Graph Algorithms for Large-Scale and Dynamic Natural Language Processing

KAMBIZ GHOORCHIAN

Doctoral Thesis in Information and Communication Technology
School of Electrical Engineering and Computer Science
KTH Royal Institute of Technology
Stockholm, Sweden 2019

Information and Communication Technology
School of Electrical Engineering
and Computer Science
KTH Royal Institute of Technology
SE-164 40 Kista
SWEDEN

TRITA-EECS-AVL-2019:85
ISBN 978-91-7873-377-4

Akademisk avhandling som med tillstånd av Kungliga Tekniska högskolan framlägges till offentlig granskning för avläggande av doktorsexamen i informations ock kommunikationsteknik torsdagen den 17 December 2019 klockan 10.00 i sal C, Electrum, Kungliga Tekniska högskolan, Kistagången 16, Kista, Stockholm.

Tryck: Universitetsservice US AB

Abstract

In Natural Language Processing, researchers design and develop algorithms to enable machines to understand and analyze human language. These algorithms benefit multiple downstream applications including sentiment analysis, automatic translation, automatic question answering, and text summarization. Topic modeling is one such algorithm that solves the problem of categorizing documents into multiple groups with the goal of maximizing the intra-group document similarity. However, the manifestation of short texts like tweets, snippets, comments, and forum posts as the dominant source of text in our daily interactions and communications, as well as being the main medium for news reporting and dissemination, increases the complexity of the problem due to scalability, sparsity, and dynamicity. Scalability refers to the volume of the messages being generated, sparsity is related to the length of the messages, and dynamicity is associated with the ratio of changes in the content and topical structure of the messages (e.g., the emergence of new phrases). We improve the scalability and accuracy of Natural Language Processing algorithms from three perspectives, by leveraging on innovative graph modeling and graph partitioning algorithms, incremental dimensionality reduction techniques, and rich language modeling methods. We begin by presenting a solution for multiple disambiguation on short messages, as opposed to traditional single disambiguation. The solution proposes a simple graph representation model to present topical structures in the form of dense partitions in that graph and applies disambiguation by extracting those topical structures using an innovative distributed graph partitioning algorithm. Next, we develop a scalable topic modeling algorithm using a novel dense graph representation and an efficient graph partitioning algorithm. Then, we analyze the effect of temporal dimension to understand the dynamicity in online social networks and present a solution for geo-localization of users in Twitter using a hierarchical model that combines partitioning of the underlying social network graph with temporal categorization of the tweets. The results show the effect of temporal dynamicity on users' spatial behavior. This result leads to design and development of a dynamic topic modeling solution, involving an online graph partitioning algorithm and a significantly stronger language modeling approach based on the skip-gram technique. The algorithm shows strong improvement on scalability and accuracy compared to the state-of-the-art models. Finally, we describe a dynamic graph-based representation learning algorithm that modifies the partitioning algorithm to develop a generalization of our previous work. A strong representation learning algorithm is proposed that can be used for extracting high quality distributed and continuous representations out of any sequential data with local and hierarchical structural properties similar to natural language text.

Keywords: Natural Language Processing; Lexical Disambiguation; Topic Modeling; Representation Learning; Graph Partitioning; Distributed Algorithms; Dimensionality Reduction; Random Indexing;

Sammanfattning

Inom naturlig språkbehandling utformar och utvecklar forskare algoritmer för att möjliggöra för maskiner att förstå och analysera mänskligt språk. Dessa algoritmer möjliggör tillämpningar som sentimentanalys, automatisk översättning, automatiska frågesvarssystem och textsammanfattning. Ämesmodellering är en sådan algoritm. Den löser problemet med att kategorisera dokument i grupper, med målet att maximera den inbördes likheten mellan dokumenten. Korta texter som Twitter-inlägg, utdrag, kommentarer och foruminlägg har blivit den huvudsakliga källan till text i våra dagliga interaktioner och kommunikation. De utgör även källan till nyheter, samt hur nyheter genereras och sprids. Detta har emellertid ökat forskningsproblemets komplexitet på grund av skalbarhet, knappheten i texterna, samt deras dynamik. Skalbarhet hänvisar till volymen av de meddelanden som genereras, knapphet avser meddelandets längd och dynamiken är associerad med förhållandet mellan förändringarna i innehållet och den aktuella strukturen hos meddelandena (till exempel den ständiga uppkomsten av nya fraser). Vi förbättrar skalbarheten och noggrannheten hos algoritmer för naturlig språkbehandling i tre perspektiv som utnyttjar innovativa grafmodellerings- och grafpartitioneringsalgoritmer, inkrementella dimensionsreduceringstekniker och rika språkmodelleringsmetoder. Vi börjar med att presentera en lösning för disambiguering i korta meddelanden. Lösningen använder en enkel grafrepresentationsmodell för att presentera aktuella strukturer i form av täta partitioner i den motsvarande grafen och tillämpar disambiguering genom att extrahera de aktuella strukturerna med en innovativ distribuerad grafpartitioneringsalgoritm. Därefter utvecklar vi en skalbar ämnesmodelleringsalgoritm som nyttjar en ny tät grafrepresentation och en effektiv algoritm för grafpartitionering. Vi analyserar sedan effekterna av temporala dimensioner för att förstå dynamiken i sociala nätverk online. Vi presenterar också en lösning för problemet med Twitter-användares geolokalisering, med användning av en hierarkisk modell som kombinerar partitioneringen av den underliggande sociala nätverksgrafen och temporala kategoriseringar av Twitter-inläggen. Resultaten visar på effekten av temporal dynamik på användarnas rumsliga beteende. Detta resultat leder till försök att utveckla och implementera en dynamisk ämnesmodellösning, vilken innehåller en partitioneringsalgoritm för en online-graf, samt ett väsentligt starkare språkmodelleringssystem baserat på skip-gram-teknik. Resultaten visar stark förbättring på både skalbarhet och noggrannhet i resultaten jämfört med etablerade modeller. Slutligen beskriver vi en dynamisk grafbaserad representationsinlärningsalgoritm som ändrar vår partitioneringsalgoritm för att presentera en generalisering av vårt tidigare arbete. En stark inärningsalgoritm föreslås, vilken kan användas för att extrahera högkvalitativa distribuerade och kontinuerliga representationer enligt godtycklig sekvensiell data med lokala och hierarkiska strukturella egenskaper som faktiskt liknar naturligt språk i form av text.

Nyckelord: Natural Language Processing; Lexilal disambiguering; Ämnesmodellering; Representationsinlärning; Grafpartitionering; Distribuerade algoritmer; Dimensionalitetsreduktion; Random Indexing;

To the sweet memories of my father, Hassan
and
the beautiful smiles of my son, Avid

Acknowledgments

"If I have seen further it is by standing on the shoulders of Giants."
Isaac Newton

First and foremost, my deepest gratitude goes to my primary supervisor Magnus Boman, for his selfless support and valuable advice. I would like to thank Magnus for trusting me and being patient towards me, which made me calm and confident to get through during the hardest time in the development of my study and to become an independent researcher. Without his wise and mature mentorship, I could never have been able to finish this PhD. *"Magnus, I'm lucky to know you!"*

Next, I would like to thank my secondary advisor Magnus Sahlgren, for the fascinating and fruitful discussions and his suggestions, advice, guidance, and devoted time over different research projects during my PhD. I want to acknowledge that Magnus is a great scientist and researcher in the area of natural language processing whose deep expertise and knowledge helped me to broaden my understanding and expand my view in this interesting field of science.

Further, I want to thank my advanced reviewer Jussi Karlgren, for wise and valuable comments on my doctoral thesis; my ex-advisor Sarunas Girdzijauskas, for providing a part of the financial support and helping me to push my boundaries beyond my imaginations; my (ex-) secondary supervisor Fatemeh Rahimian, for the thorough and valuable discussions that helped broadening my view over the areas of graph analytics and distributed systems, and her accurate reviews and careful comments on my scientific publications.

My very special gratitude and appreciation goes to my lovely beautiful wife Azadeh, for all her support, caring and sharing during the entire process of my PhD. I would like to mention that without her continuous patience and modest accompaniment I would never be able to finish this journey, while I was trying to be a good father. Also, I would like to thank my caring and compassionate mother Giti, my one and the only sister Kathrine and my clever and humble brother Saeed for all their spiritual and mental support and accompaniment.

Besides, I would like to thank my friends and colleagues at the school of Electrical Engineering and Computer Science (EECS) at KTH, Amir Hossein Payberah, Amira Solaiman, Anis Nasir, Hooman Peiro Sajjad, Kamal Hakimzadeh Harirbaf, Ananya Muddukrishna, Shatha Jaradat and Leila Bahri for all the mind-stretching discussions and debates every now and then and for all the amazing time we spend together on different occasions during conferences, meetings and seminars. Furthermore, I would like to thank all the managers and advisors involved in administrative issues at EECS: Ana Rusu, Christian Schulte, Alf Thomas Sjöland and Sandra Nylén for providing all necessary physical and mental support to facilitate my research.

Last but not least, I would like to acknowledge all the support received from the *iSocial* Marie Curie project and thank my friends and co-workers at the University of Insubria in Italy and The University of Nicosia in Cyprus for hosting me and providing all the facilities and mentorship over the course of the two great internships during my PhD.

Kambiz Ghoorchian,
December 17, 2019

List of Papers

This thesis is based on the following papers, with the author of this thesis as the main contributor in all of them. The details of the contributions on each paper are provided in Section 1.4.

I Semi-supervised multiple disambiguation [1]
Kambiz Ghoorchian, Fatemeh Rahimian and Sarunas Girdzijauskas
Published in 9th IEEE International Conference on Big Data Science and Engineering (BigDataSE), 2015.

II DeGPar: Large scale topic detection using node-cut partitioning on dense weighted graphs [2]
Kambiz Ghoorchian, Sarunas Girdzijauskas and Fatemeh Rahimian
Published in 37th International Conference on Distributed Computing Systems (ICDCS), 2017.

III Spatio-temporal multiple geo-location identification on Twitter [3]
Kambiz Ghoorchian and Sarunas Girdzijauskas
Published in IEEE International Conference on Big Data (Big Data), 2018.

IV GDTM: Graph-based dynamic topic models [4]
Kambiz Ghoorchian, Magnus Sahlgren
Submitted.

V An efficient graph-based model for learning representations [5]
Kambiz Ghoorchian, Magnus Sahlgren, Magnus Boman
Submitted.

List of Acronyms

LDA Latent Dirichlet Allocation

BTM Bi-Term Topic Model

LSA Latent Semantic Analysis

PLSA Probabilistic Latent Semantic Analysis

DTM Dynamic Topic Models

CDTM Continuous-time Dynamic Topic Models

PYPM Pitman-Yor Process Mixture

DNN Deep Neural Network

LSTM Long Short Term Memory

CRF Conditional Random Field

RL Representation Learning

RI Random Indexing

Contents

Part I

Thesis Overview

Chapter 1

Introduction

1.1 Fundation

Understanding natural language text is complex labour for computers. Most of this complexity is related to understanding ambiguities [6]. Ambiguity is an inherent and essential property in natural language that gives it the flexibility to form an infinite space of semantic structures using a finite number of elements. Interpreting the true meaning behind ambiguities is a difficult problem known as disambiguation [7]. Disambiguation is an important pre-processing task in NLP that benefits a large group of downstream problems like question answering, machine translation, automatic text summarization, information retrieval, knowledge-base population, opinion mining, and semantic search [8].

Looking at the context and extracting further information is a common method for disambiguation in documents with long and cohesive contextual structures like news articles. However, most of the documents in today's social communication are in the form of short messages. For example, companies publish their latest news as blog posts or tweets. People share their opinions through comments on social media. They answer each other's questions via collaborative services like forums and community networks using short messages. These new types of documents impose multiple challenges when it comes to disambiguation. The challenges include: (I) **Sparsity**, (II) **Scalability**, and (III) **Dynamicity**. Sparsity is related to the short length of the documents that makes it difficult to resolve ambiguities by looking at the context. Scalability is related to the large volume of short messages being generated every day. For example, Twitter alone has reported a volume of 500 million tweets per day during 2014 [9]. Dynamicity refers to the ratio of the changes in short messages including temporal dynamics of the topics or emergence of new phrases.

Earlier solutions to disambiguation based on handcrafted feature engineering have been overcome by statistical methods based on text classification, also known as topic modeling [8]. Topic modeling refers to classifying a set of documents to multiple similarity groups to maximize the intra-group similarity among all documents. Traditional approaches based on term frequency were negatively affected by the natural highly skewed distribution of words, known as *Zipf's word frequency law* [10] [11] in natural language. More specifically, Zipf's law states that the most frequent term is twice as frequent as the second most, which is consequently twice as frequent as the third most, etc. This skewness causes the function words (like *"the"*, *"and"*, *"or"*, etc) that do not carry any semantic information to be selected as topic representatives. *TF-IDF* [12] was developed to mitigate this effect using the inverse document frequency as a dampening factor. This method has remarkable properties concerning capturing local inter-document relevance characteristics of words in the documents but adds very little with respect to cross-document co-occurrence structure of the words. More advanced methods like *Latent Semantic Indexing (LSI)* [13] were proposed to account

for global co-occurrence structures by factorizing the word-document representation matrix using methods like *Singular Value Decomposition (SVD)* [14]. The main limitation of factorization-based methods is that they draw sharp lines between the boundaries of the topics in low dimensional space and extract topics as orthogonal representation vectors, while it is clear that topics in natural language text are not totally distinctive. Instead, they are overlapping and share common spaces in both syntactic and semantic dimensions. Probabilistic approaches, like PLSI [15] and LDA [16], were developed to solve this problem by modeling topics as a k dimensional latent space between words and documents and extracting those topics by inferring the parameters of the model using methods based on *Maximum Likelihood Estimation* like *Gibbs Sampling* [17] or *Stochastic Variational Inference* [18].

