TEMPORAL MODELS IN DATA MINING:

ENRICHMENT, SUMMARIZATION AND RECOMMENDATION

Von der Fakultät für Elektrotechnik und Informatik
der Gottfried Wilhelm Leibniz Universität Hannover
zur Erlangung des Grades

DOKTOR DER NATURWISSENSCHAFTEN

Dr. rer. nat.

genehmigte Dissertation
von

M.Sc. Anh Tuan Tran

geboren am 24. März 1985, in Hanoi, Vietnam

2018

Referent: Prof. Dr. techn. Wolfgang Nejdl
Korreferent: Prof. Dr. Vinay Setty
Korreferent: Prof. Dr. Heribert Vollmer
Tag der Promotion: 07. August 2018

ABSTRACT

Time plays a major and multifaceted role when studying digital collections and their relationship with the user. This is especially true when digital contents are digested long after their time point of generation, creating rooms where numerous events can be observed: The contents are revised or overwritten, users are exposed to other contents in the collection with related information and thereby update their interpretation and interest. Furthermore, the evolution of context and new events beyond the collection can also influence user perception of relevance or values of the contents. Therefore, good information processing and retrieval systems should take into account these effects of time, not only within a single object but also from intra- as well as inter-collections. At this collective level, it is important to address the user cognitive behaviour in processing information, as humans have the ability of connecting different pieces of information from various sources, and they often exercise this power, consciously or unconsciously, when generating and consuming digital contents.

Despite two decades of active research in temporal data mining and information retrieval, little attention has been paid to the effects of the cognitive aspects on time-aware information access at the collection level. This includes aspects such as how users perceive and memorise long-running events reported in online news, or how human forgetfulness affects their behaviour in finding their own digital material, etc.

In this thesis, we investigate several research questions in temporal data mining in the new perspective, inspired from the human cognitive processes in creating, organizing, exchanging, and seeking temporal information. In particular, we address the tasks of: (1) identifying temporal topics from texts and enriching them with semantic annotations; (2) summarizing text data using timeline, as well as via cognitive-based models; (3) helping users in finding their own documents through studying the influence of time on their memory at work. In more detail, we introduce a novel method to annotate topics for textual data that leverages social and temporal signals, and demonstrate its effectiveness for trending topics in social media posts such as in Twitter. We also address the scalability issue, both in algorithmic perspective and computational architecture perspective. Furthermore, we introduce the concept of entity-centric timeline summarization for long-running events in news collection, again exploiting social and temporal signals from encyclopedic resources such as Wikipedia together with features from text. For this purpose, we propose a novel adaptive learning algorithms that leverage both the relevance and novelty of news articles. As another contribution in content summarization from a more cognitive perspective, we study the domain of summarizing dialogues for decisions, taking into account the foundations in human decision making theory. Finally, we make contributions in studying human memories in enterprise domain, and design a new graph learning method to recommend professional contents for a temporal task at hand.

Keywords: *information extraction, temporal data analysis, semantic data, cognitive models, summarization, recommender system, learning to rank, structured learning*

ZUSAMMENFASSUNG

Die Zeit spielt eine wichtige und vielseitige Rolle dabei, die digitalen Sammlungen unter Berücksichtigung der Beziehung zu den Nutzern zu studieren. Dies gilt inbesondere, wenn digitale Inhalte lange nach ihrem Erstellungszeitpunkt verarbeitet werden und die Zeitraum, in dem zahlreiche Ereignisse beobachtet werden können, erzeugt wird. Zum Beispiel: Die Dokumente werden geändert oder überarbeitet; Benutzer werden anderen verbundenen Information in der Sammlung ausgesetzt und damit ihr Wissen, ihre Interpretation und Ihre Interesse aktualisieren. Ein weiteres Beispiel können die Veränderung des Kontexts und die Entstehung neuer Ereignisse die Wahrnehmung der Nutzer von Relevanz oder Werten der Inhalte beeinflussen. Daher sollte hochwertige Informationsverarbeitungs -und- aufrufsysteme diese Auswirkungen der Zeit berücksichtigen, nicht nur innerhalb eines einzelnen Objekt, sondern auch von Intra- sowie Intersammlungen. Auf dieser kollektiven Ebene ist es wichtig, das kognitive Verhalten des Nutzers bei der Verarbeitung von Informationen einzugehen. Der Grund dafür ist, dass Menschen die Fähigkeit haben, Informationen aus verschiedenen Quellen zu verbinden, und dass sie diese Befugnis oft bewußt oder bewußtlos ausüben, wenn sie digitale Inhalte erstellen oder erfassen.

