convolutional network-based face embedding technique FaceNet [103] that produces state-of-the-art face embedding in a compact Euclidean space. Other popular image feature extraction methods like Eigenface [123], Fisherface [75], Local Binary Pattern [7], and Curvelet [94] work well for face recognition if there are enough number of labeled faces for many orientations of a person. There are rotation invariant face detection techniques [54, 130, 93] that provide good results in recognizing faces but do not provide high-quality descriptors of the faces that can be leveraged for mapping features with large number of possible labels. Hassner *et al.* [48] propose a face frontalization technique that produces frontal views of non-front facing faces. It assumes a single 3D facial shape as an approximation to the shape of all faces. As opposed to the approach of Hassner *et al.*, we generate extended features based on a frontalization technique that does not assume a single 3D template, rather the technique uses facial key-points to realize face orientations. Some researchers also designed algorithms to generate rotation invariant image descriptors using

holistic Fourier feature [7, 70] but those methods do not outperform the recently developed deep learning-based mechanisms for face recognition [118, 88].

2.3 Context generation for face

The following three subsections describe the related works for face-context generation.

2.3.1 Aligning image and text

There are different approaches to generate contexts of images using textual informations that co-exist with the images. The simplest approach uses the full text of a document as the context of an image [62]. Some systems leverage the text in neighborhood of the image as the context [36, 132, 19]. The limitation of these approaches is the assumption that the image and its contextual information co-exist in a single document. Our application offers a holistic approach to generate contexts of faces using information from all documents instead of limiting the context to a local view of the documents where the faces were found.

2.3.2 Context extraction for image

Since deep learning techniques perform extraordinarily well for object detection [117], it opens a new window to researchers to describe the human interactions and semantic relationships among the objects of an image [64, 125, 131]. All these methods generate text descriptions of images based on local contextual understanding of fragments of images commonly called *objects*. A few attempts that map image fragments to a limited number of words [107, 65] suffer from the limitation of required labeled or training data. Dictionary of visual words, visual elements extracted from image, combined with text features is used to enhance semantic representations of words [18, 17]. While such enhancements give better word-representations, the systems do not generate textual words for visual elements of the images in contrast to our approach.

While there has been significant efforts in describing images by summarizing relationships of image fragments, contextual face annotation [121] has received less attention. The authors in [23] proposed a semi-automated annotation technique for faces that leverages similarity based search and relevance feedback concepts to annotate photos of a personal collection. In [25, 26], faces are automatically annotated and clustered based on visual and geo-temporal contexts for a personal photo collection. A more robust and flexible approach to face annotation is the use of auxiliary textual information [39] for automatic annotation. Image captions are used in various ways [46, 38] to annotate images and faces. All these approaches rely on labeled information, coordinates associated with images, and text near the images (e.g., caption). Instead of tagging faces by names, our approach provides a holistic context of each face as a probabilistic distribution to map a face to textual entities and geography. Moreover, our approach does not require any labeled information or metadata about the images.

In [127, 73], an unsupervised approach for automatic face annotation is introduced that retrieves a short list of weakly labeled faces based on textual query. A limitation of this approach is that the training data is prepared based on queries composed of known names. Our approach is more robust in that it does not assume that the name of the person is present in the dataset. Our approach generates a context rather than explicitly providing a name tag. Context extracted from social networks substantially increases face recognition quality [113] for auto tagging faces in personal photographs. A requirement of social presence of all the people limits the capability of this approach. Our approach, on the other hand, is able to generate a context for each face from publicly available unstructured documents.

While most of the literature studies local context in images and mapping faces to a few keywords or names, our approach harnesses its strength by providing context of a face as a probabilistic ordering of all entities found in the entire dataset, as well as a context represented as a geographical distribution.

2.3.3 Geographical context

Geo-tag of documents and user's location are sometimes leveraged as key features of geographical contexts in news recommendation systems [108]. Location information and preferences captured from the hand-held devices are widely used as contexts to recommend points of interest [77, 87]. The extraction of geographical context from the textual resources has been studied in a few frameworks. The systems use different strategies, such as, building a contextual dictionary for all cities using Wikipedia data [84], computing geographic scope through ranking algorithms [106], and modeling locations using geotagged twitter data [66]. Our context generation is nontrivial because we compute the geographical scope of each of the faces detected in the images of a news archive as a probability distribution over all countries of the world. The geographical context generated in the form of a probability distribution of countries overlaid on a map helps analysts build a mental models of the scope of a person of interest.

2.4 Contextual embeddings for text fragments

Due to the superiority of distributed representation in capturing generalized view of information over local representations, it has been successfully used in diverse fields of scientific research [21, 50, 34, 57, 91]. A pioneering work of [99] on distributed representation in language modeling targets learning of representations by back-propagating errors using a neural network. Later, a more sophisticated neural probabilistic model [11] was proposed by Bengio et al., which uses a sliding window based context of a word to generate compact representations. Recently [81] introduced continuous bag-of-words (CBOW) and skip-gram models to compute continuous vector representations of words efficiently from very large data sets. The skip-gram model was significantly improved in [82], both in terms of speed and quality of the generated vectors, by replacing hierarchical softmax with a more efficient negative sampling technique and including phrase vectors along with words. [74] then extended the CBOW model to learn distributed representation of higher level texts

like paragraphs and documents. Unlike the word embedding methods discussed above that produce a singular representation of a word or a phrase, [55] propose a language model that incorporates both local and global document context and learn multiple embeddings per word to account for homonymy and polysemy.

Although these word embedding frameworks use different approaches and address multiple aspects of a language to generate better context of the words, they completely rely on the textual content of the documents. In our framework, we look beyond text merely appearing withing a document by incorporating temporal, geographical and topical information. We argue that these additional information pieces are useful to understand the context of a unit (word or entity) better, and thus can be used to generate word embeddings capturing subtle difference in the context.

Chapter 3

Contextual Embedding for Distributed Representations of Entities

Modern text mining tasks extensively rely on lower dimensional representations of documents. Many systems consider words as the unit of text, as well as many frameworks leverage language ontology [22], sentence structures [40, 9], annotations [122], and natural language processing techniques to conceptualize text for better reflection of the context, thus making the tools heavily language-dependent. The task of generating contextual representations of text requires a generalized approach that can capture latent relationships between information pieces without any exhaustive usage of dictionary or linguistic tools. Language independent mechanisms are gradually becoming essential with the increasing appearance of domain-specific terminologies and derivative acronyms in modern text data. With complex textual information, meta data, and latent themes, it has become more challenging to compute relationships between entities because co-occurrence is no more the sole indicator of *relevance* between entities. From the perspective of document similarity, the use of overlap of terms to compute the similarity is not sufficient to capture contextual relevance. This work aims at generating distributed representations of elements of text, especially entities, to capture latent but contextual relevance even when entities do not appear in the same document.

The general aim of a distributed representation is to capture syntactic and semantic relationships. Current distributed representation generation techniques for text datasets,

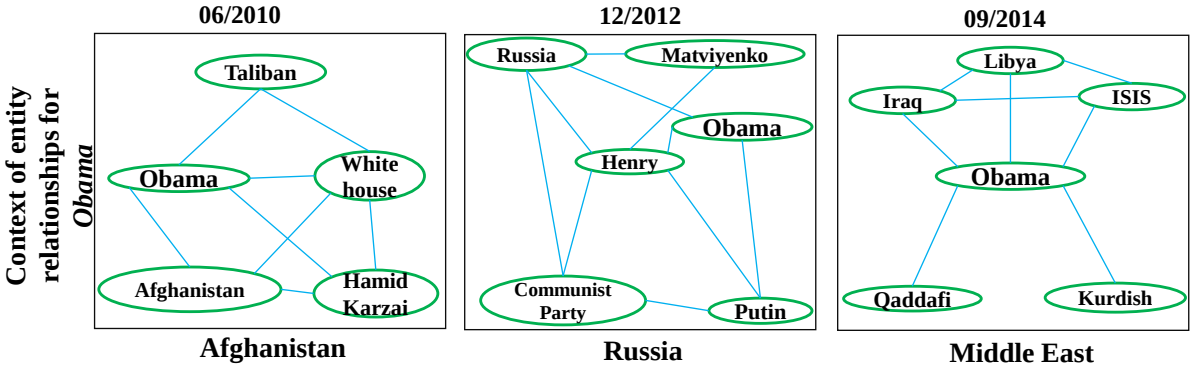


Figure 3.1: Contextual pieces of information around entity *Obama*. Three entity relationship graphs show three different geographical contexts (Afghanistan, Russia, and Middle East) of three different years (2010, 2012, and 2014).

e.g., *word2vec* [82, 83], *doc2vec* [74], and *topic2vec* [105], rely on a sliding window over the contents of the documents to create a context. This context window is used to create the input and output samples for a neural network. The most prominent feature of these frameworks is the ability to generate word vectors that preserve syntactic context of the words. The use of a sliding window as the context still limits the potential of the techniques because of the assumption that contextual words lie solely within a window or within a document. While training the model for generating the distributed vectors, *word2vec*-family of algorithms look into one document (one line, to be precise) at a time — thus ignoring the order and interdependence of the documents. In reality, and also based on our observation, an event or a topic is historically covered by a group of articles, and inclusion of that group information in training could improve the quality of the word vectors that are contextually relevant to a particular event. In addition, time plays an important role in contextual drift of the vocabulary. Most text datasets (such as, news articles and scientific publications) are nowadays time-stamped. As an example of how context of a word may change over time — the context of the word *cloud* before the year 2000 was relevant to weather, while today it might be more relevant to cloud computing and cloud storage. Moreover, the context is tightly coupled with the topics of the documents where the word

cloud was seen.

In addition to time, geographical locations related to a document may have a great influence on the context of the entities involved. For example, Figure 3.1 shows entities surrounding President Barack Obama in three different articles published in 2010, 2012, and 2014. Notice that the entities surrounding *Obama* in the three entity relationship graphs of Figure 3.1 create geographical contexts — Afghanistan, Russia, and Middle East. This is an indication that the geographical scope and context related to an entity may vary over time.

In this chapter, we describe a new mechanism to compute contextually relevant entities of each document of a corpus. The contextual information is bound by temporal, geographical, thematic information retrieved from each document. Our proposed framework generates distributed vectors taking the contextual information into account. As a result of the contextual relevance of the vectors, our method is able to discover causal and evidential relevance between entities. For example, *Cholera* and *Flood* or *Storm* are relevant. Contextually, entities like *Burma* and *Myanmar* are the same, which is better captured by our framework over state-of-the-art methods.

In summary, the contributions of this chapter are as follows.

- We propose an optimization framework that can, for each document of a corpus, flexibly generate contextual information constrained by time, geographical location and latent topics. The retrieved contextual information pieces do not solely rely on co-occurrence of the entities in other documents, rather they depend in relationships of entities seen in other relevant pairs of documents.

- We demonstrate two techniques to effectively generate low dimensional vector representations of entities by leveraging the discovered contextual relationships.

- We conduct a set of experiments to evaluate the generated distributed entity vectors. We also demonstrate how to leverage the generated vectors for traditional clustering and classification problems. The quality of the vectors are evaluated using a

benchmark word-analogy dataset as well.

3.1 Problem Description

From our empirical analysis (Section 3.3.2), we observe that the context of a document is influenced by the topics published recently. As such context detected around an entity may change over time as the relevant topics surrounding that entity change, we exemplify this phenomena in Figure 3.2 which displays four news articles related to President Barack Obama and a relevant entity relationship-graph for each of these four documents generated by our proposed system. Our system uses each document as a seed to retrieve relationships between entities from recently published relevant documents (the mechanism is described later in Section 3.2). As a result, the entities in the relationship-graphs of Figure 3.2 may not appear in the seed documents shown in the figure. The figure shows that a document published in November 2008 describes the relationships between contemporary Senator Barack Obama and Senator Hillary Clinton. The relationship graph reveals contextual relationship between President Bush, White House and Illinois, Barack Obama, and Hillary

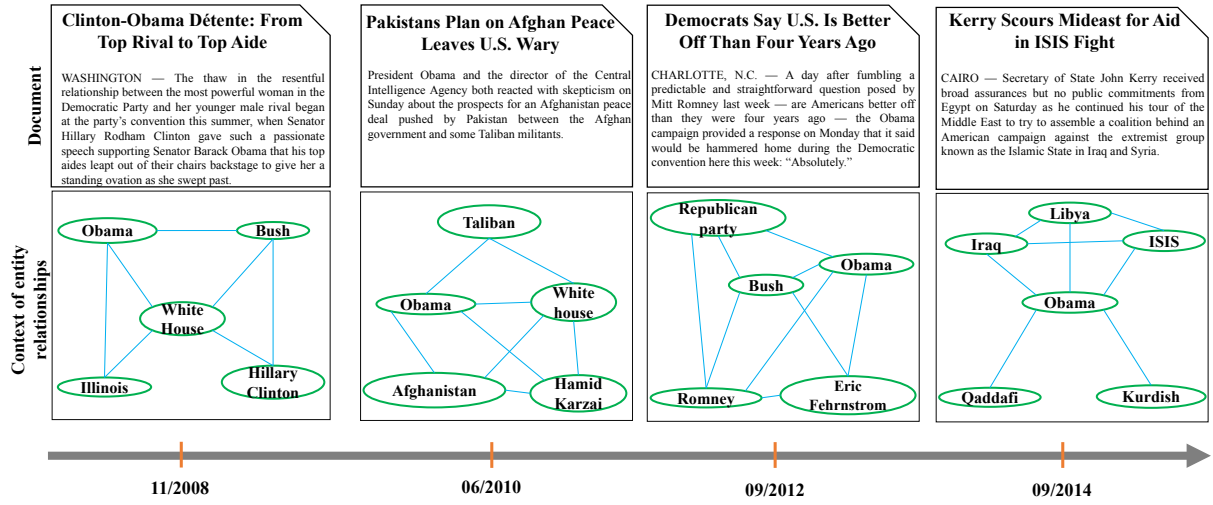


Figure 3.2: Variation of context of the entity *Obama* from November 2008 to September 2014.

Clinton. Another document published in June 2010, which contains President Obama, shows a relationship graph that is different than the one published in 2008. This is because President Obama's context relevant to the document published in 2010 is surrounded by different entities than in 2008. In September 2012, the surrounding context of the President Barack Obama switches to the election where Mitt Romney was the opponent leader from the Republican Party. The relationship graph in Figure 3.2 includes journalist Eric Fehrnstrom who was related to Romney as a top donor for the election campaign. The document placed in 2012 in Figure 3.2 does not contain the entity *Eric Fehrnstrom* but our system includes him because of his relevance with the topic of the document. President Obama's surrounding context through a seed document published in 2014 shows that the concentration shifted toward entities like *Libya*, *Iraq*, *ISIS*, *Qaddafi*, and *Kurdish*.

The example of Figure 3.2 demonstrates that the surrounding context around an entity may change over time. Along with many parameters, the context is influenced by the topic of the documents where an entity is observed. Our framework retrieves the relevant entities through evidence seen in recently published articles, establishes relationships between pairs of entities seen in different (but relevant) documents, and finally builds a holistic contextual representation for each entity leveraging the relationships. We formally describe the associated problem in the following subsection.

3.1.1 Problem Formulation

Let $\mathcal{D} = \{d_1, d_2, \ldots, d_{|\mathcal{D}|}\}$ be the set of documents and $\mathcal{E} = \{e_1, e_2, \ldots, e_{|\mathcal{E}|}\}$ be the set of entities in the corpus. Each document $d \in \mathcal{D}$ has a set of entities $\mathcal{E}_d \subset \mathcal{E}$, which we refer to as the textual content of d. In addition to its textual content, every document also includes a set of extra information. Let the geographical location related to each document $d \in \mathcal{D}$ be G_d and the publication date of d be T_d. Also, inspired by topic modeling [16] techniques, we assume that every document is a mixture of l latent topics. Let $\mathcal{T}_d$ be the topic distribution in document d.

Our primary focus in this work is on news articles. The higher level task from an analytic

point of view is to generate vectors for all entities relevant to a subset of documents $\mathcal{D}_s \subset \mathcal{D}$ that represents a particular event of interest, e.g., *Ebola outbreak* or *Cholera*. That is, $\mathcal{D}_s$ is the user input for the generation of relevant entities. Ideally, $\mathcal{D}_s$ can be the set of returned documents from a search query, or a set of documents prepared by an expert. Notice that $\mathcal{D}_s$ is the set of seed documents but the scope of relevant entities of the seed documents may span the entire corpus $\mathcal{D}$. Combining the seed information with textual, geographical and temporal relevance of each document, we define a series of tasks to generate distributed vectors for each entity.

1. For each given seed document $d \in \mathcal{D}_s$, identify the set of nearest neighbors $\mathcal{N}_d \subset \mathcal{D}$.
2. For a given seed document d, find a set of documents $\vartheta_d \subset \mathcal{D}$ that are published before d. Each document in ϑ_d satisfies some topical, geographical, and temporal constraints. For each nearest neighbor of d, $d' \in \mathcal{N}_d$, list documents $\vartheta_{d'} \subset \mathcal{D}$ that are published before d' and that satisfy the same topical, geographical, and temporal constraints.
3. Identify a set of entity relationships $R = \{\rho_1, \rho_2, \ldots, \rho_{|R|}\}$ where each relationship $\rho_i \in R$ is a pair of entities (e_1, e_2) such that e_1 is observed in d and d' where d is the seed document and d' is a nearest neighbor of d. Additionally, e_2 is observed in a document of ϑ_d and another document of $\vartheta_{d'}$. The more evident ρ_i is among d' and documents of $\vartheta_{d'}$ the stronger ρ_i is.
4. Transform the entity relationship set R generated for every seed document $d \in \mathcal{D}_s$ to generate distributed representations of every entity traced by the steps above.

In the next section we describe our proposed framework for carrying out these tasks.

3.2 Methodology

The proposed framework consists of a number of components. First, we develop a document model where each document is represented as a probability distribution over the set of entities in the corpus. Second, for each seed document, we find a set of nearest

neighbor documents. Third, for a seed document and each of its nearest neighbors, we generate another set of documents constrained by some criteria. Fourth, we formulate an optimization problem to extract the relationships between the entities in the sets of documents retrieved using the previous steps. Finally, we compute distributed vectors for the entities encountered in all the documents selected for all seeds. We leverage two methods to generate the vectors, one is focused on a graph based approach and the other one is driven by the machinery commonly seen in neural network based distributed vector generation [82, 105]. In the following subsections we describe each of these steps in more detail. For the convenience of the readers, we list the symbols used in this chapter in Table 3.1.

3.2.1 Document Modeling

Our approach focuses on entities detected from the text instead of considering words as the primary feature unit. The motivation behind the use of entities comes from the ana-

Table 3.1: List of symbols

Symbol	Description
$\mathcal{D}$	Set of documents
$\mathcal{E}$	Set of entities
$\mathcal{D}_s$	Set of seed documents
G_d	Geo location of document d
T_d	Publication date of document d
l	Number of latent topics in the corpus
$\mathcal{T}_d$	Topic distribution of document d
$\mathcal{N}_d$	Nearest neighbors of d
ϑ_d	Documents published before d bound by topical, geographical, and temporal constraints
α	Geographical context threshold
β	Temporal context threshold

lytic necessity of proximity measures among pairs of entities like people, organization, and location. The process described in this chapter is generic in nature and can be adapted for unigrams or words without any modification. We use standard Named Entity Recognizers [8, 111] to extract entities from each news articles of the corpus. The probability distribution $P^d = \{p_1^d, p_2^d, \ldots, p_{|\mathcal{E}|}^d\}$ of each document $d \in \mathcal{D}$ over the set of entities $\mathcal{E}$ can be computed as:

$$P_i^d = \frac{W(e_i, d)}{\sum_{e' \in \mathcal{E}} W(e', d)} \tag{3.1}$$

where e_i is the i^{th} entity of the entity set $\mathcal{E}$ and $W(e, d)$ is the weight reflecting the associativity between document d and entity e. We compute the association between each document $d \in \mathcal{D}$ and each entity $e \in \mathcal{E}$ using a normalized form of *TF-IDF* [51].

$$W(e, d) = \frac{(1 + log(tf_{e,d}))(log\frac{|\mathcal{D}|}{df_e})}{\sqrt{\sum\limits_{e' \in \mathcal{E}_d} \Big((1 + log(tf_{e',d}))(log\frac{|\mathcal{D}|}{df_{e'}})\Big)^2}} \tag{3.2}$$

where $tf_{e,d}$ is the frequency of entity e in document d, df_e is the number of documents containing entity e, and $\mathcal{E}_d$ is the set of entities detected in document d.

3.2.2 Expansion from a Seed Document

As described earlier, the basic idea of a seed document comes from the fact that context of any entity appears from a document. The context of the same entity seeding from two different documents may vary. Later in Section 3.2.3, we describe how our framework discovers relevant entities (or, entity relationships, to be more precise) given a seed document. Figure 3.3 outlines the process of expanding a seed document. For each seed document $d \in D_s$ where $D_s \subset D$, we select k nearest neighbors $\mathcal{N}_d = \{d_1, d_2, \ldots, d_k\}$ from D. In Figure 3.3, the seed document d is denoted by d_0 for consistency in pictorial representation. The k-nearest neighbors are selected based on KL-divergence [68] between the probability distribution of d (Equation 3.1) and the distribution of each of the documents in D. The set of $k+1$ documents, $\mathcal{N}_{d_0} = \{d_0, d_1, d_2, \ldots, d_k\}$, ideally represents a coherent set of textually

similar documents. For example, the seed document d_0 in Figure 3.3 (illustrated in the half right side of the figure) is about *Cholera outbreak in Haiti*, and the other three documents are the nearest neighbors containing similar events.

Once we have documents $\mathcal{N}_{d_0}$ similar to the seed document, our approach seeks a prior event or entity relationships that most likely led to the event described in the seed document. As described later in Section 3.3.2, the theme of a particular news article is more prominent in its recent past. We use a popular topic modeling algorithm, Latent Dirichlet Allocation (LDA) [16], to estimate the topic distribution $\mathcal{T}_d$ in each document $d \in D$. For every document $d_i \in \mathcal{N}_{d_0}$ we create a set of candidate documents $\vartheta_{d_i} = \{d_1^i, d_2^i, \ldots, d_{|\vartheta_{d_i}|}^i\}$ where each candidate document $d_j^i \in \vartheta_{d_i}$ satisfies the following three constraints.

- **Topical divergence:** Relevant events are expected to have some commonality in their topics. Therefore, d_i should have a certain level of topical similarity to $d_j^i \in \vartheta_{d_i}$. d_j^i is included in ϑ_{d_i} only if $KLDiv(\mathcal{T}_{d_i}, \mathcal{T}_{d_j^i}) \leq \alpha$, where α is the topical context threshold.
- **Geographical context:** Relevant events are likely to happen around similar geographical locations. G_d represents the set of all location entities in document $d \in \mathcal{D}$. d_j^i is included in ϑ_{d_i} only if $|G_{d_i} \cap G_{d_j^i}| \geq \beta$, where β is the geographical context threshold.
- **Temporal Order:** Based on our observation regarding dominance of a topic of a document in the recent past, our time constraint for the selection of d_j^i is $T_\theta < T_{d_j^i} < T_{d_i}$, where $T_\theta = T_{d_i} - \theta$ is the date θ days prior to T_{d_i}.

The left half part of Figure 3.3 represents the process described in this subsection and the other half provides sample documents.

3.2.3 Construction of Entity Relationships

The intuition behind generating entity relationships using separate documents is that, if a relationship $\rho = (e_1, e_2)$, where $e_1 \in d_i$ and $e_2 \in d_j^i$, is repeatedly observed between

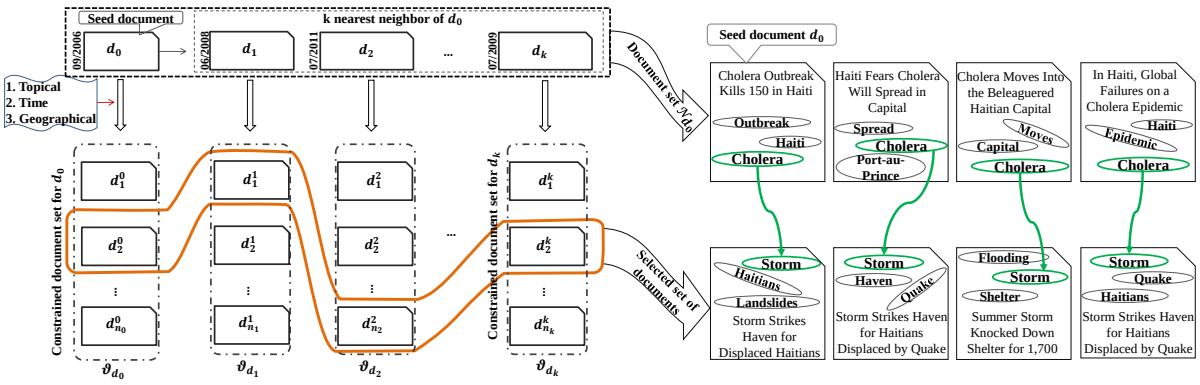


Figure 3.3: *(left)* Candidate generation process from a seed document d_0. *(right)* Cholera outbreak and its preceding related events.

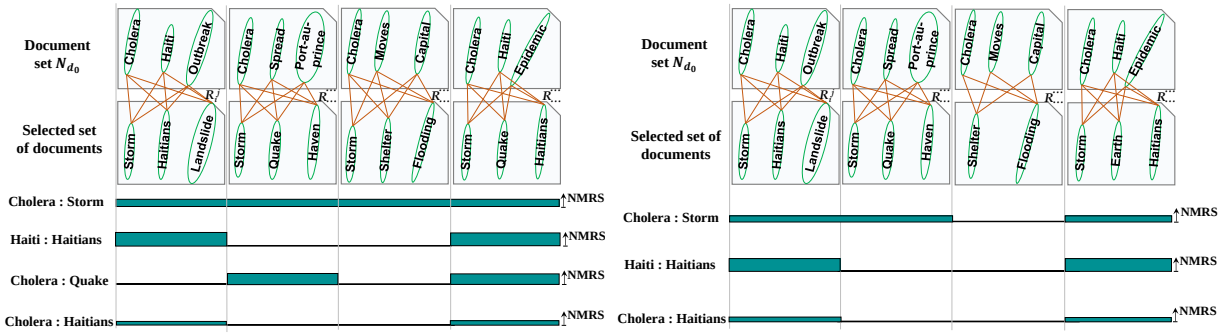


Figure 3.4: *(left)* The relationship *Cholera:Storm* is found in both documents of each of the k columns indicating a strong contextual relevance. *(right)* None of the relationships is present in all k pairs of documents indicating that the relationships are not very evidential. The objective function will favor the left set of selected documents because it reveals coherent relationships.

document pairs (d_i, d_j^i) such that $d_i \in \mathcal{N}_{d_0}$ and $d_j^i \in \vartheta_{d_i}$, then the relationship ρ is an important one because multiple pairs of documents support ρ. In the top row of the right half of Figure 3.3, we present four documents of $\mathcal{N}_{d_0} = \{d_0, d_1, d_2, d_3\}$. From the corresponding sets of ϑ_{d_0}, ϑ_{d_1}, ϑ_{d_2}, and ϑ_{d_3} we select one document d_j^i from each set ϑ_{d_i}. The four best representative documents, d_j^i, are presented in the bottom row of the right half of Figure 3.3.

Figure 3.4 shows two scenarios with two different sets of selected documents. In the left side, each document pair (d_i, d_j^i) contains the entity relationship $\{$*Cholera:Storm*$\}$ whereas in the right side none of the entity relationships exists in all pairs of (d_i, d_j^i) documents. This indicates that the set of documents selected in the left side of Figure 3.4 provides a more coherent evidence of entity relationships than the one in the right side. Now, a crucial question that can be asked is why we advocate selection of at most one document from each ϑ_{d_i} to prepare the selected set of documents. Based on empirical studies during the development of the objective function described in this subsection, rarely seen entity relationships between d_i and documents of ϑ_{d_i} that are evident in all i's are more important than frequently seen entity relationships even if they are observed in all i's of d_i and ϑ_{d_i} document-pairs. For example, the relationship $\{$*Cholera:Basket Ball*$\}$ might be very frequent for most i's of (d_i, d_j^i) document pairs but the abundance of such relationship results from the fact that regardless of time and event under analysis there will be always sports, fashion, technology sections in most news papers. Our objective here is not to find a context using the most frequent relationships observed many times, rather to discover more accurate entity relationships within ϑ_{d_i} that are observed many times for documents similar to d_i, i.e., $\mathcal{N}_{d_0}$, through the selection of rare entities. Therefore, during the selection process, our objective function should seek for a document $d_j^i \in \vartheta_{d_i}$ that creates a rare set of entity pairs with d_i that are evident at similar level of scarcity in other $(d_{i'}, d_j^{i'})$ document pairs where $d_j^{i'} \in \vartheta_{d_{i'}}$ and $i \neq i'$.

This subsection outlines how we select the best representative document d_j^i from ϑ_{d_i} to best capture the entity relationships. Notice that each (d_i, d_j^i) pair, in the example of Figure 3.3, repetitively contains the relationship *(Cholera, Storm)* indicating that *Cholera* appeared after *Storm* because each document $d_i \in \mathcal{N}_{d_0}$ is published after any document $d_j^i \in \vartheta_{d_i}$ was published.

Given a hypothetical probability distribution over all entities X that can be considered as a synthetic document, we can construct a membership probability distribution $\mathbf{v}_i^X = \{\mathbf{v}_{i1}^X, \mathbf{v}_{i2}^X, \ldots, \mathbf{v}_{in_i}^X\}$ for the documents in ϑ_{d_i}. Each $\mathbf{v}_{ij}^X$ will represent how probable it is that

X and $P^{d^i_j}$ are the same compared to all the documents in ϑ_{d_i}.

$$\mathbf{v}^X_{i_j} = \frac{exp(-\|X - P^{d^i_j}\|)}{\sum_{j'=1}^{n_i} exp(-\|X - P^{d^i_{j'}}\|)} \tag{3.3}$$

Since our aim is to select one document from each set ϑ_{d_i} our objective function should reward for a non-uniform distribution of $\mathbf{v}^X_i$. We measure the non-uniform nature of a distribution using the following formula:

$$C(\mathbf{v}^X_i) = \frac{\|U(\frac{1}{n_i}) - \mathbf{v}^X_i\|_1}{2 - \frac{2}{|\mathbf{v}^X_i|}} \tag{3.4}$$

$C(\mathbf{v}^X_i)$ will generate a scalar in the range from 0 to 1 where larger scores indicate high probabilities associated with only a few documents of ϑ_{d_i}.