Various solutions strive to tackle some of the new challenges during recent years. For example, BTM [19] used a stronger language model based on bigrams to account for sparsity or DTM [20] and CDTM [21] added temporal dimension to their inference model to solve dynamicity. The main limitation of these methods is in their underlying assumption regarding the fixed number of topics, which reduces their power to account for dynamicity. The next generation of approaches [22], [23] and [24] proposed more complex probabilistic methods based on *Multinomial Mixture Processes* [25] [26] to account for dynamicity by removing the limitation on the number of topics. However, even these methods are still limited on scalability since they rely on the same iterative optimization approaches as LDA.

Another group of solutions, known as *Representation Learning (RL)*, strives to solve disambiguation using dimensionality reduction methods based on Deep Neural Networks (DNN). They enable disambiguation by creating a low-dimensional unique representation vector, called embedding [27] for each word. RL-based methods have received a large amount of attention during recent years [28] due to their simple implementation and their high quality results. Importantly, word2vec [29] was one of the first proposed models for representation learning that mainly focused on the distributional representation of words [30] and therefore, did not account for long-distance relations between words in documents. Another solution, known as doc2vec [31], was developed to account for this problem by including document-level information in training of their model. The main limitation of these approaches, however is on the scalability of their training phase. Other methods like GloVe [32] and FastText [33], [34] even though able to account for scalability were still limited to local document level information. In lda2vec [35], it was attempted to consider topical level information. However, the main problem with lda2vec is again that it lacks scalability, as it is fundamentally based on LDA and word2vec.

In this thesis, we aim to design and develop methods and algorithms to overcome the challenges for scalable and dynamic NLP. We argue that combining graph representations with strong language modeling and dimensionality reduction techniques can overcome the limitations of centralized approaches, using decentralized optimization methods based on distributed graph analytics. In particular, we model the problems in the form of graph representations and propose efficient algorithms based on graph analytics to overcome scalability issues. The graph-based representation enables us to solve the problem in a decentralized manner using local optimization by reaping benefits from local information. Moreover, we improve the accuracy by solving sparsity and dynamicity problems using an incremental dimensionality reduction technique that allows us to efficiently apply strong language models using complex compositions of low-dimensional vector representations. Finally, we generalize the proposed topic models to design and develop a solution for efficient extraction of high-quality contextual representations to serve other downstream applications.

1.2 Research Objectives

The main goal of this thesis is to address significant challenges in named entity disambiguation over short messages in natural language text. We begin by simplifying the challenges related to the disambiguation itself and the best statistical solutions to this problem including topic modeling and representation learning. With respect to these areas, the primary objectives of this thesis are set towards:

- overcoming the scalability issues for disambiguation on short messages,
- alleviating the effect of sparsity in short messages as a barrier against the improvement of the accuracy in various NLP problems,
- mitigating the effect of dynamicity on disambiguation over short messages.

We hypothesize that scalability can be improved using localized and distributed optimization algorithms based on graph theory, which requires novel and innovative solutions for graph modeling. In addition, we believe that combining this choice of modeling and optimization with an incremental dimensionality reduction technique simplifies the application of complex language modeling methods over an online optimization mechanism. This consequently paves the way to meeting the sparsity and the dynamicity. We design innovative and efficient algorithms and develop robust systems and applications to meet the goals.

1.3 Research Methodology

This thesis follows research methodology for empirical analysis in NLP. We study state-of-the-art solutions, including topic modeling and representation learning, to identify their limitations and challenges. Then, we propose models to address those challenges and limitations and develop machine learning algorithms to evaluate our models. We exploratively improve the models and extend the solutions to meet all the challenges. Finally, we design and implement multiple experiments to evaluate and compare our solutions with the contemporary state-of-the-art approaches.

The main purpose of machine learning algorithms is to achieve generalization over a set of features in the dataset. A combination of the features is modeled as a function over a finite set of parameters and the generalization is induced by training the model following a minimization mechanism. Finally, the model is evaluated using evaluation metrics. A widely accepted categorization of approaches in machine learning divides them into *supervised* and *unsupervised* groups. In supervised methods, the goal is to minimize the error between induced and desired level of generalization. Unsupervised methods aim to extract a low-dimensional generalization of the internal structure of the data. In this research, we use both supervised and unsupervised approaches to evaluate the accuracy of our solutions. The standard method in supervised learning is to first divide the data into three distinctive sets for training, validation, and testing. Then, to train the model using the training and the validation sets, and evaluating the results of the classifications over the test set using evaluation metrics (Section 2.6). Unsupervised learning follows a similar optimization mechanism during the training phase but its evaluation is less straightforward by contrast. The reason is that no gold standard is available and therefore it is not possible to indicate the desired level of generalization. Based on this fact, indirect evaluation methods (Section2.6) have been developed that enable the evaluation of the model in terms of the characteristics of the results (e.g., the quality of the generalizations in terms of their coherence and distinctiveness).

1.4 Research Contributions

This thesis is a collection of the following contributions:

- Paper I proposes an algorithm to improve the accuracy and scalability of disambiguation on short messages by applying multiple disambiguation of ambiguous words on a scalable graph-based algorithm. The algorithm leverages graph modeling to present co-occurrence patterns in the documents and designs an efficient optimization mechanism to extract those patterns using autonomous localized communications among direct co-occurrences in the graph.

  Contribution. The author of this thesis is the main author of the paper, who designed the graph-based language representation model and developed the decentralized partitioning algorithm. He also designed and performed all of the experiments and evaluations in the paper. In addition, he wrote the majority of the paper and created all the charts and figures.

- To account for the scalability and sparsity of topic modeling on short messages we developed a solution based on dimensionality reduction and graph analytics (Paper II). First, we used an incremental dimensionality reduction technique to encode context structure of words into dense vector representations, as the building blocks of the algorithm. Then, we combined those vector representations using a bi-gram language modeling technique to extract high quality document vectors. Next, we combined all document vectors into a single highly dense and weighted graph structure, which encodes topic structures in the form of dense weighted sub-graphs. Finally, we developed a novel distributed graph partitioning algorithm to extract those topics, following a localized constraint optimization function.

  Contribution. The author of this thesis is the main author of this paper. He suggested the main idea of dense graph representation and designed the graph partitioning algorithm. He also ran multiple experiments to ensure the correctness of the model and the convergence of the partitioning algorithm. The author wrote the majority of the content, and created charts and figures in the paper. Moreover, he developed the code for cleaning and constructing the tagged dataset used for the supervised experiments in the paper.

- To examine the effect of temporal dimension on users' geo-localization in social networks, and to gain insight about the temporal dynamics of the topics of their short messages, we chose Twitter as our target social network and developed a solution for spatio-temporal multiple geo-location identification (Paper III). The solution was developed in a hierarchical structure by combining the graph partitioning of the underlying social network graph with the temporal categorization of the tweets. The results has improved the accuracy of user geo-localization on Twitter, showing the effect of temporal dimension on users' geographical mobility patterns, and the topical dynamics of their messages as a consequence.

  Contributions. The author of this thesis is the main author of the paper who contributed a thorough literature review and designed the hierarchical model for geo-localization. He implemented the code for collecting and cleaning the required dataset, including the underlying social network graph of a large number of Twitter users, together with the history of their tweets.

- In this work, we carefully designed and implemented a dynamic topic modeling algorithm (Paper IV) to account for all three requirements of topic modeling on short messages including scalability, sparsity, and dynamicity. The algorithm benefited from the incremental property of the dimensionality reduction technique presented in (Paper II). We developed a dynamic community detection algorithm that autonomously finds the true number of topics to account for dynamicity, in contrast to the common clustering algorithms that require a fixed k number

of topics to be specified. In addition, we used a stronger language model based on skip-gram to account for sparsity.

Contributions. The author was the main contributor of this work, including design and implementation of the graph representation model and the online graph community detection algorithm. He wrote the majority of the contents of the paper and developed all the experiments, and created all the figures and charts in paper.

- The last paper in this thesis (Paper V) is a solution for scalable contextualized representation learning as an important pre-processing task in various down-stream NLP applications. The algorithm is used to address ambiguity by extracting rich contextualized representations in a scalable manner. The same dimensionality reduction and feature vector composition techniques are used as in Paper IV. However, the community detection algorithm was modified to improve accuracy. In addition, a new component was implemented for extracting the contextual representations from feature vectors.

  Contributions. The author was the main contributor of the paper, who suggested the overall idea of converting the streaming topic modeling solution into a scalable contextual representation learning model. He designed and implemented the localized graph community detection algorithm for extracting the topics and representations. Also, the majority of the paper writing and the entire set of experiments was designed and completed by the main author.

1.5 Thesis Disposition

The rest of this thesis is organized as follows. Chapter 2 presents the required background for understanding the problem of disambiguation, together with a general overview of its strongly related solutions, including text classification and representation learning. The chapter concludes by presenting the background related to our proposed solutions including approaches in graph analytics and dimensionality reduction. Chapter 3 summarizes the contents of our publications in three sections. Section 3.1 presents our solution for multiple disambiguation on short texts. Section 3.2 covers the topic modeling solutions and describes the transition from static to dynamic topic modeling by explaining the effect of temporal dimension. Section 3.3 presents the details of our last paper on contextual representation learning. Chapter 4 concludes the thesis by providing an overall summary of our contributions, and by presenting potential future possibilities achievable by following the line of research explored in this thesis.

Chapter 2

Background

In this chapter, we present the background material required to cover the technical parts in the rest of the thesis. The chapter begins by explaining the ambiguity in natural language, positioning the type of ambiguity under study in this thesis, and defining the problem of disambiguation as a fundamental pre-processing task in NLP. We then describe the two problems of topic modeling and representation learning as well as the most widely applicable solutions to disambiguation. The rest of the chapter presents the details related to graph analytics and dimensionality reduction as the dominant tools and techniques used in our proposed solutions.

2.1 Disambiguation

Ambiguity is a characteristic of natural language that is inherited as an internal property from human conversation [36]. It refers to situations where sentences or phrases can have multiple interpretations. Ambiguity appears at all linguistic levels from morphemes, words, and phrases, to sentences or paragraphs [37]. A common classification groups ambiguity into five different categories [38]: (i) Lexical, (ii) Syntactic, (iii) Semantic, (iv) Discourse, and (v) Pragmatic. Table 2.1 shows a taxonomy together with a short description and a clarifying example of each group. In this thesis, we focus on lexical ambiguity as one of the most common and difficult types of ambiguities in natural language.

Lexical ambiguity is concerned with ambiguity in individual words. A word can either be ambiguous with respect to its syntactic category (e.g., *Noun*, *Verb*, *Adjective*, etc.) or its semantic reference (e.g., the meaning of the word). For example, in the sentence *"We saw her duck."* the ambiguity comes from the fact that the word *"duck"* can be either a verb or a noun, whereas, in the sentence *"The bat hit the ball!"* the ambiguity is related to the entity behind the word *"bat"*, which can either refer to a flying mammal or a baseball bat.

In both cases, identifying the true category, sense, or entity behind an ambiguous word is the problem known as lexical disambiguation. In the first case, the problem is called *Category Disambiguation*, whereas in the second case it is referred to as *Sense/Entity Disambiguation* [40]. The first problem is not difficult since there is a limited number of lexical categories that can be assigned to each word using rule-based methods based on part-of-speech tagging. The second problem, in contrast, is significantly complex. It has been shown to be an NP-hard problem [41], [42] since there is in theory an infinite number of possible sense/entity assignments for each word in natural language. This thesis specifically focuses on the second problem of entity disambiguation.

In linguistic terms, any object or thing (like a person, country, organization, etc.) in the real world is called *Entity* and the word referring to that entity is called *Mention*. Thus, entity disambiguation is defined as the problem of identifying the true entity behind an ambiguous mention in a

Table 2.1: Different types of ambiguity in natural language text.

Ambiguity	**Description**	**Example**
Lexical	Ambiguity in words (**The type of ambiguity studied in this thesis**).	Syntactic Category: *"We saw her duck"*. Semantic Reference: *"She has a lot of fans"*
Syntactic	Ambiguity in the structural hierarchy behind a sequence of words	Scope: *"Every student did not pass the exam."*. [39] Attachment: *"Bob saw Alice with a telescope"* [37].
Semantic	Ambiguity in a sentence despite the disambiguation of all lexical and syntactic ambiguities.	*"Fruit flies like banana."*.
Discourse	Ambiguity among shared words or shared knowledge across multiple documents, which is transferred through context.	*"The horse ran up the hill. It was very steep. It soon got tired"*.
Pragmatic	Ambiguity related to the processing of users intention, sentiment, belief or generally the current state of the world, also known as the world model. It happens when there is a lack of complete information during a conversations. [37]	Tourist (checking out of the hotel): *"Waiter, check if my sandals are in my room."* Waiter (running upstairs and coming back panting): *"Yes sir, they are there."*

document. Entity disambiguation is a fundamental pre-processing task for a large number of downstream applications in NLP like *Entity Linking* [43], *Relation Extraction* [44], and *Knowledge-base Population* [45]. It also benefits multiple applications in domain-specific tasks, as in clinical and biomedical domains [46], [47].

Supervised methods [48] use automatic or manually generated dictionaries of entities, like WordNet [49], [50] or BabelNet [51], tagged with multiple senses to assist in disambiguation. Nevertheless, these methods have difficulties with scaling, since the complexity of ambiguity is correlated with the dimension of the word-spaces [52] involved in the text. For example, in the sentence *"Mary liked Alice's photo from the party."* the verb *"like"* shows that either *"Mary"* literally liked the photo or she has pushed the like button on the photo uploaded in *"Alice's"* social media page. This is an example of the increase in the complexity and scope of ambiguity that affects outcomes from dictionary-based supervised approaches, where senses are discretely listed independently.

Based on that, ambiguity is considered a statistical problem that requires a statistical model of extrinsic information, including the context, in order to reflect the probabilities associated with an assertion. Following this line of thinking, unsupervised methods [53–60] try to infer the meaning directly using the context information in a corpus of documents. These methods apply clustering of the documents to extract similar examples, and given a sentence containing ambiguous mention they try to induce the correct entity by comparing the context with different clusters and choosing the most similar cluster. The two main limitations of these methods lie in their (i) iterative and global optimization method, and (ii) single disambiguation modeling approach, which respectively impose challenges related to their scalability and accuracy. In Paper I, we design and develop a solution for multiple disambiguation based on unsupervised topic modeling (Section 2.2) and graph analytics (Section 2.4) to address those challenges.