Trotz jahrzehntelanger Forschung in zeitlichem Data Mining und Information Retrieval wurde bislang den Auswirkungen der kognitiven Aspekte auf den zeitabhängigen Informationsverarbeitung auf der Sammlungsebene nur wenig Aufmerksamkeit geschenkt. Dazu gehören Aspekte wie die Art und Weise, wie Nutzer Dauerereignisse z.B. in Online-Nachrichten verfolgen oder auswendig lernen, oder wie sich menschliche Vergesslichkeit auf ihr Verhalten bei der Suche nach ihrem eigenen digitalen Material auswirkt.

In dieser Dissertation legen wir verschiedene Forschungsfragen im zeitlichen Data Mining aus einer neuen Perspektive fest, inspiriert von den menschlichen kognitiven Prozessen bei den Erstellungen, Organisierungen, Austauschen und Suchen nach zeitlichen Informationen. Insbesondere richten wir auf folgende Fragestellung aus: (1) Wie die zeitlichen Themen von Textinhalte ermittelt werden, und damit wie die Inhalte um semantische Information angereichert werden; (2) Wie die Textdaten anhand der Zeitleiste sowie von kognitiv basierten Modellen richtig zusammengefasst werden; (3) Wie ein System die Nutzer bei der Such nach ihrem eigenen Dokumente unterstützen kann, indem die Auswirkung der Zeit auf ihre Gedächtnis berücksichtigt wird. Zur ersten Frage führen wir eine neue Methode für Anreicherung der zeitlichen Themen von Textdaten ein, die soziale und zeitliche Merkmale verwendet, und zeigen ihre Wirksamkeit, um Social-Media-Trending z.B. in Twitter anzureichern. Außerdem befassen wir uns mit dem Thema Skalierbarkeit, sowohl in Bezug auf die algorithmischen Modellen als auch auf die Infrastruktur. Zur zweiten Frage setzen wir das neue Konzept der Entity-basierten Zeitleiste als Zusammenfassung der Dauerereignisse ein, und entwickeln wir neue Methodenansätzen zur effektiven Zusammenfassen von Online-Nachrichtenartikeln. Unsere Methode kombiniert soziale und zeitliche Merkmalen, die aus Enzyklopädischen wie Wikipedia entstehen, mit herkömmliche Merkmalen der Textinhalte. Die Methode kann auch anhand eines neuartigen adaptiven Lernalgorithmus die Relevanz und die Neuheit von Nachrichtenartikeln ausgleichen. Neben Online-Nachrichten untersuchen wir weiteren Bereich des Zusammenfassens von gesprochenen Dialogen unter

Berücksichtigung der menschlichen Entscheidungsverfahren. Zur dritten Frage leisten wir Beiträge zur Erforschung menschlicher Erinnerungen im Unternehmensbereich und entwerfen eine neue Graph-Lernmethode, um professionelle Inhalte für eine zeitliche Aufgabe zu empfehlen.

Schlagwörter: *Informationsextraktion, Zeitliche Datenanalyse, semantischen Daten, Kognitives Modell, Textzusammenfassung, Empfehlungsdienst, Lernen auf Rang, strukturiertes Lernen*

ACKNOWLEDGMENTS

In the course of this thesis, I am grateful to many people for their support and assistance. First of all, I would like to acknowledge my advisor Prof. Dr. Wolfgang Nejdl, for giving me the opportunity to work in L3S Research Center together with the many excellent scientists in the area of Web Science, and on many interesting projects. I also thank associate Prof. Dr. Vinay Setty, for agreeing to consider and evaluate my PhD thesis. In addition, I would like to thank Ms Anca Vais and Ms Marion Walters for their organizational efforts during the beginning and last phases of my PhD study. I truly appreciate all of their time and work.