If X is the free variable of an optimization routine then the following objective function would result in a high probability document in each set ϑ_{d_i}.

$$f(X) = \sum_{i=0}^{k} C(\mathbf{v}^X_i)$$

$f(X)$ will basically provide the best X for which there is a relevant document (without any confusion) in each set ϑ_{d_i}. If each ϑ_{d_i} has an importance factor that is additionally determined as a free variable $A = \{a_1, a_2, \ldots, a_k\}$ such that $\|A\|_1 = 1$, then the objective function becomes

$$f(X, A) = \sum_{i=0}^{k} a_i \times C(\mathbf{v}^X_i) \tag{3.5}$$

Equation 3.5 is a suitable objective function to ensure a common theme between the selected documents of each set ϑ_{d_i}, given that each selected document has the highest $\mathbf{v}^X_{i_j}$ after the optimization routine converges. However, this does not guarantee that the entity relationships R^j_i observed between $d_i \in \mathcal{N}_{d_0}$ and a selected $d^i_j \in \vartheta_{d_i}$ for a particular i are also observed for other i values. At this stage, we will modify Equation 3.5 to incorporate such relationships.

A set of relationships R^j_i between two documents $d_i \in \mathcal{N}_{d_0}$ and $d^i_j \in \vartheta_{d_i}$ is composed of the set of all possible relationships $\rho = (e_1, e_2)$ such that $e_1 \in d_i$ and $e_2 \in d^i_j$. We compute

the shared information between two sets of relationships R_i^j and R_l^k using Normalized Mutual Relationships Score (NMRS):

$$NMRS(R_i^j, R_l^k) = \sum_{\rho \in R_i^j \cup R_l^k} p(\rho|R_i^j, R_l^k) log \frac{p(\rho|R_i^j, R_l^k)}{p(\rho|R_i^j)p(\rho|R_l^k)} \tag{3.6}$$

where the probability $p(\rho|R_i^j)$ of a relationship $\rho = (e_1, e_2)$ given the set of relationships R_i^j is computed using the following formula

$$p(\rho|R_i^j) = \frac{f_{e_1,d_i} * f_{e_2,d_j^i} + 1}{\sum_{\rho' \in R_i^j}(f_{e'_1,d_i} * f_{e'_2,d_j^i} + 1)} \tag{3.7}$$

where f_{e_1,d_i} is the frequency of entity e_1 in document d_i.

Similarly, the probability $p(\rho|R_i^j, R_l^m)$ of the relationship ρ given the set of relationships R_i^j and R_l^m is calculated by

$$p(\rho|R_i^j, R_l^m) = \frac{min(f_{e_1,d_i} * f_{e_2,d_j^i}, f_{e_1,d_l} * f_{e_2,d_m^l}) + 1}{\sum_{\rho' \in R_i^j \cup R_l^m}(max(f_{e'_1,d_i} * f_{e'_2,d_j^i}, f_{e'_1,d_l} * f_{e'_2,d_m^l}) + 1)} \tag{3.8}$$

We modify the objective function in Equation 3.5 to incorporate the relationships in the following new objective function.

$$\begin{aligned} f(X, A) &= \sum_{i=1}^{K} C(\mathbf{v}_i^X) \sum_{j=1}^{n_i} C(\mathbf{v}_{(i+1)}^X) \times \sum_{m=1}^{n_{i+1}} a_i \mathbf{v}_{i_j}^X a_{i+1} \mathbf{v}_{(i+1)_m}^X NMRS\Big(R_i^j, R_{i+1}^m\Big) \\ &= \sum_{i=1}^{K} C(\mathbf{v}_i^X) C(\mathbf{v}_{(i+1)}^X) a_i a_{i+1} \times \sum_{j=1}^{n_i} \sum_{m=1}^{n_{i+1}} \mathbf{v}_{i_j}^X \mathbf{v}_{(i+1)_m}^X NMRS\Big(R_i^j, R_{i+1}^m\Big) \end{aligned} \tag{3.9}$$

Similar to the objective function of Equation 3.5, the objective function of Equation 3.9 will result in a common theme between the selected documents of each set ϑ_{d_i}, given that each selected document has the highest $\mathbf{v}_{i_j}^X$. In addition, the objective function of Equation 3.9 maximizes the entity relationships R_0^j observed between $d_0 \in \mathcal{N}_{d_0}$ and a selected $d_j^0 \in \vartheta_{d_0}$ over all R_i^j sets with subsequent i values. The objective function is smooth and continuous and any local optimization routine will be able to maximize it over the set of variables X and A. We used Python to implement the objective function and leveraged `scipy.optimize.minimize` as our optimization routine.

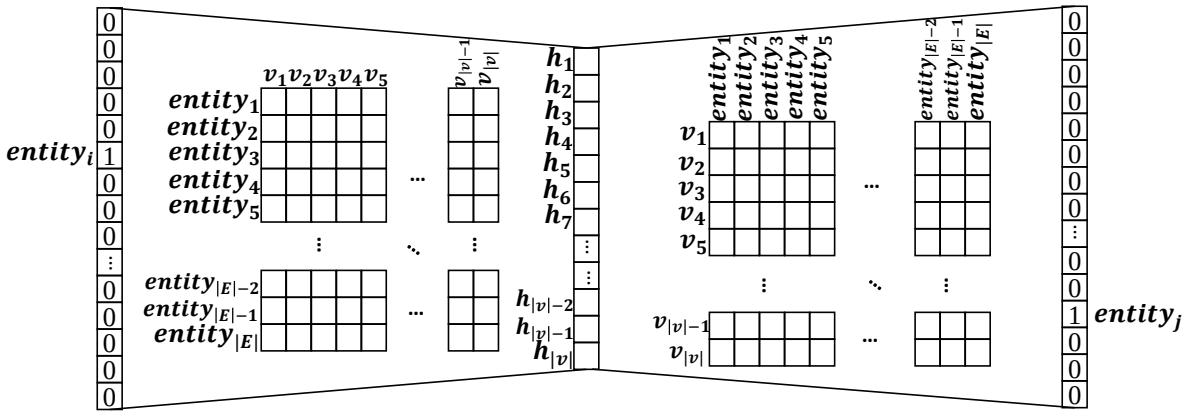


Figure 3.5: Two layers neural network for entity vectorization.

3.2.4 Vector Generation from Relationships

Using the optimization formula described in Section 3.2.3, after the selection of the best (d_i, d_j^i) pairs of documents, we obtain a set of entity relationships. The objective function was maximized for this set of relationships. These entity relationships form a context and can be represented as edges of a graph for every seed document, as shown in Figures 3.1 and 3.2. These transformations are done to extract the latent features contained by the aggregated relationships. Now, the task of vector generation for each entity, given the contextual set of entity relationships for every seed document, can be performed in one of the two ways, (a) compose all the entity relationships in a weighted graph and apply an orthogonal transformation of the weighted graph adjacency matrix to form vectors for the entities, and (b) use the entity relationships discovered for every seed document to train a neural network to generate neural entity embeddings. The first approach uses spectral graph theory [27] and Principal Component Analysis (PCA) to transform the $|\mathcal{E}| \times |\mathcal{E}|$ adjacency matrix to a $|\mathcal{E}| \times C$ matrix of C principal components. The second approach resembles the method used in Word2Vec [82, 83]. At each step of the training of Word2Vec, a set of consecutive words from a document is given to the network where it takes one word from that set as input and attempts to predict the remaining words in the set. We leverage

this model to create vectors of entities by feeding each observed entity relationship (a pair of entities) to the network — one entity is used as input to predict the other one. Figure 3.5 shows that $entity_i$ is given as the input of the two-layer neural network to predict $entity_j$ for a relationship $\rho = (entity_i, entity_j)$.

3.3 Experimental Results

In this section, we seek to answer the following questions.

1. What is the justification for using the temporal, geographical and topical constraints during the optimization relevant to each seed document? (Section 3.3.2)
2. How effective are the generated entity relationships? (Section 3.3.3)
3. How good are the generated vectors in capturing the context of entities? (Section 3.3.4)
4. Can the entity vectors be used to produce high-quality clusters? (Section 3.3.5)
5. How useful are the entity vectors in classifying documents? (Section 3.3.6)

3.3.1 Data Collection

We downloaded news articles from the New York Times archive [4] by using python script and several python modules, e.g., urllib2 and Beautiful Soup, to handle HTTP request and parsing HTML data. We used more than 54,000 New York Times articles that are categorized as politics. For supervised evaluations, we used the 20 Newsgroups dataset [72], which contains approximately twenty thousand documents.

3.3.2 Significance of Constraints

In Section 3.2.2, we explained how a seed document can be expanded by first taking its k-nearest documents and then generating a set of candidate documents for each of those k documents. The candidate documents are selected by enforcing temporal, geographical,

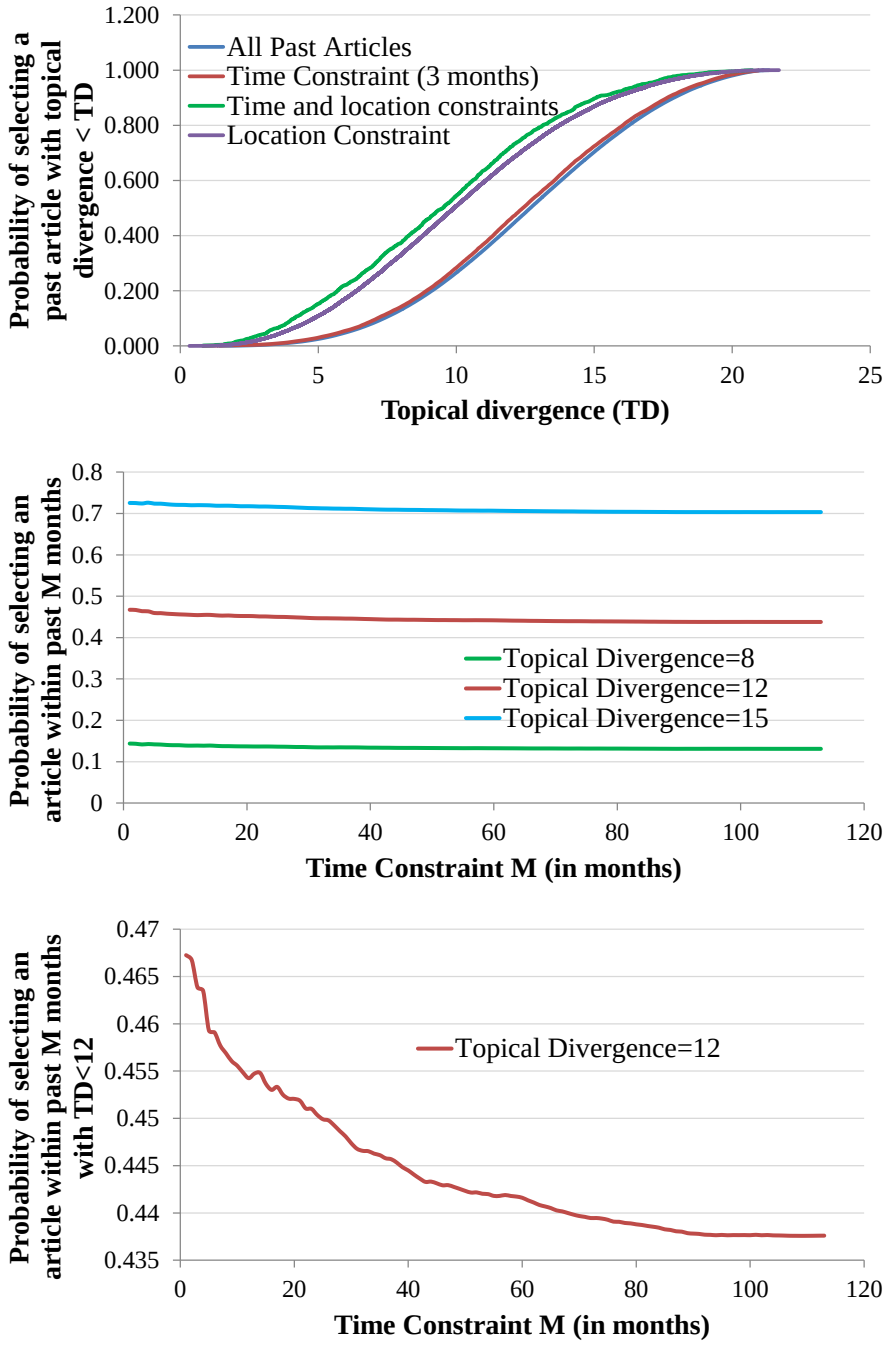


Figure 3.6: (top) Addition of constraints increases the likelihood of having topically similar documents. (middle and bottom) The effect of time constraints on topical evolution.

and topical constraints. In this section, we provide empirical justification for using such constraints while generating the candidate documents. Figure 3.6 (top) shows that the probability of selecting a topically similar document published prior to a seed document increases when selection is constrained by both time and location, as evident through the first line from the top of the plot. Figure 3.6 (middle) demonstrates that longer spans in time as the temporal constraint dilute topics resulting in higher topical divergence with the seed. A similar evidence is found in the experiment with Figure 3.6 (bottom). It shows that longer temporal span in the past for the selection of the candidate documents leads to lower probability of finding topically similar documents. The probability is the ratio, number of documents satisfying topical constraint to total number of documents satisfying time constraint. The topical divergence between the seed document and a document published in the past is measured by computing the KL-divergence between the topic distribution of these two documents. These divergences are averaged over the number of pairs observed during each experiment.

All the experiments of Figure 3.6 illustrate that the selection process of candidate documents from a seed is well founded by natural topical trends observed in news articles.

3.3.3 Contextual Relationships for a Seed Document

After we select the k-nearest documents and the corresponding sets of candidate documents for each seed document d_0, we formulate an optimizer in Section 3.2.3 that produces highly probable entity relationships and the corresponding set of selected documents that carry a common theme. An example of such a set of entity relationships is shown in the second column of Table 3.2 for a Cholera related seed document. The relationships in i-th row of the table are characterized by high $NMRS(R_{i-1}^j, R_i^k)$ scores in Equation 3.6, i.e., they share significant mutual information in the document pairs of i-th row and $(i-1)$-th row. The first document in the first column of Table 3.2 is the seed document that describes Cholera outbreak in Haiti. The other documents in the first column are the k nearest neighbors of the seed document. The third column records a selected document, which yields highly

probable relationships, from the candidate pool of each document in the first column. A few notable relationships are *'the world health organization : world health organization'*, *'unicef : the world health organization'*, and *'health : borders'*. The last column shows the final importance factor or weight of each row, as determined by the optimizer (variable A in Equation 3.9). In this specific case, for the nine pairs of documents in nine rows of the table, the weights varied from 0.08 to 0.12.

Given a seed document, our system is able to discover contextual entity relationships from an automatically crafted set of documents selected from the entire corpus. Table 3.2 shows the outcome for one seed document. For every seed document, our system generates

Table 3.2: Selected set of documents and corresponding relationships for a seed document that describes cholera outbreak.

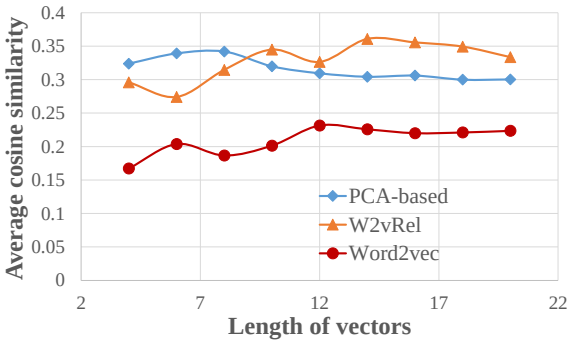
$d_i \in \mathcal{N}_{d_0}$	Set of relationships	Selected set of documents	a_i
Cholera Outbreak Kills 150 in Haiti	–	New Flood Warnings Raise Fears in Pakistan	0.12
Haiti Fears Cholera Will Spread in Capital	'the world health organization : the world health organization', 'the world health organization : health', 'the world health organization : world health organization', 'world health organization : the world health organization', 'world health organization : health'	Evacuations Continue in Southern Pakistan	0.12
Vaccinations Begin in a Cholera-Ravaged Haiti	'world health organization : cholera', 'health : cholera', 'health : port-au-prince', 'world health organization : port-au-prince', 'world health organization : health'	In Haiti, Global Failures on a Cholera Epidemic	0.12
Pattern of Safety Lapses Where Group Worked to Battle Ebola Outbreak	'balakrish nair : haitians', 'balakrish nair : haitian', 'balakrish nair : paul farmer', 'balakrish nair : h.i.v', 'balakrish nair : haiti'	Botswana Doctor Is Named to Lead W.H.O. in Africa	0.12
In Haiti, Global Failures on a Cholera Epidemic	'thomas r : partners', 'thomas r : sierra leone', 'thomas r : ebola', 'thomas r : sierra leones', 'thomas r : he'	In a Gang-Ridden City, New Efforts to Fight Crime While Cutting Costs	0.12
Ebola Could Strike 20,000, World Health Agency Says	'montereys : balakrish nair', 'montereys : nepal', 'montereys : blame', 'montereys : the lancet', 'montereys : tropical medicine'	Health Officials Try to Quell Fear of Ebola Spreading by Air Travel	0.08
Cholera Moves Into the Beleaguered Haitian Capital	'the world health organization : health', 'world health organization : health', 'titus naikuni : ebola', 'titus naikuni : liberia', 'titus naikuni : the world health organization'	Amid Cholera Outbreak in Haiti, Misery and Hope	0.09
Medical Need Climbs Alongside Death Toll in Yemen	'health : borders', 'diarrhea emergency : humanitarian', 'diarrhea emergency : the world health organization', 'diarrhea emergency : marie-evelyne louis', 'diarrhea emergency : christine antoine'	Pakistani Lawmakers Urge Diplomacy in Yemen Conflict but Decline Combat Role	0.12
U.N., Fearing a Polio Epidemic in Syria, Moves to Vaccinate Millions of Children	'unicef : yemens', 'unicef : yemenis', 'unicef : the world health organization', 'unicef : abdu rabbu mansour hadi', 'unicef : houthi'	40 Years After War, Israel Weighs Remaining Risks	0.11

pairs of contextual entities that might not directly appear in the seed document, or the relationships might not even appear in one single document in the entire corpus.

3.3.4 Evaluation through Entity Analogy

In this subsection, we compare the generated vectors for entities using two methods as described in Section 3.2.4, PCA and neural network based approaches, to Google's Word2vec in terms of contextual analogy of entities.

In the first experiment, we evaluate the ability of the distributed vectors obtained from our methods in capturing context. We leverage the database of capital-country, country-currency and city-state pairs provided with the code base of Word2vec [82] as ground truth in this experiment. Every entry of the database was created and verified by the human. We calculate the average cosine similarities among entity pairs of all capital-country, country-currency and city-state pairs. Figure 3.7 (a) shows that our methods, referred as *PCA-based* and *W2vRel*, outperform the Word2vec method in terms of average



(a) Evaluation using set of analogous words shows that our methods perform significantly better for making similar vectors for entities that are contextually analogous.

Entity pair	*Analogous?*	*PCA-based*	*W2vRel*	*Word2vec*
burma : myanmars	✓	***0.84***	***0.99***	***0.64***
burma : yangon	✓	0.713	0.86	0.66
myanmars : yangon	✓	0.954	0.86	0.93
myanmars : tripoli	✗	***0.02***	***-0.06***	***0.45***
burma : tripoli	✗	0.056	-0.06	0.17
republican : al gore	✗	-0.124	0.08	0.7
republican : obama	✗	***-0.111***	***0.09***	***0.31***

(b) Sample cosine similarities between pairs of vectors generated by three methods. Our approaches capture similarities/dissimilarities better than Word2vec both for analogous and non-analogous pairs.

Figure 3.7: Experimental results for analogous pairs of entities.

similarity between the pairs. Interestingly, after a certain length of vector, our *W2vRel* method surpasses *PCA-based* method but the Word2vec method still performs worse than both of our methods. Figure 3.7 (a) provides an evaluation using ground truth analogous entities. As an additional analysis, we examined pair samples to evaluate vectors generated both for analogous and non-analogous pairs. In Figure 3.7 (b), we present cosine similarity scores of seven pairs of entity vectors, out of which three pairs are analogous and four pairs are non-analogous. The table shows that the cosine similarity between *Burma* and *Myanmars* is high using both our approaches than Word2vec. Given that *Burma* is the former name of *Myanmars*, our approaches tend to capture this relationship better than

Table 3.3: Top 10 contextually similar entities for *Qaddafi*.

Qaddafi					
PCA-based		**W2vRel**		**Word2vec**	
colonel qaddafi	0.983	**colonel qaddafi**	0.99	**tripoli**	0.81
the a.p	0.969	**tripoli**	0.973	zimbabwe african national union-patriotic front	0.79
tripoli	0.938	zliten	0.96	ice	0.77
libyan	0.909	alain jupp	0.959	daniel malan	0.770
monica garca prieto	0.824	laurence hart	0.958	keeb	0.768
libyans	0.816	baghdadi al-mahmoudi	0.955	curiosity of ice	0.759
solidarity	0.811	the a.p	0.948	guantnamo	0.756
thirachai phuvanat-naranubala	0.807	jupp	0.947	kabul international	0.754
nature	0.722	mustapha abdul jalil	0.934	**colonel qaddafi**	0.734
jay carney	0.708	bad boy	0.933	james g	0.724

Word2vec. Similarly, our approaches capture better analogous relationships than Word2vec for cases, {burma : yangon} and {myanmars : yangon}. For all the non-analogous samples — {myanmars : tripoli}, {burma : tripoli}, {republican : al gore}, and {republican : obama} — cosine similarity scores resulting from pairs of vectors using our approaches are lower than the scores using Word2vec. This indicates that our approaches are able to distinguish non-contextual pairs better than Word2vec.

We also examined entities of interest by computing their 10-nearest neighbors using all three methods. Table 3.3 compares the top ten contextually similar entities of *Qaddafi* retrieved by these three methods. In Table 3.3, *PCA-based* and *W2vRel* refer to the two approaches we use to generate vectors for entities. Obviously, *Word2vec* refers to Google's Word2vec approach. To make the systems comparable, we made the text input for Word2vec a list of entities as they appear in the text documents instead of using word units. All three methods retrieve correlated entities to some extent as the most similar entities to *Qaddafi*. Cosine similarity between vectors was used to compute proximity. The entities retrieved by two of our methods produced better results than the ones retrieved by the baseline Word2vec approach. For example, *Colonel Qaddafi* appears as the most similar entity to *Qaddafi* using both our approaches but Word2vec lists *Colonel Qaddafi* as the ninth nearest entity to *Qaddafi*. Our observation in this case is that the PCA-based method retrieved most contextual entities for *Qaddafi*. Highly relevant entities to *Qaddafi* are marked in the table in bold.

Similarly, Table 3.4 shows the top ten entities contextually similar to *Burma*, the former name of the country *Myanmar*. The PCA-based and W2vRel methods retrieved several related entities that are highlighted in bold. Word2vec could not retrieve any entity that is related to *Myanmar*, to the best of our knowledge.

3.3.5 Evaluation using Clusters

The previous section describes that our approaches generate vectors that are easily distinguishable for non-analogous pairs, as well as detectable for analogous pairs. Vectors with

Table 3.4: Top 10 contextually analogous entities for *Burma*.

Burma					
PCA-based		**W2vRel**		**Word2vec**	
myanmar.he	0.973	student generation	0.999	teams	0.853
association of **southeast asian** nations	0.973	**myanmars**	0.999	clegg	0.837
nobutaka machbimura	0.972	kenji nagai	0.998	stanford hospital	0.827
min zaw	0.972	**burma** media association	0.998	asahi glass foundation	0.813
kenji nagai	0.971	association of **southeast asian** nations	0.998	shaw	0.803
ibrahim gambari	0.971	lee hsien loong	0.998	van	0.802
shwe	0.84	gambari	0.998	mcdonnell young	0.801
myanmars	0.84	**myanmar.he**	0.997	central district of california	0.792
sheik nabil qaouk	0.84	u nyan win	0.997	yavlinsky	0.788
tyre	0.84	min zaw	0.996	kenji nagai	0.786

such capabilities tend to produce good clustering outcomes. In this section we evaluate the generated vectors in terms of clustering quality. We cluster the entities, given a generated vector for each entity, using k-means clustering. We apply k-means on three different sets of entity vectors generated by three methods (a) our *PCA-based* approach, (b) our neural network based approach referred to as *W2vRel* in the figures, and (c) benchmark *Word2vec* approach. We measure the quality of clustering outcomes using two standard cluster evaluation measures: Silhouette coefficient [97] and Dunn index [32]. For both the measures, larger values are better. Figure 3.8 (left) shows that our two proposed meth-

ods outperform Word2vec in terms of the average Silhouette coefficient. Negative average Silhouette coefficient for Word2vec indicates lack of structure in the clustering outcome. Both our approaches have positive Silhouette coefficients. Figure 3.8 (right) shows that our neural network based method, referred as *W2vRel* in the figure, performs better than the Word2vec and our PCA based in terms of Dunn index. Our PCA based method performs marginally better than baseline Word2vec method.

3.3.6 Evaluation using Classification

In this section, we compare the quality of the vectors by the three methods through a classification task. We use 20newsgroups [72] dataset for this purpose that contains 18,828 news articles divided into 20 exclusive classes related to topical categories. The purpose of our methods and Word2vec is to generate vectors for entities. We construct feature vectors for the documents for classification by first clustering the entity vectors into c groups using k-means clustering algorithm. Then we create a c-dimensional feature vector for each document d_i where the j^{th} element of the feature vector is the number of entities in document d_i that belong to the j^{th} cluster of entities.

We use Support Vector Machine (SVM) to classify the documents. We use 10-fold

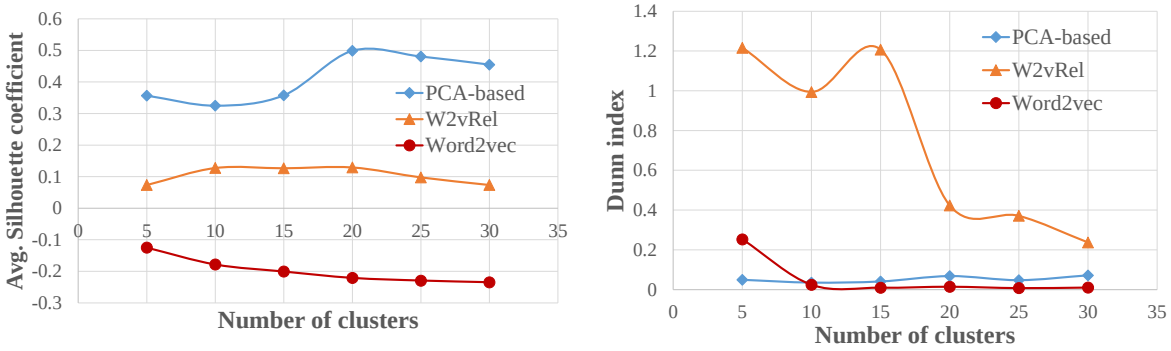


Figure 3.8: *(left)* Our approaches exhibit positive and higher average Silhouette coefficient than Word2Vec. *(right)* Vectors generated by our neural network based method provides the best Dunn index.

cross validation for the evaluation. In a previous section we have observed that the entity vectors generated by our methods return better clustering of entities. As a result, the entity vectors contribute towards better document classification as shown in Figure 3.9. The top and middle plots in Figure 3.9 show that our methods (marked as PCA-based and W2vRel) outperform the Word2vec method in terms of classification accuracy and F-measure. Figure 3.9 (bottom) shows the corresponding ROC curve for each method. To combine multiple class ROC we use macro averaging. Macro averaging is appropriate in this example because the 20 newsgroups dataset contains almost equal number of documents for each group. Both our approaches result in higher Area Under the Curve (AUC) than that of the Word2vec method.

3.4 Summary

Our framework leverages contextual information available in a corpus to generate distributed representations for entities observed in each document. Experimental results in this chapter depict comparative analyses of different word embedding techniques, studies of effectiveness of the generated distributed vectors in several data mining applications, and qualitative analyses of the contexts generated for entities.