2.2 Topic Modeling

Topic modeling has initiated from the *Topic Detection and Tracking (TDT)* [61] task. TDT was defined as the task of finding and tracking topics from sequences of news articles. In linguistic terms, topic modeling is a dimensionality reduction problem that enables computers to reduce the large syntactic space of documents into a significantly smaller space of words with similar semantic representations. In addition to entity disambiguation (Section 2.1), topic modeling can benefit a large number of other applications [36] like information retrieval, knowledge extraction from scientific papers, text summarization, etc.

Early statistical approaches, like Latent Semantic Analysis (LSA) [13, 62], model the problem of topic modeling as a simple word-document co-occurrence frequency matrix in euclidean space and use factorization-based methods, like Singular Value Decomposition (SVD) [14] to extract the orthogonal projections of the co-occurrence matrix as semantic topic representations across the documents. The euclidean assumption makes these methods inefficient with respect to accuracy and complexity, however, on both memory and computation [16]. Moreover, studies [63] show language units share spaces and have similarities therefore, drawing a sharp line between the boundaries decreases the accuracy of the model. For example, two words like *Orange* and *Apple* share meanings in different semantic spaces like *fruit names* and *organization names* thus, given a sentence containing these words, it is not always possible to assign the exact topic of the sentence (e.g., fruits or organizations). Instead, it is better to soften the restrictive orthogonal assumption and allow wider decision boundaries using probabilistic approaches like Probabilistic Latent Semantic Indexing (PLSI) [15] and Latent Dirichlet Allocation (LDA) [16]. These approaches, model the topics as a latent low dimensional space over the space of words and documents. They develop a generative model [64], [65] in which a limited number of topics are considered as a distribution over a set of words, and each document belongs to a subset of the topics and is generated as a combination of a set of words sampled from the corresponding topics. The main linguistic assumption behind these models is that documents with the same words are similar. Therefore, they do not consider the order of appearance of the words in the documents, an approach known as *Bag-of-Words (BOW)*. This choice of language model is much stronger in modeling the global relations of the documents like the co-occurrence structures of the words compared to basic statistical models. However, it still ignores the syntactic information in the documents like the grammatical structures of the sentences and the order of the words.

Moreover, the probabilistic topic models were initially designed to account for modeling topics over long documents with large context sizes [61], like news articles. However, most of the documents generated in today's social media are in the form of short messages, like tweets. This new type of documents not only increases the current challenges related to sparsity and scalability but also adds new challenges related to the dynamicity. The latter is an inherent property of the new environments, like the online social networks where these messages are generated.

Different groups of solutions have been developed to address these challenges during recent years. A group of approaches known as *Relational Topic Models (RTM)* [66], [67], [68] [69] has proposed to solve sparsity by relating documents using external information like network details, content, or profile information of the users. However, they were only moderately successful since attaining such information is difficult due to various limitations [70]. Another group of solutions that includes Correlated Topic Models (CTM) [71] and Biterm Topic Model (BTM) [19] tried to address sparsity related to the two underlying assumptions in LDA, namely the Bag-Of-Words (BOW) language model and the independence of the topics in the underlying *Dirichlet* topic mixture distribution. BTM reconstructed the LDA to incorporate a bigram language model, known as the *Bag-of-Bigrams*, to improve the language modeling issues. CTM used a logistic normal distribution that allows for overlapping generalization of patterns over the parameters in the latent space. More specifically, since LDA assumes that each document belongs to multiple topics and all topics in

the dataset are totally independent from each other, the algorithm draws multiple topics for each document from a random Dirichlet distribution assuming the same probability for all topics to be selected. CTM, in contrast, assumes that topics are not totally independent from each other. For example, a document about cinema is more probable to appear in the entertainment and gaming topics than pharmacy or medicine.

These approaches even though successful in improving the sparsity but still limited in terms of scalability. The reason is that their inference model is based on Gibbs Sampling [17] that is the same model used by LDA. Gibbs sampling is a centralized and iterative optimization algorithm that requires global information on each iteration and therefore, faces scalability issues when it comes to large number of documents. In Paper II, we developed a solution based on graph modeling and dimensionality reduction to account for both sparsity and scalability. We designed our model to use a bi-gram language model together with a dense and weighted graph representation, and a scalable localized graph partitioning algorithm. We showed that our model is able to significantly outperform the state-of-the-art models (LDA and BTM) on scalability, while maintaining accuracy.

As time passes, the characteristics of topics of short messages are changing, and a large number of words and phrases are generated and added to the vocabulary. These changes are referred to as dynamicity in topic modeling on short messages. In Paper III, we studied the effect of temporal dynamics on the behavior of Twitter users. The study showed that the temporal dynamics strongly affects users' spatial behavior, which consequently influences the topical structure of their messages. Thus, to capture these temporal dynamics we need a model that extracts topics in an online setting, while tackling sparsity and scalability. A group of solutions known as *Dynamic Topic Models(DTM)* [20], [21] were developed to model the dynamicity as the correlation between the elements of the co-variance matrix of the parameters of the Dirichlet distribution. These approaches, similar to LDA, are based on the same assumptions related to a fixed number of topics that limit their power to account for sparsity. To account for this limitation, another group of solutions [22], [23] and [24] proposed more complex probabilistic models based on *Multinomial Mixture Processes* [25] [26], which allow for an infinite number of topics. However, even these methods are still limited with respect to scalability, since they rely on the same iterative optimization approaches as LDA.

All the above models try to address one or more challenges in topic modeling on short messages. However, to the best of our knowledge a solution that meets all the three challenges at once is missing. Therefore, in Paper IV we developed GDTM, a graph-based solution for dynamic topic modeling on short messages, to meet all three challenges. The solution were based on localized graph partitioning and dimensionality reduction, similar to DeGPar. However, GDTM used a much stronger language modeling by applying the incremental property in the dimensionality reduction technique. This approach empowers the solution to improve over sparsity while accounting for dynamicity and scalability.

2.3 Representation Learning

Representation Learning (RL) is a dimensionality reduction technique for transforming the high dimensional feature space into abstract low dimensional information-rich representations, which yield the same degree of explanation consuming lower amount of information [28]. The fundamental theory behind RL methods is based on distributional hypothesis [72] which states that words that appear in the same contexts share semantic meaning, following the famous quote *"you shall know a word by the company it keeps"* [73]. Based on that, RL-generated abstractions not only (i) improve the performance on various down-stream tasks and (ii) save enormous human labour behind feature engineering, but also (iii) has the ability to infer generalizations [74]. Generalization is defined as a property that enables inductive extraction of a belief over an unseen phenomenon using the

representations of a similar concept [75]. For example, after observing many times that *"monkeys like bananas"* this observation converts into a belief in our brain and afterwards by observing just a few observations of chimpanzees and noting their similarities to monkeys, we tend to believe that chimps also like banana. RL received a large amount of attention during recent years [28] and has become one of the fundamental reprocessing tasks in NLP.

One of the initial solutions to RL, known as *Word2Vec (W2V)* [29] creates continuous low-dimensional vector representations called embeddings [27], for words in a dataset by looking at the vicinity context in their corresponding documents. The authors developed a supervised classification sequence learning algorithm based on shallow neural networks [6] and implemented two models based on bag-of-words and skip-gram approaches. Their solution successfully encoded the local syntactic regularities in documents [76]. However, it was less successful in capturing more complex semantic relations between the words. For example, it can correctly identify linear word relations like: $king - queen \approx man - woman$, but it has difficulty identifying more complex semantic relations for example antonymy: $good - bad \approx small - large$.

The next group of solutions proposed different approaches like incorporating global information in the word-document co-occurrence matrix [32], adding sub-word information using N-gram language models [33], [34], or adding document-level [31] [77] or topic-level information [35] to the corresponding word vectors. This aims to improve the quality of the extract representations in terms of complex semantic structures. However, these approaches do not consider the dynamicity in the semantic behavior of the words and therefore, their extracted representations usually encode coarse grained representations, which create misleading results especially in difficult downstream tasks, like co-reference resolutions. A popular example of such problem is the retrieval of advertisements related to the *amazon fire tablets* created by the *Amazon* company, rather than the significantly important news articles about the devastating expansions of fire through the natural amazon rain-forests as the result of searching *amazon fire* keywords on *Google* search engine during the corresponding time [78]. The next generation of approaches like: ELMo [79], BERT [80], ULMFiT [81], Semi-supervised Sequence Learning [82], GPT [83] and GPT2 [84], also known as contextualized representation learning models, incorporate context sensitive information, which allows for the creation of task-specific representations adaptive to the context.

All the above mentioned solutions are based on deep neural networks which use inference models based on the backpropagation optimization algorithm. The main problem with this optimization mechanism is with scalability, especially when it comes to short messages. In Paper V we design and develop an efficient algorithm for contextual representation learning. We design an innovative graph-based community detection algorithm that allows us to model the topical structure of the documents and extract vector representations for words. We later use the vector representations and the constructed topic model to extract and use document representation vectors to improve the results in multiple downstream NLP tasks.

2.4 Graph Analysis

Graphs provide a means to model the relations and interactions among a group of discrete individual entities, and graph analysis refers to the set of models and algorithms to study and extract useful information from those relations [85], [86]. A graph $G\langle V, E\rangle$ is often defined as a set of *vertices* V that represent the entities, and a set of *edges* E that connect pairs of vertices if there is a relation between their corresponding entities. For example in the Facebook graph, vertices represent users and edges show their friendship relation. Graphs can encode various information related to the characteristics of the entities and the relations. Edges can be weighted or unweighted and directed or undirected. A weight is typically used to represent the strength of the relation between the adjacent objects, whereas the direction shows the orientation of the relation. The Twitter network is

an example of a directed graph where the edges are in the direction of the following in contrast to Facebook where users are either mutual friends or not friends.

Graph algorithms are used for modeling and solving many real world problems. *Pagerank* [87] is one of the most famous graph algorithms developed to facilitate search in the Google search service. *Sub-graph isomorphism* [88] is another graph algorithm that is used for solving different problems like pattern matching in graph databases [89] to retrieve the results of queries or pattern recognition in malware detection [90] and biological and chemical applications [91]. Graph partitioning is another widely used application of graphs to model and solve many real world problems like load balancing in distributed systems or topic modeling and disambiguation in NLP. The main motivation behind using graph algorithms lies in their simple and straightforward design structure that allows for developing efficient algorithms able to infer reasonable generalizations using highly scalable localized optimization mechanisms [92]. The localization enables the deployment and running of the algorithms on distributed data analysis platforms like Spark [93] or Storm [94], and take advantage of large-scale parallel and distributed data processing paradigms like MapReduce [95].

In this thesis, we mainly focus on graph partitioning and community detection algorithms. In particular, we model the problems in the form of dense sub-graphs that represent dissimilar groups of similar objects and we design and develop graph partitioning and community detection algorithms to solve the problems. In the next section, we explain the problems of graph partitioning and community detection, and then we present graph modeling approaches for NLP, and topic modeling in particular.

2.4.1 Graph Modeling for NLP

Graphs provide a rich and flexible means to model and solve NLP problems in an efficient manner [96]. There are different methods to model NLP problems in the form of graphs [97], depending on the characteristics and requirements of the problem. The most typical graph representation method for topic modeling 2.2 is to create a graph, known as a *co-occurrence graph [97]*, by assigning a single node to each unique word in a given corpus of documents and connect the vertices upon observing the co-occurrence of their corresponding words among the documents in the corpus. In Paper I we leveraged this method to develop a solution for multiple disambiguation as topic modeling problem. We constructed a co-occurrence graph representation that encodes topics in the form of dense sub-graphs in the graph and developed a graph partitioning 2.4.2 algorithm to extract those sub-graphs for entity disambiguation.

The main problem with co-occurrence graphs is the sparseness which causes scalability issues. To overcome that problem we used another graph representation model based on dense and weighed graph representation. In this model we used dimensionality reduction to first encode contextual syntactic structures in the documents into low dimensional dense contextual vector representation using a fast and incremental dimensionality reduction technique known as *Random Indexing* 2.5. Then we used a dense and weighted graph modeling approach to aggregate the corresponding vector representations to encode the frequent co-occurrence patterns. Then, developed a two localized graph partitioning 2.4.2 to extract the topics.

Figure 2.1 shows a sample of both sparse and dense graph representation models. As we can see, in the sparse model the number of vertices and consequently the size of the graph increases with the number of unique words in the data. In contrast, the dense model has a fixed number of vertices and an upper bound on the theoretical possible number of edges which makes it a fixed size graph.

After constructing the graph model the next step is to develop algorithms to extract the topics from those representations.

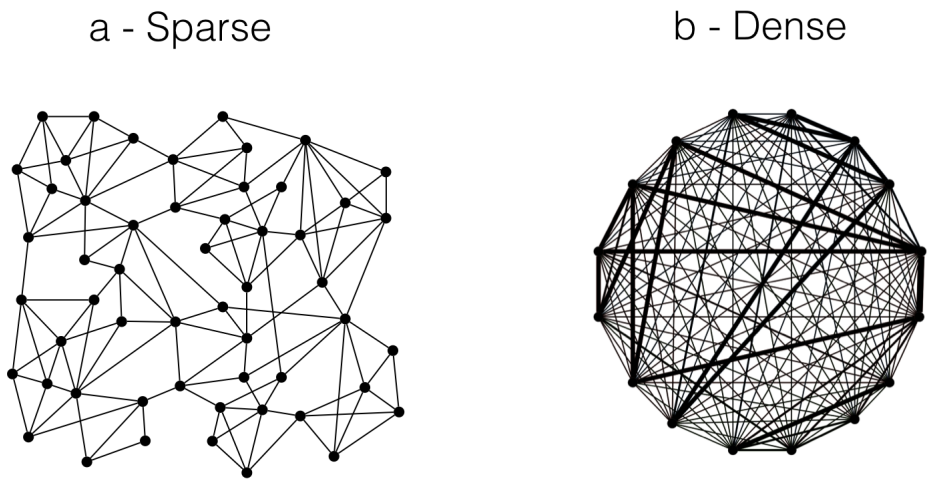

Figure 2.1: Sparse vs Dense graph representation for topic modeling in NLP.