A special thank to Dr. Claudia Niederée, who has been both my project manager and my research mentor since the very first day of my PhD. Dr. Niederée has given me invaluable guidance during my time at L3S, both professional and personal, which has shaped my skills in conducting research, managing time, writing scientific and business works, etc. In this thesis, I would like to give her heartfelt thanks for all of her support.

Additionally, I would like to express my gratitude to my colleagues and friends at L3S for many fruitful discussions, as well as for great experiences that we share in this house. It has been amazing six years with a lot of fun and lessons, without which I would not have been able to accomplish many of the achievements in this thesis.

I would like to extend my sincerest thanks to my parents, my sisters Lan Anh and Hai Anh for their encouragement and continuous support during my time in Germany. Finally, I am indebted to my wife Nhung and my daughter Vy, for their unconditional love, and for being the source of my motivation and power every single day. This thesis would not have been possible without them.

FOREWORD

The methods and algorithms presented in this thesis have been published at various conferences, as follows.

Chapter 3 addresses the problem of enrichment of collections of temporal data using annotation and indexing scheme. The results have been published in:

- Tuan Tran, Nam Khanh Tran, Asmelash Teka Hadgu, Robert Jäschke. *Semantic Annotation for Microblog Topics Using Wikipedia Temporal Information*. In Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, **EMNLP'15**, pages 97–106, Lisbon, Portugal, 2015. ACL. [TTTHJ15]

- Tuan Tran, Tu Ngoc Nguyen. *Hedera: Scalable Indexing, Exploring Entities in Wikipedia Revision History*. In Proceedings of the 13th International Semantic Web Conference, **ISWC'14**, pages 297–300, Riva del Garda, Italy, 2014. Springer. [TN14]

Chapter 4 focuses on one application of enriched data collection, which is text summarization using two fashions: Timeline and memory models. It discusses two temporal and cognitive approaches, resulting in two summarization methods: Timeline-based and neural network-based summarization. The results of this chapter are discussed in:

- Tuan Tran, Nattiya Kahnabua, Claudia Niederée, Ujwal Gadiraju, Avishek Anand. *Balancing Novelty and Salience: Adaptive Learning to Rank Entities for Timeline Summarization of High-impact Events*. In Proceedings of the 24th ACM International Conference on Information and Knowledge Management, **CIKM'15**, pages 1201–1210, New York, NY, USA, 2015. ACM. [TNK+15]

- Tuan Tran, Francesca Bonin, Léa A. Deleris, Debasis Ganguly, Killian Levacher. *Preparing a Dataset for Extracting Decision Elements from a Meeting Transcript Corpus*. (**IBM Research Report**, No. IRE1803-019. March 21, 2018.)

Chapter 5 studies another application of semantically enriched data collection: Recommendation. Specifically, we discuss various issues in recommendation in

semantic information systems, where documents are not isolated but are connected via manifold relationships. We focus on enterprise data domain as the case study. The works in this chapter appear in:

- Tuan Tran, Sven Schwarz, Claudia Niederée, Heiko Maus, Nattiya Kanhabua. *The Forgotten Needle in My Collections: Task-Aware Ranking of Documents in Semantic Information Space*. In Proceedings of the 1st ACM SIGIR Conference on Human Information Interaction and Retrieval, **CHIIR'16**, pages 35-44, New York, NY, USA, 2016. ACM. [TSN+16]

During the Ph.D. study, I have also published and co-authored a number of papers touching different aspects of entity mining, temporal topics, cognitive learning, and event analytics. Not all aspects are discussed in this thesis due to space limitation. The complete list of publications is as follows:

- Claudia Niederee, Nattiya Kanhabua, Tuan Tran, Kaweh Djafari Naini. *Preservation Value and Managed Forgetting*. In Personal Multimedia Preservation: Remembering or Forgetting Images and Video. The Springer Series on Cultural Computing (**SSCC'18**), pages 101–129, 2018. Springer. [NKTN18]

- Nam Khanh Tran, Tuan Tran, Claudia Niederee. *Beyond Time: Dynamic Context-aware Entity Recommendation*. In Proceedings of the 14th Extended Semantic Web Conference, **ESWC'17**, pages 353–368, Portoroz, Slovenia, 2017. Springer. [VTN+16]