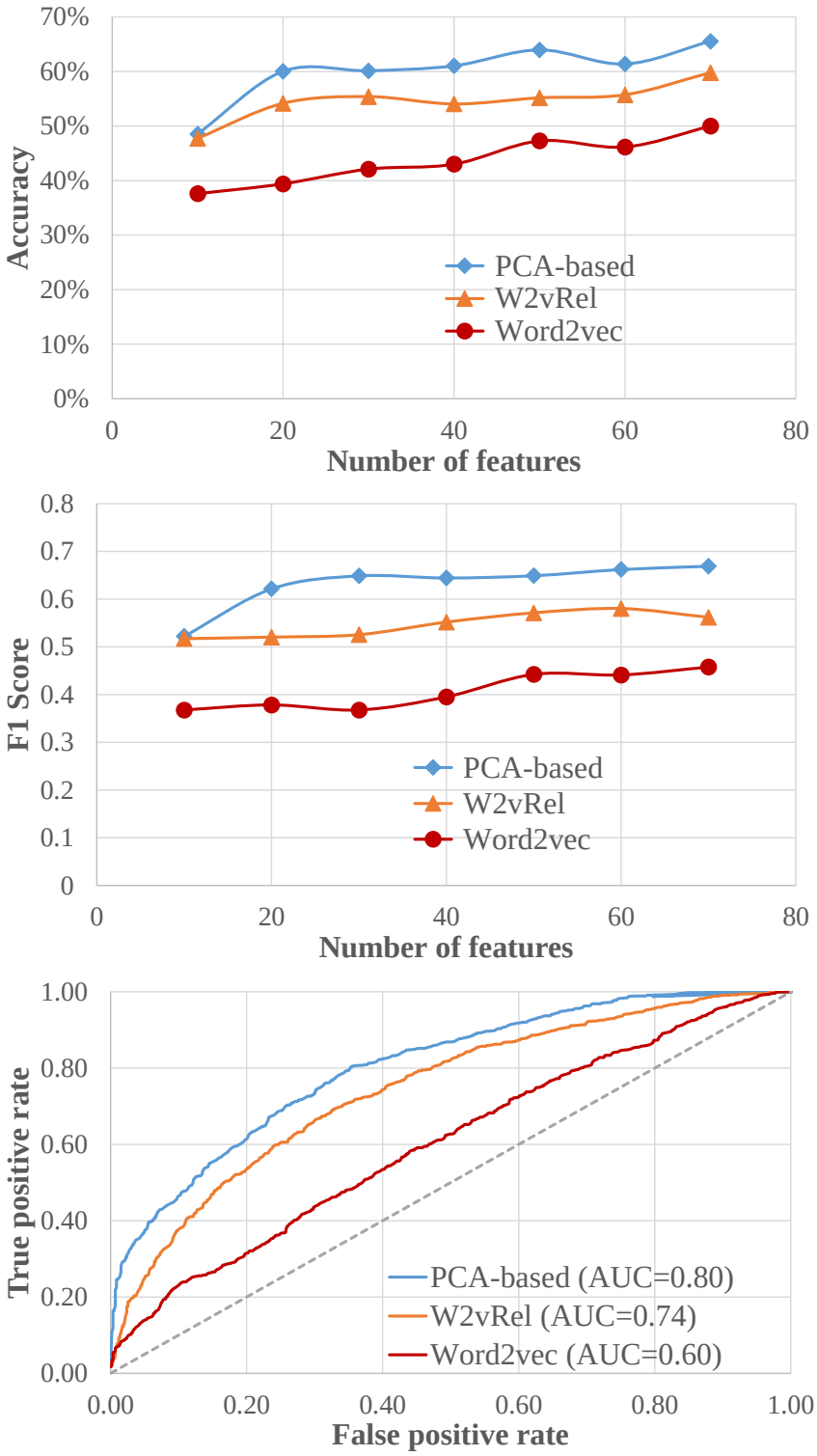


Figure 3.9: Accuracy, F1 score and ROC curve for classifying documents from 20newsgroups dataset based on the entity vectors produced by the methods.

Chapter 4

Synthesizing Front Facing Views of Faces

In recent times, we see a rapid growth of multimedia content in publicly available documents. It is highly likely nowadays that a news article, a Wikipedia page or a blog post will contain some kinds of multimedia contents, e.g., images, video and audio clip. Among other types of multimedia, image is the most prevalent one. The explosion of social media and its impact in electronic media introduces a new way of embedding user's content in publications such as news article. Nowadays, instagram posts and tweets are very commonplace in electronically published news articles. For example, Figure 4.1 shows an article published by CNN [1] in which a tweet with an image is embedded in the content of the article.

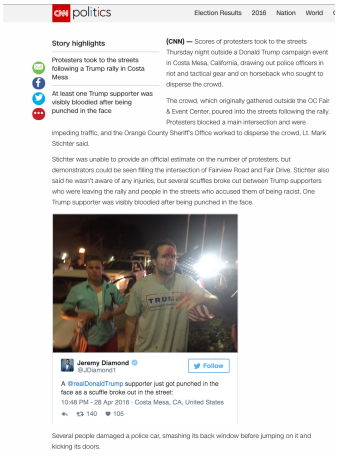


Figure 4.1: An embedded tweet in a news snippet.

The influence of social media content and the unconstrained nature of photography in news articles bring new challenges for the applications that require to analyze faces of people. Most of the faces appearing in news articles are not guaranteed to be front facing. For example, a screen-shot of CNN home page at a particular date is shown in Figure 4.2. Three persons are appearing in the Figure 4.2 showing three different orientation of their faces and none of them are front facing. Documents containing images of people are used

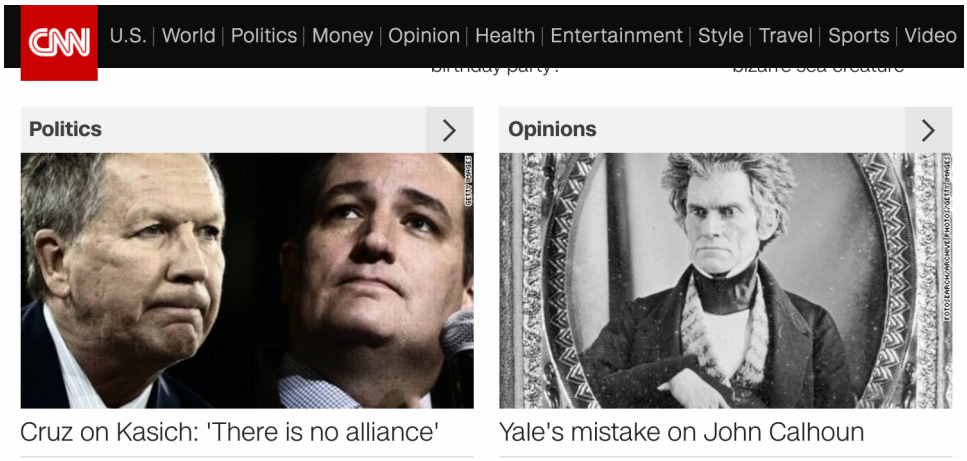


Figure 4.2: Three faces with three different orientations.

in a variety of applications, such as surveillance, situational awareness and disaster management. These applications require extraction of high-quality facial features. Extraction of effective features from the faces is a very important task since the quality of the features later dominates the performance of the applications that use faces. Popular feature extraction methods like Eigenface [123], Fisherface [75], and Local Binary Pattern [7] commonly used in face recognition perform reasonably well when most of the faces are front-facing. Since many of the faces in news articles are non-front-facing, we propose a novel frontalization method to be able to capture positions of some key facial points in a projected plane where the non-front-faced photo represents a front-posing face. This enables us to bring non-front-facing faces to a common space where all faces are considered front-facing. To the best of our knowledge, there is a related work proposed by Hassner *et al.* [48] that frontalizes only side-posed faces by cloning one side to other. Another limitation of the

work is the use of single 3D face model for all kinds of faces. In contrast, we estimate the 3D shape of every face in terms of facial key points and our method is capable of tackling any kinds of face orientations. Our observation is that the frontalized facial points can be used to generate complementary features to combine with other state-of-the-art features extraction methods such as a deep learning based face embedding framework FaceNet [103]. The combined features can be used to construct contextual information described in Chapter 5 for faces.

The contributions are as follows:

- We propose a facial key points frontalization technique that is capable to address all kinds of faces and their orientations.
- We demonstrate a technique to extract facial features from the frontalized facial key points.

4.1 Methodology

The proposed facial key points frontalization and feature extraction methods has four operational stages: (1) facial key points detection, (2) angle prediction for non-front facing faces, (3) frontalization of the facial points, and (4) feature generation from frontalized facial points. The following subsections describe all the stages in detail.

4.1.1 Face Detection and Feature Extraction

We leverage Convolutional Neural Network (CNN) based state-of-the-art face detection approaches [133, 76] to detect faces from a set of images. The face detection model is a deep cascade architecture built on convolutional neural networks. We use a pre-trained model that jointly performs face detection and alignment using multi-task cascaded convolutional networks [133].

Popular feature extraction methods like Eigenface [123], Fisherface [75], and Local Binary Pattern [7] commonly used in face recognition perform reasonably well when most of the faces are front-facing. Most of the images in a news corpus are not taken in a studio or laboratory environment and the detected faces are not always front facing. Recently, deep learning based face embedding approaches [103, 115] have started outperforming traditional facial feature extraction methods. We use a pre-trained model of FaceNet [103], a CNN-based face embedding framework, to extract face features in low-dimensional euclidean space. Since many of the faces in the images of datasets are side-facing, we use a frontalization method to be able to capture positions of some key facial points in a projected plane where the side-faced photo represents a front-posing face. Our frontalization method estimates a 3D face for each of the faces, whereas the technique in [48] assumes a single 3D face shape for all faces. We frontalize facial key points unlike [48] to be able to extract facial features from the angles of the facial points and distances between them. Frontalization of facial key points enables us to enhance the lower dimensional CNN-based embeddings by including the additional frontalization features.

4.1.2 Facial Key Points Detection

We detect five facial key points – two eye centers, nose and two mouth corners, as shown in Figure 4.3 – using a pre-trained cascaded deep convolutional neural network [116]. The model takes linear time in terms of the number of faces. It was trained using the LFW dataset [56]. The neural network utilizes texture information over the entire face and the geometric constraints among key points with high accuracy. Since a face can be in any pose in an image, we need to frontalize the facial key points to be able to extract the actual geometric properties of a face. Even before the frontalization, it is necessary to estimate the angles of a face by which it is deviated from a front facing position.

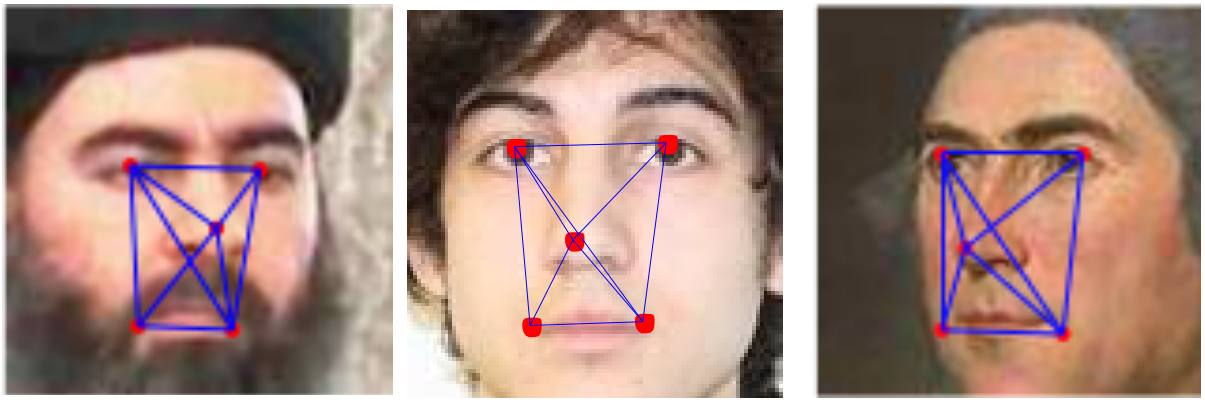


Figure 4.3: Detected eyes, nose and mouth corner points using Cascaded Deep Convolutional Neural Network for few faces.

4.1.3 Angle Prediction

We use Generalized Linear Model (GLM) to predict vertical and side angles of a face. We generate a synthetic dataset using six 3D-face models, which is created from six face images covering various ethnicities (e.g., Caucasian, Scandinavian, Japanese). Since the dimension of faces in the knowledge base, κ is 100×100 pixels, we imagine a cube of $100 \times 100 \times 100$ for the 3D face models where the center of head is at the origin $(0, 0, 0)$. Five facial key points of the faces are then marked on these models. We made three assumptions for simplicity in placing the five key points in the model.

1. Facial key points for eye pair and mouth corners are in the same plane, which is +35 unit ahead of and parallel to xz-plane.

2. Facial key point for nose is in plane $\mathcal{P}$ parallel to xz-plane and +15 ahead of the plane of mouth corners and eye pair.

3. Nose point $\pi_{nose} = (0, 50, 0)$ is fixed for all the models, but eye pair and mouth corner points are placed by maintaining relative distance of those points.

To create a synthetic training dataset of faces with known facial key points and known angles, we rotate the 3D models by varying angles around z-axis and x-axis and projecting

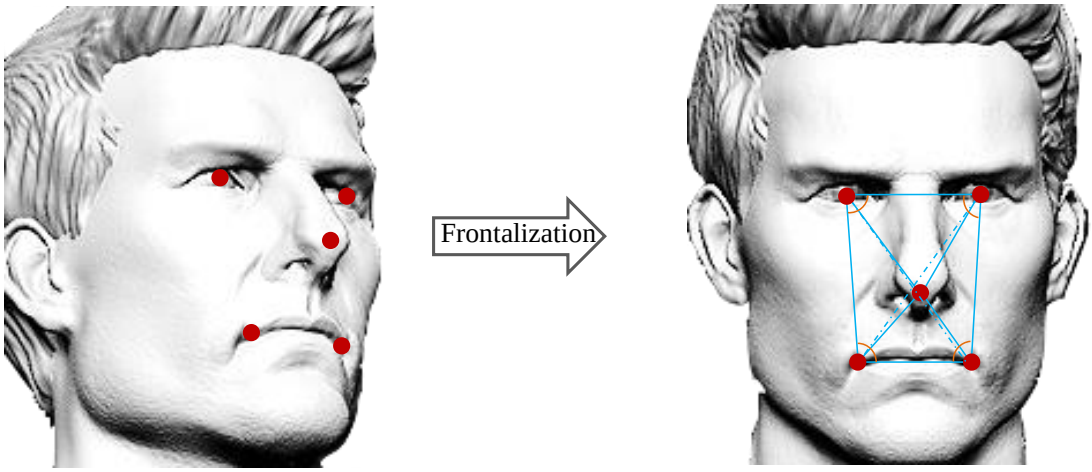


Figure 4.4: Facial key-points frontalization.

them back on a 2D xz-plane. We rotate the models from $-45°$ to $+45°$ at $3°$ intervals around z-axis and from $-15°$ to $+15°$ at $5°$ intervals around x-axis. This produces a training set of 1302 instances. Each instance consists of 30 features as described in **"Face Feature Generation"** part later. Two class labels are azimuth angle az around z-axis and elevation angle el around x-axis. We train two GLM models using these two sets of class labels. After the training, the models can predict the vertical and side angles for any five facial key points of a face. This enables us to apply frontalization by those predicted angles.

4.1.4 Facial Key Points Frontalization

We predict azimuth angle (side angles), az and elevation angle, el for each detected face by using GLMs from the extracted facial key points. Let $\rho = \{\rho_1, \rho_2, ..., \rho_5\}$ be the set of five facial key points extracted from a face. Algorithm 1 describes the steps for frontalizing the set of key points ρ. The algorithm uses a function *Rotate* that rotates a 3D point around a particular axis by a certain angle. Figure 4.4 shows a sample of frontalization applied on five facial key points.

Algorithm 1: Facial Key Point Frontalization

Input : $\rho, \pi_{nose}, \mathcal{P}, az, el$

Output: $\rho^{frontalized}$

$\mathcal{P}' \leftarrow$ Plane $\mathcal{P}$ rotated by az around z-axis then by el around x-axis

$\pi'_{nose} \leftarrow Rotate(Rotate(\pi_{nose}, axis = z, az), axis = x, el)$.

for all $\rho_k \in \rho$ **do**

  $\rho'_k \leftarrow \rho_k + (\pi'_{nose_x}, \pi'_{nose_z}) - \rho_3$

  $L_k \leftarrow$ Line perpendicular to xz-plane going through ρ'_k

  $t_k \leftarrow$ Intersecting point between the plane $\mathcal{P}'$ and the line L_k

  $t'_k \leftarrow Rotate(Rotate(t_k, axis = x, el), axis = z, az)$

  $\rho_k^{frontalized} \leftarrow t'_k$ orthogonally projected on xz-plane

end for

return $\rho^{frontalized}$

4.1.5 Face Feature Generation from Frontalized Points

An important criterion in building a good feature set of a face is that the produced features vary little for a particular person in different lighting conditions, expressions, and occlusions. Our method for facial key point generation and frontalization aid in generating features with such properties. We generate two sets of features from the frontalized facial key points. The first set is composed of pairwise relative distances between all five facial key points and the second set comprises of twenty angles as shown in Figure 4.5. Combination of these two sets of features makes a feature vector of length thirty where the first ten are the pairwise relative distances and the last twenty are the angles.

In addition to these thirty features, we leverage a pre-trained convolutional neural network based face embedding technique [103] that produces 128 dimensional embedding for each face. We did not consider other embedding dimensionalities because [103] demonstrates that the optimal embedding dimension is 128. Previously extracted thirty features using frontalized key points are concatenated with this 128 dimensional embedding forming

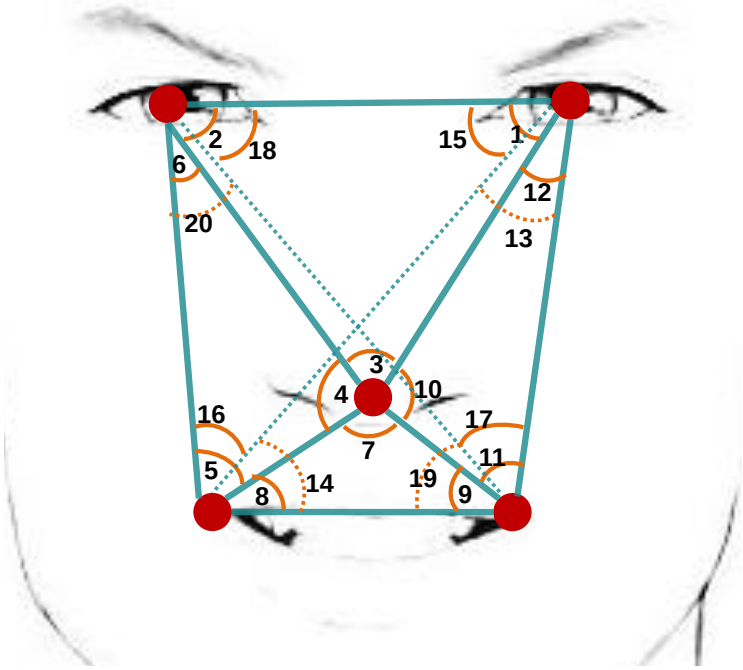


Figure 4.5: Twenty angles generated from frontalized five facial key points.

a vector of length 158 for each face.

4.2 Experimental Results

The specific questions we seek to answer in this section are:

1. Does frontalization of key points result in better face recognition accuracy? (Section 4.2.1)

2. Which method is more capable of detecting facial key points for this particular data set? (Section 4.2.2)

3. Which learning mechanism we should use to attain greater performance in predicting the proper rotation angles for frontalization? (Section 4.2.3)

We use New York Times news articles for our experiments. The dataset contains 54,371 articles and 86,966 images with around 98,914 faces and 69,829 entities. All these news

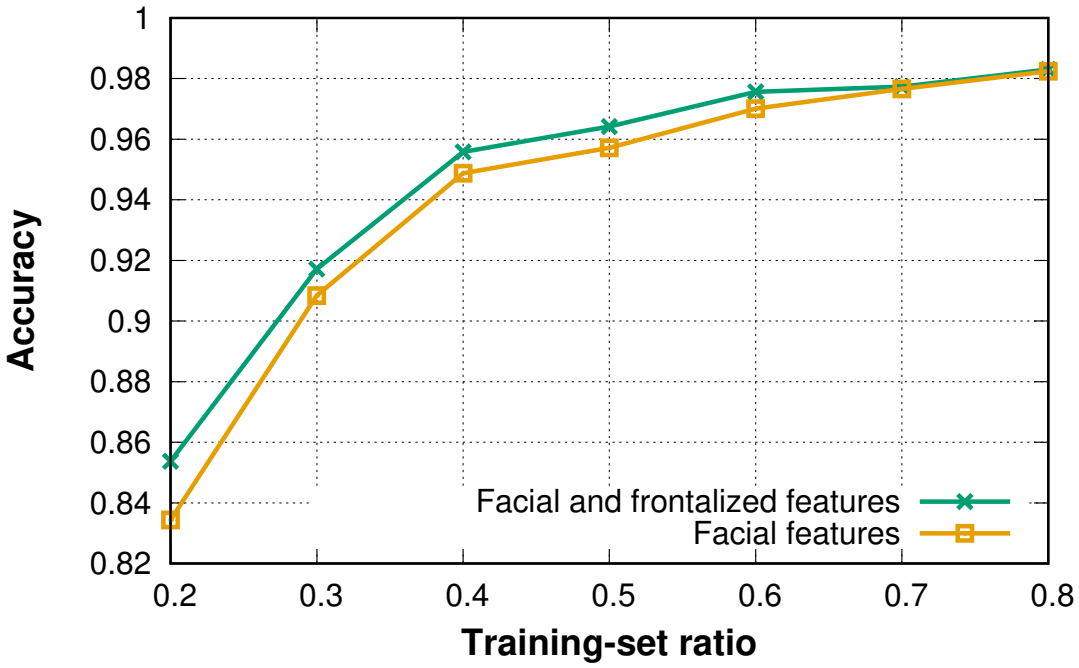


Figure 4.6: Comparison of face recognition accuracies with and without Frontalization technique.

articles are time stamped. For all the classification based experiments, we used logistic regression, one-vs-rest when appropriate, with L2-regularization, and 10-fold cross-validation unless mentioned explicitly.

4.2.1 Frontalization for Face Recognition

Although the main target of the frontalization is to help context generation methods described in Chapter 5, we evaluate the frontalization technique using face recognition to study its impact. In most face recognition literature, ground truth faces of each people are taken in different poses under the same environment (e.g., illumination). However, faces detected from the New York Times images are not annotated and limited in number and poses. The experiment in this section compares face recognition accuracies with and without frontalization. We picked 5,690 faces of 401 persons from the Labeled Faces in

the Wild (LFW) dataset [56], which has annotated face images that are not taken in a controlled environment. We picked only those persons for whom at least five face images were available so that we can experiment using different training-test ratio. We used an one-vs-rest logistic regression with L2-regularization. Figure 4.6 compares face recognition accuracies at different training and test splits with and without frontalization. The figure shows that inclusion of frontalized features with facial features yields better accuracy. The Figure 4.6 also depicts that our frontalization technique has a greater impact on face recognition when training data is limited. On average, the standard deviation of the data for each generated point of Figure 4.6 was less than 0.005.

4.2.2 Facial Key Point Selection

Detection of an appropriate number of key points is challenging because of different poses faces can have. We applied two methods, Boosted Regression with Markov Networks (BoRMaN) [124] and a Deep Convolutional Network Cascade (CNN) [116] method, for facial key point extraction. The former discovers twenty facial key points and the later finds five. Our observation is that the BoRMaN method performs well with faces taken under controlled environment, e.g., photos taken in laboratories. When we used detected faces from our dataset, the BoRMaN method was able to detect facial points properly for 70% of the faces. For this experiment, we randomly picked up 300 faces from our database and manually checked if the points detected are in the vicinity of the expected pixels. With the same 300 faces, the accuracy of the CNN method in detecting five facial key points was 99%. Figure 4.7 shows two examples where five key points are detected correctly by CNN but the twenty points detected by the BoRMaN method are cluttered in one region of the face.

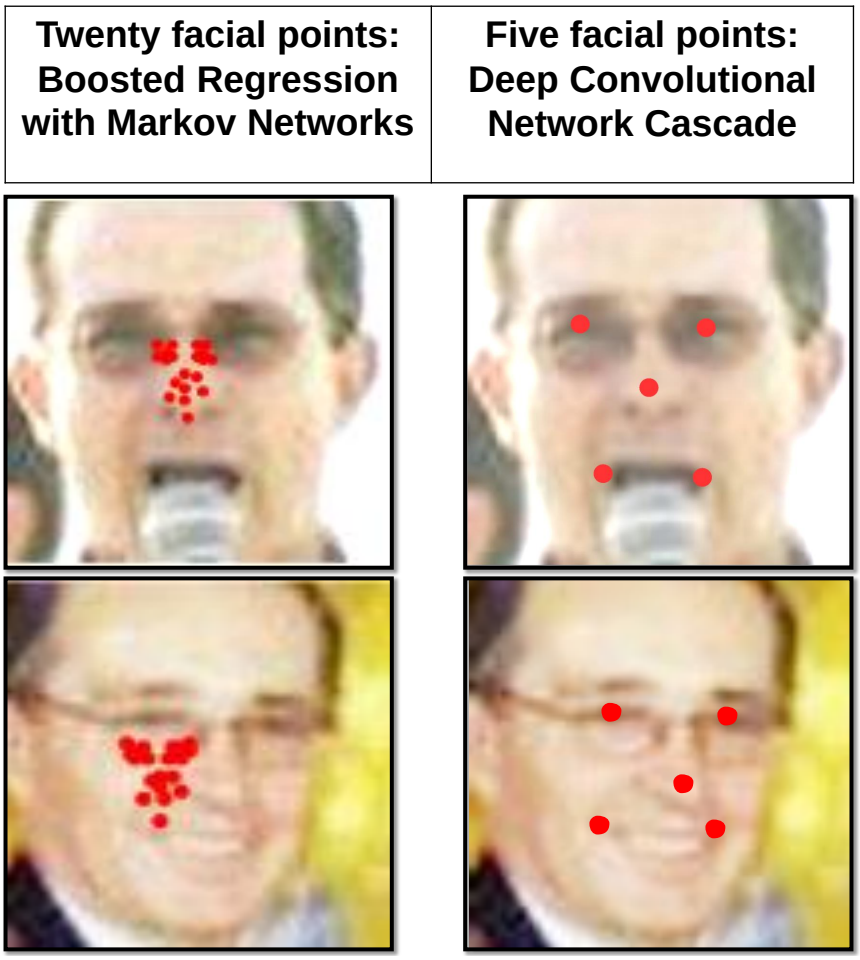
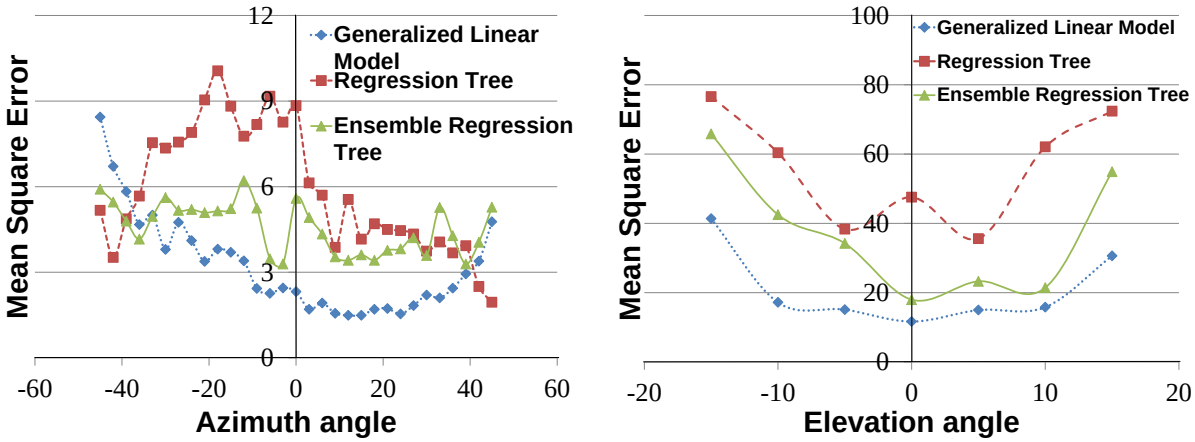


Figure 4.7: Comparison of two facial key point detection techniques.

4.2.3 Prediction of Frontalization Angles

Our frontalization technique relies on a generalized linear model based classifier to compute the azimuth and elevation angles of a face in an original image. To compare the linear model based classifier with a few other alternative mechanisms, we created a synthetic face model and rotated it in different angles to create different poses. That is, the ground truth angles are known for all the poses and an error can be computed for prediction of those angles. Figure 4.8 shows a comparison of mean square errors using three methods: Generalized Linear Model (GLM), Regression Tree and Ensemble Regression Tree to predict rotation angles. Figure 4.8(a) is for sideway rotations and Figure 4.8(b) shows the errors with

elevation of the faces. In both the cases, GLM has lower errors than any other method at most of the angles. In this experiment, the sideway angles were varied from $-45°$ to $+45°$ and the elevations were varied from $-15°$ to $+15°$.



(a) (b)

Figure 4.8: Mean square errors of (a) azimuth angle predictors and (b) elevation angle predictors.

4.3 Summary

This chapter presents a novel facial key points frontalization technique that complements other state-of-the-art methods for effective facial feature extraction. This method uses 3D surface as an approximation to the shape of all faces and rotates the facial key points after placing them on the 3D surface. Experimental results show that our facial feature extraction method helps to improve the face recognition performance for the labeled faces in the wild.

Chapter 5

F2ConText: How to Extract Holistic Contexts of Persons of Interest

Building a mental model and establishing contextual phenomena is central to many exploratory analysis work, especially for improved situational awareness [49]. Data analysts face many challenges while fusing disparate streams of data and rapidly prototyping event-scenarios for quantitative predictions to help policy makers arrive at analytical conclusions [28]. Publicly available imagery data and textual information are widely leveraged during rescue missions, disaster management, surveillance, and in other scenarios to gather insights for making informed decisions. Although there are existing systems to aid exploratory analysis (e.g., see [58, 86, 85]), the growing volume of public data feeds and the evolving demand of analytical capabilities necessitate further aide in situational awareness.

David Cameron

Person Context	Location Context
david	**kingham**
david_cameron	**britain**
queen_elizabeth_ii	**sarsden**
samantha_cameron	**medieval_britain**
shaun_bailey	peasemore
david and samantha	**unin_europea**
nick_houghton	**daylesford**
shaun_bailey	hot_shot
george_zambellas	**windsor_hockey_club**
phil_redmond	speen

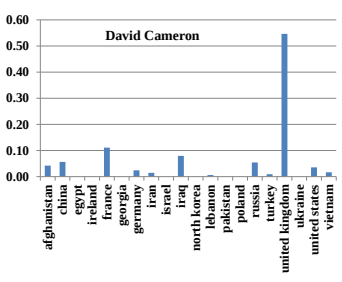


Figure 5.1: Generated contexts of a face. From left to right: the face image for which contexts are generated, person context, location context, geographical context using a bar chart, and geographical context laid on a map. The geographical context shows that the United Kingdom has the highest probability for the British Prime Minister, David Cameron.