2.4.2 Graph Partitioning

Graph partitioning is the problem of dividing a graph into a pre-defined number of sub-graphs, of equal or nearly equal sizes, following an optimization function. The optimization is defined in terms of the minimization of the inter-partition communication and the maximization of the intra-partition density. Both communication and density can be defined in terms of the number, weight and even the direction of the edges between and among the partitions. Intuitively, we want to partition a graph such that the vertices in each partition are more strongly connected to each other than to vertices in other partitions. This optimization criterion is useful for many applications. For example, in distributed computing, a good partitioning of the data can significantly improve the efficiency by minimizing the communication cost between machines, and by maximizing the localization.

There are two approaches for partitioning a graph: (i)*edge-cut*, and (ii)*node-cut*. Edge-cut partitions the graph by removing the edges while node-cut divides the vertices into multiple other vertices, called replicas, where each replica takes a subset of the edges connected to the original node. Figure 2.2 shows the difference between the two approaches. As we can see, edge-cut is good for partitioning sparse graphs while node-cut performs better on dense graphs. The reason is that in both cases a lower number of cuts are required to partition the graph. This is in line with the communication minimization criteria in the optimization function. Thus, graph partitioning is defined as the problem of finding a minimum cut that partitions the graph into a pre-defined number of sub-graphs with maximum density among partitions.

Finding the minimum cut is an NP-hard problem and there are a large number of proposed solutions [98]. In Paper II, we develop a graph partitioning algorithm for topic modeling. The topics are modeled in the form of a dense and weighted graph representation. Therefore, we design and implement and efficient node-cut graph partitioning algorithm to extract the topics. We define a localized utility function based on the *Max-Flow Min-Cut* [99] theorem and design an optimization mechanism using on localized label propagation [100] to extract the partitions.

2.4.3 Graph Community Detection

Community detection is the problem of extracting communities in a graph [101]. Communities, similar to partitions are groups of strongly connected vertices (entities) in the graph. But in contrast

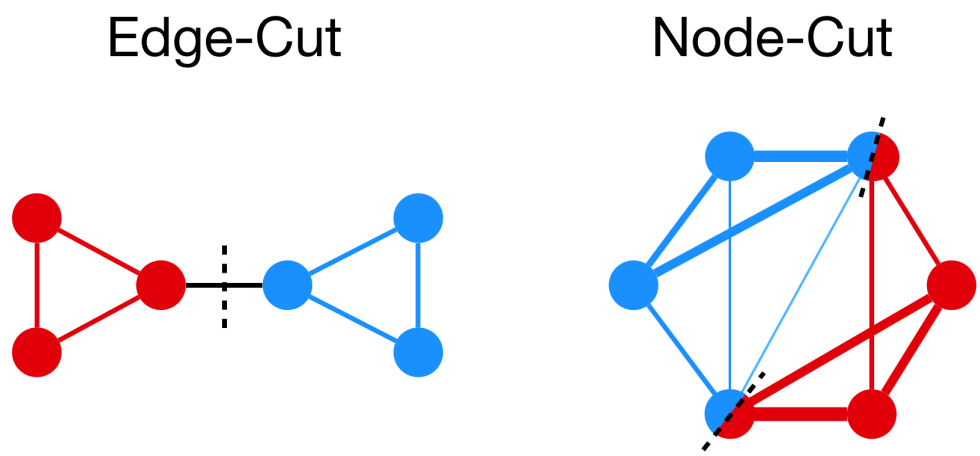

Figure 2.2: Node-cut vs Edge-cut approaches for graph partitioning on sparse and dense graphs. As we can see, both approaches satisfy the main criteria to find the minimum cut that partitions the graph into same/similar size sub-graphs with maximum intra-partition density.

to partitions, the number and consequently the size of the communities is not specified in advance. Therefore, sub-graphs extracted in community detection tend to follow a more natural structure of the group formation in the graph. For example, the friendship network around a single user in Facebook can often be divided into multiple groups of strongly connected vertices representing different community memberships like family, friends, and colleagues in the real world. Thus, communities can have arbitrary numbers and sizes. This additional degree of freedom significantly increases the difficulty of community detection, as compared to partitioning. Therefore, defining a metric to measure the quality of partitioning is difficult.

Different approaches have been proposed to measure the quality of partitioning [102]. *Modularity* [103] is the most well-known and accepted measure to calculate the quality of a partitioning. The main idea behind modularity is to measure the quality of a partitioning by comparing the density of the extracted partitions against the expected value of the same quantity in the same community structure but with a random permutation of the edges. In formal representation the value of modularity is calculated as follows:

$$Q = \frac{1}{2m} \sum_{vw} [A_{vw} - \frac{k_v . k_w}{2m}] \delta(c_v, c_w)$$

where m is the number of edges in the graph, $\frac{k_v . k_w}{2m}$ is the probability of an edge between two vertices v and w in a random graph with m edges. A_{vw} is 1 if v is connected to w and 0 otherwise. δ indicates community membership of the two vertices, v and w, and its value is 1 if both are in the same community and 0 otherwise.

Following this method, in Paper I we develop a localized graph community detection algorithm to extract the topics. The algorithm defines communities as colors and initializes by randomly assigning a unique color to each node. Then, the algorithm proceeds to extract the communities following a localized optimization mechanism based on modularity. The optimization is applied through local communication between vertices where each node tries to form and expand a local community by diffusing a portion of its color into the adjacent vertices in its vicinity.

In Paper II, we used graph partitioning to extract the topics. The main problem with this approach is the pre-defined number and size of the partitions that limit the power of the algorithm to

account for sparsity in topic modeling on short messages (2.2). Community detection is a remedy to account for this limitation. However, dynamicity is another problem that iterative community detection algorithms (like the one presented and used in Paper I) are not able to solve. A suitable method that accounts of dynamicity needs to extract and update the partitions in a continuous manner. This is a new approach called *Streaming Graph Partitioning/Community Detection*. There is a large body of research in this area [104], [105], [106], and [107]. However, these solutions are mainly designed to extract partitions/communities on very large and sparse networks with power-law [108] degree distributions. To the best of our knowledge, there are no solutions that can be applied on highly dense and weighted graphs similar to those presented and used in our solutions. Based on that, to overcome those limitations related to sparsity and dynamicity for topic modeling in Paper II, we design and develop a novel streaming graph community detection algorithm to extract the topics in online mode in paper Paper IV. We design a utility function based on maximization of the global average density of the communities in the graph. The algorithm follows a localized optimization mechanism based on modularity to extract the communities.

2.5 Random Indexing

Random Indexing (RI) [63] is an incremental dimensionality reduction technique based on random projections and context windowing method. The fundamental idea behind random projections is based on the *Johnson-Lindenstrauss* lemma [109] which states that if points in a vector space are projected onto a randomly selected subspace of suitably high dimension, then the distances between points are approximately preserved. RI is an efficient method compared to traditional factorization based methods, while it creates sufficiently accurate results [63]. The efficiency comes from the random projection that allows the model to be applied incrementally to each instance of the data independently and the accuracy owes to the context windowing model that provides a stronger language representations compared to traditional sparse word-by-document representation [52].

RI constructs unique low-dimensional vector representations for each unique word in a dataset as follows [110]. To each word, RI assigns two vectors including a *random vector* that keeps a unique low-dimensional representation for the word and a *context vector* that is responsible for capturing the surrounding contextual structure of word. The random vector is created once by initializing a vector of all zeros and randomly choosing a specific number of elements and assigning their values to $\{1, -1\}$ at random. Whereas, the context vector is constantly updated by moving a window of a specific size around the word and aggregating the random vectors of the surrounding context words.

RI is used as a common preprocessing step in down-stream data-mining applications to gain scalability. However, representations created by RI are only capable of capturing local syntactic structures like synonymy while complex tasks like disambiguation and topic modeling require a representation model that is able to capture long range semantic structures in the documents. Therefore, in Paper II, Paper III, and Paper V we use RI as an initial step in our language modeling to create low-dimensional representations. We slightly change the vector construction model to only capture the co-occurrence patterns in the documents and not their significance. Later in our language model in the algorithms, we develop a more complex document representation model to capture long range dependencies based on skip-graph techniques [111].

2.6 Evaluation Metrics

This section presents the details of different evaluation metrics used in our research. The evaluation methods are generally divided into intrinsic and extrinsic. Intrinsic evaluation is where it is possible to directly measure the quality of the results (e.g., measuring the quality of clustering using B-cubed [112]). In contrast, extrinsic evaluation is a methodology where the quality of the results is

being measured by applying the output of the primary task in some form of an input into a secondary task with capability of intrinsic evaluation. We use three intrinsic metrics, including *B-Cubed* [112], *Coherence Score* [113] and *V-Measure* [114], to evaluate the results in disambiguation and document classification problems. For RL, we leverage on extrinsic evaluation methods by applying the results to secondary standard NLP tasks [115]. The reset of this section will present the details of the intrinsic evaluation methods. The extrinsic methods will be covered later in Section 3.3.

2.6.1 B-Cubed:

is a statistical measure for evaluating the accuracy of the results of topic modeling in text. It calculates the accuracy of a classification for each document compared to a gold standard. The result is reported as the average over all documents in the dataset. In particular, given a dataset D with n documents, tagged with k hand-labels, $L = \{l_1, \ldots, l_k\}$ and a classification of the documents into k class-labels, $C = \{c_1, \ldots, c_k\}$, the B-Cubed of a document d with hand-label l_d and class-label c_d is calculated as:

$$B(d) = 2 \times \frac{P(d) \times R(d)}{P(d) + R(d)},$$

where P and R stand for *Precision* and *Recall*, respectively, and are calculated as follows:

$$P(d) = \frac{|d'|_{\{d' \in D : c_{d'} = c_d, l_{d'} = l_d\}}}{|d'|_{\{d' \in D : c_{d'} = c_d\}}}$$

$$R(d) = \frac{|d'|_{\{d' \in D : c_{d'} = c_d, l_{d'} = l_d\}}}{|d'|_{\{d' \in D : l_{d'} = l_d\}}}.$$

Precision shows the likelihood of documents correctly classified in a specific class c, with respect to the total number of documents in that class. Recall represents the likelihood with respect to the total number of documents in a specific label l. The total B-Cubed score is calculated as the average over all documents in the dataset:

$$B_{total} = \frac{1}{n} \times \sum_{i=1}^{n} B(d_i).$$

2.6.2 Coherence Score:

is an evaluation metric for measuring the quality of extracted topics in a topic classification problem. It assumes that the most frequent words in each class have higher co-occurrences among the documents in that class than among the documents across multiple classes. Thus, given a set of documents classified into k topics, $T = \{t_1, \ldots, t_k\}$, first, the coherency of each topic, z, with top m probable words, $W^z = \{w_1, \ldots, w_m\}$, is calculated as,

$$C(z, W^z) = \sum_{i=2}^{m} \sum_{j=1}^{i-1} \log \frac{D({w_i}^z, {w_j}^z)}{D({w_j}^z)},$$

where $D({w_i}^z, {w_j}^z)$ is the co-occurrence frequency of the words v_i and v_j among documents in z and $D({w_j}^z)$ is the total frequency of w_j in z. Then, the total coherency of the partitioning is calculated as:

$$C(T) = \frac{1}{k} \times \sum_{z \in T} C(z, W^z).$$

V-Measure:

2.6.3 V-Measure:

is an entropy based evaluation metric for measuring the quality of the clustering with respect to *Homogeneity* and *Completeness* of the results. Assume the set of true classes $C = \{c_1, c_2, ...\}$ and the clustering result $K = \{k_1, k_2, ...\}$. Homogeneity measures the distribution of the classes given a specific clustering; whereas, completeness shows the integration of the members of each class. A clustering with a single cluster for each class results in the highest homogeneity and the lowest completeness. In contrast, a single clustered result features the highest completeness and the lowest homogeneity. V-Measure is computed as the weighed (β) harmonic mean of the two values:

$$V_\beta = \frac{(1+\beta) * h * c}{(\beta * h) + c}.$$

According to [114], V-Measure contemplates the conditional distribution of all the classes and therefore, it captures the irregularities and matching of the classes across the result. In other words, V-Measure is sensitive to the diversity of classes in all the clusters and prefers the result with lower diversity of classes in each cluster.

Chapter 3

Summary of Publications

This chapter presents a summary of the papers included in the thesis. The papers cover a set of solutions focusing on ambiguity as a property of natural language text. The solutions, following an empirical approach, propose various models ranging from simple count-based models to static and dynamic text classification and complex contextual representation learning methods for disambiguation as a general pre-processing step for solving various NLP problems.

3.1 Semi-Supervised Multiple Disambiguation

Clustering one-hot encoded document vector representations is a common approach for disambiguation in natural language text [53]. This approach does not scale, regardless of a variety of clustering algorithms, like pairwise similarity [53–55], distributed pairwise similarity [56–58], inference-based probabilistic [59] or multi-pass clustering [60]. The main limitation is in their inefficient underlying vector representation model. More specifically, one-hot vectors are highly sparse, meaning that they contain a large number of zero elements that make them computationally inefficient with respect to memory and processing.

The other limitation in contemporary solutions (even in the more advanced approaches [116–118]), to the best of our knowledge lies in their assumption regarding the single disambiguation. More specifically, they assume a single ambiguous word per document and design models for disambiguation either by removing the other ambiguous words from the analysis or by ignoring their ambiguity, and accepting only one of their senses. Both of these methods eliminate some of the information available in the data, as shown in Figure 3.1, and therefore, reduce the efficiency of the corresponding algorithms.