- Khoi Duy Vo, Tuan Tran, Tu Ngoc Nguyen, Xiaofei Zhu, Wolfgang Nejdl. *Can We Find Documents in Web Archives without Knowing their Contents ?*. In Proceedings of the 8th International ACM Conference on Web Science, **Websci'16**, pages 173–182, New York, NY, USA, 2011. ACM. [VTN+16]

- Holger Eichelberger, Claudia Niederee, Apostolos Dollas, Ekaterini Ioannou, Cui Qin, Grigorios Chrysos, Christoph Hube, Tuan Tran, Apostolos Nydriotis, Pavlos Malakonakis, Stefan Burkhard, Tobias Becker, Minos Garofalakis. *Configure, Generate, Run: Model-based Development for Big Data Processing*. In European Project Space on Intelligent Technologies, Software engineering, Computer Vision, Graphics, Optics and Photonics, **SCITEPRESS'16**, Roma, Italy, 2016. SCITEPRESS. [END+16]

- Tuan Tran, Andrea Ceroni, Mihai Georgescu, Kaweh Djafari Naini, Marco Fisichella. *WikipEvent: Leveraging Wikipedia Edit History for Event Detection*. In Proceedings of 15th International Conference on Web Information System Engineering, **WISE'14**, pages 90–108, Thessaloniki, Greece, 2014. Springer. [TCG+14]

- Tuan Tran, Mihai Georgescu, Xiaofei Zhu, Nattiya Kanhabua. *Analysing the duration of trending topics in Twitter using Wikipedia*. In Proceedings of the 2014 ACM Conference on Web Science, **WebSci'14**, pages 251–252, New York, NY, USA, 2014. ACM. [TGZK14]

- Khaled Hossain Ansary, Tuan Tran, Nam Khanh Tran. *A pipeline tweet contextualization system at INEX 2013*. In Working Notes of Conference and Labs of the Evaluation Forum, **CLEF'13**, Valencia, Spain. 2013. CEUR-WS.org. [ATT13]

- Giang Binh Tran, Tuan Tran, Nam Khanh Tran, Mohammad Alrifai, Nattiya Kanhabua. *Leveraging Learning To Rank in an Optimization Framework for Timeline Summarization*. In Proceedings of Workshop on Time-aware Information Access, the 36th International ACM SIGIR Conference on Research and Development in Information Retrieval, **TAIA'13**, 2013. [TTT+13b]

- Tuan Tran. *Exploiting temporal topic models in social media retrieval*. In Proceedings of the 35th International ACM Conference on Research and Development in Information Retrieval, **SIGIR'12**, pages 999–999, New York, NY, USA, 2012. ACM. [Tra12]

Contents

List of Figures

List of Tables

1 Introduction

With the explosive production of digital devices and proliferation of online platforms, digital contents become ubiquitous and norms of information consumption every day. People generate data using their smart phones, record their lives in social media, and meanwhile are exposed to a great deal of online news media. Companies increasingly advocate digitization of their assets to reduce maintenance costs and improve collaboration. Such trends are inevitable. Among the different types of digital contents, textual contents play an important role, as reading is still one of the most popular cognitive process for human to acquire information. With the rapid growth data generated every day, it is an acute mission to assist user in consuming texts with automated data mining methods and systems.

Yet humans rarely examine and interpret a content in isolation. A human brain naturally establishes connections between neurons to facilitate processing, transmitting of information, and to form memories. This innate ability is exercised all the time when people generate, exchange, search and explore data. However, it is more pronounced when people navigate a collection than a single document for a certain information need. Example digital collections include a news storyline consisting articles discussing the same story, or an enterprise database. When processing such collection, people consciously or unconsciously connect information pieces from different contents to obtain desired knowledge. Grouping digital contents into collections is indeed a common means to ease exploration: Photo albums, news topics, document folders classified by events, projects, or by sharing policy, etc. Even in online social media, when the contents are often exchanged in unplanned ways, there are mechanisms to organize information, such as using hashtags, constructing social groups, or discussion threads.