This chapter presents a framework called F2ConText (*Face to Context using Text*) that helps analysts build contextual templates for persons of interest using co-occurring images and textual data. A contextual template is composed of mappings between faces, names, geographical locations, and many other entities. While developing a contextual sense about events and persons of interest is a natural process for human beings, automatic generation of context using publicly available data to aid the analytic process is still a challenge due to the massiveness and multimodal nature of the datasets. Moreover, the heterogeneous feature elements that create a meaningful context are not well defined in most publicly available datasets. For example, the Wikipedia entry for David Cameron, the former Prime Minister of United Kingdom, contains a few faces of his supporters whose contexts are not related to the use of his free time described in the page.

The images in publicly available news articles do not have labeled faces as found in many social media photos. With news archives containing hundreds of thousands of articles and images, analysts cannot label all the faces in the images manually. Additionally, many of the faces in the news images are of unknown people that may appear with other known persons of interest. For example, the names of the security personnel of the prime minister of a country may never appear in any news article but the faces of the security team may be seen with the country's prime minister in an image of a news article. Theoretically, the prime minister and a security staff will have the same context if the face of the security staff appears in all the images where the prime minister appears. In reality, the prime minister's face will appear in more images than the face of a particular security person. This indicates that the context of the prime minister will span more articles than the context of a security staff, which will result in similar but ultimately different contexts. The foundation of our system to generate context for faces is driven by this concept.

Situational awareness requires harnessing high quality contextual information regarding location of people of interest. In many instances the location information of a known person is apparent from her/his social presence in media like linkedin, twitter, and facebook. Unfortunately, faces of people in images of news articles are not name-tagged. Moreover,

the location context of a person is beyond the name of the current location or birth place of the person. A person of one country may be discussed well in another country. The geographical context can be the distribution of degree of association of a person with all locations. This chapter describes a methodology to generate a geographical context at country level for every face detected in every image of a news corpus. The benefit of the ability to generate country-level geographical context is two-fold. First, the geographical context answers the "where" question during an analysis relevant to insurgency or any event of interest. Second, a geographical context is traceable over time, which allows analysts to study how focus of a person of interest change over time and over locations. The proposed solution does not rely on any coordinate data, rather it generates a geographical context for every face image using news content, and a publicly available city and country list. The experiments discussed in this chapter exclusively use open source and publicly available data.

5.1 Associated Analytic Challenges

Most of the existing analytic tools (e.g., Entity Workspace [12], Jigsaw [112], NetLens [63], and Sentinel Visualizer [41]) require development of context in the mind of the analyst through extensive manual exploration of the data and connection building effort. Challenges grappled by analysts associated with contextual analysis of people of interest using image and text data are outlined below:

Challenge 1: *Limited or no knowledge about the images:* Associated images in news articles and many other public feeds do not have labeled information. Lack of labels and meta-data makes querying difficult, which results in manual effort of tagging the faces with textual comments and remembering images for future use.

Challenge 2: *Lack of mappings between image and text:* Intelligence analysts struggle in connecting text granules with images in presence of documents containing both text and images. While existing software tools help detect entities (e.g., person and organization)

from text [10, 8], detect faces in images [126], and extract facial features from the detected faces [123], the task of mapping the extracted faces to text granules to provide a sense of context is still unsolved.

Challenge 3: *Connecting content-driven context to location:* For many decision making tasks, analysts are required to represent the contexts discovered from the content of the documents in a different space. For example, the location entities found in a document might not always be countries, rather the locations can be region names, towns, cities, or even some organizations that are representative of some areas. Additionally, the same region name can be present in multiple countries. How can an analyst quickly retrieve a geographical context at country level to disambiguate locations or to down scope the analysis to a particular part of the globe?

Challenge 4: *Lack of support for contextual grouping of people of interest:* Detecting groups of people with similar activities or context containing suspicious entities of interest is key to many intelligence analysis tasks in order to reveal latest social associations [45]. Lack of software support to detect such contextual grouping of potential pool hampers generation of high quality intelligence in a timely manner.

Challenge 5: *Lack of support to study evolving nature of context:* Another limitation is the lack of algorithmic support to assist analysts in coping with the changing nature of the reasoning tasks they routinely tackle [119]. In the space of contextual analysis, reasoning requires extensive understanding of how each context evolves over time. Current literature lacks such context-tracking mechanism.

Challenge 6: *Complex nature of the diffusion of events:* Detection of the evolution of the context of an entity provides the ability to track individual contexts. However, the study of an event is more complex than the analysis of a specific context because an event is a composition of interactions between many individuals. The rapid growth of textual and imagery data makes it quite challenging for analysts to trace the genealogy of all actors involved in an event of interest.

5.1.1 Contributions

The challenges outlined above motivate our context generation mechanism. Specific contributions are as follows:

- Our framework leverages a mechanism to enhance facial features by complementing state-of-the-art techniques. These features are later connected to entities using a probabilistic model to avoid manual labeling.

- We formulate and solve the problem of holistically mapping faces to textual entities (e.g., person and location) to build a context for each face detected in the images of a news archive.

- F2ConText generates geographical context at the country level for each face found in each of the images of a news archive. None of the state-of-the-art methods to generate geographical context has the ability to compute such a nontrivial mapping between faces and geography.

- We demonstrate that the generated contexts help identify meaningful contextual clusters of faces.

- We demonstrate that a geographical context generated by our framework is traceable over a time-line. This allows analysts reason how a geographical context of a certain image may evolve over time.

- We introduce a new event summarization mechanism that leverages text and images to explain diffusion and evolution of events as chains of documents. The proposed method traverses a similarity network of news articles without materializing the network entirely and by constraining consecutive documents with certain cohesion threshold, context overlap requirement, and temporal ordering.

Figure 5.1 shows a sample contextual template generated by F2ConText. The figure shows the person context, the location context, and the generated geographical context of

the British Prime Minister, David Cameron.

5.2 Problem Formulation

Let $\kappa = \{\mathcal{A}, \mathcal{E}, \mathcal{I}, \mathcal{F}, R_{AE}, R_{AI}, R_{IF}\}$ be a collection of articles containing images $\mathcal{I} = \{i_1, i_2, \ldots, i_{|\mathcal{I}|}\}$ and textual descriptions $\mathcal{A} = \{a_1, a_2, \ldots, a_n\}$. Text descriptions may contain entities from $\mathcal{E} = \mathcal{E}^N \cup \mathcal{E}^L = \{e_1, e_2, \ldots, e_{|\mathcal{E}|}\}$ where $\mathcal{E}^N$ is the set of person entities and $\mathcal{E}^L$ is the set of location entities. $\mathcal{F} = \{f_1, f_2, \ldots, f_{|\mathcal{F}|}\}$ is the set of faces extracted from images $\mathcal{I}$. R_{AE} represents entities within articles, $\{\{a_q, e_r\} : a_q \in \mathcal{A}, e_r \in \mathcal{E}\}$. Similarly, R_{AI} is the set of relationship $\{\{a_q, i_r\} : a_q \in \mathcal{A}, i_r \in \mathcal{I}\}$ and R_{IF} is the set of relationships $\{\{i_q, f_r\} : i_q \in \mathcal{I}, f_r \in \mathcal{F}\}$ representing images within articles and faces of people within images, respectively. The task of context generation in F2ConText is two pronged. For each face $f \in \mathcal{F}$,

1. generate a person context $\mathcal{C}^N(f) = \{\psi_1(f), \psi_2(f), \ldots, \psi_{|\mathcal{E}^N|}(f)\}$. $\psi_r(f)$ is a tuple $\{\{f, e_r, P(e_r|f)\} : f \in \mathcal{F}, e_r \in \mathcal{E}^N)\}$, where $P(e_r|f)$ is the probability of a person entity e_r given a face f. In practice, $\mathcal{C}^N(f)$ is arranged in descending order and records only a feasible number of probabilities. Similar to the person entity context, generate the location context $\mathcal{C}^L(f) = \{\chi_1(f), \chi_2(f), \ldots, \chi_{|\mathcal{E}^L|}(f)\}$ where $\chi_q(f) = \{\{f, e_q, P(e_q|f)\} : f \in \mathcal{F}, e_q \in \mathcal{E}^L\}$ and $P(e_q|f)$ is the probability of location entity e_q given a face f.
2. generate a geographical context, $D(f) = \{d_1(f), d_2(f), \ldots, d_m(f)\}$, as a probability distribution of m countries.

5.3 Methodology

F2ConText uses three operational stages to generate contexts for faces: (1) feature extraction and modeling, (2) generation of entity based contexts — person context ($\mathcal{C}^N(f)$) and location context ($\mathcal{C}^L(f)$), and (3) construction of geographical context, $D(f)$, as a

Table 5.1: List of frequently used symbols

Symbol	Description
κ	Collection of articles containing images and text
$\mathcal{A}$	Set of textual documents
$\mathcal{I}$	Set of images
$\mathcal{F}$	Set of faces
$\mathcal{E}$	Set of entities
$\mathcal{C}^N(f)$	Person context for face f
$\mathcal{C}^L(f)$	Location context for face f
$D(f)$	Geographical context for face f
$D^1(f)$	Geographical context for face f based on person name entities
$D^3(f)$	Geographical context for face f based on location entities
$D^T(f)$	Ground truth geographical context for face f
$D^B(f)$	Geographical context for baseline method for face f
λ	User settable parameter for adjusting sensitivity of entity level context
θ	Maximum allowed document distance for finding story
τ	Maximum allowed distance for face context between documents

probability distribution over all countries.

The following subsections describe these computational stages. For the convenience of the readers, we provide a list of most frequently used symbols in Table 5.1.

5.3.1 Features Extraction and Modeling

F2ConText requires extraction of features from both images and texts that coexist in the documents of a collection.

Facial Feature Generation in F2ConText:

To detect faces from the set of images $\mathcal{I}$, we use Convolutional Neural Network (CNN) based face detection approaches [133, 76]. A pre-trained model of [133] is used to jointly performs face detection and alignment using multi-task cascaded convolutional networks. As the face features, we use the concatenation of facial features extracted using our technique in Chapter 4 and face embedding produced by a pre-trained model of FaceNet [103], a CNN-based face embedding framework.

Entity Extraction and Document Modeling:

F2ConText combines the outputs of a number of entity extractors including LingPipe [8], OpenNLP [10], and Stanford NER [110] to identify entities within the textual contents of the articles in $\mathcal{A}$. Although we extracted all standard entity types including person name, organization, and location, this work scopes down the analysis to person and location entities only, especially because the images are explained using detected human faces. The weight of person-name entity $e \in \mathcal{E}^N$ in the article $a \in \mathcal{A}$ is computed as:

$$W_N(e,a) = \frac{(1 + log(tf_{e,a}))(log\frac{|\mathcal{A}|}{af_e})}{\sqrt{\sum_{e' \in \mathcal{E}_a^N} \Big((1 + log(tf_{e',a}))(log\frac{|\mathcal{A}|}{af_{e'}})\Big)^2}} \tag{5.1}$$

where $tf_{e,a}$ is the frequency of entity e in article a, af_e is the number of articles containing a connection with entity e, and $\mathcal{E}_a^N$ is the set of person entities that are connected to article a. Equation 5.1 is a variant of TF-IDF modeling with cosine normalization [78]. TF-IDF stands for Term Frequency-Inverse Document Frequency. It is a term weighing mechanism intended to reflect the importance of a term in a document within a corpus. The articles in $\mathcal{A}$ have descriptions of different sizes. In general, longer descriptions have higher term frequencies because many terms are repeated. The cosine normalization helps lessen the impact of size of the descriptions in the modeling. The Weight, $W_L(e,a)$, of a location entity $e \in \mathcal{E}^L$ in an article $a \in \mathcal{A}$ is calculated using the same formula as Equation 5.1.

5.3.2 Generation of Entity based Context

F2ConText generates two separate contexts, one using person name entities and the other using location entities, for each face $f \in \mathcal{F}$. The person context $\mathcal{C}^N(f)$ of a face f is expressed as a probability distribution over the set of person entities $\mathcal{E}^N$. Similarly, the location context $\mathcal{C}^L(f)$ is expressed as a probability distribution over the set of location entities $\mathcal{E}^L$. Since the mechanism to generate person context $\mathcal{C}^N(f)$ and location context $\mathcal{C}^L(f)$ are similar, we present the process for computing person context only.

Using Bayes' rule, the probability of an entity $e \in \mathcal{E}^N$ for each given face f can be expressed as

$$P(e|f) \propto P(e) \times P(f|e) \tag{5.2}$$

where $P(e)$ is the prior probability of e and $P(f|e)$ is the likelihood. Let $f = \{f^1, f^2, \ldots, f^V\}$ be the feature representation of face f obtained by the method described in Section 5.3.1. V is the length of the feature vector of f. With an assumption of independence between the features, we can rewrite Equation 5.2 as

$$P(e|f) \propto P(e) \times \prod_{l=1}^{V} P(f^l|e) \tag{5.3}$$

A person entity e can appear in multiple articles of $\mathcal{A}$. Let $\mathcal{A}^e \subseteq \mathcal{A}$ be the set of articles that contain entity e. Each article $a \in \mathcal{A}^e$ may in turn contain a number of faces (as expressed by R_{AI} in κ). Let $\mathcal{F}^a \subseteq \mathcal{F}$ be the set of faces in article a. The relationships between the face-features and person-name entities in articles can be computed by the entity weights, $W_N(e, a)$ and face-feature weights in the articles. The likelihood of the lth feature of a face given an entity e can be computed as

$$P(f^l|e) = \left(\frac{\sum_{a \in \mathcal{A}^e} P(f^l|a) \times W_N(e, a)}{\sum_{a \in \mathcal{A}^e} W_N(e, a)} \right)^{f^l} \tag{5.4}$$

where $P(f^l|a)$, which is calculated by Equation 5.5, is the probability of the face feature f^l given article a.

$$P(f^l|a) = \frac{\sum_{\phi \in \mathcal{F}^a} \phi^l}{\sum_{\phi \in \mathcal{F}^a} \sum_{l'=1}^{V} \phi^{l'}} \tag{5.5}$$

where ϕ^l is the lth feature of a face $\phi \in \mathcal{F}^a$. Now, replacing $P(f^l|e)$ of Equation 5.3 by this expression we obtain

$$P(e|f) \propto P(e) \times \prod_{l=1}^{V} \left(\frac{\sum_{a \in \mathcal{A}^e} P(f^l|a) W_N(e,a)}{\sum_{a \in \mathcal{A}^e} W_N(e,a)} \right)^{f^l} \tag{5.6}$$

Taking logarithm on both sides of Equation 5.6:

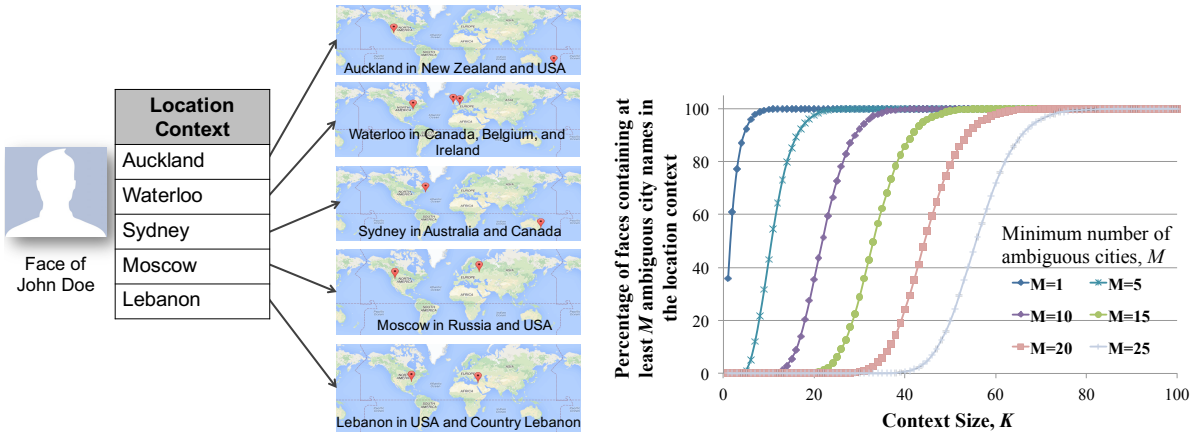
$$\begin{aligned} \log(P(e|f)) \propto \log(P(e)) &- \sum_{l=1}^{L} \left(f^l \times \log \left(\sum_{d \in D^e} W(e,d) \right) \right) \\ &+ \sum_{l=1}^{L} \left(f^l \times \log \left(\sum_{d \in D^e} P(f^l|d) \times W(e,d) \right) \right) \end{aligned} \tag{5.7}$$

We generate context for each face using Equation 5.7, which produces a probability distribution over all entities for each face f. In practice, we do not record the full probability distributions, rather we keep record of a maximum of L_C entities with highest probabilities as the context of a face.

5.3.3 Geographical Context Generation

Notice that the location context $\mathcal{C}^L(f)$ is generated leveraging the same method used to generate the person context $\mathcal{C}^N(f)$. Analysts generally have deep knowledge about the actors of relevant events but location entities are difficult to interpret because they might contain village, city, county, or even community names. To aid an analyst with a more abstract sense of location context, our framework generates a geographical context in the form of a country distribution, which demonstrates prominence of countries as seen in a document.

Our observation is that The New York Times dataset has more than 85% articles containing at least three ambiguous city names. As a result of such ambiguities, the location entity based context is not sufficient to generate a geographical context. Figure 5.2(a) shows that each of the prominent location entities related to John Doe can be found in multiple



(a) Location context of a person.

(b) Location ambiguity in the location-context of the faces from NY Times dataset.

Figure 5.2: Location ambiguities in dataset.

countries. A closer look on the maps reveal that John Doe's geographical context is more focused on North America. Figure 5.2(b) depicts high number of ambiguous locations in the location contexts of the detected faces of NY Times dataset.

To address the location ambiguity problem, our framework generates a template of probability distribution $D(f)$ of countries for each face $f \in \mathcal{F}$ by combining entity level contexts $\mathcal{C}^N(f)$ and $\mathcal{C}^L(f)$. This probability distribution $D(f)$ is the geographical context generated for each face. To identify ambiguous locations, the framework uses a publicly available database [44] that maps all city names with countries.

Generation of Geographical Context, $D(f)$: Let $\Phi = \{\phi_1, \phi_2, \ldots, \phi_m\}$ be the set of m countries, ρ be the set of all cities, and ρ_{ϕ_i} be the set of all cities in country ϕ_i. The country distribution of a face $f \in \mathcal{F}$ using the person entity based context $\mathcal{C}^N(f)$ alone can

be defined as:

$$D^1(f) = \{d^1_{\phi_1}(f), d^1_{\phi_2}(f), ..., d^1_{\phi_{|\Phi|}}(f)\}, \text{ where}$$
$$d^1_{\phi_i}(f) = \sum_{e \in \mathcal{E}^N} \left[P(\phi_i|e) \times \exp\left(-\lambda \frac{\max\limits_{e' \in \mathcal{E}^N}(LL(e', f)) - LL(e, f)}{\max\limits_{e' \in \mathcal{E}^N}(LL(e', f)) - \min\limits_{e' \in \mathcal{E}^N}(LL(e', f))} \right) \right] \tag{5.8}$$

where λ is a user settable parameter vary the number of top entities to consider for analytic purpose. Larger values of λ results in lesser number of entities.

The country distribution of f using location context is:

$$D^2(f) = \{d^2_{\phi_1}(f), d^2_{\phi_2}(f), ..., d^2_{\phi_{|\Phi|}}(f)\} \tag{5.9}$$

where $d^2_{\phi_i}(f)$ is computed using the same formula as $d^1_{\phi_i}(f)$ with the only exception that $d^2_{\phi_i}(f)$ uses the location entities and location contexts instead of person entities and contexts.

In practice, the entity recognizers do not realize location tokens with 100% accuracy. To reduce the impact of erroneous locations we define another distribution, $D^3(f)$, by validating the existence of the location context entities in the database of cities and countries. $D^3(f)$ is defined by:

$$D^3(f) = \{d^3_{\phi_1}(f), d^3_{\phi_2}(f), ..., d^3_{\phi_{|\Phi|}}(f)\}, \text{ where}$$
$$d^3_{\phi_i}(f) = \sum_{e \in \mathcal{E}^L \cap (\rho \cup \Phi)} \left[P(\phi_i|e) \times \exp\left(-\lambda \frac{\max\limits_{e' \in \mathcal{E}^L \cap (\rho \cup \Phi)}(LL(e', f)) - LL(e, f)}{\max\limits_{e' \in \mathcal{E}^L \cap (\rho \cup \Phi)}(LL(e', f)) - \min\limits_{e' \in \mathcal{E}^L \cap (\rho \cup \Phi)}(LL(e', f))} \right) \right] \tag{5.10}$$

Probability, $P(\phi_i|e)$, is computed by

$$P(\phi_i|e) = \frac{\eta_e^{\phi_i}}{\eta_e} \tag{5.11}$$

where $\eta_e^{\phi_i}$ is the number of articles containing both country ϕ_i and entity e. η_e is the number of articles containing e.

Notice that $D^3(f)$ is an improved version of $D^2(f)$ that resolves errors of location entity detection. Therefore, the final composition of the geographical context is:

$$D(f) = ln(D^1(f)) + ln(D^3(f)) \tag{5.12}$$

Equation 5.8, 5.10 and 5.12 produce country probability distribution for a face f based on person context, location context and a composition of Equation 5.8 and 5.10, respectively. In Section 5.4, we show a comparison of the effectiveness between several combinations of $D^1(f)$, $D^2(f)$, and $D^3(f)$.

5.3.4 Complexity Analysis

Generation of the entity-based context using Equation 5.7 for all faces costs $\mathcal{O}(|\mathcal{F}| \times N_d \times |\mathcal{E}|)$, where N_d is the average number of documents associated with each entity. Since the face features are pre-computed and of fixed length, we consider that the length of the features is a constant. For every pair of face and entity, Equation 5.7 requires N_d repetitions. To generate the geographical context for every face in $\mathcal{F}$, Equations 5.8, 5.10 and 5.12 need to iterate over all the entities for each country in the world. The geographical context generation (Equations 5.8, 5.10 and 5.12) for all faces has a time complexity of $\mathcal{O}(|\mathcal{F}| \times |\Phi| \times |\mathcal{E}|)$, where Φ is the set of all countries in the world.

5.3.5 Analytical Task Extensions

The geographical and entity based contexts have a wide variety of applications in exploratory analysis. In this section, we present three extensions of the framework to demonstrate different capabilities of the framework: (a) finding genealogy of events, (b) tracking geographical context of people over time, and (c) discovering contextual clusters of faces. These extensions are described below. Empirical studies on all the extensions are provided in Section 5.4.

Finding Genealogy of Event

We investigate the potency of the entity-level contexts of faces by introducing a new mechanism to summarize events that evolve over time. The purpose of time-evolving summarization of an event is to give the analyst an overview as a chain of documents with

Figure 5.3: The story contains five news documents associating Boston bombers' involvement in the Waltham triple murder. The story includes trial phase.

accompanying faces relevant to the documents. An example of the summarization of an event is shown in Figure 5.3. The figure explains the aspects of the Boston Marathon Bombing tragedy. Figure 5.3 is further explained later in Section 5.4.8.

The summarization task focuses on forming a chain of {article, face-set} pairs using a set of news documents $\mathcal{D}$ and the generated face contexts using knowledge base created from Wikipedia articles. Summarization requires extraction of entities from $\mathcal{D}$ to form $\mathcal{E}^{\mathcal{D}}$, discovering faces relevant to each new document based on similarity between a news document and context of images in the knowledge base, and designing a path-finding algorithm in a similarity network of news documents where discovered paths are constrained by text coherence, context similarity, and progression of time in the story. While in reality news documents may contain images but many of those images are repeated from the past to provide a visual context. Many of the news documents do not contain any image. Since the knowledge base covers broad aspects of everything, face images can be reproduced from the images of the knowledge base. In our set of news documents, $\mathcal{D}$, we considered that there is no image. We reproduce relevant faces for each document of a story using the knowledge base. We order faces for each news document $d \in \mathcal{D}$ based on relevance between d and all faces. We compute this relevance using a context matching score between a news document d and the context $C(f)$ of a face f:

$$\beta(f,d) = \sum_{e \in \mathcal{E}_d^{\mathcal{D}} \cap C(f)} V(e,d) \times (|C(f)| - R(e,f)) \tag{5.13}$$

where V is a function similar to W defined in Equation 5.1. The weights of V are computed over news documents of $\mathcal{D}$. $R(e,f)$ is the rank of e in the list of entities $C(f)$ of face f. That is, we have two kinds of information associated with each news document. One is the weighted list of entities extracted from the text of the document and the other type is the most relevant faces (or contexts) from the knowledge base. Both the types are leveraged in a heuristic search algorithm to build a path of {article, face-set} pairs between two documents $\{d_s, d_t\} \in \mathcal{D}$. We use a variation of A* search algorithm that uses Soergel distance as the heuristic. Soergel distance is an admissible heuristic for A* search. The Soergel distance between two documents d and d' is calculated using the following formula.

$$SrgDist(d,d') = \sum_{e \in \mathcal{E}_d^{\mathcal{D}} \cup \mathcal{E}_{d'}^{\mathcal{D}}} \frac{|V(e,d) - V(e,d')|}{max(V(e,d), V(e,d'))} \tag{5.14}$$

Equation 5.14 is also used to compute distance between the combined contexts of the relevant faces of two documents.

Our heuristic algorithm maintains the following properties during exploration.

1. The complete search space, i.e., the network of documents is not precomputed. Instead, neighboring documents during the search are generated on-the-fly by looking up a precomputed ball tree of the documents dataset to compute b-nearest neighbors.

2. Any two consecutive documents during the search must maintain a maximum allowable distance θ.

3. Face contexts of one document, as combined from certain number of most relevant faces retrieved by using Equation 5.13, cannot be more than τ distant from the face contexts of a neighboring document.

4. The search must have a progression over time, i.e., $Timestamp(d_i) \leq Timestamp(d_{i+1})$ for two consecutive articles d_i and d_{i+1}.

Notice that all these constraints can be applied during a candidate evaluation phase of any heuristic search algorithm. We generate the b nearest neighbors based on text content of the documents, and rank the candidate documents for exploration based on their contexts. b is also considered the branching factor for the heuristic search algorithm. Empirical studies with different b, θ, and τ values are shown in Section 5.4.9.

Tracking Geographical Context

Traceability of faces of interest in terms of geographical context help an analyst understand the spatial nature of an actor of an event. This includes how political campaigns spread over the world, or how an actor of one country influences the public sentiments of the neighboring countries.

To trace the geographical context of faces, we divided the New York Times dataset into buckets of two consecutive years in such a way that each bucket has one year in common with the next bucket in the sequence. This is to ensure that the time series generated from the dataset do not have sudden spikes due to discrete year-wise division of the data. After the division of the data, we copy a face image of a person of interest to all the articles containing the name of the person. That is, the copy is an addition to all other images that already exist in the data for each bucket. Then, we generate a geographical context for each of the copies within the scope of each bucket. A divergence between the geographical context generated for a person and a uniform country distribution is recorded for each bucket. This results in a signal and the higher value of the divergence indicates an association of a person to a few countries. A fall in the signal indicates globalization because the geographical distribution is closer to the uniform distribution.

Later, in the experimental results (Section 5.4.5), we demonstrate traces of faces of a few political leaders.

Contextual Clustering of Faces

The focus of this application is to form groups of faces with high inter-cluster contextual similarity. This application helps discover the community of a person (face) based on shared contextual similarity. We leverage a density based clustering algorithm, DBSCAN [35], in this application. Unlike k-means clustering, DBSCAN does not require prior specification on the number of clusters. DBSCAN has the ability to avoid outliers and form the intrinsic clusters. We used Soergel distance [52] to compute the dissimilarity between two contexts of two faces. Section 5.4.6 explains some of the contextual clusters discovered using DBSCAN.

5.4 Experimental Results

We use New York Times news articles for our experiments. The dataset contains 54,371 articles and 86,966 images with around 98,914 faces and 69,829 entities. All these news articles are time stamped. The data collection procedure is described in Section 3.3.1 of Chapter 3. For all the classification based experiments, we used logistic regression, one-vs-rest when appropriate, with L2-regularization, and 10-fold cross-validation unless mentioned explicitly. In this section, we seek to answer the following questions to justify the capabilities of our F2ConText framework. Our experiments and case studies are divided into two categories: (1) contextual analysis, where we evaluate different types of contexts quantitatively and qualitatively and (2) runtime analysis for context generation method.

1. **Contextual analysis:**

   (a) How good are the contexts generated for the face images? (Section 5.4.1)

   (b) How well do the generated face-contexts complement a solution of a face recognition problem? (Section 5.4.2)

   (c) How do person, location and geographical contexts provide a sense about a face image? (Section 5.4.3)

(d) How well does the geographical context $D(f)$ perform compared to its different compositions and any baseline? (Section 5.4.4)

(e) Are geographical contexts traceable for analytic purpose? (Section 5.4.5)

(f) Are the generated contexts suitable for computing distance to be able to cluster the faces? (Section 5.4.6)

(g) What is the impact of face context on the quality of the stories? (Section 5.4.7)

(h) Do the generated stories provide meaningful genealogy of events? (Section 5.4.8)

(i) How do the search parameters control the characteristics of the stories? (Section 5.4.9)

2. **Runtime analysis:**

(a) Does the context generation mechanism scale well with increasing data size? (Section 5.4.10)

5.4.1 Quality of Entity Level Face-Contexts

We present three different experiments in this section to evaluate entity level contexts of persons. Each of the experiments evaluates a distinct aspect of the context.

In the first experiment, we evaluate our context generation method in terms of the capability of capturing the actual person name of a face within the context. The person context of a face is a list of entities in descending order of association probabilities (Eq. 5.7). For comparison, we use a baseline method, which creates a context of a face by combining all entities of the document where the face was found. The entities in the context of a face using the baseline method is ordered by the TF-IDF weights of the entities in the document containing the face. In Figure 5.4 we compare our context generation method, *F2ConText*, against the baseline method. The x-axis represents the number of top entities considered as the context. The y-axis represents percentage of faces for which the context of the face contained the actual name of the person.