In Paper I we developed a multiple disambiguation algorithm to address the two above-mentioned limitations. The algorithm uses a novel graph-based feature representation model and a diffusion-based community detection algorithm (see Section 2.4.2). In our model, we consider multiple disambiguation in contrast to the common approach for single disambiguation. The main assumption is that different senses belong to different topics. We use a sparse and unweighted graph representation to model documents such that the topics create dense sub-graphs in that graph. Then, we develop a distributed graph partitioning algorithm to extract those topics.

Examining the accuracy of our algorithm requires an appropriate dataset containing documents tagged with multiple ambiguous mentions from specific topics. Since the two standard datasets in the field, including John Smith [53] and Person-X [55] did not satisfy this property, we created a synthetic dataset by crawling all *Wikipedia* pages containing a URL to 6 chosen entities. Then we trained our model in two different settings for single and multiple disambiguation. We used V-measure [114] (see Section 2.6.3) for evaluating the quality of the clustering and B-Cubed [112] (see

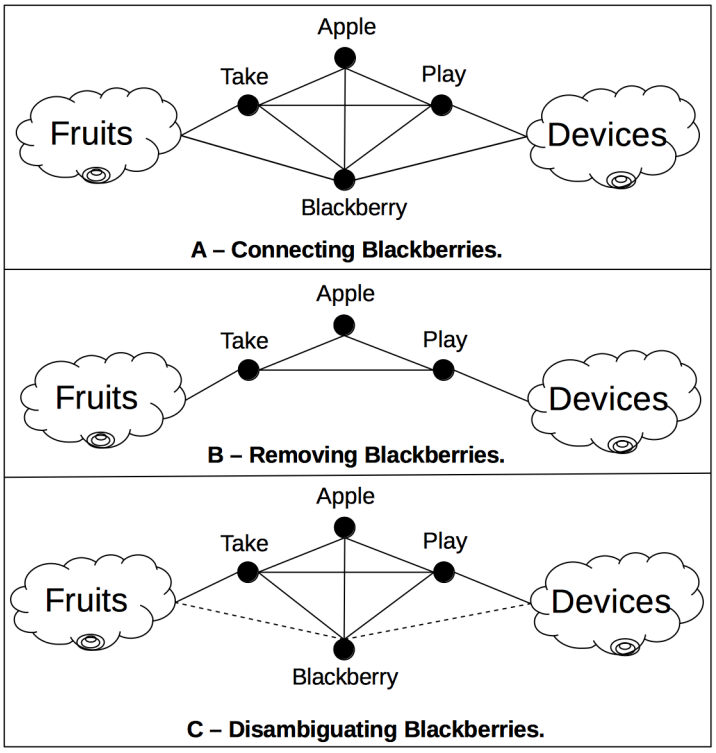

Figure 3.1: Single vs Multiple disambiguation. The figure presents three different approaches to apply disambiguation when there are multiple ambiguous words in a document. (A) and (B) show two approaches from single disambiguation perspective. (A) assumes no ambiguity on one of the words (*Blackberry* in this example) and assigns the word to one of the topics, while (B) proposes to remove one of the ambiguous words (*Blackberry*) and only focus on the other. In contrast, (C) shows the approach taken in our research where we develop a model to apply multiple disambiguation at parallel.

Section 2.6.1) to compare the accuracy of the model on both single and multiple disambiguation. The results, as expected, confirm the efficiency of our multiple disambiguation approach.

This was the initial step in our journey through disambiguation which gave us a strong clue to the overall structure of the problem, and directed us towards examining more complex solutions based on topic modeling and text classification in the papers that followed.

3.2 Topic Modeling

Topic modeling is a well-established approach for disambiguation in NLP. However, it was originally designed for application on natural language texts with large contexts like newswire documents. Therefore, using topic modeling for disambiguation on short messages imposes various challenges (see Section 2.2) including: (i) scalability, (i) sparsity, and (iii) dynamicity. In this section, we present the Paper II achievements as a solution for large scale static topic modeling with graphs to address the first two challenges, scalability and sparsity. Then, we continue to present the analysis of the effect of temporal properties of the messages on users' geo-location identification in Paper III. The latter paper highlights the importance of incorporating the temporal dimension to address the dynamicity. Finally, we present Paper IV a solution for dynamic topic modeling on short messages that accounts for all three challenges in one single approach.

3.2.1 Static Topic Modeling

We developed DeGPar to account for sparsity and scalability of topic modeling on short messages. Figure 3.2 shows the overall protocol of the algorithm which is applied in two steps: (i) Graph Construction, and (ii) Graph Partitioning. In the first step, the algorithm converts all documents into a dense graph representation as follows. First, DeGPar uses a dimensionality reduction technique called random indexing (see Section 2.5) to extract a feature vector representation for each word in the dataset. Then, it uses a graph modeling approach to combine feature vectors of the words into a dense document graph representation. Next, the algorithm combines all document graph representations into a single, highly dense and weighted graph representation called *knowledge graph*. The knowledge graph is designed so that it encodes the complex topical structures of the documents as highly dense and weighted partitions with node-cut boundaries. In the second step, the algorithm extracts the topics by partitioning the knowledge graph into multiple disconnected dense sub-graphs. We developed a localized partitioning algorithm to extract those topics following a node-cut minimization approach based on modularity optimization (see Section 2.4.2).

To compare the efficiency and scalability of DeGPar, we developed and ran two sets of experiments against two SotA approaches including: LDA [16] and BTM [19]. For efficiency, we created a tagged dataset using the *SNAP-Twitter* [119] dataset that contains $476M$ trending topics published in 2009. For scalability experiments, we used a *NIST* standard dataset, called *TREC-2011* [120]. Then, we created 5 different test datasets containing different number of tweets to compare the scalability of DeGPar with BTM against variations in the number of documents. The results show that DeGPar is significantly better than both BTM and LDA in terms of accuracy. Also, the scalability results show that DeGPar is at least an order of magnitude faster than the fastest SotA approach BTM in terms of the execution time.

DeGPar uses random indexing as an incremental dimensionality reduction technique. However, we are not taking the full benefit of its incremental property in DeGPar and we are using it as a batch approach. Therefore, DeGPar is not fully attacking the problem of dynamicity for topic modeling on short messages. In the next steps, we examine the effect of temporal dimension of the short messages on their underlying topical structures, and then we develop a dynamic topic modeling solution to account for this important property for improving the accuracy of the models.

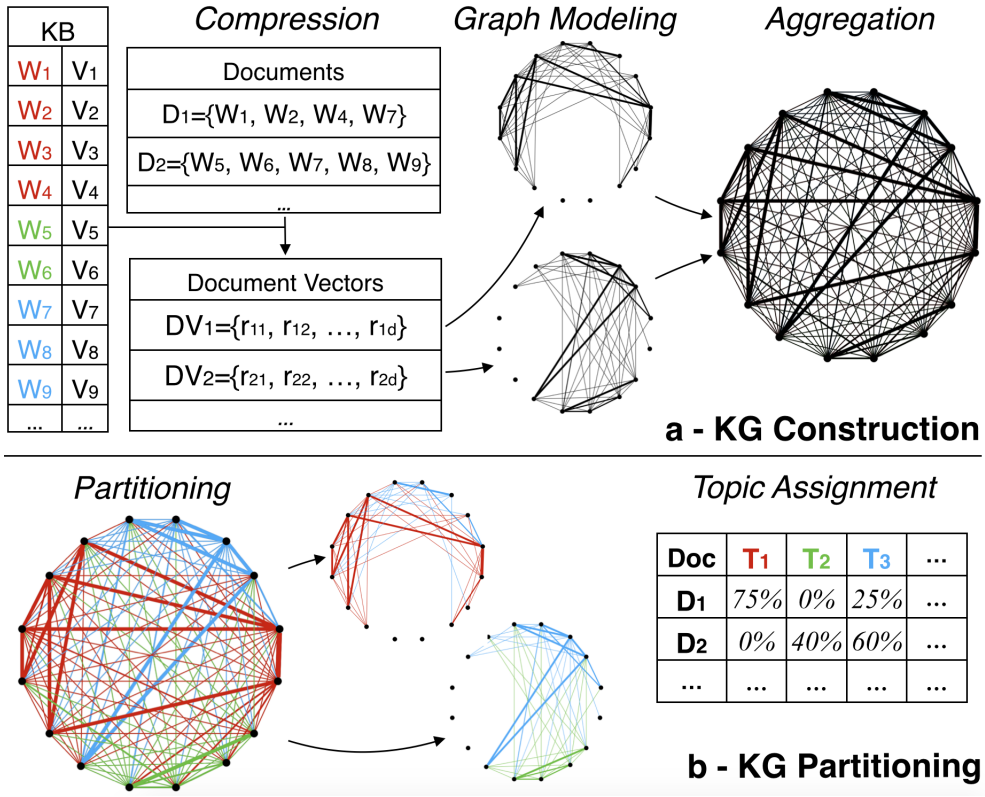

Figure 3.2: DeGPar, a graph-based algorithm for topic modeling on short messages. The fundamental idea is to encode topics as dense sub-graphs in a highly dense and weighted graph representation, we call *Knowledge Graph* and extract the topics by partitioning that graph. Thus, the overall process of the algorithm is divided into two phases: (a) knowledge graph construction: that leverage random indexing and bigram language modeling to construct and aggregate document feature graphs into the knowledge graph, and (b) knowledge graph partitioning to extract those topics using an innovative graph partitioning algorithm.

3.2.2 Temporal Analysis

In Paper III, we developed a solution for analyzing the effect of temporal dimension of the messages on their topical structure as a function of users' locality. Traditional methods either used content information [121–125] or leveraged on social network information [126–131] to predict the true geo-location of users of online social networks.

We developed a hierarchical classification model based on graph partitioning and temporal categorization to predict users' locations based on temporal dynamics of their messages and their social network friendship relations. Figure 3.3 shows the overall structure of the two phases of the model including: (i) Partitioning and (ii) Categorization. In the first step, we partition users and their corresponding tweets into multiple groups based on the topological structure of their underlying social network graph to account for their friendship locality. Then, in the second step, we categorize the tweets in each partitioned group into multiple sub-groups depending on the time-stamp of their tweets to consider the temporal dynamics of their behavior. Finally, we use those categories to analyze the effect of temporal categorization by predicting users' locations in each temporal group.

We ran a set of experiments on a large-scale Twitter dataset. The dataset was collected from 2010

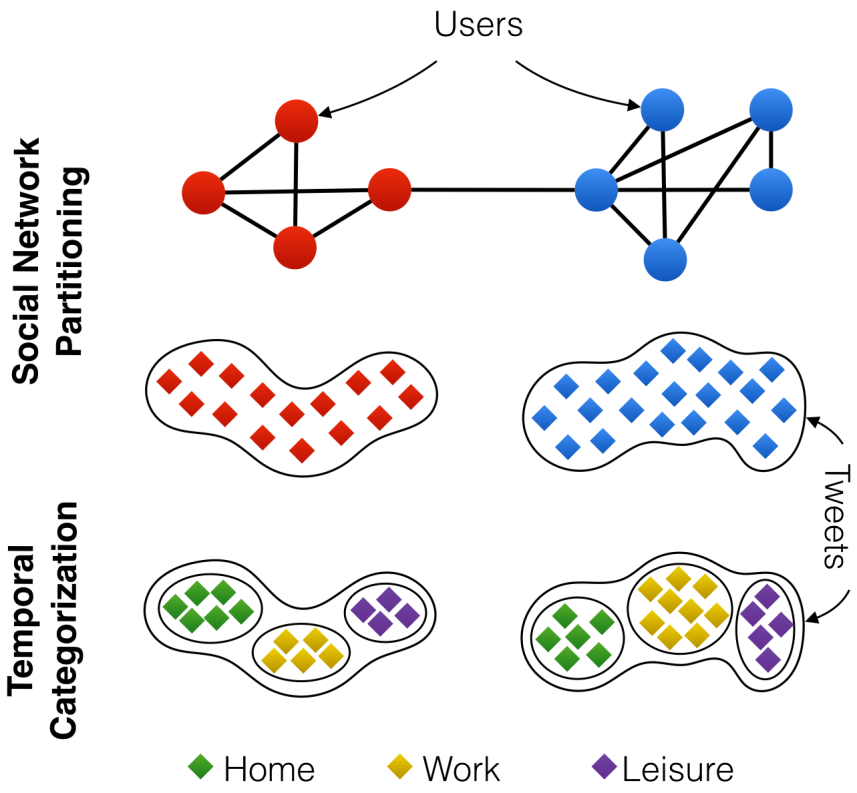

Figure 3.3: A hierarchical algorithm for multiple geo-location identification on Twitter constructed from two layers: *Social Network Partitioning* and *Temporal Categorization*. The goal is to develop a model for identifying geo-location of users in Twitter using their underlying social network graph and the time of their tweets, in presence of limited geo-tagged information (e.g., only less than 6% of the users have their geo-tagging active). The idea is to divide users into geographically close groups using hierarchical clustering of their social network graph and their temporal activities.

to 2014 using a cloud-based mechanism for crawling the data in a distributed manner. We compare the results with the best SotA model [130], which only used the partitioning of the underlying social network graph in their prediction model. The experiment results show that the incorporation of the temporal dimension significantly affects the quality of location prediction. The results show that the temporal dimension is highly correlated with user geographic locality which intuitively affects the topical structure of the messages. Following this intuition, in our next paper we developed a dynamic topic modeling solution that accounts for dynamicity in topic modeling in short message, in addition to sparsity and scalability.

3.2.3 Dynamic Topic Modeling

The study on temporal analysis brought us to develop Graph-based Dynamic Topic modeling (GDTM). We realized that an online solution is needed that can address all properties at once. However, we made significant modifications to the algorithm and the language modeling approach. To account for sparsity we developed a much stronger language model based on the skip-gram language model and leveraged a composition model based on the ideas from skip-gram (non-consecutive n-grams) model.