It is stunning to observe on the one hand the growth of online digital collections to assist quick and convenient digestion, and on the other hand, the increasing difficulties and frustration users face in consuming and comprehending content of true interest. In addition to the increasing size of the collections, which is recognized and popularized as the main cause of an "information overload" issue, the complexity and heterogeneous of digital collections also play a role in the obstruction. Among dimensions that contribute to this

complexity, there is one which becomes the target of much study in data mining, and of this thesis as well: Time.

1.1 Time and Digital Collections

Time characterizes a fundamental and independent dimension in which events are sequenced. With time, everything evolves: Knowledge is expanded, topics are changed, contexts get shifted, etc. Time also plays a critical role in the way humans perceive such development, or the way they remember or forget things. These virtues of time have a special influence on how contents in a digital collection are created and interpreted:

Firstly, the value and interpretation of contents are sensitive to contexts and can change over time. Much of the digital contents are generated at a certain time point, but are consumed at other time points, in many cases long after. Thus, there are time gaps between generation and consumption activities, and during these gaps, contexts might well change: New concepts are defined, the background that makes one topic dominant becomes no longer the case. This can alter the user interpretation for such contents. For example, a news article about a sport event could draw high attention after a few days of its creation, but it quickly becomes less relevant the week after. The time gap between creation and consumption also changes the concept interpretation. For example, the word "gay" was heavily used in old English to mean "joyful", "carefree"[1], and only includes sexual connotation until the 20th century. Digital archives of former contents such as the ballet "The Gay Parisian" (*Gaîté Parisienne*, Léonide Massine, 1938) would be difficult to understand without consideration of their time of premiere.

Secondly, a digital collection is not formed at one point, but rather accumulated over time, as different documents are generated, and grouped or re-grouped together. This is especially the case when the collection is constructed on the basis of a temporal topic such as an event: New documents are added when the event gets updated. One implication is that when users get exposed to different contents in sequence, they can update their own understanding about the subject. For instance, when subscribing to a stream of social media posts about a story, users' understanding can change according to new unfolded details.

Thirdly, unlike physical object, each digital content can be overridden: A Web page is updated, Office documents are modified, etc. Over time, this creates different versions of a digital resource, each version can be consumed by different users and at different time. For example, if someone examines the Wikipedia page of Barack Obama in 2007 and 2012, they will get completely different revisions. In fact, the low cost of online text editing has enabled the rapid development of the Web, and approaches attempting to capture temporal development of the Web (e.g. Wayback Machine[2] needs to archive multiple revisions of Web pages at different point in time.

Last but not least, images of contents seen by humans stay in their memory for a certain

[1] https://en.wikipedia.org/wiki/Gay
[2] https://archive.org/

amount of time. When presented with new contents, humans have the ability to connect with the old content, within or beyond the collection, to form an overall picture, or even to reason new fragments of information. Because of this cognitive capability, a digital collection of items presented to users in the course of time can create additional information or even sentiments to the users. This can be seen in how some social media platforms such as Twitter [3] report a long-running event, where each post in isolation might be terse and may not make sense, but together they can draw a good overview picture of the event. Other example is a transcript of a conversation: It is often impossible to understand what one participant mean by their utterances without connecting to those of previous ones. Even in a narrative article such as a Wikipedia page, references can be used to infer who is being referred in the later parts of the text.

With the recent development of the Web, the above influence of time on digital texts is increasingly pronounced. Online contents become more temporally dynamic and also quicker to saturate [HNA16, SH10]. The creation and editing of digital contents become now much easier with the presence of many user-contributed online platforms such as YouTube, Facebook, Instagram, Twitter, etc. In addition, the Internet also changes human habits in remembering, organizing and seeking information [SLW11].

1.2 Time-aware Information Access

The past two decades have witnessed tremendous advances in improving and revolutionizing access to digital data. The efforts initiated by the database and information retrieval (IR) communities led to novel architectures, methods and technologies in organizing data, as well as in effective access and filtering of information. Much of these advances are still crucial to today's search engines and management systems. Recently, researchers start to recognize the role of time on digital information access, and identify special issues and desired quality when dealing with time-sensitive contents.

In Web search, it is shown that neglecting the recency of data in news and social media might bring negative value to user experience, due to issues such as obsolete information or time-inconsistency of a collection [Met07]. Apart from that, there is also *temporal information need*, for instance when the need for information access is elicited within a particular time period (searching for materials while on an assignment), such time period should also be taken into account in well-performing search systems [KBN+15]. Other example is the searching about external events and topics, either directly queried by the user, or propagated within a social network [KLPM10].