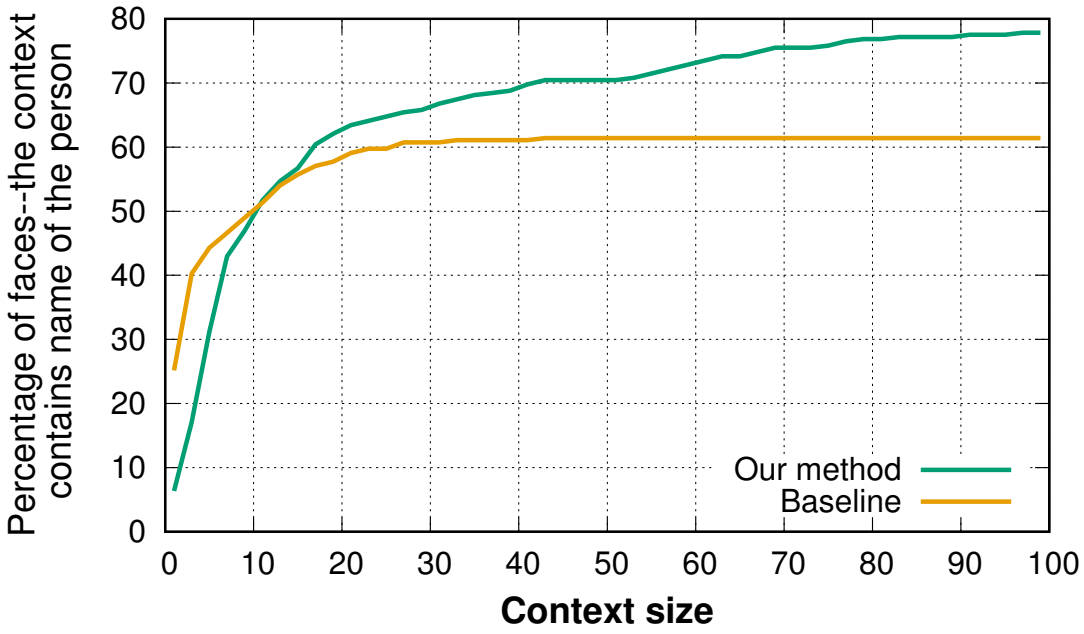


Figure 5.4: Quality of context in terms of the appearance of the person name in the context of a face.

The line for baseline method in Figure 5.4 suggests that the name and face of a person appear in the same document for only around 60 percents of the faces. The line becomes horizontal after a context size of around 30 because there was no document containing more than 30 name entities. Figure 5.4 shows that *F2ConText* is capable of producing context containing the actual name of the persons for more faces than the baseline method when the context size is greater than 10. When the context size is greater than 30, *F2ConText* keeps improving its performance as opposed to the baseline method. This indicates that *F2ConText* can bring the actual person name of the face even if the name appears in other documents but not in the document that contains the face. The result is based on a random 1200+ faces for which a human analyst labeled the faces with their actual names. This benchmark data is available in this link: `http://dal.cs.utep.edu/projects/storyboarding/KAIS/LabeledFaces.zip`

In the second experiment, we evaluate the quality of the context of a face by comparing

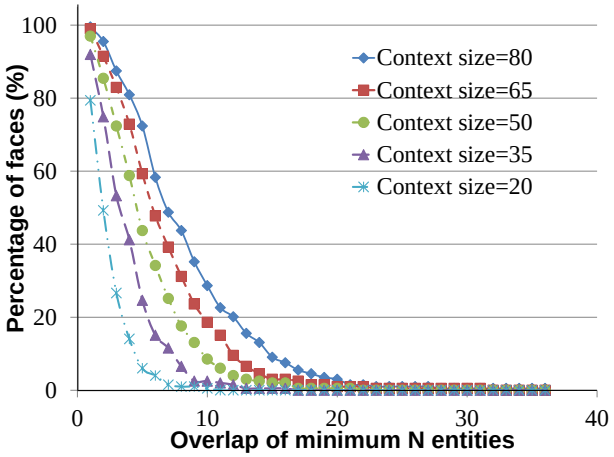


Figure 5.5: Resemblance between a face context and a relevant Google image search.

the context with Google image search results. Although Google search has a different objective than ours, we are interested in verifying whether some of the entities in the context of a face can be discovered using Google image search by uploading the face image. Since Google image search API limits the number of queries and the process is time consuming, we randomly picked up to 300 faces and computed how many of the terms of the context of each face were found in the list of titles and summaries in the first page, where the search result was returned by Google after uploading the face. Then we calculate the number of entities it has in common with our context of that face. Figure 5.5 shows a distribution of percentage of faces for different number of overlaps of context and Google search. As expected, the percentage of faces goes down when we look for more common entities. The plot shows that almost 75% of the faces have at least five entities in common with corresponding Google search result when we pick a maximum of 80 most probable entities.

In an additional experiment to evaluate the quality of the generated contexts against ground truth information, we sample 21 faces from $\mathcal{F}$, and manually attach most appropriate person entities to each of them with the help of human experts. We then compare these ground truth contexts with our automatically generated context. Figure 5.6 demonstrates a comparison between two approaches using vanilla face features and face features combined

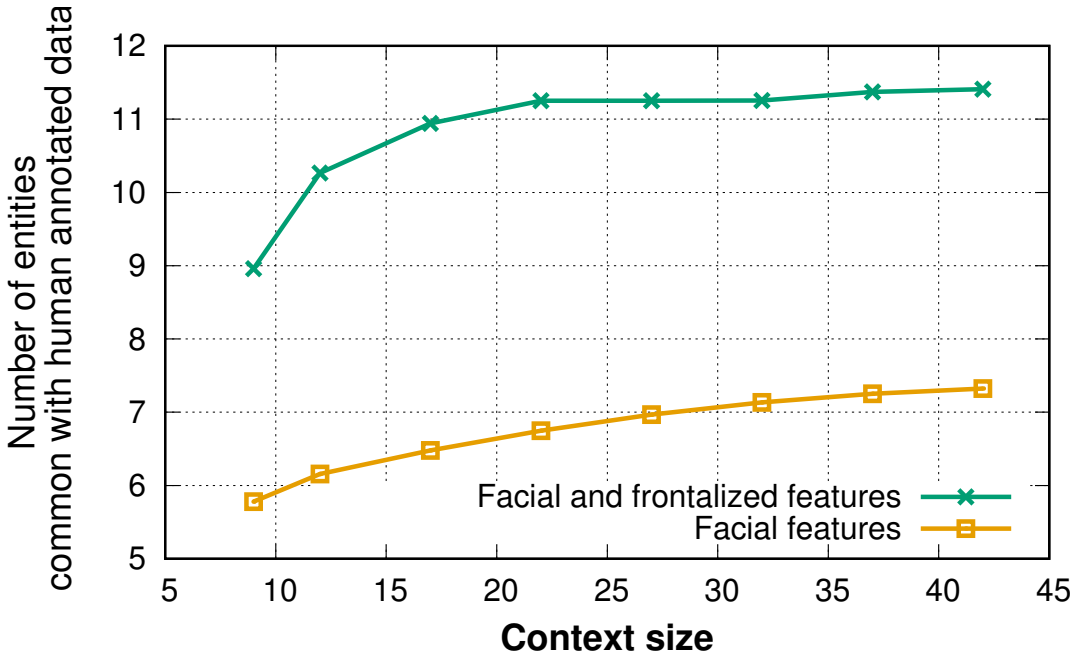


Figure 5.6: Comparison of context generation methods for human annotated test data. Adding frontalized features improve quality of context.

with frotalized features, in terms of number of entities in common with the human-made context. Face feature combined with frontalization was found to be providing the best context for all context sizes.

5.4.2 Context in Face Recognition

The contexts generated for faces can be used as features to complement a face recognition solution. In this study, we create three sets of features for face images, (1) features generated using faceNet and frontalization described in Section 5.3.1, (2) the context features generated for each face using our framework (3) a combination (concatenation) of the context features and the face-features. Figure 5.7 shows that incorporation of context features with face-features improves the face-recognition accuracy in terms of F1-score when compared with face recognition using face-features alone, or face recognition using context features alone. For this experiment, we randomly selected around 1,200 faces which re-

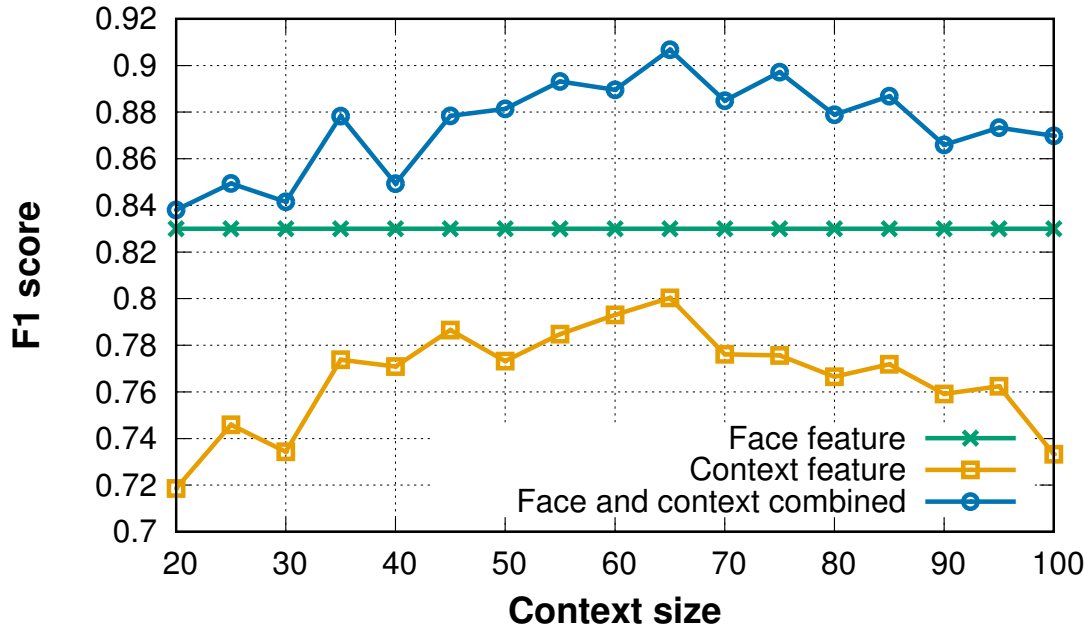


Figure 5.7: Context in face recognition.

sulted in around 500 people. A human expert labeled these faces. For the classification, we used an one-vs-rest logistic regression with L2-regularization and 2-fold cross validation.

It is noticeable that the accuracy of face recognition varies with different context size. A general observation is that if the context size is smaller the F1 score is lower; slowly with increasing context size, the F1 score becomes higher with a peak at around size 65. The F1 score decreases as the context size is increased further. This indicates that the selection of the best context size is an analytic choice for any dataset-specific face-recognition task.

5.4.3 Person, Location, and Geographical Contexts

Figure 5.1 in Section 5.1.1 shows an example of a face, the generated person context, location context, geographical context using a bar chart, and the geographical context laid on a map. The face was of David Cameron, a British politician and the former Prime Minister of the United Kingdom. The person context captures the name of the Prime Minister as well as the names of a few other related people. Entities in the generated

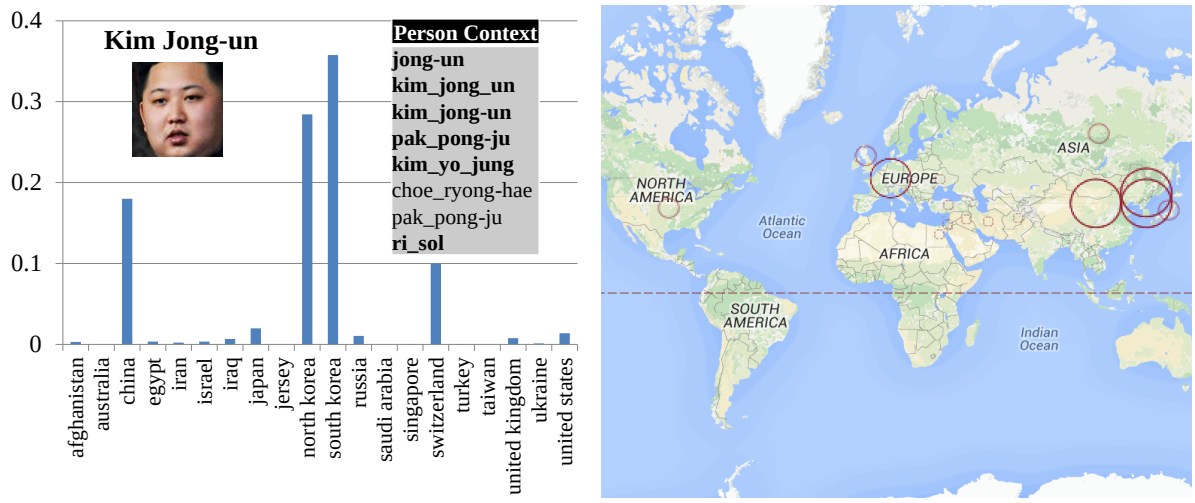


Figure 5.8: An example of geographical context: (left) Probabilities of top twenty countries with highest values, and (right) probabilities are highlighted in a map.

location context provide an idea about the areas related to the face, such as "kingham", "britain", and "medieval britain". The bar chart in Figure 5.1 presents the geographical context, which is the probability distributions of countries computed by Equation 5.12. In addition, we provide another representation in which circles with size proportional to those country-probabilities are laid on a map.

Figure 5.8 shows the geographical context of a face of Kim Jong-un, the supreme leader of North Korea. The person context is set into the bar chart. Both the bar chart and the map portray that Kim Jong-un's geographical context is focused on the region of North Korea, South Korea and China. The reason behind Switzerland's appearance in the geographical context is a controversial piece of information about Kim Jong-un's school attendance in Switzerland.

5.4.4 Comparison between Different Methods to Generate Geographical Context

The absence of a ground truth dataset to evaluate the generated geographical contexts for each face makes our evaluation challenging. To address this issue, we develop a ground truth set by manually labeling one hundred faces by the name of the person corresponding to each face, and then by detecting countries and cities from the documents where the labeled person name is found. This allows us to generate a ground truth country distribution, $D^T(f)$, for each face labeled manually. Let ϱ^f be the set of person names manually identified for face f. The ground truth country distribution of f is:

$$\begin{aligned} D^T(f) &= \{d^T_{\phi_1}(f), d^T_{\phi_2}(f), ..., d^T_{\phi_{|\Phi|}}(f)\}, \text{ where} \\ d^T_{\phi_i}(f) &= \sum_{\varrho \in \varrho^f} \sum_{a \in a^\varrho} \sum_{l \in \mathcal{E}^L_a \cap \rho_{\phi_i}} P(\phi_i|l) * W_L(l, a) \end{aligned} \tag{5.15}$$

Here, a^ϱ is the set of articles where name ϱ appears, ρ_{ϕ_i} be the cities in country ϕ_i and $\mathcal{E}^L_a$ is the set of location entities in article a. A geographical context $D(f)$ of a manually labeled face f is evaluated as a high quality context if $D(f)$ is close to the ground truth geographical context, $D^T(f)$, which is computed using the labels.

A baseline geographical context $D^B(f)$ of a face f is a country distribution that is generated using the cities and countries found in the same article which contains f. $D^B(f)$ is computed using the following equation.

$$\begin{aligned} D^B(f) &= \{d^B_{\phi_1}(f), d^B_{\phi_2}(f), ..., d^B_{\phi_{|\Phi|}}(f)\}, \text{ where} \\ d^B_{\phi_i}(f) &= \sum_{l \in \mathcal{E}^L_{a_f}} P(\phi_i|l) \end{aligned} \tag{5.16}$$

where a_f is the article where face f appears.

In this section, we demonstrate the effectiveness of our approach and the baseline approach as compared to the ground truth. From Equation 5.12, $D(f)$ is a composition of $D^1(f)$ and $D^3(f)$. In this experiment, we compare the resulting error of $D(f)$ and all combinations of $D^1(f)$, $D^2(f)$ and $D^3(f)$ using the ground truth data. The error is derived

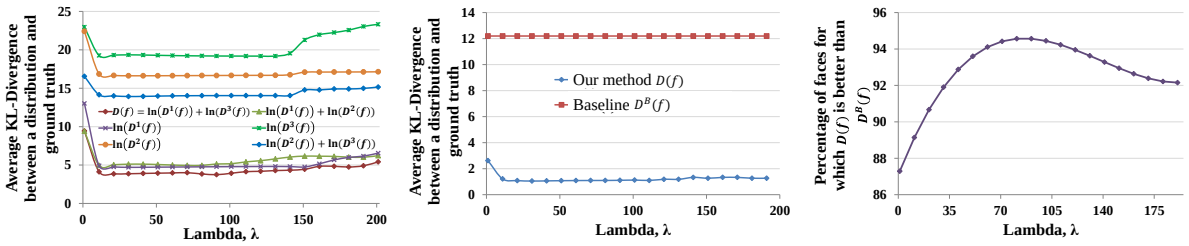


Figure 5.9: (left) The combination of country distributions $D^1(f)$ and $D^3(f)$ performs best for various values of λ. (middle) Performance of $D(f)$ against baseline $D^B(f)$: lower average KL-divergence using our method indicates better results. (right) $D(f)$ performs better than $D^B(f)$ more than 87% of the times even during the worst choice of λ.

by computing the average KL-divergence between the distribution under consideration and the ground truth distribution $D^T(f)$ of one hundred labeled faces. A lower average KL-divergence indicates a better distribution because it is closer to the ground truth. Figure 5.9(left) shows average KL-divergences using different combinations with varying λ. Lower values of λ will allow inclusion of more entities. Figure 5.9 (left) shows that $D(f)$ has the lowest KL-divergence with the ground truth using any value of λ. This indicates that $D(f)$ performs the best among all the combinations.

In addition, we compute average KL-div$(D(f), D^T(f))$ and KL-div$(D^B(f), D^T(f))$ with different λ. The baseline, $D^B(f)$, is computed using Equation 5.16. Figure 5.9 (middle) shows that the geographical context $D(f)$ generated by our method has lesser KL-divergence than the baseline $D^B(f)$ indicating that our approach provides closer results to the ground truth. KL-div$(D^B(f), D^T(f))$ is constant in Figure 5.9(left) because it does not depend on λ.

One observation is that average KL-div$(D(f), D^T(f))$ is the highest when λ is too small. This is because small values of λ indicate the use of a long list of entities in the context. Our observation is that the best performance is found near $\lambda = 80$, which is evident in Figure 5.9(right). The plot also shows that even in the worst case, $D(f)$ performs better than $D^B(f)$ more than 87% of the times.

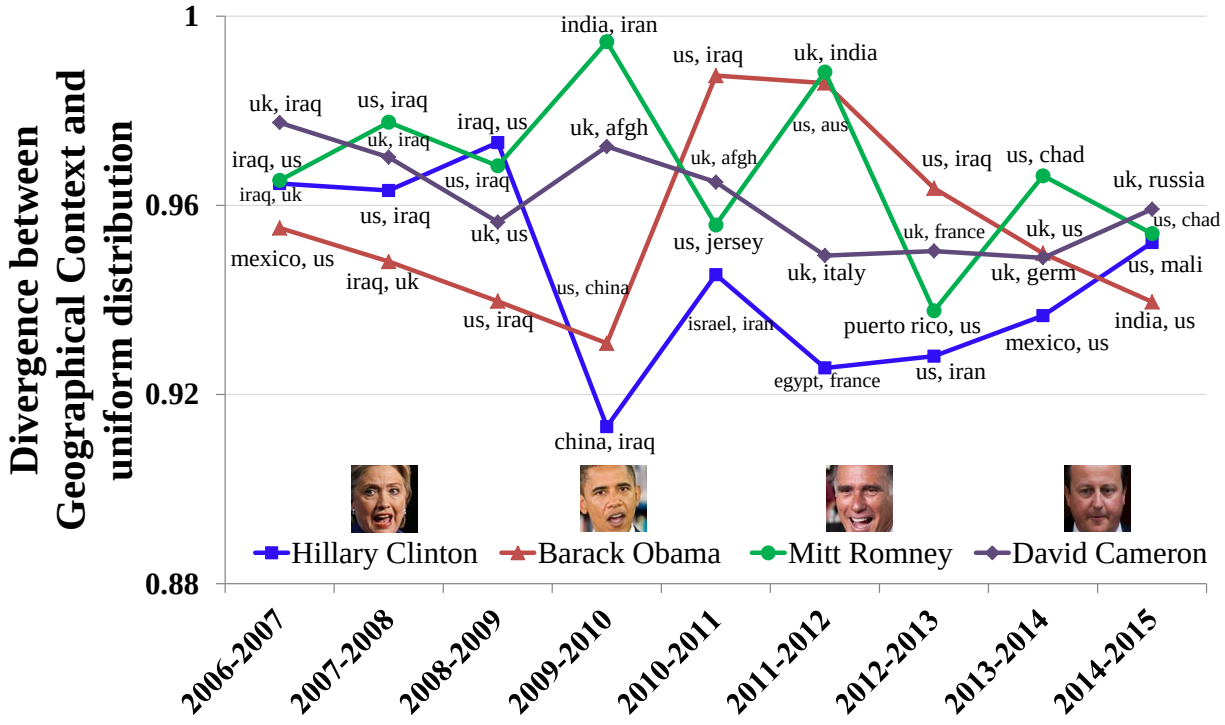


Figure 5.10: Geo-contextual trends of leaders over time.

5.4.5 Tracking Geographical Contexts

As explained in Section 5.3.5, geographical context of each person is a traceable distribution. In Figure 5.10, we outline the trends of geographical contexts of four political leaders: Hillary Clinton, Barack Obama, Mitt Romney, and David Cameron. Top two countries are used as a label for each data point in the plot. Hilary Clinton served as the United States Secretary of State from 2009 to 2013, in which part the trend line has comparatively lower values indicating a focus on affairs around the globe. The upward movement of Hilary Clinton's trend line from 2013 indicates that her focus is centralizing toward the United States. While Mitt Romney's trend line exhibits somewhat more centralizations toward the United States, President Barack Obama's trend line has some interesting patterns — gradual globalization from 2006 to 2010, centralization from 2010 to 2012, and again gradual globalization from 2012 to 2015. David Cameron's trend line has similar trends

to President Barack Obama but David Cameron's trend line has lesser fluctuations. Such tracking capability will allow social scientists and analysts study the dynamics of persons of interest.

5.4.6 Context based Clustering of Faces

Generated contexts can be used to create a feature space for the faces. For the person contexts, it is possible to create vectors using entities as features. The distribution of the geographical context can be directly leveraged as the feature space. This leads to the ability to compute pairwise distance between contexts and hence opens up the opportunity to contribute to many machine learning applications. In this subsection, we provide examples and analysis of clustering outcomes using person and geographical contexts. We leveraged DBSCAN to group the faces based on context. Figure 5.11 shows two clusters of faces generated by DBSCAN using person context elements as features. In Cluster 1 of Figure 5.11, there are eleven faces of three people. One of these three people is the Turkish president, Recep Tayyip Erdogan. Cluster 1 contains five faces of president Recep Tayyip Erdogan. These five faces were detected from five different news articles of the New York Times dataset. The faces of other two people were found in the vicinity of the images where the president's face was present. All these faces are in the same cluster because their person context has high similarity. Cluster 2 brings together a total of thirteen faces of eight people from five documents. These faces are either connected to the bombing in London's transit system in 2005, or charged with terrorism and murder relevant to September 11 hijacking of commercial airliners. In both Cluster 1 and 2, faces with similar person contexts are brought together.

Figure 5.12 shows a cluster generated by DBSCAN that leverages the geographical context. The cluster contains seven faces of David Cameron, and five faces of five different people. All of their geographical contexts have focus on Europe, especially, the United Kingdom. This example shows the potential in bringing persons of interest with similar geographical context in the same group.

We use a Latent Dirichlet Allocation (LDA) [14] based technique to evaluate the contexts of each face cluster generated by our framework and compare this with a baseline approach where context of each face is generated using person entities found only in the article where the image of the face is located. We first apply LDA to generate the topics of each of the documents in the corpus. A good face cluster should bring the context from documents of the same topic. For a face cluster c_i, clustered using contexts of the faces, we find the documents $\delta(c_i)$ from which the faces of c_i were retrieved. If the contexts of the faces of c_i are good, then the documents of $\delta(c_i)$ should be from a small number of topics. If $\delta(c_i)$ comes from too many topics, this would indicate that the contexts of the faces that formed c_i are scattered over many topics and hence faces in c_i are less contextual. The documents relevant to a baseline face cluster come from too many topics, where the faces in a person-context based cluster come from low number of topics. The weight in the vertical axis of Figure 5.13 is a representation of the number of clusters distributed to T topics and is computed by $\sum_{c \in C^T} \frac{|c|}{|\mathcal{F}|}$, where C^T is the set of face clusters where the documents of the faces of each cluster are distributed to a total of T topics. Larger values with low T and lower weight with larger T represent a better quality of context in the clusters. Figure 5.13 shows that the person context based face clustering (green line) ended long before

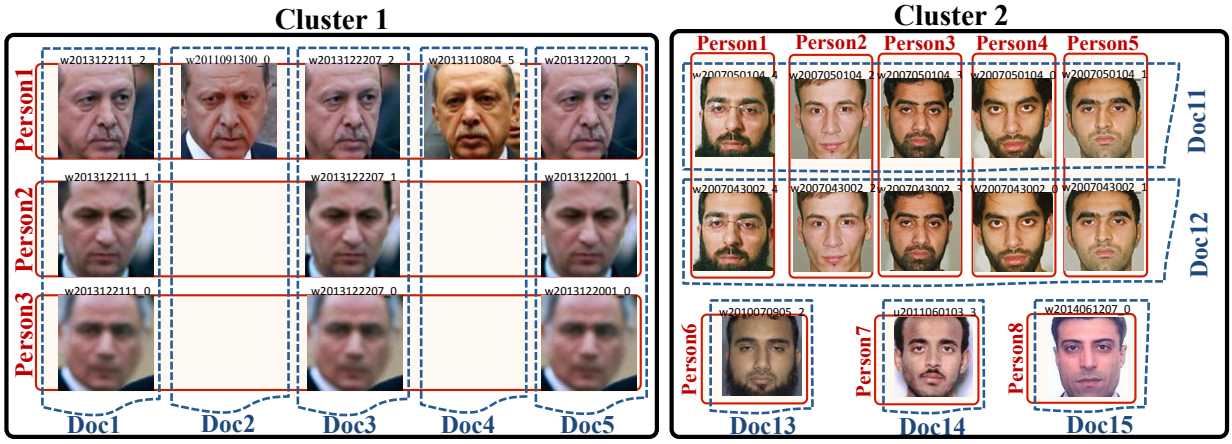


Figure 5.11: Two clusters of faces generated by DBSCAN using name context elements as the features of the faces.

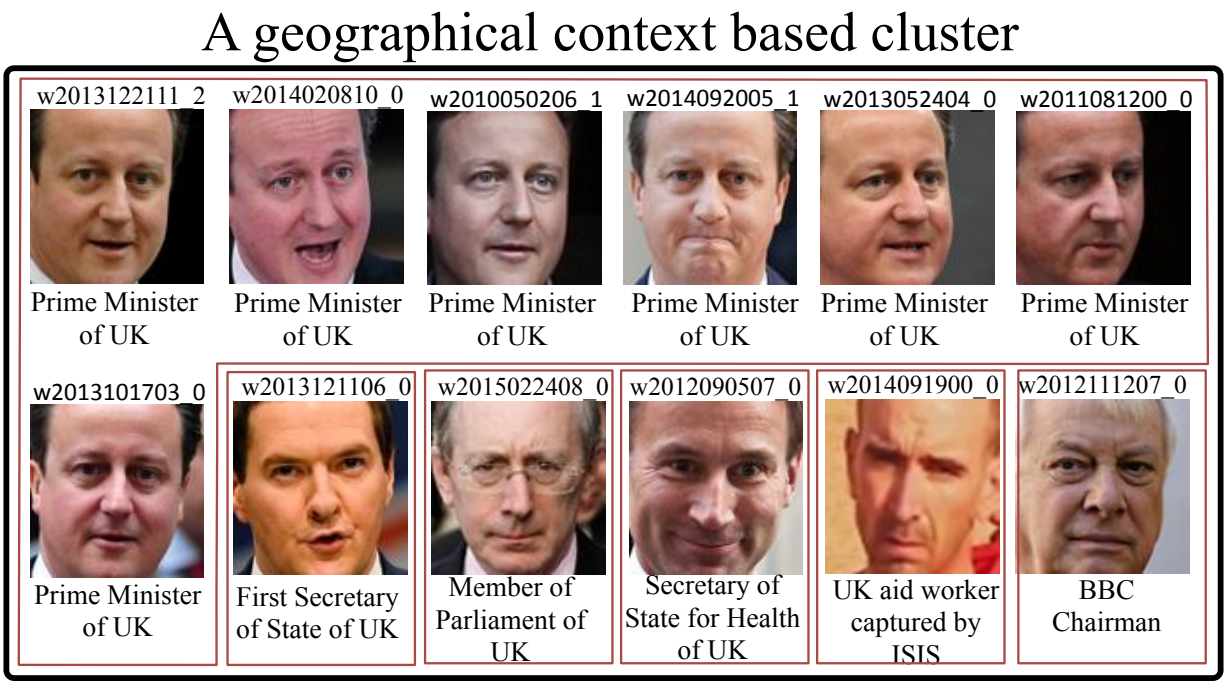


Figure 5.12: An example of a geo-context based clustering.

the baseline clustering (red line) in the x-axis indicating that the clusters produced by our method are more contextual.