Figure 3.4 shows the structure of the algorithm that is comprised of four different components which are designed in a pipe-line approach. The stream of documents pass through these components to model and extract their complex topical structure in the form of a dense weighted graph as follows. The first component uses an incremental dimensionality reduction technique to extract simple lexical representations that only encode paradigmatic information (e.g., synonymy) of the words in each document. The second component then combine these representations to construct more complex contextual document vector representations. To this stage the algorithm has only created discrete representations that can express the contextual structure of each document in an isolated manner. However, the true topical structure across the stream requires further similarity analyses to cluster those representations into latent semantic spaces of topics. To achieve this goal we used a graph-based representation approach as the third component of the algorithm to convert each document context vector into a graph representation. This graph representation allows us to encode all documents into a single dense graph representation (e.g., the knowledge graph as in DeGPar, Section 3.2.1) that is capable of localized optimization. However, the main goal of GDTM was to develop an online approach to account for dynamicity as well as scalability and sparsity. Based on that we designed an innovative online graph partitioning algorithm as the fourth component in the pipeline, which is capable of extracting the topics while aggregating stream of documents into the knowledge graph.

We ran two sets of experiments to show the accuracy and scalability of our solution compared to SotA models. For accuracy, we used a tagged Twitter dataset and we compared the accuracy of GDTM with four SotA algorithms. For scalability, we used a large-scale Twitter dataset and compared all the models both in terms of the quality of the clustering and the time required for making the clustering using different approaches. We used B-cubed 2.6.1 for evaluation, and for quality of the clusters in scalability experiments we used another standard evaluation measure called coherence score 2.6.2. GDTM strongly outperforms the SotA models, both on accuracy and scalability.

The next section will present how this language modeling decision together with the inherent incremental properties of random indexing allow us to develop a state-of-the-art contextual representation model capable of scalable representation learning on short messages, with very high quality considering both sparse and continuous representation models.

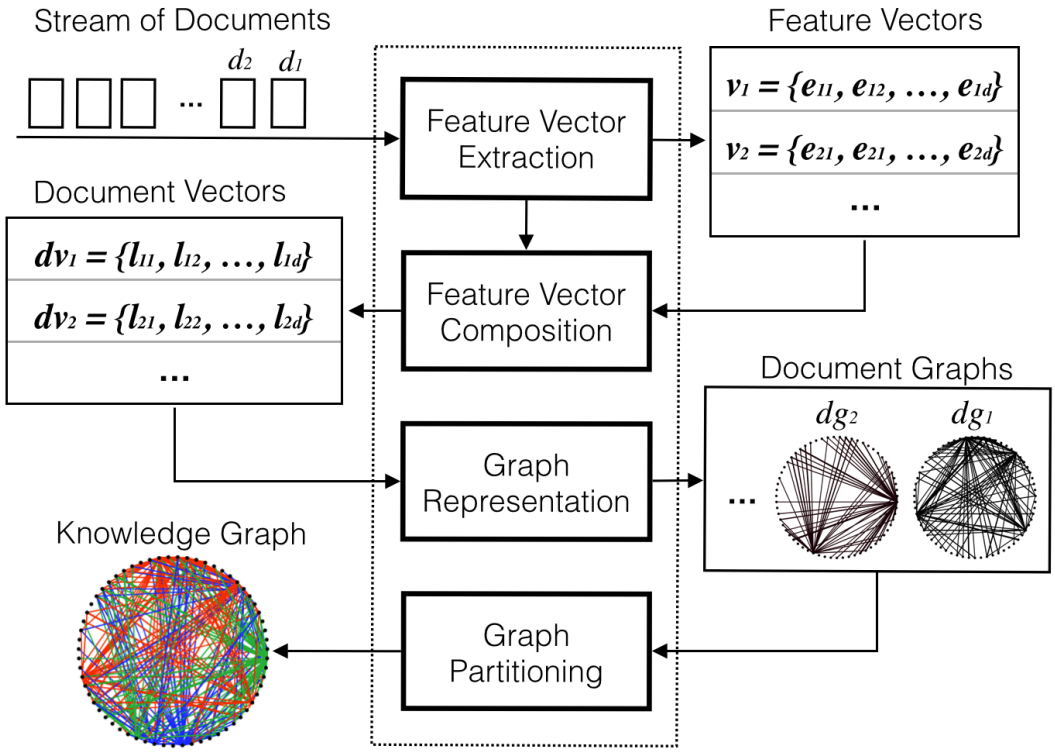

Figure 3.4: The overall protocol of GDTM, a graph-based dynamic topic modeling algorithm. A stream of short texts passes through multiple components where feature vectors are extracted and converted into feature graphs. The feature graphs are then aggregated into a single graph representation, which is partitioned at the same time to extract the topics.

3.3 Representation Learning

The dense graph structure that we designed and created in DeGPar and improved in GDTM encodes a large amount of contextual information which we used to indicate the topics of the short messages. However, from our experiments we realized that it can be used for extracting rich semantic contextual representations of the words in those documents, addressing the problem known as *representation learning* (see Section 2.3). Representation learning has recently received significant attention as a means to capturing the contextual information as low-dimensional vector representations that can considerably improve accuracy and scalability in various downstream NLP applications. Large-scale language models like ELMO [79], BERT [80], and GPT [83, 84] need significant resources to train. Even more computationally efficient approaches like Word2Vec [29], GloVe [32], Doc2Vec [31], FastText [33], LDA2Vec [35], and ULMFiT [81] are still based on iterative back propagation, which limits their scalability.

In Paper V, we proposed a solution to address the problem of scalability on contextual representation learning, using the models built and established in our previous research. Figure 3.5 shows the details of the algorithm that we call ReLeGraph. ReLeGraph is composed of five components where each component applies a set of modifications to the documents, until the contextual representations are generated. The main idea is to design and train a topic model (similar to GDTM and DeGPar) and then generate contextual representations for words by extracting the distribution of the topics

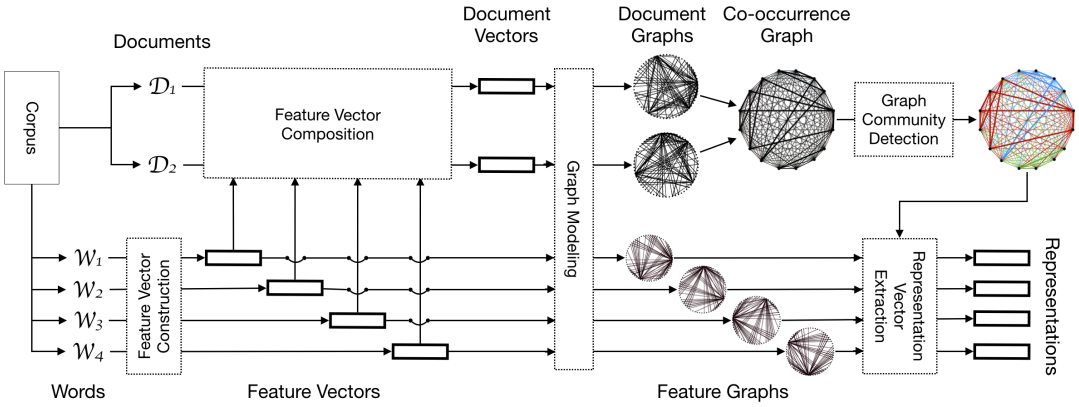

Figure 3.5: ReLeGraph, a model for large scale representation learning with graphs. The algorithm is constructed of five different components that are designed in a pipeline approach which can be applied totally independently. A stream of documents pass through multiple components and the model extracts a representation for each word upon the appearance of the word in a document and keeps updating the representation upon further appearances of the word in the stream. Each word is considered as a feature and is assigned a feature vector using random indexing. The algorithm uses a similar model as DeGPar to extract the topics among documents and then applies that model to extract feature representations.

over their individual feature graph representations. ReLeGraph benefits from a loop-free pipeline approach for feature generation, which makes it an incremental single-pass algorithm. In addition, due to the independence of different components in the pipeline, the algorithm can be implemented in a fully distributed way, which makes it massively scalable, relative to the number of documents to be processed.

We ran a set of experiments to compare ReLeGraph with three state-of-the-art models on a large-scale Twitter dataset and took an extrinsic evaluation approach to measure the quality of the results using a standard evaluation toolkit called SentEval [132]. We compared the scalability and accuracy of ReLeGraph with FastText [33], GloVe [32], and Word2Vec [29] as baseline. The results show that ReLeGraph performs at least 3 times faster than the fastest baseline model (FastText) on generating the representations, while it maintains a reasonably high accuracy compared to all the baseline models.

Chapter 4

Conclusion

4.1 Summary

Even though machines are still in early stages of their development with respect to understanding, let alone comprehension, of the complexities in human natural language, significant improvements have been appeared during recent years. The work reported on in this thesis proposes strong contributions in this direction by tackling one of the most fundamental problems, known as disambiguation, against natural language understanding for machines.

To understand at depth the complexity and the challenges of disambiguation, we made a through study of the current solutions and proposed a solution for multiple disambiguation. We showed that our proposed solution not only improved the overall accuracy of the model compared to the contemporary single disambiguation approaches, but also pin-pointed various important challenges and limitations of those approaches with respect to disambiguation on short messages. We identified three main challenges: sparsity, scalability, and dynamicity as the natural consequence of applying traditional approaches to short messages. Following that, we based our studies on an empirical and exploratory approach, and proposed and developed various solutions to gradually overcome the challenges. Next, we turned our focus to the problem of topic modeling on short messages as the most common approach for disambiguation on natural language text. We developed DeGPar, an efficient graph-based topic modeling solution that combines dimensionality reduction with graph analytics to achieve scalability and overcome sparsity. DeGPar showed significant improvement of the scalability aspect, as compared to the contemporary state-of-the-art solutions on topic modeling. To overcome the problem of dynamicity, we studied the temporal dynamics on short messages in Twitter and identified the strong effect of the spatio-temporal movement of the users on the dynamics of their topics. Following that, we developed GDTM as the dynamic topic modeling solution to account for all three challenges at once. GDTM takes advantage of all the characteristics in DeGPar, including the incremental property of the dimensionality reduction technique to account for dynamicity by developing an online-streaming topic modeling solution.

Finally, we developed ReLeGraph as an scalable and efficient solution for creating contextual representations. ReLeGraph takes advantage of the approaches developed in DeGPar and GDTM to design a multi-purpose disambiguation method based on representation learning that can be used as a generic solution to facilitate multiple down-stream NLP problems. The model has been evaluated on accuracy and scalability following an extrinsic evaluation approach using a standard evaluation toolkit.

All the presented solutions in this thesis achieve one or more of the issues in natural language processing on modern large scale, dynamic and sparse short messages. It is also important to note that a common advantage of all the solutions is that they are all language agnostic; all models have

the capability to run on any set of documents regardless of the language of their text. This property enables the models to be applied on highly linguistically divergent texts generated in online social networks.

4.2 Limitations

The main barrier to this research was related to the availability of standard datasets for development of experiments and evaluation of the models. In the first step, we decided to use available data with valid publication requirements, such as short messages published in Twitter, in our experiments. However, Twitter has limited the access to their public data after 2010, which made it difficult for us to simply crawl and use the data. Therefore, in order to be able to have publishable data we had to collect our own dataset. However, this is time-consuming and Twitter API had very restrictive regulations regarding data collection. To solve this problem, we tried to use a famous Twitter dataset from 2009, called *Twitter7* [133], which has been used in various contemporary solutions, but we could not find it online. Therefore, we contacted the authors of the data and received limited internal access to the data, which has been used in our Paper I. In the next step, we developed a system for data collection and let it run for three years, in parallel with our research, in order to collect the required data to support our further research.

The other barrier to this research was related to the availability of resources for running our large-scale experiments in a distributed environment. We overcame this limitation by using a moderate dedicated cluster computer, and a redesigning of our models to re-scale the proposed solutions and comparing models into reasonable and applicable size, matching our computational resources.

4.3 Future Work

The solutions presented in this thesis open up a new horizon towards design and development of efficient algorithms for large scale and dynamic natural language processing using graph analytics. The combination of an incremental dimensionality reduction technique with complex graph modeling and partitioning algorithms allows us to extract representations in an efficient and scalable way that can benefit various other down-stream NLP problems to gain significant improvement with respect to both accuracy and computational complexity. However, we believe that the proposed approaches can still be improved in multiple ways. First, by designing more complex language models like character-level encoding of the contextual patters to expand the complexity of the representations. Second, by designing domain-specific community representation metrics with stronger relations to the community structures in natural language text. Moreover, we believe that the proposed solutions are domain-agnostic and can be further modified and customized to be applied to similar sequence learning problems from other domains, such as image processing and biological sequence analysis learning.

References

[1] K. Ghoorchian, F. Rahimian, and S. Girdzijauskas, "Semi-supervised multiple disambiguation," in *2015 IEEE Trustcom/BigDataSE/ISPA*, vol. 2, pp. 88–95, Aug 2015.

[2] K. Ghoorchian, S. Girdzijauskas, and F. Rahimian, "DeGPar: Large scale topic detection using node-cut partitioning on dense weighted graphs," in *2017 IEEE 37th International Conference on Distributed Computing Systems (ICDCS)*, pp. 775–785, June 2017.

[3] K. Ghoorchian and S. Girdzijauskas, "Spatio-temporal multiple geo-location identification on twitter," in *2018 IEEE International Conference on Big Data (Big Data)*, pp. 3412–3421, Dec 2018.

[4] K. Ghoorchian and M. Sahlgren, "GDTM: Graph-based dynamic topic models," in *Submitted*, 2019.

[5] K. Ghoorchian, M. Sahlgren, and M. Boman, "An efficient graph-based model for learning representation," in *Submitted*, 2019.

[6] C. D. Manning and H. Schütze, *Foundations of Statistical Natural Language Processing*. Cambridge, MA, USA: MIT Press, 1999.

[7] R. Grishman and B. Sundheim, "Message understanding conference-6: A brief history," in *Proceedings of the 16th Conference on Computational Linguistics - Volume 1*, COLING '96, (Stroudsburg, PA, USA), pp. 466–471, Association for Computational Linguistics, 1996.