In information filtering, the timeliness of data to be presented to the user, taking into account the specific temporal constraints, is of particular concern [BC92]. For example, for users who have followed an event in news media, it is desirable not to display contents that they have already seen, but novel ones instead, ideally in an intuitive way to help users visualize the development of the stories. In recommendation, a good recommender system

[3] https://twitter.com/

should not only suggest contents similar to the user profile, but also relevant to the temporal context, for example under special occasions (Christmas, sport events, etc.)

In text understanding, providing the temporal background when a document or collection was generated will make the contents much more accessible. For the ballet example "The Gay Parisian" mentioned above, the disambiguation of the word "gay" ("joyful") helps clarify the confusion. Other challenge is to enrich contents with annotations of named entities such as persons, organizations or locations, taking into account the temporal evolution of such entities [TGK+12]. For instance, when mentioning "The US president" in a news article published after 20 January 2017, the user should be presented with the entity Donald Trump, and not Barack Obama.

The above examples of time-aware information access tasks, although not comprehensive, illustrate the importance of incorporating time into modern data mining systems. In fact, in information retrieval (IR) research, temporal information retrieval[4] has become a vey active topic recently with several fruitful results, as illustrated in [Met07, BC92, KLPM10]. While early work focused on extracting and indexing time-related information explicit from the contents [MW00, SG10b, BBNW07], recent study has started to realize the connection between this information and the human memory [DCC+03, MMJ05, AYJ11] . In many scenarios, there are different patterns in the way users search, retain, or recall information over time, and such patterns can be explained via studies related to human remembering and forgetting [PDR13]. This is also the premise on which this thesis has been grounded.

1.3 Motivation

In this thesis, we aim to study the impact of time on assisting user in accessing digital contents *at the collection level*. Our study is conducted in a new perspective, which has drawn little attention in time-aware information access research: The influence of time reflects the dynamics of human cognitive behaviour in creating, organizing, exchanging, and seeking temporal information. More specifically, we aim to model the time dimension in digital collections, both from development of the underlying context in contents, and from the view points of human perception to such development. We study how this new way of modelling differs from traditional approaches, which model documents individually.

We focus our study on textual contents to get benefits from advanced low-level technologies in text understanding. To this extent, we frame our work in particular application domains: In online social media, and in business domains. The choice of two contrasting domains of studies - public contents and professional contents - have shed light on the multifaceted roles of time in how the text data is generated and exchanged, where the human cognitive process is largely different. In short, our research question is the following:

Research Goal. *How can the access to a text collection driven by temporal information needs be improved, if the system gets information about human cognitive behaviour in pro-*

[4] https://en.wikipedia.org/wiki/Temporal_information_retrieval

cessing such collections ?

The term "access" here both means passive consumption, where the user is presented and recommended with information from the collection, and active consumption, where the user performs search or navigation through the colleciton. The concept "Temporal information need" is very broad, and in our work, is studied in the context of two domains, social media and businesss, as mentioned above. In the social media domain, we focus on the need of users following popular events reported in news. In the business domain, we study the need of a professional user performing temporal tasks at hand, either alone or collaboratively with other colleagues (for instance, in a meeting). In both domains, we assume, and if possible, develop an environment in which the human cognitive process can be automatically observed by a computer system. We believe such information is valuable in improving time-aware access in the collection level. To evaluate this idea, we investigate new methods which combines studies of human remembering and forgetting with temporal models in text mining, and compare them with established models in literature. The research goal of this thesis can be divided into further questions such as follows.

First, we seek the ideas to set up an environment in which human behaviour in creating and exploring text collections can be understood by a computer system. There are different approaches to record the observations of such information in text contents, and in our work, we aim to set up an *enrichment* process, where the metadata are embedded into contents in a computer-processable format.

Second, once the collection has been enriched, we study how to improve different tasks in exploring its documents under certain temporal information need. Here we follow two directions of application domains as mentioned above and study two specific tasks: The summarization of text collections, and the recommendation of business documents for a temporal task. Each task leads to one research sub-goal which is studied separately in this thesis.