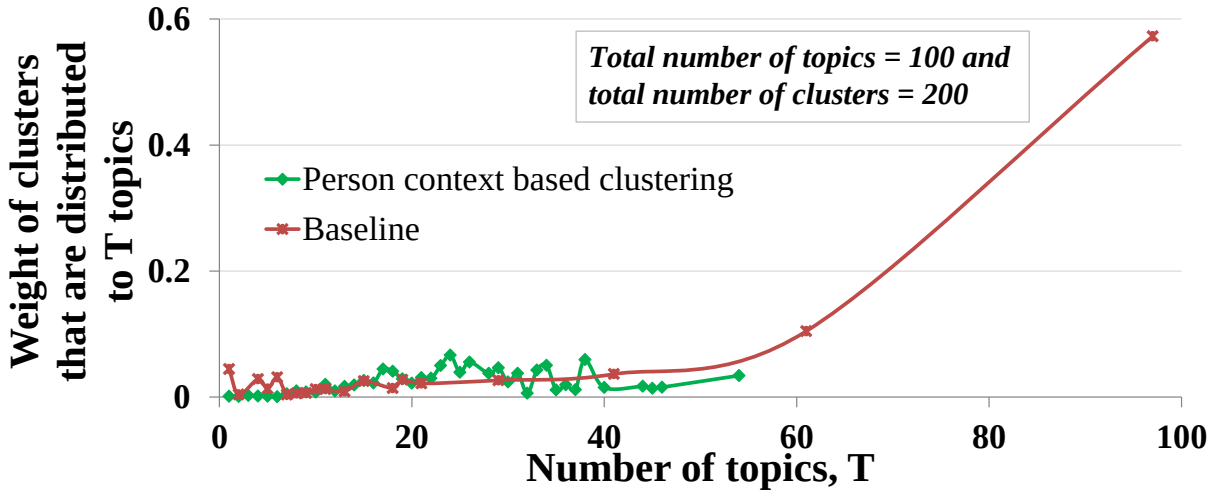


Figure 5.13: Person context clustering has higher quality (small values with large number of topics) than the baseline.

5.4.7 Impact of the use of Context in Story Generation

While inclusion of image context in the core heuristic path finding part of the summarization process imposes an additional constraint, the outcome of the use of this constraint becomes evident when we compare the stories side by side. For a fair comparison we make sure that both the methods have the same parameters (e.g., same θ, branching factor b, and

Table 5.2: Sample stories generated with and without context. (URLs of the original news documents are provided with the article IDs in blue and can be reached by clicking on the IDs.)

Story with context	Story without context
[NY435 → NY317 → NY175 → NY427 → NY642 → NY552 → NY525] The story describes the request for delay and change of trial location.	[NY435 → NY201 → NY525] The intermediate article digresses from the focus and brings investigation in Russia into consideration.
[NY178 → NY370 → NY334 → NY609] The story connects triple murder with current cases.	[NY178 → NY505 → NY458 → NY609] The intermediate articles are about trial announcements.
[NY265 → NY129 → NY279] The story focuses on the trial of a friend of Tsarnaev and the jury selection for the suspected friends.	[NY265 → NY129 → NY124 → NY279] One of the intermediate articles is off-topic and is about a citation of winning a video game at trial of the accused Boston bomber's friend.
[NY158 → NY379 → NY206 → NY280] This story describes former Governor's testimony for Tsarnaev's friend Robel Phillipos.	[NY158 → NY280] The testimony of the former Governor does not show up in the story without context.

start-end pairs). Table 5.2 shows four pairs of sample stories generated with and without face contexts.

Our observation is that our method generates more coherent stories when the context is used, which is to be expected because context overlaps between consecutive articles reinforces a constraint that each story must be weaved using a certain theme. For all start-end pair of documents in Table 5.2, the stories with context are more coherent than the stories without context. From the table, we observe that each story without the use of context has some off-topic documents that may be relevant but the flow of the theme is broken. Sometimes we have the same story with and without the use of context but those stories are not reported here.

5.4.8 Examples of Generated Stories using Boston Marathon Bombing Data

New York Times returns 1028 documents with the query "Boston Marathon Bombing". The summarization mechanism discovered a number of sub-events that provide a fine mental model of branches of the Boston Marathon Bombing tragedy and happenings afterwards. Table 5.3 lists a few of the stories discovered by our mechanism. The first story of Table 5.3 is illustrated in Figure 5.3 on a storyboard using term clouds and faces.

The story of Figure 5.3 describes a connection between Boston bombers and Waltham triple murder. The story moves forward till the penalty phase. The term cloud of the storyboard highlights a person named *Todashev*, a victim of triple murder, along with the bombers Tamerlan and Tsarnaev. Each related face list automatically selected from the knowledge base by the framework captures faces of relevant people very well. For example, the first face of the second document of the story is Todashev, whose face is found repeatedly in the consecutive articles. Rest of the faces in the story are of the Boston bombers Tamerlan and Tsarnaev, police, and rescue crews.

The theme of the second story in Table 5.3 is focused on the conviction of Tsarnev's

Table 5.3: Stories using Boston Marathon Bombing data. (URLs of the original news documents are provided with the article IDs in blue and can be reached by clicking on the IDs.)

Story	Explanation
NY286 (2014/04/15) → NY370 (2014/10/24) → NY334 (2014/11/12) → NY609 (2015/03/02) → NY677 (2015/04/10)	This story associates Boston bombers' involvement in the Waltham triple murder. The story includes trial phase.
NY158 (2014/10/16) → NY379 (2014/10/16) → NY206 (2014/10/28) → NY280 (2014/10/28)	Former governor of Massachusetts testifies for Tsarnaev's friend Robel Phillipos. Phillipos was found guilty of making false statement to authorities.
NY435 (2014/05/02) → NY317 (2014/06/18) → NY340 (2014/08/14) → NY525 (2015/02/06)	Tsarnaev's lawyers urge appeals to move trial location to Washington, delaying trial date. The story shows that the request was denied.
NY121 (2014/04/22) → NY273 (2014/07/10) → NY177 (2014/08/20) → NY338 (2014/09/27)	This story highlights the trials of three friends of Tsarnaev. All three were accused of obstructing justice by lying and destroying evidence.

friend, Robel Phillipos, for lying to the FBI. Therefore, Robel Phillipos was found guilty of making false statement to authorities.

The third sub-events of Table 5.3 summarizes Tsarnaev's lawyers' appeal to move trial location to Washington which delayed the trial date. The fourth story describes the trials of three friends of Tsarnaev who were accused of obstructing justice, lying, and destroying evidences.

5.4.9 Characteristics of Stories in Terms of User-Settable Parameters

Two important user-settable parameters in our method are maximum allowable distance θ and branching factor (nearest neighbor), b. Figure 5.14 shows the impact of θ and b on the statistical significance, average length and number of stories. To calculate the statistical significance, p-value, we randomly pick up b documents from the entire candidate pool and check if the documents picked satisfy the distance threshold θ, iterating the test 5,000 times. We repeat this process for every junction-article of a discovered story. The overall p-value of story is calculated by multiplying all the p-values of every document of the story except for the last one. Figure 5.14(top) shows that the significance decreases (i.e. p-value increases) with higher values of θ and b. This is an expected outcome since higher θ values imply less stringent overlap of content between consecutive articles. Less stringent constraint may result in stories with loose connections between consecutive articles. Similar argument can explain the plot in Figure 5.14(middle). Increasing the θ value and branching factor b leads to shorter stories with loosely connected neighbors. The curve for branching factor 20 and 35 are exception which gives even shorter stories than larger branching factor until $\theta = 0.75$. This exception is justified by the bottom plot where we see that there were not enough stories for those two branching factors until $\theta = 0.75$. For other branching factors, number of stories follow a similar upward trend with increasing θ.

The summary is as follows:

- The statistical significance of the distance threshold used in our story generation method decreases with the higher value of distance threshold.
- The higher value of branching factor and distance threshold leads to shorter stories with loosely connected neighbors.

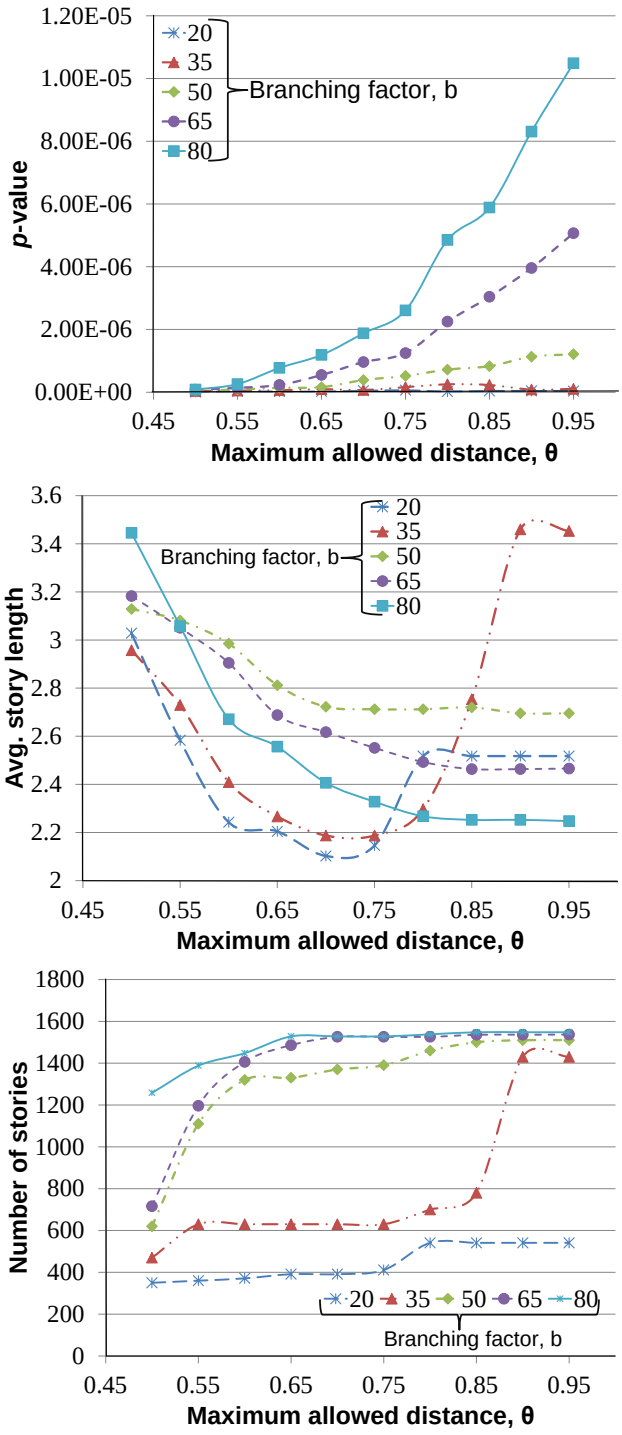


Figure 5.14: Impact of search parameters on characteristics of stories.

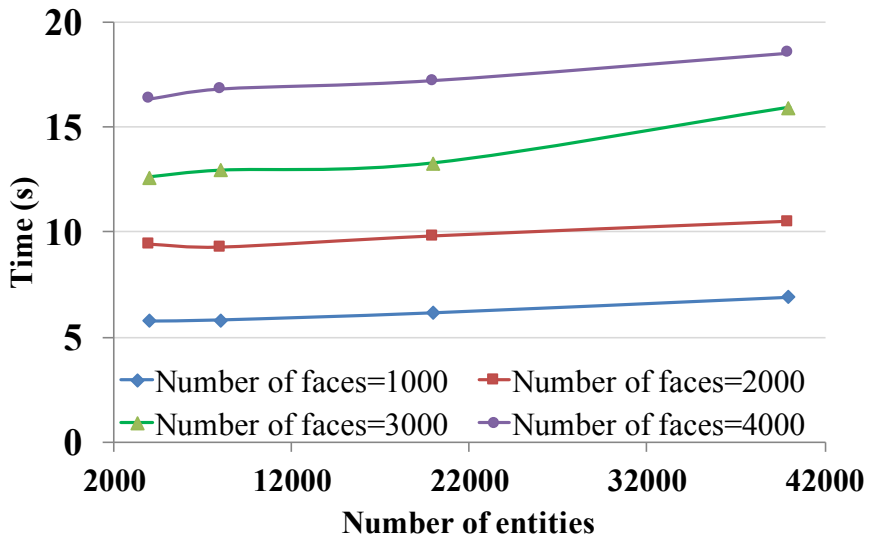


Figure 5.15: Runtime to generate entity contexts for faces.

5.4.10 Runtime Analysis

The plot in Figure 5.15 shows that as the number of entities increases for a fixed number of faces, the runtime increases almost linearly. Additionally, the runtime increases almost linearly along the vertical axis as the number of faces grows. The plot indicates that the context generation mechanism is scalable for large datasets. The person context generation time for 98,914 faces detected in the New York Times Dataset was around five hours. The geographical contexts of all these detected faces were generated in 15 minutes. The textual content in the dataset contained 65,240 person and 4,589 location entities. The run time is obtained using a regular desktop computer with Intel Core i7 Quad Core CPU @ 3.40GHz and 24GB RAM.

5.5 Summary

This chapter presents an automated system F2ConText that effectively retrieves holistic contextual phenomenon from news articles. F2ConText fuses face features with textual entities to provide a better understanding of the contextual scope of persons of interest.

The framework does not require any human supervision for mapping image features to textual snippets. Results show that our system captures meaningful contextual features that can be leveraged by other machine learning applications.

Chapter 6

An Unsupervised Framework for Representing Documents Containing Images and Text

An overwhelming number of publicly available documents are generated in digital form by different people and machines, and may contain different types of data including text, image, audio, video, and tweet. The coexistence of different kinds of data is a great source of contextual information but brings new challenges for information retrieval (IR) tasks [61]. One of the major challenges is to exploit disparate content of documents in a variety of applications, such as text classification [104], document clustering [80], information retrieval [79] and question answering [120]. Most of the applications can not directly interpret the content of documents, therefore, greatly rely on a numerical representation of documents. Moreover, the performance of the applications is directly affected by the quality of a document representation.

We observed that the texts of the documents containing images are relatively less informative compared to the documents containing no images. This observation is based on the news articles collected from the New York Times. In Figure 6.1, two histograms of the entropies of documents are presented: documents with images and documents with no images. The entropy [2] is calculated based on the textual content. We use a discrete random variable *WordChoice* with possible values being all the words in a document. The

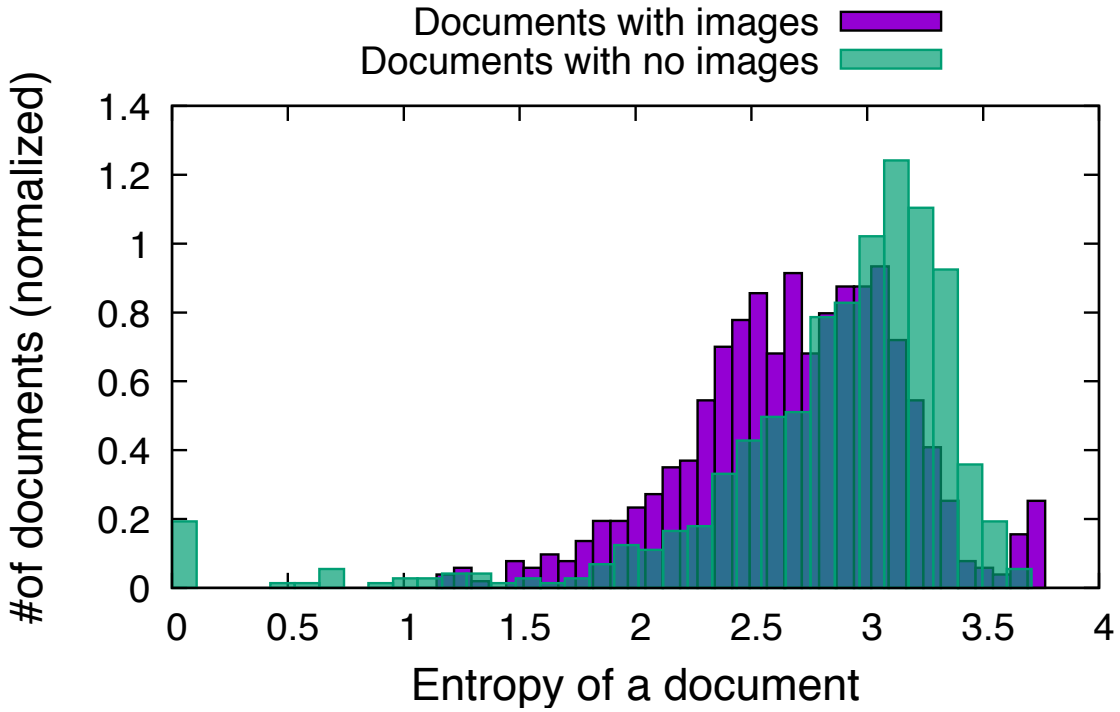


Figure 6.1: Histograms of the entropies of documents.

probability of a word w_i in a document is calculated as

$$P(WordChoice = w_i) = \frac{\text{Frequency of } w_i \text{ in a document}}{\text{Sum of frequencies of all words in that document}}$$

From this figure we see that more documents without images have relatively higher entropy compared to the documents with images. In other words, documents containing images possess less information in its textual content. Therefore, dealing with documents with imagery content poses multiple challenges in representing documents: complexity to understand the images and reduced amount of information in the textual content. To address these problems, this chapter presents methods to exploit contextual information of text fragments and visual objects, i.e., faces, found in images for document representation.

Document representation aims to map a document into a condensed representation of its content. There have been proposed a lot of document representation models, such as Boolean model [71], bag-of-words [47], vector space model [102], latent semantic analysis [31]. All of these models emphasize the frequencies of lexical units while mostly neglect the

order and context. Neural language models [11] utilize local contexts of words and their relationships in representing documents in a lower dimensional continuous space. Recently, local context window based variants of the neural language models, such as doc2vec [74] and skip-gram [81], have been shown to capture syntactic and semantic word relationships from the unstructured text corpus and perform better than the traditional methods of document representation for lots of IR related tasks. Neural language models are usually very effective when there is a huge amount of data available. However, the context window based methods sometimes suffer from lack of context in short documents, e.g., tweets, news headlines, and titles of research papers. In spite of the numerous research efforts on text-based document modeling, very few attempts have been made to harness non-textual content in document representation. A few researchers have approached this problem by utilizing user-tagged images and text together in a joint model for document representation [109, 96]. However, the exploitation of contextual information generated for images and text in document representation remains an open challenge.

In this chapter, we present (1) a neural language model that employs contextual information of persons depicted in images along with textual content for document representation and (2) a text representation to complement that neural representation by utilizing contextual information of text fragments.

In summary, the main contributions are as follows:

- We present a local context window based multimodal neural language model that exploits contextual information of persons depicted in images and textual content for document representation.
- We present a complementary text representation technique that utilizes contextual information of text fragments.
- We conduct a set of experiments to evaluate the proposed representations. We demonstrate how to leverage the representations in traditional classification and clustering problems.

6.1 Problem Formulation

Let $\mathcal{D} = \{d_1, d_2, \ldots, d_{|\mathcal{D}|}\}$ be the set of documents containing text and images, $\mathcal{E} = \{e_1, e_2, \ldots, e_{|\mathcal{E}|}\}$ be the set of entities, and $\mathcal{I} = \{i_1, i_2, \ldots, i_{|\mathcal{I}|}\}$ be the set of images in the corpus. We extract faces from the images to be able to find the context of persons. Let $\mathcal{F} = \{f_1, f_2, \ldots, f_{|\mathcal{F}|}\}$ be the set of faces.

In this work we consider only the documents that contain both text and images. Our goal here is to generate high-quality distributed vector representation of the documents for document classification and clustering. We divide the big problem into the following subproblems:

1. Generating context F_i for every face $f_i \in \mathcal{F}$. The context of a face is a probability distribution over all the person names found in a corpus.

2. Generating vector representation V^e of every entity e based on the textual content of the corpus.

3. Learning document vector D^d for each document $d \in \mathcal{D}$ using imagery and textual content of the documents.

6.2 Methodology

The proposed framework comprises three main stages: (1) context extraction for persons, where we build a probabilistic model to associate entities with every face, (2) building contextual embedding of entities and (3) generating vector representation of documents. An overview of the complete framework is presented in Figure 6.2. In the following subsections we describe each of these stages in more detail.

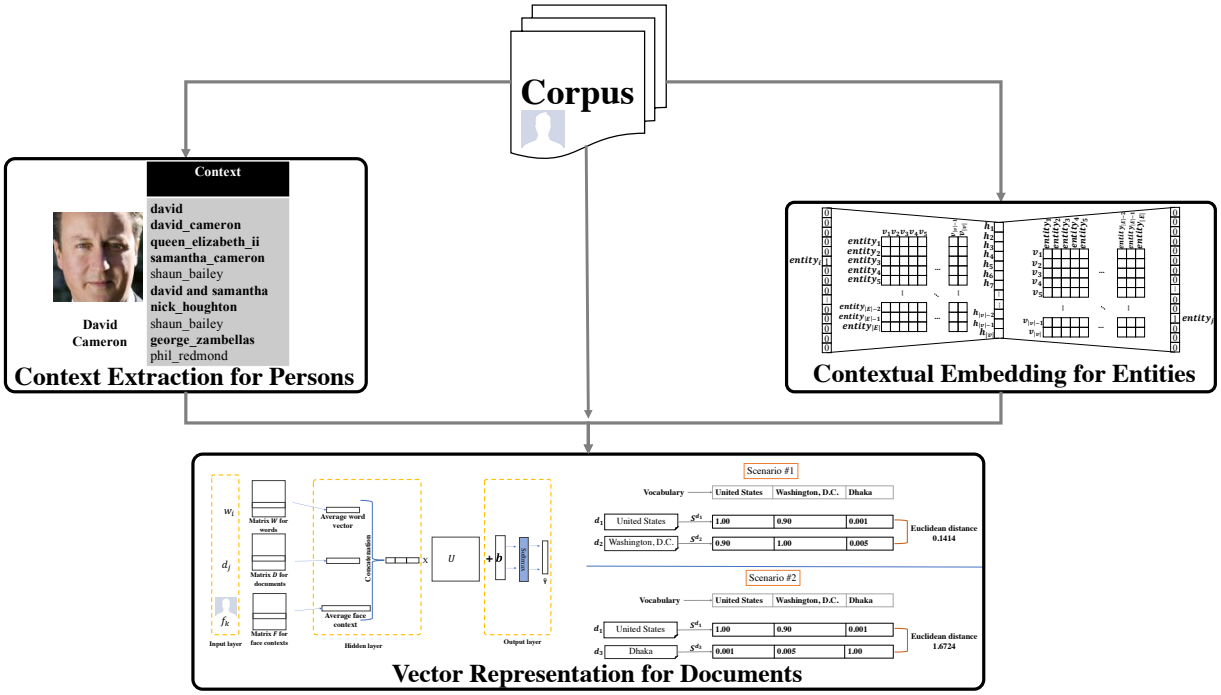


Figure 6.2: Overview of the proposed framework.

6.2.1 Context Extraction for Persons

Since context of a person is a probabilistic mapping between face and textual entities, we need to detect faces from images, extract facial features from faces and identify entities from text. We leverage Convolutional Neural Network (CNN) based state-of-the-art face detection approaches to detect faces from images $\mathcal{I}$ [133, 76]. The face detection model is a deep cascade architecture built on convolutional neural networks. We use a pre-trained model of [133] that jointly performs face detection and alignment using multi-task cascaded convolutional networks. To extract high quality face features, we use a pre-trained model of FaceNet [103], a CNN-based face embedding framework, to extract face features in low-dimensional euclidean space. Since many of the faces in our datasets are side-facing, we use frontalization [61] method to be able to capture positions of some key facial points in a projected plane where the side-faced photo represents a front-posing face. This enables us to bring side faces to a common space where all faces are considered front-facing. The

details of the frontalization method are described in Chapter 4.

The extraction of the entities from the textual content of documents $\mathcal{D}$ rely on a number of entity extractors including LingPipe [8], OpenNLP [10], and Stanford NER [110]. Although we extracted all standard entity types including person name, organization, and location, this chapter scopes down the analysis to person entities only, especially because the images are explained using detected human faces. We compute the weight of every entity in documents by using TF-IDF [78] weighting with cosine normalization. TF-IDF stands for Term Frequency-Inverse Document Frequency. It is a term weighing mechanism intended to reflect the importance of a term in a document within a corpus. Section 5.3.1 describes the weighting mechanism in more detail.

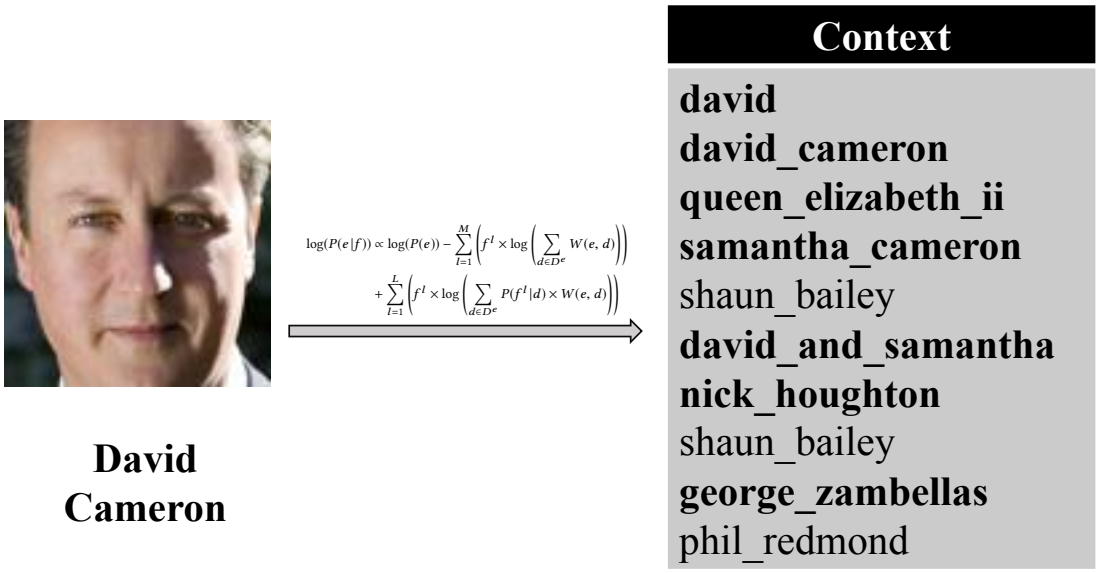


Figure 6.3: Context of David Cameron, the former Prime Minister of United Kingdom.

Context of a face is a probability distribution over all the entities in a corpus. We use a probabilistic model to generate context for every face. The details of the context generation process and modeling are explained in Section 5.3.2 of Chapter 5. The final equation is

shown here again as:

$$\log(P(e|f)) \propto \log(P(e)) - \sum_{l=1}^{M} \left(f^l \times \log \left(\sum_{d \in D^e} W(e,d) \right) \right) + \sum_{l=1}^{L} \left(f^l \times \log \left(\sum_{d \in D^e} P(f^l|d) \times W(e,d) \right) \right) \quad (6.1)$$

In practice, we do not record the full probability distributions, rather we keep record of a certain number of entities with highest probabilities as the context of a face. Figure 6.3 shows an example of context for David Cameron, former Prime Minister of United Kingdom. The figure depicts that his context includes name of his wife, the name of the Queen of United Kingdom and names of some related persons in his cabinet.

6.2.2 Embedding Generation for Entities

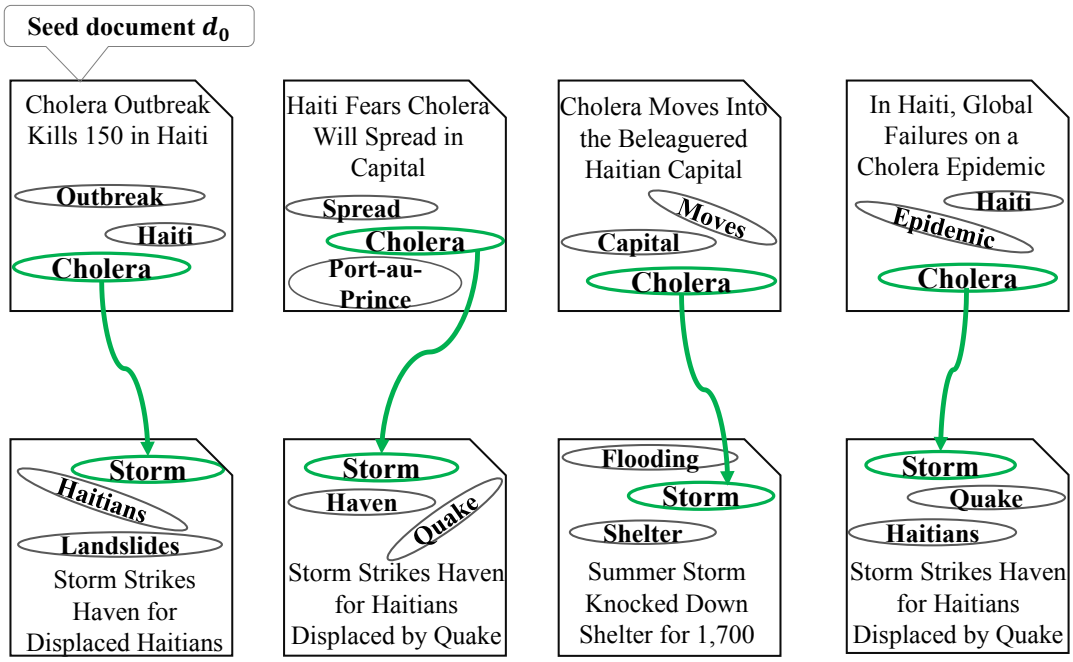


Figure 6.4: Entity relationships across documents.

Contextual information extracted from the textual content of documents can be viewed

as a multi-graph of entities. To construct the multi-graph, we find relations between entities by utilizing temporal, geographical and thematic information associated with each document. We presented an objective function in Section 3.2.3 of Chapter 3 that is used to extract relationships between two entities. Figure 6.4 shows an example where the objective function decides to pick a relationship between the entities *cholera* and *storm*, which appear in different documents in different time.