[8] D. Nadeau and S. Sekine, "A survey of named entity recognition and classification," *Lingvisticae Investigationes*, vol. 30, 01 2007.

[9] G. Stricker, "The 2014 #yearontwitter." https://blog.twitter.com/official/en_us/a/2014/the-2014-yearontwitter.html, 2014.

[10] G. Zipf, *The Psycho-Biology of Language: An Introduction to Dynamic Philology*. Cognitive psychology], Routledge, 1999.

[11] G. Zipf, *Human behavior and the principle of least effort: an introduction to human ecology*. Addison-Wesley Press, 1949.

[12] G. Salton and M. J. McGill, *Introduction to Modern Information Retrieval*. New York, NY, USA: McGraw-Hill, Inc., 1986.

[13] S. Deerwester, S. T. Dumais, G. W. Furnas, T. K. Landauer, and R. Harshman, "Indexing by latent semantic analysis," *JOURNAL OF THE AMERICAN SOCIETY FOR INFORMATION SCIENCE*, vol. 41, no. 6, pp. 391–407, 1990.

[14] C. Eckart and G. Young, "The approximation of one matrix by another of lower rank," *Psychometrika*, vol. 1, pp. 211–218, Sep 1936.

[15] T. Hofmann, "Probabilistic latent semantic indexing," in *Proceedings of the 22Nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '99, (New York, NY, USA), pp. 50–57, ACM, 1999.

[16] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent dirichlet allocation," *J. Mach. Learn. Res.*, vol. 3, pp. 993–1022, Mar 2003.

[17] S. Geman and D. Geman, "Stochastic relaxation, gibbs distributions, and the bayesian restoration of images," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 6, pp. 721–741, Nov. 1984.

[18] M. D. Hoffman, D. M. Blei, C. Wang, and J. Paisley, "Stochastic variational inference," *Journal of Machine Learning Research*, vol. 14, pp. 1303–1347, 2013.

[19] X. Yan, J. Guo, Y. Lan, and X. Cheng, "A biterm topic model for short texts," in *Proceedings of the 22Nd International Conference on World Wide Web*, WWW '13, (New York, NY, USA), pp. 1445–1456, ACM, 2013.

[20] D. M. Blei and J. D. Lafferty, "Dynamic topic models," in *Proceedings of the 23rd International Conference on Machine Learning*, ICML '06, (New York, NY, USA), pp. 113–120, ACM, 2006.

[21] C. Wang, D. Blei, and D. Heckerman, "Continuous time dynamic topic models," in *Proceedings of the Twenty-Fourth Conference on Uncertainty in Artificial Intelligence*, UAI'08, (Arlington, Virginia, United States), pp. 579–586, AUAI Press, 2008.

[22] J. Yin and J. Wang, "A dirichlet multinomial mixture model-based approach for short text clustering," in *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '14, (New York, NY, USA), pp. 233–242, ACM, 2014.

[23] J. Yin and J. Wang, "A text clustering algorithm using an online clustering scheme for initialization," in *Proceedings of the 22Nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '16, (New York, NY, USA), pp. 1995–2004, ACM, 2016.

[24] J. Qiang, Y. Li, Y. Yuan, and X. Wu, "Short text clustering based on pitman-yor process mixture model," *Applied Intelligence*, vol. 48, pp. 1802–1812, jul 2018.

[25] K. Nigam, A. K. Mccallum, S. Thrun, and T. Mitchell, "Text classification from labeled and unlabeled documents using em," *Machine Learning*, vol. 39, pp. 103–134, May 2000.

[26] I. Sato and H. Nakagawa, "Topic models with power-law using pitman-yor process," in *Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '10, (New York, NY, USA), pp. 673–682, ACM, 2010.

[27] T. Mikolov, I. Sutskever, K. Chen, G. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," in *Proceedings of the 26th International Conference on Neural Information Processing Systems - Volume 2*, NIPS'13, (USA), pp. 3111–3119, Curran Associates Inc., 2013.

[28] Y. Bengio, A. Courville, and P. Vincent, "Representation learning: A review and new perspectives," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, pp. 1798–1828, Aug 2013.

[29] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," *CoRR*, vol. abs/1301.3781, 2013.

[30] J. L. Elman, "Finding structure in time," *Cognitive Science*, vol. 14, pp. 179–211, 1990.

[31] Q. Le and T. Mikolov, "Distributed representations of sentences and documents," in *Proceedings of the 31st International Conference on International Conference on Machine Learning - Volume 32*, ICML'14, pp. II–1188–II–1196, JMLR.org, 2014.

[32] J. Pennington, R. Socher, and C. Manning, "Glove: Global vectors for word representation," in *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 1532–1543, Association for Computational Linguistics, 2014.

[33] P. Bojanowski, E. Grave, A. Joulin, and T. Mikolov, "Enriching word vectors with subword information," *CoRR*, vol. abs/1607.04606, 2016.

[34] A. Joulin, E. Grave, P. Bojanowski, and T. Mikolov, "Bag of tricks for efficient text classification," *CoRR*, vol. abs/1607.01759, 2016.

[35] C. E. Moody, "Mixing dirichlet topic models and word embeddings to make lda2vec," *CoRR*, vol. abs/1605.02019, 2016.

[36] D. Jurafsky and J. H. Martin, *Speech and Language Processing (2Nd Edition)*. Upper Saddle River, NJ, USA: Prentice-Hall, Inc., 2009.

[37] A. Franz, *Automatic Ambiguity Resolution in Natural Language Processing: An Empirical Approach*. Berlin, Heidelberg: Springer-Verlag, 1996.

[38] P. Bhattacharyya, "Natural language processing: A perspective from computation in presence of ambiguity, resource constraint and multilinguality," *CSI Journal of Computing*, vol. 1, 03 2012.

[39] B. Aljoscha, W. Stephan, K. Alexander, K. Michael, B. Patrick, and B. Johan, "What are scope ambiguities?." http://www.coli.uni-saarland.de/projects/milca/courses/comsem/html/node92.html, 2003.

[40] A. Chang, V. I. Spitkovsky, C. D. Manning, and E. Agirre, "A comparison of named-entity disambiguation and word sense disambiguation," in *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)* (N. C. C. Chair), K. Choukri, T. Declerck, S. Goggi, M. Grobelnik, B. Maegaard, J. Mariani, H. Mazo, A. Moreno, J. Odijk, and S. Piperidis, eds.), (Paris, France), European Language Resources Association (ELRA), may 2016.

[41] N. Ide and J. Véronis, "Introduction to the special issue on word sense disambiguation: The state of the art," *Comput. Linguist.*, vol. 24, pp. 2–40, Mar. 1998.

[42] R. Navigli, "Word sense disambiguation: a survey," *ACM COMPUTING SURVEYS*, vol. 41, no. 2, pp. 1–69, 2009.

[43] M. Francis-Landau, G. Durrett, and D. Klein, "Capturing semantic similarity for entity linking with convolutional neural networks," in *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 1256–1261, Association for Computational Linguistics, 2016.

[44] P. Verga, E. Strubell, and A. McCallum, "Simultaneously self-attending to all mentions for full-abstract biological relation extraction," in *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pp. 872–884, Association for Computational Linguistics, 2018.

[45] M. Dredze, P. McNamee, D. Rao, A. Gerber, and T. Finin, "Entity disambiguation for knowledge base population," in *Proceedings of the 23rd International Conference on Computational Linguistics*, COLING '10, (Stroudsburg, PA, USA), pp. 277–285, Association for Computational Linguistics, 2010.

[46] P. Bhatia, K. Arumae, and B. Celikkaya, "Dynamic transfer learning for named entity recognition," *CoRR*, vol. abs/1812.05288, 2018.

[47] Y. Sasaki, Y. Tsuruoka, J. McNaught, and S. Ananiadou, "How to make the most of ne dictionaries in statistical ner," *BMC Bioinformatics*, vol. 9, p. S5, Nov 2008.

[48] A. Chisholm and B. Hachey, "Entity disambiguation with web links," *Transactions of the Association for Computational Linguistics*, vol. 3, pp. 145–156, 2015.

[49] C. Fellbaum, "A semantic network of english: The mother of all wordnets," *Computers and the Humanities*, vol. 32, pp. 209–220, Mar 1998.

[50] P. T. Vossen, "The global wordnet association." http://globalwordnet.org/, 2019.

[51] R. Navigli and S. P. Ponzetto, "BabelNet: The automatic construction, evaluation and application of a wide-coverage multilingual semantic network," *Artificial Intelligence*, vol. 193, pp. 217–250, 2012.

[52] M. Sahlgren and S. universitet. Institutionen för lingvistik, *The Word-Space Model: Using Distributional Analysis to Represent Syntagmatic and Paradigmatic Relations Between Words in High-dimensional Vector Spaces*. SICS dissertation series, Department of Linguistics, Stockholm University, 2006.

[53] A. Bagga and B. Baldwin, "Entity-based cross-document coreferencing using the vector space model," in *Proceedings of the 17th International Conference on Computational Linguistics - Volume 1*, COLING '98, (Stroudsburg, PA, USA), pp. 79–85, Association for Computational Linguistics, 1998.

[54] J. Mayfield, D. Alexander, B. J. Dorr, J. Eisner, T. Elsayed, T. Finin, C. Fink, M. Freedman, N. Garera, P. McNamee, S. Mohammad, D. W. Oard, C. D. Piatko, A. B. Sayeed, Z. Syed, R. M. Weischedel, T. Xu, and D. Yarowsky, "Cross-document coreference resolution: A key technology for learning by reading," in *AAAI Spring Symposium: Learning by Reading and Learning to Read'09*, pp. 65–70, 2009.

[55] C. H. Gooi and J. Allan, "Cross-document coreference on a large scale corpus," in *HLT-NAACL'04*, pp. 9–16, 2004.

[56] J. Dean and S. Ghemawat, "Mapreduce: Simplified data processing on large clusters," *Commun. ACM*, vol. 51, pp. 107–113, Jan. 2008.

[57] P. Pantel, E. Crestan, A. Borkovsky, A.-M. Popescu, and V. Vyas, "Web-scale distributional similarity and entity set expansion," in *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 2 - Volume 2*, EMNLP '09, (Stroudsburg, PA, USA), pp. 938–947, Association for Computational Linguistics, 2009.

[58] L. Kolb, A. Thor, and E. Rahm, "Dedoop: Efficient deduplication with hadoop," *Proc. VLDB Endow.*, vol. 5, pp. 1878–1881, Aug. 2012.

[59] S. Singh, A. Subramanya, F. Pereira, and A. McCallum, "Large-scale cross-document coreference using distributed inference and hierarchical models," in *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies - Volume 1*, HLT '11, (Stroudsburg, PA, USA), pp. 793–803, Association for Computational Linguistics, 2011.

[60] L. Sarmento, A. Kehlenbeck, E. Oliveira, and L. Ungar, "An approach to web-scale named-entity disambiguation," in *Proceedings of the 6th International Conference on Machine Learning and Data Mining in Pattern Recognition*, MLDM '09, (Berlin, Heidelberg), pp. 689–703, Springer-Verlag, 2009.

[61] J. Allan, J. Carbonell, G. Doddington, J. Yamron, Y. Yang, J. A. Umass, B. A. Cmu, D. B. Cmu, A. B. Cmu, R. B. Cmu, I. C. Dragon, G. D. Darpa, A. H. Cmu, J. L. Cmu, V. L. Umass, X. L. Cmu, S. L. Dragon, P. V. M. Dragon, R. P. Umass, T. P. Cmu, J. P. Umass, and M. S. Umass, "Topic detection and tracking pilot study final report," in *In Proceedings of the DARPA Broadcast News Transcription and Understanding Workshop*, pp. 194–218, 1998.

[62] T. K. Landauer and S. T. Dumais, "A solution to plato's problem: The latent semantic analysis theory of the acquisition, induction, and representation of knowledge," *Psychological Review*, vol. 104, pp. 211–240, 1997.

[63] P. Kanerva, J. Kristoferson, and A. Holst, "Random indexing of text samples for latent semantic analysis," in *In Proceedings of the 22nd Annual Conference of the Cognitive Science Society*, pp. 103–6, Erlbaum, 2000.

[64] D. M. Blei, "Probabilistic topic models," *Commun. ACM*, vol. 55, pp. 77–84, Apr. 2012.

[65] C. M. Bishop, *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Berlin, Heidelberg: Springer-Verlag, 2006.

[66] J. Chang and D. M. Blei, "Relational topic models for document networks," in *Proceedings of the Twelfth International Conference on Artificial Intelligence and Statistics (AISTATS-09)* (D. V. Dyk and M. Welling, eds.), vol. 5, pp. 81–88, Journal of Machine Learning Research - Proceedings Track, 2009.

[67] D. A. Cohn and T. Hofmann, "The missing link - a probabilistic model of document content and hypertext connectivity," in *Advances in Neural Information Processing Systems 13* (T. K. Leen, T. G. Dietterich, and V. Tresp, eds.), pp. 430–436, MIT Press, 2001.

[68] J. Weng, E.-P. Lim, J. Jiang, and Q. He, "Twitterrank: Finding topic-sensitive influential twitterers," in *Proceedings of the Third ACM International Conference on Web Search and Data Mining*, WSDM '10, (New York, NY, USA), pp. 261–270, ACM, 2010.

[69] L. Hong and B. D. Davison, "Empirical study of topic modeling in twitter," in *Proceedings of the First Workshop on Social Media Analytics*, SOMA '10, (New York, NY, USA), pp. 80–88, ACM, 2010.

[70] S. Stieglitz, M. Mirbabaie, B. Ross, and C. Neuberger, "Social media analytics – challenges in topic discovery, data collection, and data preparation," *International Journal of Information Management*, vol. 39, pp. 156 – 168, 2018.

[71] D. M. Blei and J. D. Lafferty, "Correlated topic models," in *Proceedings of the 18th International Conference on Neural Information Processing Systems*, NIPS'05, (Cambridge, MA, USA), pp. 147–154, MIT Press, 2005.

[72] Z. S. Harris, "Distributional structure," WORD, vol. 10, no. 2-3, pp. 146–162, 1954.

[73] J. Firth, "A synopsis of linguistic theory 1930-1955," in *Studies in Linguistic Analysis*, Philological Society, Oxford, 1957. reprinted in Palmer, F. (ed. 1968) Selected Papers of J. R. Firth, Longman, Harlow.

[74] D. E. Rumelhart, J. L. McClelland, and C. PDP Research Group, eds., *Parallel Distributed Processing: Explorations in the Microstructure of Cognition, Vol. 1: Foundations*. Cambridge, MA, USA: MIT Press, 1986.

[75] G. E. Hinton, J. L. McClelland, and D. E. Rumelhart in *Parallel Distributed Processing: Explorations in the Microstructure of Cognition, Vol. 1* (D. E. Rumelhart, J. L. McClelland, and C. PDP Research Group, eds.), ch. Distributed Representations, pp. 77–109, Cambridge, MA, USA: MIT Press, 1986.

[76] K. Ethayarajh, D. Duvenaud, and G. Hirst, "Towards understanding linear word analogies," in *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, (Florence, Italy), pp. 3253–3262, Association for Computational Linguistics, July 2019.

[77] O. Melamud, J. Goldberger, and I. Dagan, "context2vec: Learning generic context embedding with bidirectional lstm," in *Proceedings of The 20th SIGNLL Conference on Computational Natural Language Learning*, (Berlin, Germany), pp. 51–61, Association for Computational Linguistics, Aug. 2016.

[78] Z. SchlangerAugust, "Why you won't see much news about the devastating amazon rainforest fires on google news." https://qz.com/1692755/amazon-fire-tablet-upstages-amazon-rainforest-fires-on-google/, 2019.

[79] M. E. Peters, M. Neumann, M. Iyyer, M. Gardner, C. Clark, K. Lee, and L. Zettlemoyer, "Deep contextualized word representations," *CoRR*, vol. abs/1802.05365, 2018.

[80] J. Devlin, M. Chang, K. Lee, and K. Toutanova, "BERT: pre-training of deep bidirectional transformers for language understanding," *CoRR*, vol. abs/1810.04805, 2018.

[81] J. Howard and S. Ruder, "Fine-tuned language models for text classification," *CoRR*, vol. abs/1801.06146, 2018.

[82] A. M. Dai and Q. V. Le, "Semi-supervised sequence learning," *CoRR*, vol. abs/1511.01432, 2015.

[83] A. Radford, K. Narasimhan, T. Salimans, and I. Sutskever, "Improving language understanding by generative pre-training," 2018.

[84] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, and I. Sutskever, "Language models are unsupervised multitask learners," 2018.

[85] J. Bondy and U. Murty, *Graph Theory*. Springer Publishing Company, Incorporated, 1st ed., 2008.

[86] E. David and K. Jon, *Networks, Crowds, and Markets: Reasoning About a Highly Connected World*. New York, NY, USA: Cambridge University Press, 2010.

[87] L. Page, S. Brin, R. Motwani, and T. Winograd, "The pagerank citation ranking: Bringing order to the web.," November 1999. Previous number = SIDL-WP-1999-0120.

[88] S. Gay, F. Fages, T. Martinez, S. Soliman, and C. Solnon, "On the subgraph epimorphism problem," *Discrete Appl. Math.*, vol. 162, pp. 214–228, Jan. 2014.

[89] J. Lee, W.-S. Han, R. Kasperovics, and J.-H. Lee, "An in-depth comparison of subgraph isomorphism algorithms in graph databases," in *Proceedings of the 39th international conference on Very Large Data Bases*, PVLDB'13, pp. 133–144, VLDB Endowment, 2013.

[90] D. Bruschi, L. Martignoni, and M. Monga, "Detecting self-mutating malware using control-flow graph matching," in *Proceedings of the Third International Conference on Detection of Intrusions and Malware & Vulnerability Assessment*, DIMVA'06, (Berlin, Heidelberg), pp. 129–143, Springer-Verlag, 2006.

[91] B. Balasundaram, S. Butenko, and S. Trukhanov, "Novel approaches for analyzing biological networks," *Journal of Combinatorial Optimization*, vol. 10, pp. 23–39, Aug 2005.

[92] J. Suomela, "Survey of local algorithms," *ACM Computing Surveys*, vol. 45, no. 2, pp. 24:1–40, 2013.

[93] M. Zaharia, M. Chowdhury, M. J. Franklin, S. Shenker, and I. Stoica, "Spark: Cluster computing with working sets," in *Proceedings of the 2Nd USENIX Conference on Hot Topics in Cloud Computing*, HotCloud'10, (Berkeley, CA, USA), pp. 10–10, USENIX Association, 2010.

[94] A. Toshniwal, S. Taneja, A. Shukla, K. Ramasamy, J. M. Patel, S. Kulkarni, J. Jackson, K. Gade, M. Fu, J. Donham, N. Bhagat, S. Mittal, and D. Ryaboy, "Storm@twitter," in *Proceedings of the 2014 ACM SIGMOD International Conference on Management of Data*, SIGMOD '14, (New York, NY, USA), pp. 147–156, ACM, 2014.

[95] J. Dean and S. Ghemawat, "Mapreduce: Simplified data processing on large clusters," *Commun. ACM*, vol. 51, pp. 107–113, Jan. 2008.

[96] R. F. Mihalcea and D. R. Radev, *Graph-based Natural Language Processing and Information Retrieval*. New York, NY, USA: Cambridge University Press, 1st ed., 2011.

[97] S. S. Sonawane and P. A. Kulkarni, "Graph based representation and analysis of text document: A survey of techniques," *International Journal of Computer Applications*, vol. 96, pp. 1–8, June 2014. Full text available.

[98] A. Buluç, H. Meyerhenke, I. Safro, P. Sanders, and C. Schulz, "Recent advances in graph partitioning," *CoRR*, vol. abs/1311.3144, 2013.

[99] L. R. Ford and D. R. Fulkerson, *Maximal Flow Through a Network*, pp. 243–248. Boston, MA: Birkhäuser Boston, 1987.

[100] K. I. Kim, J. Tompkin, H. Pfister, and C. Theobalt, "Context-guided diffusion for label propagation on graphs," *CoRR*, vol. abs/1602.06439, 2016.

[101] S. Fortunato, "Community detection in graphs," *Physics Reports*, vol. 486, no. 3-5, pp. 75 – 174, 2010.

[102] C. Schulz, S. K. Bayer, C. Hess, C. Steiger, M. Teichmann, J. Jacob, F. Bernardes-lima, R. Hangu, and S. Hayrapetyan, "Course notes: Graph partitioning and graph clustering in theory and practice," 2015.

[103] M. Newman and M. Girvan, "Finding and evaluating community structure in networks.," *Physical review. E, Statistical, nonlinear, and soft matter physics*, vol. 69 2 Pt 2, p. 026113, 2004.

[104] C. Tsourakakis, C. Gkantsidis, B. Radunovic, and M. Vojnovic, "Fennel: Streaming graph partitioning for massive scale graphs," in *Proceedings of the 7th ACM International Conference on Web Search and Data Mining*, WSDM '14, (New York, NY, USA), pp. 333–342, ACM, 2014.

[105] I. Stanton and G. Kliot, "Streaming graph partitioning for large distributed graphs," in *Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '12, (New York, NY, USA), pp. 1222–1230, ACM, 2012.

[106] I. Stanton, "Streaming balanced graph partitioning algorithms for random graphs," in *Proceedings of the Twenty-fifth Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA '14, (Philadelphia, PA, USA), pp. 1287–1301, Society for Industrial and Applied Mathematics, 2014.

[107] A. Hollocou, J. Maudet, T. Bonald, and M. Lelarge, "A linear streaming algorithm for community detection in very large networks," *CoRR*, vol. abs/1703.02955, 2017.

[108] M. Newman, "Power laws, pareto distributions and zipf's law," *Contemporary Physics*, vol. 46, no. 5, pp. 323–351, 2005.

[109] W. B. Johnson, J. Lindenstrauss, and G. Schechtman, "Extensions of lipschitz maps into banach spaces," *Israel Journal of Mathematics*, vol. 54, pp. 129–138, Jun 1986.

[110] M. Sahlgren, "An introduction to random indexing," in *Proceedings of the Methods and Applications of Semantic Indexing Workshop at the 7th International Conference on Terminology and Knowledge Engineering (TKE)*, (Copenhagen, Denmark), 2005.

[111] D. Guthrie, B. Allison, W. Liu, L. Guthrie, and Y. Wilks, "A closer look at skip-gram modelling," in *Proceedings of the Fifth International Conference on Language Resources and Evaluation (LREC'06)*, European Language Resources Association (ELRA), 2006.

[112] A. Bagga and B. Baldwin, "Entity-based cross-document coreferencing using the vector space model," in *Proceedings of the 17th International Conference on Computational Linguistics - Volume 1*, COLING '98, (Stroudsburg, PA, USA), pp. 79–85, Association for Computational Linguistics, 1998.

[113] D. Mimno, H. M. Wallach, E. Talley, M. Leenders, and A. McCallum, "Optimizing semantic coherence in topic models," in *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, EMNLP '11, (Stroudsburg, PA, USA), pp. 262–272, Association for Computational Linguistics, 2011.

[114] A. Rosenberg and J. Hirschberg, "V-measure: A conditional entropy-based external cluster evaluation measure.," in *EMNLP-CoNLL*, pp. 410–420, ACL, Jun 2010.

[115] A. Conneau and D. Kiela, "Senteval: An evaluation toolkit for universal sentence representations," in *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC-2018)*, European Language Resource Association, 2018.

[116] B. Wellner, A. McCallum, F. Peng, and M. Hay, "An integrated, conditional model of information extraction and coreference with application to citation matching," in *Proceedings of the 20th Conference on Uncertainty in Artificial Intelligence*, UAI '04, (Arlington, Virginia, United States), pp. 593–601, AUAI Press, 2004.

[117] M. Wick, S. Singh, and A. McCallum, "A discriminative hierarchical model for fast coreference at large scale," in *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Long Papers - Volume 1*, ACL '12, (Stroudsburg, PA, USA), pp. 379–388, Association for Computational Linguistics, 2012.

[118] F. Rahimian, S. Girdzijauskas, and S. Haridi, "Parallel community detection for cross-document coreference," in *Proceedings of the 2014 IEEE/WIC/ACM International Conference on Web Intelligence*, WI'14, August 2014.

[119] J. Yang and J. Leskovec, "Patterns of temporal variation in online media," in *Proceedings of the Fourth ACM International Conference on Web Search and Data Mining*, WSDM '11, (New York, NY, USA), pp. 177–186, ACM, 2011.

[120] N. Baten, "Tweets2011." http://trec.nist.gov/data/tweets/, 2011.

[121] J. Goldenberg and M. Levy, "Distance is not dead: Social interaction and geographical distance in the internet era," *CoRR*, vol. abs/0906.3202, 2009.

[122] C. Wang, J. Wang, X. Xie, and W.-Y. Ma, "Mining geographic knowledge using location aware topic model," in *Proceedings of the 4th ACM Workshop on Geographical Information Retrieval*, GIR '07, (New York, NY, USA), pp. 65–70, ACM, 2007.

[123] J. Eisenstein, B. O'Connor, N. A. Smith, and E. P. Xing, "A latent variable model for geographic lexical variation," in *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, EMNLP '10, (Stroudsburg, PA, USA), pp. 1277–1287, Association for Computational Linguistics, 2010.

[124] Z. Cheng, J. Caverlee, and K. Lee, "You are where you tweet: A content-based approach to geo-locating twitter users," in *Proceedings of the 19th ACM International Conference on Information and Knowledge Management*, CIKM '10, (New York, NY, USA), pp. 759–768, ACM, 2010.

[125] L. Hong, A. Ahmed, S. Gurumurthy, A. J. Smola, and K. Tsioutsiouliklis, "Discovering geographical topics in the twitter stream," in *Proceedings of the 21st International Conference on World Wide Web*, WWW '12, (New York, NY, USA), pp. 769–778, ACM, 2012.

[126] R. Li, S. Wang, and K. C.-C. Chang, "Multiple location profiling for users and relationships from social network and content," *Proc. VLDB Endow.*, vol. 5, pp. 1603–1614, July 2012.

[127] J. McGee, J. Caverlee, and Z. Cheng, "Location prediction in social media based on tie strength," in *Proceedings of the 22Nd ACM International Conference on Information & Knowledge Management*, CIKM '13, (New York, NY, USA), pp. 459–468, ACM, 2013.

[128] Zhu and Z. Ghahramani, "Learning from labeled and unlabeled data with label propagation," *School Comput Sci Carnegie Mellon Univ Pittsburgh PA Tech Rep CMUCALD02107*, vol. 54, no. CMU-CALD-02-107, pp. 1–19, 2002.

[129] L. Kong, Z. Liu, and Y. Huang, "Spot: Locating social media users based on social network context," *Proc. VLDB Endow.*, vol. 7, pp. 1681–1684, Aug. 2014.

[130] R. Compton, D. Jurgens, and D. Allen, "Geotagging one hundred million twitter accounts with total variation minimization," *CoRR*, vol. abs/1404.7152, 2014.

[131] D. Jurgens, "That's What Friends Are For: Inferring Location in Online Social Media Platforms Based on Social Relationships," in *Proceedings of the 7th International AAAI Conference on Weblogs and Social Media*, pp. 273–282, 2013.

[132] A. Conneau and D. Kiela, "Senteval: An evaluation toolkit for universal sentence representations," *CoRR*, vol. abs/1803.05449, 2018.

[133] J. Yang and J. Leskovec, "Patterns of temporal variation in online media," in *Proceedings of the Fourth ACM International Conference on Web Search and Data Mining*, WSDM '11, (New York, NY, USA), pp. 177–186, ACM, 2011.