Finally, we leverage entity relationships to generate the vectors using the machinery commonly seen in neural network based distributed vector generation [82, 105]. In Section 3.2.4 of Chapter 3, we presented the details of the vector generation techniques, but here we are describing it briefly. The task of vector generation for each entity, given the set

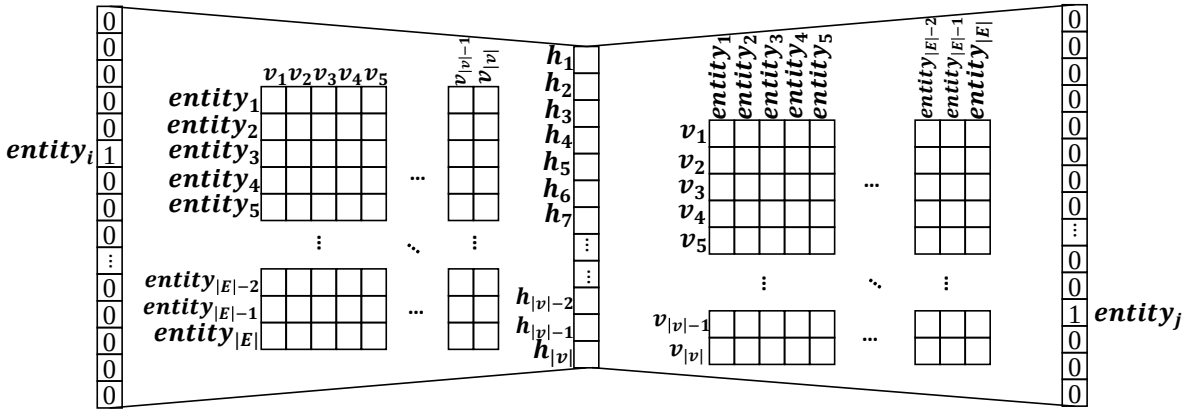


Figure 6.5: A shallow neural network for entity vectorization.

of entity relations for every document, can be performed by training a neural network that generates neural entity embeddings. This approach resembles the method used in word2vec [82, 83]. At each step of the training of word2vec, a set of words are given as input to the neural network and another word is considered as the target word to predict. We leverage this model to create vectors of entities by feeding each observed entity relationship (a pair of entities) to the network — one entity is used as input to predict the other one. Figure 6.5 shows that $entity_i$ is given as the input of the two-layer neural network to predict $entity_j$ for a relationship $\rho = (entity_i, entity_j)$.

6.2.3 Contextual Vector Representation of Documents

In this section, we introduce a contextual vector representation technique for documents containing both imagery and textual content. Every document is mapped to a unique vector of real numbers reflecting contextual information extracted from images and text.

This framework utilizes words co-occurrence probabilities in a predictive neural architecture to generate vector representation of documents. For a word w_t in a document, the set of its following and preceding words $C^t = \{w_{c_1}, w_{c_2}, \ldots, w_{c_{|C^t|}}\}$ constitute the context of that word. Given the target word w_t and its context C^t, the goal is to maximize the probability of predicting w_t given C^t, $p(w_t|C^t)$. Now for all the target words in the vocabulary $\mathcal{V}$, the objective of the framework is to maximize the average probability

$$\frac{1}{|\mathcal{V}|} \sum_{w_t \in \mathcal{V}} p(w_t|C^t)$$

To make the prediction model log linear, instead of maximizing the average probability, the framework maximize the average log probability

$$\frac{1}{|\mathcal{V}|} \sum_{w_t \in \mathcal{V}} log \quad p(w_t|C_t) \tag{6.2}$$

A log-linear classification model such as softmax is used to obtain the posterior probability distribution of words

$$p(w_t|C^t) = \frac{exp(y_t)}{\sum_i^{|\mathcal{V}|} exp(y_i)}$$

where y_i is the log probability of i-th word in the vocabulary $\mathcal{V}$. The posterior probabilities are computed as

$$y = b + UW^T x^w \tag{6.3}$$

where b and U are the parameters for the log linear model. The i-th row of the matrix W represent the vector for the i-th word in the vocabulary $\mathcal{V}$. The input vector x^w is a $|\mathcal{V}|$ dimensional vector, where x_i^w will be $\frac{1}{|C^t|}$ if $w_i \in C^t$, 0 otherwise.

The model gets trained usually by stochastic gradient descent (SGD) and back-propagation. This type of model is known as neural network based language model [11]. Although we

initialize W by random numbers, semantically similar words will have similar vector representations after the training converges.

Learning Document Vectors by Exploiting Person Context:

While learning word vectors, the vector representation of documents can also be learned by using the same objective function in Equation 6.2. The inclusion of a matrix D for documents in Equation 6.3 allows the model to learn the matrix D such that documents with semantically similar textual content get mapped to similar vectors. The i-th row of D is the vector representation of the i-th document in $\mathcal{D}$. The inclusion of D is shown in Equation 6.4 where $\|$ represents vector concatenation.

$$y = b + U((W^T x^w) \| (D^T x^d)) \tag{6.4}$$

The input vector x^d for an i-th document is a $|\mathcal{D}|$ dimensional vector where the i-th element is 1, and all other elements are zeros.

So far in Equation 6.4, we have seen the exploitation of semantic meaning of words in representing documents. But documents frequently contain non-textual content, such as images of persons, that are usually overlooked in document representation. This framework introduces a way to harness contextual information of the faces in document representation. In Section 6.2.1, we showed the methodology for extracting entity distributions F of faces. Now, we include the matrix F in Equation 6.4 to get the modified forward-propagation equation as:

$$y = b + U((W^T x^w) \| (D^T x^d) \| (F^T x^f)) \tag{6.5}$$

where x^f is the $|\mathcal{F}|$ dimensional input vector for faces.

The SGD optimizes the objective function in Equation 6.2 that uses the modified Equation 6.5. It is important to note that the matrix F does not get modified, unlike matrices D and W, during the training phase. A graphical representation of the Equation 6.5 is presented in Figure 6.6. There are three matrices W, D and F in between input and hidden

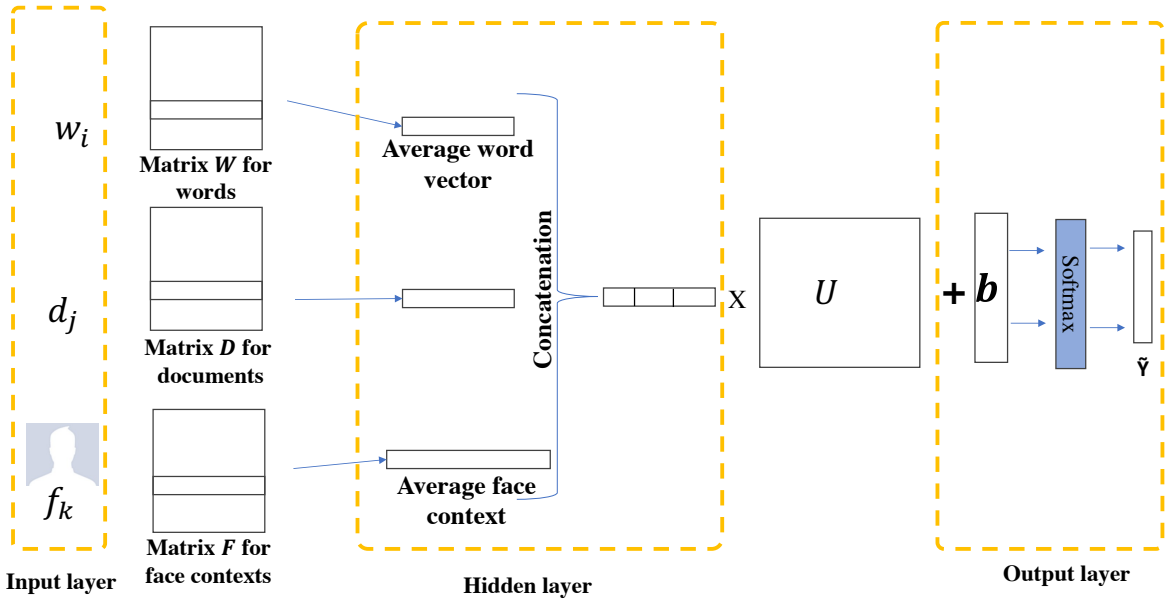


Figure 6.6: Graphical representation illustrating the model in Equation 6.5.

layers. The hidden layer consists of a concatenation of three vectors. The first vector is the average word vector of the words in C^t and can be produced by multiplying transpose of W and x^w. To avoid the expensive matrix-vector product, we usually do look up in the matrix W. The second vector is for the document d_j where the target word w_t and its context C^t belong. The third vector is the average vector for the context of faces that belong to the same document d_j. Then a matrix-vector product of softmax parameter U and the concatenated vector yields a vector for the output layer, where we add another softmax parameter b to produce un-normalized probabilities for the softmax function.

Complementary Document Vectors:

In this chapter, we deal with only the documents that contain one or more images. When there are some images in a document, the information in the textual content usually get reduced, which is evident in Figure 6.1. To exploit text fragments that are not present

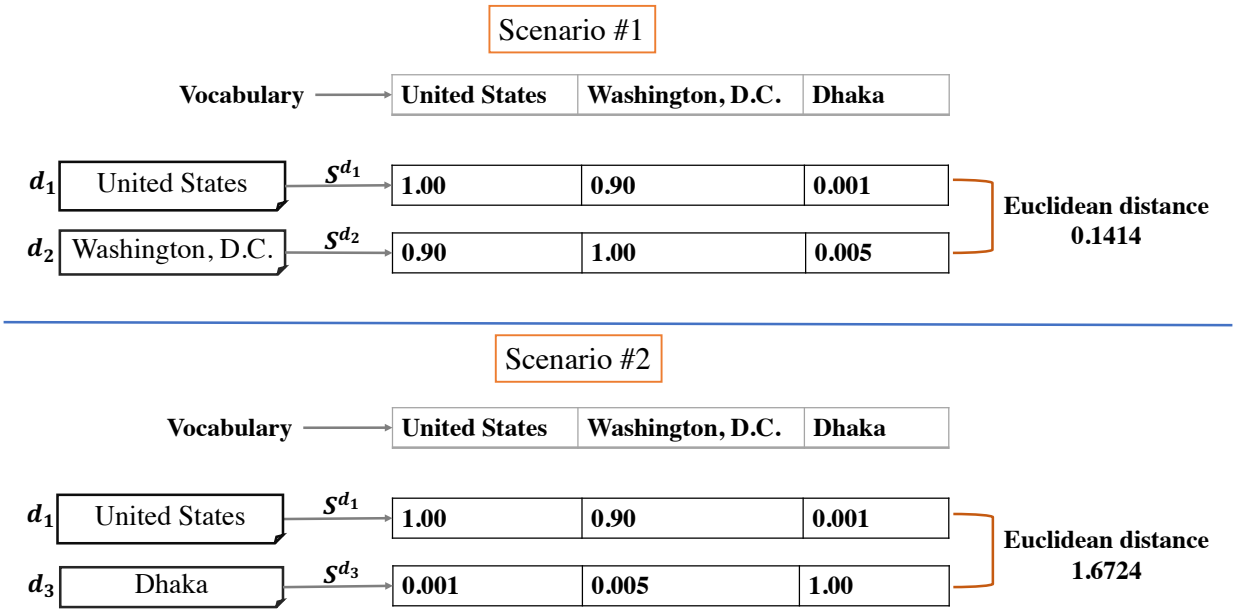


Figure 6.7: Document features modeling based on similarity of entity vectors.

in a document but similar to the text text fragments in that document, we introduce a complementary representation of text. Let $\mathcal{E}^d$ be the set of entities in document d. Then the entity similarity based document vector S^d for a document $d \in \mathcal{D}$ is defined as $S^d = \{s_1^d, s_2^d, \ldots, s_{|\mathcal{E}|}^d\}$. We compute s_i^d as

$$s_i^d = \frac{1}{|\mathcal{E}^d|} \sum_{j, e_j \in \mathcal{E}^d} \frac{V^{e_i} \cdot V^{e_j}}{\|V^{e_i}\| \|V^{e_j}\|} \tag{6.6}$$

where e_i is the i-th entity in $\mathcal{E}$. In essence, Equation 6.6 calculates average cosine similarity between the vector of e_i and the vectors of all the entities in document d.

Figure 6.7 shows two scenarios demonstrating the effectiveness of the entity based representation in distinguishing documents. There are three documents d_1, d_2 and d_3 and each of them contains only single entity. The values in the vectors are the similarity scores between the entities of these documents and entities in the vocabulary. In the first scenario, the vectors of the documents d_1 and d_2 look very close in euclidean space and this reflects well with the contents of these documents. In the second scenario, the vectors are far away

in euclidean space, which rightly indicates the contextual dissimilarity in the contents of the documents d_1 and d_3.

6.3 Experimental Results

We are using four different datasets to conduct experiments for justifying our contributions. The first dataset contains 36,000 news articles from the New York Times where each article belongs to one of the five categories: U.S., World, Sports, Arts, and Business. All the articles contain both textual and imagery content and published in 2016. We downloaded these news articles from the New York Times archive [4] by using python script and several python modules, e.g., urllib2 and Beautiful Soup, to handle HTTP request and parsing HTML data. For majority of the experiments, we use New York Times dataset unless it is mentioned explicitly. The other three datasets are the Twitter Sentiment Analysis Dataset [5], sentiment classification dataset from the University of Michigan [6] and Quora Question Pairs Dataset [3]. We evaluate our methods on three different information retrieval tasks: document classification, document clustering and document similarity. The research questions we seek to answer in this chapter are as follows:

1. How good are the document vectors in separating documents between categories? (Section 6.3.1)

2. How effective are the person contexts in representing documents for document classification? (Section 6.3.2)

3. Are document vectors generated by utilizing person context useful for document clustering? (Section 6.3.3)

4. Do entity similarity based vectors complement the person context based representation? (Section 6.3.4)

Before diving into the details of the experiments, we are describing the abbreviations used throughout the experimental section. The abbreviation *DocVecPC* and *DocVec* denote

two document representation techniques in which person context is exploited and ignored, respectively. The *DocVec* uses only textual content and is equivalent to the baseline method doc2vec [74]. The term frequency-inverse document frequency based vector space representation of documents is denoted as *TFIDF* [78] and considered as the second baseline method. The methods *DocVec* and *DocVecPC* generate document vectors using the models in Equation 6.4 and 6.5, respectively. Finally, the complementary document representation that utilizes contextual similarity of text fragments is abbreviated as *DocVecES*.

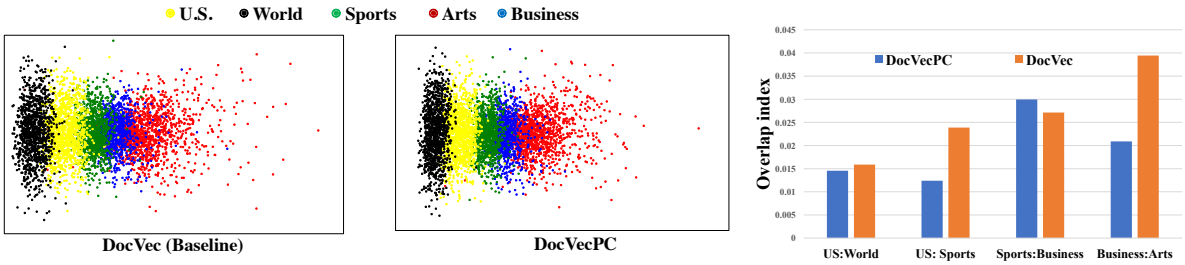


Figure 6.8: First two principal components of the document vectors and the overlap of the components between categories.

6.3.1 Separability of Document Vectors

To understand the distinguishing capability of the document vectors, we plot first two principal components of the vectors generated by *DocVec* and *DocVecPC* in Figure 6.8. The plots portray that the document vectors generated by the method *DocVecPC* contain more distinctive features and are capable of differentiating categories of documents better than that of *DocVec*. In the left plot, there are more overlaps between the categories than the middle plot. The column plot in Figure 6.8 (right) presents overlap index of four pairs of categories for both of the document vectorization methods shown in left and middle plots. The overlap index is a measure to quantify the overlaps present between the adjacent categories, and a larger value of that index indicates more overlap. We compute overlap

index value for two sets of 2-D points as follows:

$$\text{Overlap index} = \frac{\text{Number of common points}}{\text{Total number of points in both sets}}$$

where a point x is common in both sets if there is a point y in the other set and the euclidean distance between x and y is smaller than $\epsilon = 10^{-2}$. I experimented with different values of ϵ and found similar patterns for them.

The results in this section signify that contextual information of persons depicted in images can be utilized for a better representation of documents that reduces overlap between document categories. The compactness of each category in that figure also indicates that our proposed method *DocVecPC* helps to reduce the intra-category distance of documents.

6.3.2 Person Context for Document Classification

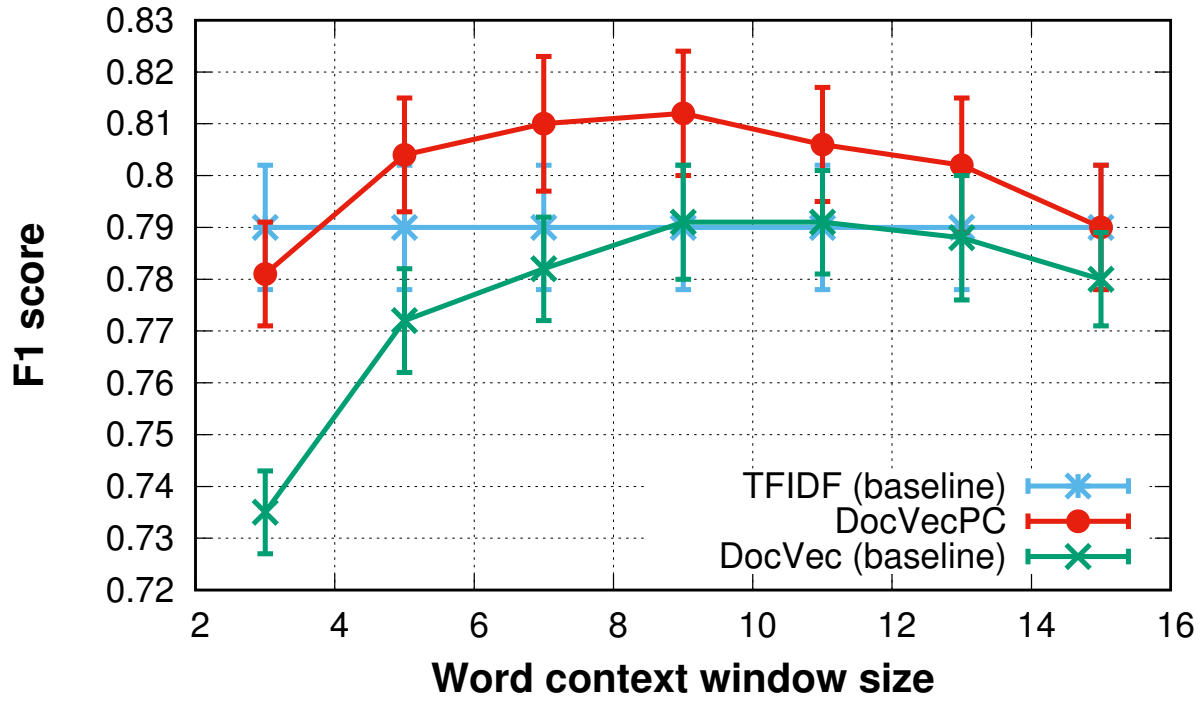


Figure 6.9: Comparison of three document representation techniques for document classification task.

In this experiment, we evaluate the impact of person contexts on document representations through a classification task. We now discuss the performance of three document representation techniques in more detail. We use one-vs-rest logistic regression classifier to classify documents using document representations *DocVecPC*, *DocVec* and *TFIDF* as feature vectors. We compare our document representation *DocVecPC* against two baseline representations *DocVec* and *TFIDF* in Figure 6.9. Figure 6.9 shows that our *DocVecPC* outperforms both of the baseline representations. The performance of *DocVecPC* and *DocVec* depends on the context window size, which is the number of words to be considered as context to predict a target word using the objective function in Equation 6.2. It is evident in the Figure 6.9 that a certain range of values for window size produces best result. Too small window can not capture enough context and too large window brings noisy context in the document representations.

6.3.3 Evaluation using Clusters

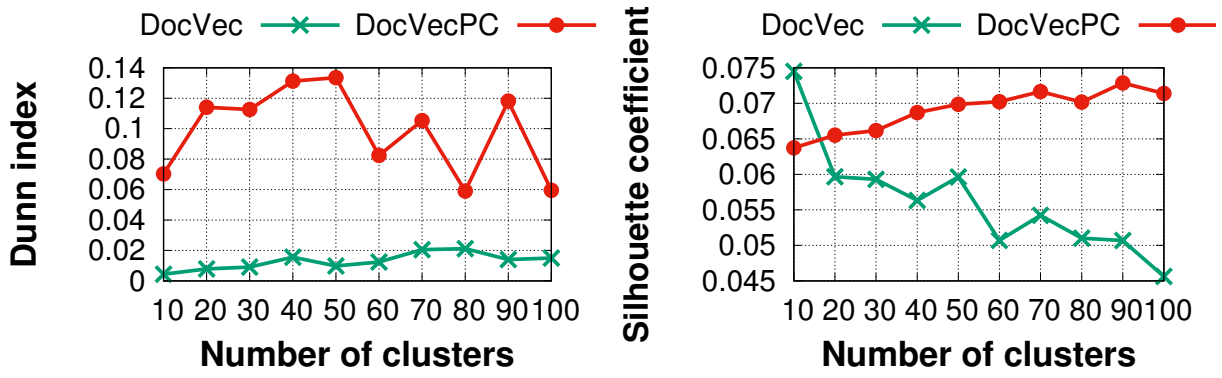


Figure 6.10: Clustering based evaluation of document vectors using two internal evaluation schemes. Both of the Dunn index and Silhoutte coefficient indicate *DocVecPC* contains useful features for better clustering of documents.

In this experiment, we evaluate the document vectors in terms of clustering quality. We cluster the documents using k-means clustering algorithm, given the vector representations of the documents. We measure the quality of the clusters using two standard cluster

evaluation measures: Silhouette coefficient [98] and Dunn index [33]. For both of the cluster evaluation methods, we use Euclidean distance as the distance metric. The higher values of Silhouette coefficient and Dunn index indicate better quality of clusters. Figure 6.10 (left) shows that our vector representation *DocVecPC* performs significantly better than the baseline representation *DocVec* in terms of Dunn index for any number of clusters. In Figure 6.10 (right), the positive coefficient values for both methods indicate that the clustering configuration is appropriate. The Figure 6.10 (right) also shows that the proposed *DocVecPC* performs better than *DocVec* when the number of clusters is more than fifteen.

6.3.4 Impact of Complementary Document Representation

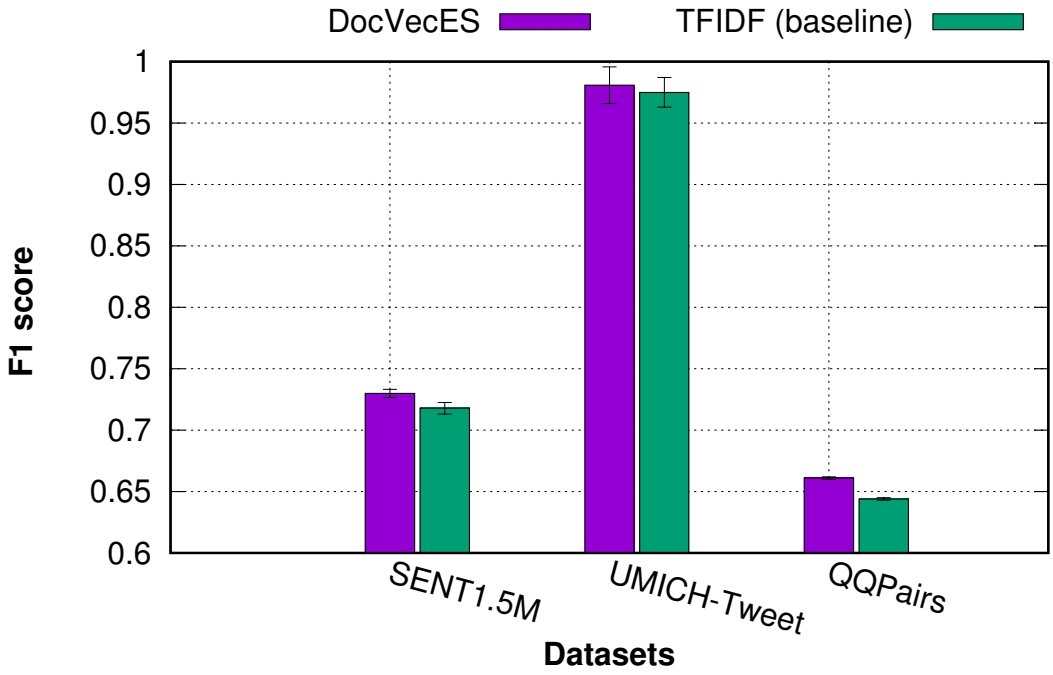


Figure 6.11: It compares the performance of *DocVecES* and *TFIDF* on three datasets of short documents for classification tasks.

In this experiment, we analyze the performance of our document representation *DocVecES* and its effectiveness. We evaluate the individual performance of *DocVecES* on three

different datasets: (1) Twitter Sentiment Analysis Dataset *SENT1.5M*, (2) Sentiment Analysis Dataset *UMICH-Tweet* by University of Michigan and (3) Question pair dataset *QQPairs* by Quora. The datasets *SENT1.5M* and *UMICH-Tweet* contain 1,578,627 and 7,086 labeled tweets, respectively. The *QQPairs* dataset contains more than 400,000 question pairs.

In Figure 6.11, we show results for predicting positive and negative sentiments for the *SENT1.5M* and *UMICH-Tweet* datasets. It also shows the performance of our *DocVecES* in identifying duplicate questions in *QQPairs* dataset. We compare our *DocVecES* representation against *TFIDF* in terms of F1 score with 5-fold cross validation and logistic regression. As can be seen in Figure 6.11, *DocVecES* produces slightly better results for classification task in all datasets. We also observed that *DocVecES* is mostly effective for short documents. A possible reason for the superiority of *DocVecES* over *TFIDF* would be that *TFIDF* depends on the frequency of words in a document and short documents such as tweet usually does not contain a word twice. However, our *DocVecES* exploits words that are not present in a document but similar to the words in that document.

We observe that a bigger impact of the complementary representation *DocVecES* can be achieved by incorporating it into *DocVecPC*. We combine *DocVecPC* with *DocVecES* and *TFIDF* to identify the combination that performs better in document classification task. The term combination in this context denotes a concatenation. We use one-vs-rest logistic regression and support vector machine with RBF kernel to classify documents in New York Times dataset. Figure 6.12 shows classification performances in terms of F1 score with micro averaging for all the combinations of document representations. The evaluation is performed with 10-fold cross validation. The combinations of document representations produce superior performance in classification task compared to the individual representations. In Figure 6.12 (left), we compare the performance of *DocVecPC+DocVecES* and *DocVecPC+TFIDF* against *DocVecPC* for the logistic regression classifier. We see that incorporating the *TFIDF* boosts the performance of *DocVecPC*. However, incorporation of our complementary representation *DocVecES* gives even better F1 score. Figure 6.12 (right) shows the performance of those combinations for SVM classifier with non linear (RBF) ker-

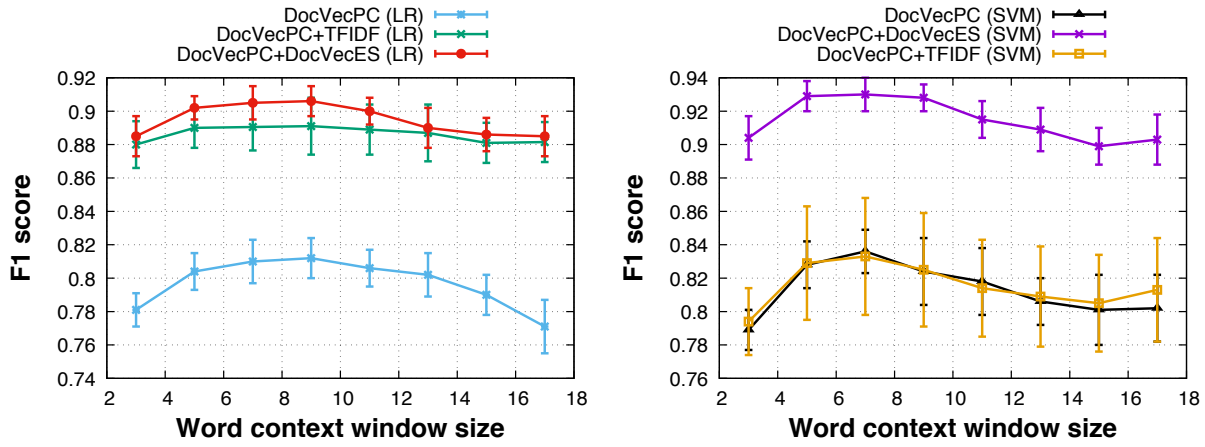


Figure 6.12: Comparison of the combinations of document representations. The name of the classifiers is given in a parenthesis with legend.

nel. In this case, the classifier could not utilize the *TFIDF* when combined with *DocVecPC*. However, the incorporation of our *DocVecES* in *DocVecPC* provides a huge boost in terms of F1 score. It is even better than the best performance in the left plot.

Finally, it is evident from the Figure 6.12 that the combination of *DocVecPC* and *DocVecES* produces better classification results and is effective for both linear and non-linear classifier.

6.4 Summary

In this chapter, we propose a neural language model for learning multimodal document representation. Our model represents each document as latent variables and exploit context of persons depicted in images. An additional text representation method is also presented that complements the multimodal language representation.

Chapter 7

Conclusions

In this dissertation, I introduced representation techniques for documents, text fragments, and images (in particular faces). The proposed document representation technique uses a neural language model that takes contextual information extracted from imagery and textual content as its input. Empirical studies presented in this dissertation demonstrate that my approach to represent documents combining content, text fragments, and images result in better outcomes in many machine learning applications. Examples of such machine learning applications include classification, document indexing, and similarity search.

In Chapter 3, I introduced an objective function that assisted contextual information extraction, i.e., relationships between text fragments, from textual content. Later in that chapter, I demonstrated two techniques for constructing embeddings of text fragments from the contextual relationships extracted by my objective functions. Most of the existing techniques for text fragment or word representation utilize co-occurrence of text fragments in a sentence or document. Therefore, they cannot capture the relationships when two text fragments are related but residing in two different documents. Moreover, the existing techniques require a huge amount of data to produce a high-quality representation of text fragments. In contrast, my technique is capable of extracting relationships between text fragments regardless of their presence in a document. My method utilizes temporal, geographical, and topical information of the documents as opposed to co-occurrence based techniques. Even though my model is suitable for and applicable to big data, it does not require massive amount of documents of the same type to generate high-quality representations for text fragments.

Chapter 4 and Chapter 5 demonstrated context generation techniques for visual objects,

in particular, human faces. In Chapter 4, I described a method that detects faces from images and a facial-feature extraction method. In Chapter 5, I have introduced several probabilistic models that construct three kinds of contexts for images of people. The three kinds of contexts include related person names, locations, and countries. There is not current literature on such context generation for faces, the best of my knowledge. The focus of the current literature is mostly on tagging a face with the name of the person when the name is present in the associated text. My technique produces a holistic context for every face and does not limit its capability to tagging by one name for each face. Moreover, my technique is capable of generating contextual information of a face even if the name of the person does not appear in the text.

In Chapter 6, I presented a neural language model for document representation. The proposed model exploits different modalities of context extracted using the techniques presented in the preceding chapters in a unified document representation. It produces a compact vector-based representation of documents in a continuous space. All existing document representation techniques utilize the textual content of the documents alone. My technique utilizes imagery and textual content together in constructing a vector representation for each document. Unlike existing document representation techniques, my technique is able to exploit the contextual information constructed for text fragments as well as faces of people. Chapter 6 also demonstrates the effectiveness of the document representations in several information retrieval tasks, including document classification and clustering.

Finally, I have two broad directions for future research work: (1) prediction of events using unstructured text and (2) construction of contextual representations of video fragments. The following two subsections outline these future directions.

7.1 Event prediction

Although this dissertation presented a technique to extract relationships between entities by not limiting their co-occurrence in a sentence or document, the question remains

unanswered whether these relationships are capable of capturing attributes to predict the appearance of an entity in the future. I would like to extend my investigation toward predicting entities that may appear in the future.

My current neural network model does not include the time parameter, which prohibits it from predicting a time-frame for each of the observed relationships. Being able to predict the time of the end-entity of a relationship given one entity, will address the *when* analytic aspect of event prediction. Additionally, inclusion of an explicit location parameter in the neural network will address the *where* analytic aspect. The objective would be to predict whether the appearance of a set of entities may result in the appearance of another set of entities in future in a particular geographical location. For example, will the appearance of some drug-lords result in drug war near US-Mexico borders? *When* and **where** aspects are crucial in such analyses.

The prediction of the appearance of an entity further can be extended to predict the occurrence of an event. Before going for the prediction of an event, it is necessary to be able to define an event and detect it from the unstructured documents. Unstructured social media posts are a great source of information for detecting events. For example, a sudden rise of tweets and facebook posts can be used to detect the occurrence of an event. The event could be defined as a set of entities associated with particular time-dependent rise of number of social media posts. My goal is to find event-event abstract relationships like *Natural disaster—Disease outbreak*, *Drug curtail—Drug-related crime*, and many other abstract relationships. This will allow me to forecast an event-type that may appear in the future when a particular event-type is observed in the present days.

7.2 Contextual representation of videos

Multimedia documents nowadays contain a lot of videos as well as images. With the growth of news media, a large archive videos, text, and images have become available on the web. A video may contain multiple topics of discussions. For example, a news broadcast may cover

news relevant to politics, fashion, sports, weather forecast, or any other topic. Current state of the art literature for video analytics does not include contextual representation for videos. How can I represent video segments of a news by a vector just like I did for documents? Such contextual representations of news segments will help classify videos contextually, index them, and search similar news videos covering the same context.

A news video is complex in the sense that the face appearing in the video is describing another context. Therefore, face similarity aspects are useless in video context. Moreover, the audio associated with video is more contextual to the news. The probabilistic model presented in this dissertation to construct contextual information of human faces can be combined with a speech-to-text system to produce a representation for videos.

The goal of the representation would be to represent every news video segments in a continuous space such that the representation captures the context of each news. An application of such a representation is a recommendation system for videos. The new video recommendation system will be able to suggest better contextual videos.

References

[1] CNN news. http://www.cnn.com/. Accessed: 2017-07-04.

[2] Entropy (information theory). https://en.wikipedia.org/wiki/Entropy_(information_theory). Accessed: 2017-07-04.

[3] First quora dataset release: Question pairs. https://data.quora.com/First-Quora-Dataset-Release-Question-Pairs. Accessed: 2017-07-04.

[4] The New York Times archive. http://www.nytimes.com/ref/membercenter/nytarchive.html. Accessed: 2017-07-04.

[5] Twitter sentiment analysis training corpus. http://thinknook.com/twitter-sentiment-analysis-training-corpus-dataset-2012-09-22/. Accessed: 2017-07-04.

[6] Umich si650 - sentiment classification. https://inclass.kaggle.com/c/si650winter11. Accessed: 2017-07-04.

[7] Timo Ahonen, Abdenour Hadid, and Matti Pietikainen. Face description with local binary patterns: Application to face recognition. *PAMI*, 28(12):2037–2041, 2006.

[8] Alias-i. LingPipe 4.1.0. Accessed: Feb 07, '16, http://alias-i.com/lingpipe/.

[9] Gabor Angeli, Melvin Johnson Premkumar, and Christopher D Manning. Leveraging linguistic structure for open domain information extraction. In *ACL-IJCNLP*, 2015.

[10] Apache Software Foundation. OpenNLP. Accessed: Feb 07, '16, http://opennlp.apache.org.

[11] Yoshua Bengio, Réjean Ducharme, Pascal Vincent, and Christian Janvin. A neural probabilistic language model. *Machine Learning Research*, 3:1137–1155, 2003.

[12] Eric A Bier, Edward W Ishak, and Ed Chi. Entity workspace: an evidence file that aids memory, inference, and reading. In *ISI*, pages 466–472. 2006.

[13] David C Blair and Melvin E Maron. An evaluation of retrieval effectiveness for a full-text document-retrieval system. *Communications of the ACM*, 28(3):289–299, 1985.

[14] David M. Blei, Andrew Y. Ng, and Michael I. Jordan. Latent dirichlet allocation. *JMLR*, 3:993–1022, March '03.

[15] David M Blei, Andrew Y Ng, and Michael I Jordan. Latent dirichlet allocation. *Journal of machine Learning research*, 3(Jan):993–1022, 2003.

[16] David M. Blei, Andrew Y. Ng, and Michael I. Jordan. Latent dirichlet allocation. *JMLR*, 2003.

[17] Elia Bruni, Giang Binh Tran, and Marco Baroni. Distributional semantics from text and images. In *Proceedings of the GEMS 2011 workshop on geometrical models of natural language semantics*, pages 22–32. Association for Computational Linguistics, 2011.

[18] Elia Bruni, Nam-Khanh Tran, and Marco Baroni. Multimodal distributional semantics. *J. Artif. Intell. Res.(JAIR)*, 49(2014):1–47, 2014.

[19] Deng Cai, Shipeng Yu, Ji-Rong Wen, and Wei-Ying Ma. Vips: a vision-based page segmentation algorithm. Technical report, MSR-TR-2003-79, 2003.

[20] William B Cavnar, John M Trenkle, et al. N-gram-based text categorization. *Ann Arbor MI*, 48113(2):161–175, 1994.

[21] David J Chalmers. Syntactic transformations on distributed representations. In *Connectionist Natural Language Processing*, pages 46–55. 1992.

[22] Danqi Chen and Christopher D Manning. A fast and accurate dependency parser using neural networks. In *EMNLP*, pages 740–750, 2014.

[23] Longbin Chen, Baogang Hu, Lei Zhang, Mingjing Li, and HongJiang Zhang. Face annotation for family photo album management. *IJIG*, 3(01):81–94, 2003.

[24] Mengen Chen, Xiaoming Jin, and Dou Shen. Short text classification improved by learning multi-granularity topics. In *IJCAI*, pages 1776–1781, 2011.

[25] Jae Young Choi, Wesley De Neve, Yong Man Ro, and Konstantinos N Plataniotis. Automatic face annotation in personal photo collections using context-based unsupervised clustering and face information fusion. *IEEE Transactions on Circuits and systems for Video Technology*, 20(10):1292–1309, 2010.

[26] Jae Young Choi, Seungji Yang, Yong Man Ro, and Konstantinos N Plataniotis. Face annotation for personal photos using context-assisted face recognition. In *Proceedings of the 1st ACM international conference on Multimedia information retrieval*, pages 44–51. ACM, 2008.

[27] Fan RK Chung. *Spectral graph theory*, volume 92. 1997.

[28] National Research Council. *Making the Nation Safer: The Role of Science and Technology in Countering Terrorism*. The National Academies Press, Washington, DC, 2002.

[29] Gabriella Csurka, Christopher Dance, Lixin Fan, Jutta Willamowski, and Cédric Bray. Visual categorization with bags of keypoints. In *Workshop on statistical learning in computer vision, ECCV*, volume 1, pages 1–2. Prague, 2004.

[30] Scott Deerwester, Susan T Dumais, George W Furnas, Thomas K Landauer, and Richard Harshman. Indexing by latent semantic analysis. *Journal of the American society for information science*, 41(6):391, 1990.

[31] Susan T Dumais. Latent semantic analysis. *Annual review of information science and technology*, 38(1):188–230, 2004.

[32] Joseph C Dunn. A fuzzy relative of the isodata process and its use in detecting compact well-separated clusters. 1973.

[33] Joseph C Dunn. A fuzzy relative of the isodata process and its use in detecting compact well-separated clusters. 1973.

[34] Jeffrey L Elman. Distributed representations, simple recurrent networks, and grammatical structure. *Machine Learning*, 7(2-3):195–225, 1991.

[35] Martin Ester, Hans peter Kriegel, Jrg S, and Xiaowei Xu. A density-based algorithm for discovering clusters in large spatial databases with noise. In *KDD*, 1996.

[36] Fariza Fauzi, Jer-Lang Hong, and Mohammed Belkhatir. Webpage segmentation for extracting images and their surrounding contextual information. In *MM*, 2009.

[37] Fangxiang Feng, Xiaojie Wang, and Ruifan Li. Cross-modal retrieval with correspondence autoencoder. In *Proceedings of the 22nd ACM international conference on Multimedia*, pages 7–16. ACM, 2014.

[38] SL Feng, Raghavan Manmatha, and Victor Lavrenko. Multiple bernoulli relevance models for image and video annotation. In *Computer Vision and Pattern Recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE Computer Society Conference on*, volume 2, pages II–II. IEEE, 2004.

[39] Yansong Feng and Mirella Lapata. Automatic image annotation using auxiliary text information. In *ACL*, volume 8, pages 272–280, 2008.

[40] Jenny Rose Finkel, Trond Grenager, and Christopher Manning. Incorporating non-local information into information extraction systems by gibbs sampling. In *ACL*, 2005.

[41] FMS Advanced Systems Group, FMS Inc. Sentinel Visualizer. Accessed: April 07, '16, www.fmsasg.com.

[42] Eibe Frank, Gordon W Paynter, Ian H Witten, Carl Gutwin, and Craig G Nevill-Manning. Domain-specific keyphrase extraction. In *16th International Joint Conference on Artificial Intelligence (IJCAI 99)*, volume 2, pages 668–673. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1999.

[43] Andrew C Gallagher and Tsuhan Chen. Using context to recognize people in consumer images. *IPSJ Transactions on Computer Vision and Applications*, 1:115–126, 2009.

[44] GeoDataSource. World Cities Database. Accessed: Feb 07,'16, www.geodatasource.com/world-cities-database.

[45] United States Government. A tradecraft primer: Structured analytic techniques for improving intelligence analysis. *CIA CSI*, 2009.

[46] Matthieu Guillaumin, Thomas Mensink, Jakob Verbeek, and Cordelia Schmid. Face recognition from caption-based supervision. *International Journal of Computer Vision*, 96(1):64, 2012.

[47] Zellig S Harris. Distributional structure. *Word*, 10(2-3):146–162, 1954.

[48] Tal Hassner, Shai Harel, Eran Paz, and Roee Enbar. Effective face frontalization in unconstrained images. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4295–4304, 2015.

[49] Richards Heuer. *Psychology of intelligence analysis*. CIA, 99.

[50] Geoffrey E Hinton. Learning distributed representations of concepts. In *CogSci'86*, 1986.

[51] M. Shahriar Hossain, Patrick Butler, Arnold P. Boedihardjo, and Naren Ramakrishnan. Storytelling in entity networks to support intelligence analysts. In *KDD '12*, 2012.

[52] M. Shahriar Hossain, Patrick Butler, Arnold P. Boedihardjo, and Naren Ramakrishnan. Storytelling in entity networks to support intelligence analysts. In *KDD '12*, 2012.

[53] Xia Hu, Nan Sun, Chao Zhang, and Tat-Seng Chua. Exploiting internal and external semantics for the clustering of short texts using world knowledge. In *Proceedings of the 18th ACM conference on Information and knowledge management*, pages 919–928. ACM, 2009.

[54] Chang Huang, Haizhou Ai, Yuan Li, and Shihong Lao. High-performance rotation invariant multiview face detection. *TPAMI*, 29(4):671–686, 2007.

[55] Eric H Huang, Richard Socher, Christopher D Manning, and Andrew Y Ng. Improving word representations via global context and multiple word prototypes. In *ACL*, 2012.

[56] Gary B Huang, Manu Ramesh, Tamara Berg, and Erik Learned-Miller. Labeled faces in the wild: A database for studying face recognition in unconstrained environments. Technical report, Technical Report 07-49, University of Massachusetts, Amherst, 2007.

[57] John E Hummel and Keith J Holyoak. Distributed representations of structure: A theory of analogical access and mapping. *Psychological Review*, 104(3):427, 1997.

[58] IBM. IBM Analytics for a Safer Planet. Accessed: Feb 07,'16, www.ibm.com/analytics/us/en/safer-planet/.

[59] Giridharan Iyengar, Pinar Duygulu, Shaolei Feng, Pavel Ircing, SP Khudanpur, Dietrich Klakow, MR Krause, Raghavan Manmatha, Harriet J Nock, D Petkova, et al. Joint visual-text modeling for automatic retrieval of multimedia documents. In *Proceedings of the 13th annual ACM international conference on Multimedia*, pages 21–30. ACM, 2005.

[60] Md Abdul Kader, Arnold P Boedihardjo, Sheikh Motahar Naim, and M Shahriar Hossain. Contextual embedding for distributed representations of entities in a text corpus. In *KDD BigMine*. PMLR, 2016.

[61] Md Abdul Kader, Sheikh Motahar Naim, Arnold P Boedihardjo, and M Shahriar Hossain. Connecting the dots using contextual information hidden in text and images. In *Thirtieth AAAI Conference on Artificial Intelligence*, 2016.

[62] P.R. Kalva, F. Enembreck, and A.L. Koerich. Web image classification based on the fusion of image and text classifiers. In *ICDAR*, volume 1, pages 561–568, 2007.

[63] Hyunmo Kang, Catherine Plaisant, Bongshin Lee, and Benjamin B Bederson. Netlens: iterative exploration of content-actor network data. *Information Visualization*, 6(1):18–31, 2007.

[64] Andrej Karpathy and Li Fei-Fei. Deep visual-semantic alignments for generating image descriptions. In *CVPR*, pages 3128–3137, 2015.

[65] Andrej Karpathy, Armand Joulin, and Fei Fei F Li. Deep fragment embeddings for bidirectional image sentence mapping. In *Advances in neural information processing systems*, pages 1889–1897, 2014.

[66] Sheila Kinsella, Vanessa Murdock, and Neil O'Hare. "i'm eating a sandwich in glasgow": Modeling locations with tweets. In *SMUC*, pages 61–68, 2011.

[67] Ryan Kiros, Ruslan Salakhutdinov, and Rich Zemel. Multimodal neural language models. In *Proceedings of the 31st International Conference on Machine Learning (ICML-14)*, pages 595–603, 2014.

[68] S. Kullback and R. A. Leibler. On information and sufficiency. *Ann. Math. Statist.*, 1951.

[69] Giridhar Kumaran and James Allan. Text classification and named entities for new event detection. In *Proceedings of the 27th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 297–304. ACM, 2004.

[70] Jian Huang Lai, Pong C Yuen, and Guo Can Feng. Face recognition using holistic fourier invariant features. *Pattern Recognition*, 34(1):95–109, 2001.

[71] Frederick Wilfrid Lancaster and Emily Gallup. Information retrieval on-line. Technical report, 1973.

[72] Ken Lang. Newsweeder: Learning to filter netnews. In *ML95*, pages 331–339, 1995.

[73] Duy-Dinh Le and Shin'ichi Satoh. Unsupervised face annotation by mining the web. In *Data Mining, 2008. ICDM'08. Eighth IEEE International Conference on*, pages 383–392. IEEE, 2008.

[74] Quoc V Le and Tomas Mikolov. Distributed representations of sentences and documents. In *ICML'14*, pages 1188–1196, 2014.

[75] Hyung-Ji Lee, Wan-Su Lee, and Jae-Ho Chung. Face recognition using fisherface algorithm and elastic graph matching. In *ICIP*, pages 998–1001, '01.

[76] Haoxiang Li, Zhe Lin, Xiaohui Shen, Jonathan Brandt, and Gang Hua. A convolutional neural network cascade for face detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5325–5334, 2015.

[77] Xutao Li, Gao Cong, Xiao-Li Li, Tuan-Anh Nguyen Pham, and Shonali Krishnaswamy. Rank-geofm: A ranking based geographical factorization method for point of interest recommendation. In *SIGIR '15*, pages 433–442, 2015.

[78] Christopher D Manning, Prabhakar Raghavan, and Hinrich Schütze. Scoring, term weighting and the vector space model. *Introduction to information retrieval*, 100:2–4, 2008.

[79] Christopher D Manning, Prabhakar Raghavan, Hinrich Schütze, et al. *Introduction to information retrieval*, volume 1. Cambridge university press Cambridge, 2008.

[80] Christopher D Manning, Hinrich Schütze, et al. *Foundations of statistical natural language processing*, volume 999. MIT Press, 1999.

[81] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*, 2013.

[82] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. Distributed representations of words and phrases and their compositionality. In *NIPS'13*, 2013.

[83] Tomas Mikolov, Wen-tau Yih, and Geoffrey Zweig. Linguistic regularities in continuous space word representations. In *HLT-NAACL*, pages 746–751, 2013.

[84] A. Mishra, N. Mishra, and A. Agrawal. Context-aware restricted geographical domain question answering system. In *CICN '10*, pages 548–553, 2010.

[85] Pacific Northwest National Laboratory. IN-SPIRE Visual Document Analysis. Accessed: Feb 07, '16, http://in-spire.pnnl.gov/.

[86] Palantir Technologies. Palantir Gotham. Accessed: Feb 07, '16, www.palantir.com/palantir-gotham/.

[87] Moon-Hee Park, Jin-Hyuk Hong, and Sung-Bae Cho. Location-based recommendation system using bayesian user's preference model in mobile devices. In *UIC*, volume 4611, pages 1130–1139. 2007.

[88] Omkar M Parkhi, Andrea Vedaldi, and Andrew Zisserman. Deep face recognition. *BMVC*, 1(3):6, 2015.

[89] Jeffrey Pennington, Richard Socher, and Christopher D Manning. Glove: Global vectors for word representation. In *EMNLP*, volume 14, pages 1532–1543, 2014.

[90] Xuan-Hieu Phan, Le-Minh Nguyen, and Susumu Horiguchi. Learning to classify short and sparse text & web with hidden topics from large-scale data collections. In *Proceedings of the 17th international conference on World Wide Web*, pages 91–100. ACM, 2008.

[91] Jordan B Pollack. Recursive distributed representations. *Artificial Intelligence*, 46(1), 1990.

[92] Princeton University. WordNet. `http://wordnet.princeton.edu`. Accessed: 2017-07-04.

[93] Shafin Rahman, Sheikh Motahar Naim, Abdullah Al Farooq, and Md Monirul Islam. Performance of mpeg-7 edge histogram descriptor in face recognition using principal component analysis. In *ICCIT*, pages 476–481, 2010.

[94] Shafin Rahman, Sheikh Motahar Naim, Abdullah Al Farooq, and Md Monirul Islam. Combination of gabor and curvelet texture features forface recognition using principal component analysis. *IACSIT*, 4(3):264, 2012.

[95] Daniel Ramage, David Hall, Ramesh Nallapati, and Christopher D Manning. Labeled lda: A supervised topic model for credit attribution in multi-labeled corpora. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 1-Volume 1*, pages 248–256. Association for Computational Linguistics, 2009.

[96] Nikhil Rasiwasia, Jose Costa Pereira, Emanuele Coviello, Gabriel Doyle, Gert RG Lanckriet, Roger Levy, and Nuno Vasconcelos. A new approach to cross-modal multimedia retrieval. In *Proceedings of the 18th ACM international conference on Multimedia*, pages 251–260. ACM, 2010.

[97] Peter J Rousseeuw. Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. *Journal of computational and applied mathematics*, 20:53–65, 1987.

[98] Peter J Rousseeuw. Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. *Journal of computational and applied mathematics*, 20:53–65, 1987.

[99] David E Rumelhart, Geoffrey E Hinton, and Ronald J Williams. Learning representations by back-propagating errors. *Cognitive Modeling*, 5, 1988.

[100] Magnus Sahlgren and Rickard Cöster. Using bag-of-concepts to improve the performance of support vector machines in text categorization. In *Proceedings of the 20th international conference on Computational Linguistics*, page 487. Association for Computational Linguistics, 2004.

[101] Gerard Salton and Michael J. McGill. *Introduction to Modern Information Retrieval*. McGraw-Hill, Inc., New York, NY, USA, 1986.

[102] Gerard Salton, Anita Wong, and Chung-Shu Yang. A vector space model for automatic indexing. *Communications of the ACM*, 18(11):613–620, 1975.

[103] Florian Schroff, Dmitry Kalenichenko, and James Philbin. Facenet: A unified embedding for face recognition and clustering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 815–823, 2015.

[104] Fabrizio Sebastiani. Machine learning in automated text categorization. *ACM computing surveys (CSUR)*, 34(1):1–47, 2002.

[105] Debakar Shamanta, Sheikh Motahar Naim, Parang Saraf, Naren Ramakrishnan, and M Shahriar Hossain. Concurrent inference of topic models and distributed vector representations. In *ECML PKDD*, pages 441–457. 2015.

[106] Mrio J. Silva, Bruno Martins, Marcirio Chaves, Ana Paula Afonso, and Nuno Cardoso. Adding geographic scopes to web resources. *CEUS*, 30(4):378 – 399, 2006.

[107] Richard Socher and Li Fei-Fei. Connecting modalities: Semi-supervised segmentation and annotation of images using unaligned text corpora. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pages 966–973. IEEE, 2010.

[108] Jeong-Woo Son, A-Yeong Kim, and Seong-Bae Park. A location-based news article recommendation with explicit localized semantic analysis. In *SIGIR*, 2013.

[109] Nitish Srivastava and Ruslan R Salakhutdinov. Multimodal learning with deep boltzmann machines. In *Advances in neural information processing systems*, pages 2222–2230, 2012.

[110] Stanford Natural Language Processing Group. Stanford NER. Accessed: Feb 07, '16, http://nlp.stanford.edu/software/CRF-NER.shtml.

[111] Stanford NLP Group. Stanford NER. Accessed: May 13, 2016, http://nlp.stanford.edu/software/CRF-NER.shtml.

[112] John Stasko, Carsten Görg, and Zhicheng Liu. Jigsaw: supporting investigative analysis through interactive visualization. *Information visualization*, 7(2):118–132, 2008.

[113] Zak Stone, Todd Zickler, and Trevor Darrell. Autotagging facebook: Social network context improves photo annotation. In *Computer Vision and Pattern Recognition Workshops, 2008. CVPRW'08. IEEE Computer Society Conference on*, pages 1–8. IEEE, 2008.

[114] Aixin Sun. Short text classification using very few words. In *Proceedings of the 35th international ACM SIGIR conference on Research and development in information retrieval*, pages 1145–1146. ACM, 2012.

[115] Yi Sun, Yuheng Chen, Xiaogang Wang, and Xiaoou Tang. Deep learning face representation by joint identification-verification. In *Advances in neural information processing systems*, pages 1988–1996, 2014.

[116] Yi Sun, Xiaogang Wang, and Xiaoou Tang. Deep convolutional network cascade for facial point detection. In *CVPR '13*, pages 3476–3483, 2013.

[117] Christian Szegedy, Alexander Toshev, and Dumitru Erhan. Deep neural networks for object detection. In *NIPS*, 2013.

[118] Yaniv Taigman, Ming Yang, Marc'Aurelio Ranzato, and Lior Wolf. Deepface: Closing the gap to human-level performance in face verification. In *CVPR*, pages 1701–1708, 2014.

[119] GHEORGHE Tecuci, Mihai Boicu, David Schum, and Dorin Marcu. Coping with the complexity of intelligence analysis: cognitive assistants for evidence-based reasoning. Technical report, LAC GMU, 2010.

[120] Stefanie Tellex, Boris Katz, Jimmy Lin, Aaron Fernandes, and Gregory Marton. Quantitative evaluation of passage retrieval algorithms for question answering. In *Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval*, pages 41–47. ACM, 2003.

[121] Yuandong Tian, Wei Liu, Rong Xiao, Fang Wen, and Xiaoou Tang. A face annotation framework with partial clustering and interactive labeling. In *CVPR'07*, pages 1–8. IEEE, 2007.

[122] Kristina Toutanova, Dan Klein, Christopher D Manning, and Yoram Singer. Feature-rich part-of-speech tagging with a cyclic dependency network. In *NAACL '03*, 2003.

[123] Matthew Turk and Alex Pentland. Eigenfaces for recognition. *Cognitive Neuroscience*, 3(1):71–86, 1991.

[124] M. Valstar, B. Martinez, X. Binefa, and M. Pantic. Facial point detection using boosted regression and graph models. In *CVPR '10*, pages 2729–2736, 2010.

[125] Oriol Vinyals, Alexander Toshev, Samy Bengio, and Dumitru Erhan. Show and tell: A neural image caption generator. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3156–3164, 2015.

[126] Paul Viola and Michael Jones. Rapid object detection using a boosted cascade of simple features. In *CVPR*, '01.

[127] Dayong Wang, Steven CH Hoi, Ying He, Jianke Zhu, Tao Mei, and Jiebo Luo. Retrieval-based face annotation by weak label regularized local coordinate coding. *IEEE transactions on pattern analysis and machine intelligence*, 36(3):550–563, 2014.

[128] Fang Wang, Zhongyuan Wang, Zhoujun Li, and Ji-Rong Wen. Concept-based short text classification and ranking. In *Proceedings of the 23rd ACM International Conference on Conference on Information and Knowledge Management*, pages 1069–1078. ACM, 2014.

[129] Kaiye Wang, Ran He, Wei Wang, Liang Wang, and Tieniu Tan. Learning coupled feature spaces for cross-modal matching. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2088–2095, 2013.

[130] Bo Wu, Haizhou Ai, Chang Huang, and Shihong Lao. Fast rotation invariant multi-view face detection based on real adaboost. In *FG*, pages 79–84, 2004.

[131] Benjamin Z Yao, Xiong Yang, Liang Lin, Mun Wai Lee, and Song-Chun Zhu. I2T: Image parsing to text description. *Proceedings of the IEEE*, 98(8):1485–1508, 2010.

[132] Tian Yong-hong, Huang Tie-jun, and Gao Wen. Exploiting multi-context analysis in semantic image classification. *JZUS-A*, 6(11):1268–1283, 2005.

[133] Kaipeng Zhang, Zhanpeng Zhang, Zhifeng Li, and Yu Qiao. Joint face detection and alignment using multitask cascaded convolutional networks. *IEEE Signal Processing Letters*, 23(10):1499–1503, 2016.

[134] Jile Zhou, Guiguang Ding, and Yuchen Guo. Latent semantic sparse hashing for cross-modal similarity search. In *Proceedings of the 37th international ACM SIGIR conference on Research & development in information retrieval*, pages 415–424. ACM, 2014.

[135] Xiaofeng Zhu, Zi Huang, Heng Tao Shen, and Xin Zhao. Linear cross-modal hashing for efficient multimedia search. In *Proceedings of the 21st ACM international conference on Multimedia*, pages 143–152. ACM, 2013.

Curriculum Vitae

Md Abdul Kader was born in Satkhira, Bangladesh as the youngest son of late Md Abdul Khaleque Gazi and Rabea Khanam. He began pursuing his Bachelor's degree in Computer Science and Engineering at the University of Dhaka, Bangladesh. After graduating from the University of Dhaka, he worked at Samsung R&D institute in Bangladesh for around a year and half. In 2012, he started his doctoral studies in Computer Science at the University of Texas at El Paso. During the course of doctoral studies, he did internships at IBM Watson Ltd. and at a startup named CloudlyIO in Silicon Valley. He will be working full-time at IBM Watson after his graduation. His list of publications is as follows.

- Md Abdul Kader, Arnold P. Boedihardjo, Sheikh Motahar Naim, M. Shahriar Hossain, *Contextual Embedding for Distributed Representations of Entities in a Text Corpus*, KDD BigMine (PMLR v53), 2016.
- Md Abdul Kader, Sheikh Motahar Naim, Arnold P. Boedihardjo, M. Shahriar Hossain, *Connecting the Dots using Contextual Information Hidden in Text and Images*, SA, AAAI-2016.
- Md Abdul Kader, Arnold P. Boedihardjo, M. Shahriar Hossain, *F2ConText: How to Extract Holistic Contexts of Persons of Interest*, submitted to a journal, KAIS, 2017.
- Md Abdul Kader, Arnold P. Boedihardjo, M. Shahriar Hossain, *An Unsupervised Framework for Representing Documents Containing Images and Text*, scheduled to be submitted in IEEE BigData, 2017.
- Sheikh Motahar Naim, Md Abdul Kader, Arnold P. Boedihardjo, M. Shahriar Hossain, *Encoding Lineage in Scholarly Articles. Workshop on Scholarly Big Data*, AAAI 2016.

Contact Information: mkader@utep.edu

This thesis was typed by Md Abdul Kader.