1.4 Contributions of the Thesis

During this thesis, we have developed a number of machine learning-based methods to target different aspects of the above research questions. The major contributions of this thesis are:

First, we propose and advocate the use of semantic annotations in combination with the temporal models. The semantic annotations help encoding computer-processable information of the context into textual contents, which can better facilitate the temporal models in different text mining applications. This contribution is mainly discussed in chapter 3.

Second, we define novel summarization techniques both in social media and business domains. In social media, we propose a novel news summarization approach based on semantic annotations (entity), which provides a higher-level overview of the story. The details of our methods are described in chapter 4. In addition, we also propose a novel adaptive learning framework that can summarize different kinds of news stories (social events, natural disasters, etc.) in an automated and unified way. In the business domain, we study a

new task of decision summarization from enterprise meeting transcripts, with novel methods based on neural networks, inspired by human long- and short-term memory.

Third, we study the application of recommending documents for enterprise data in daily professional scenarios, modeled via semantic graph ranking problem. Not only entity information but also their relationships are considered using novel supervised graph learning methods. Details about this study are given in chapter 5.

We also have made additional contributions as follows:

Human Remembering Dynamics as Implicit Feedback in Machine Learning: Besides developing new machine learning methods, we also propose effective approaches to infer the human implicit feedback to automatically, or semi-automatically gather training data. Our methods rely on several observations on patterns of human memory. Since manually crafting data labels is a tedious task, our proposed process is helpful to scale up the training data and improve the learning quality significantly. These approaches are discussed in chapter 3 (section 3.2.4), chapter 4 (secion 4.2.3) and chapter 5 (section 5.2.1) respectively.

Implementation of the frameworks as an open-source projects: As the final contribution, we have developed and open-sourced some of the algorithms discussed in this thesis in achieving scalable computation via big data framework that runs on Hadoop infrastructures. The details are given in chapter 3 (section 3.3).

1.5 Thesis Structure

The remainder of the thesis is organized as follows:

In chapter 2 we discuss selected general background techniques and algorithms that build a basis to achieve the goals of this thesis. In particular, we focus on selected techniques from the areas of Machine Learning, Information Retrieval, and Information Extraction.

Chapter 3 discusses the study and proposed model for enrichment of temporal topics in text collections, taking into account social in addition to linguistic features. We also briefly discuss the issue of scalability through one study of high-performance indexing of large-scale corpus.

Chapter 4 describes our proposed framework for summarization of text using two approaches: Timeline-based summarization and memory-based summarization, each on different domains of text: Online news and multi-party dialogues in business scenarios.

Chapter 5 discusses the temporal models used in documents recommendation for business scenario. Specifically, we introduce the new system to suggest an associate to find their own documents at work according to a task at hand.

Finally, we conclude the contributions of this thesis again and point out directions for future research in chapter 6.

2 General Background

In this chapter we briefly introduce general background techniques and algorithms that form the basis of this thesis. In particular, we focus on selected techniques from the areas of Information Extraction (IE) and Information Retrieval (IR). In addition, as the thesis is heavily built upon several machine learning models, we also discuss several related machine learning (ML) backgrounds in a separate section.

First, we present relevant Information Extraction on entity recognition and linking. Second, we discuss relevant Information Retrieval background, specifically temporal information retrieval. This includes indexing temporal collections, coping with temporal information need, as well as other advanced techniques. Third, we discuss in more details the related background in the area of machine learning, upon which our various models are built. This includes both a discussion about supervised and weakly-supervised machine learning techniques, for unstructured and structured (graph) data. Finally, we review the background in cognitive science, in particular human remembering and forgetting, focusing on relevant literature to our work.

2.1 Relevant Information Extraction Background

During this dissertation, we devised and used a number of methods to annotate semantic information from unstructured data. These methods lie in the broad research area of information extraction, and the unstructured data can be text, images or arbitrary types. This section discusses the relevant foundations and methods in extracting textual data, and focuses on the most pertinent sub-tasks: named entity recognition and linking, and sequence tagging.

2.1.1 Basic Concepts

Named Entities. This is the core concepts in many parts of this thesis, and it accounts for most of the semantic information presented in the data. One definition which has been popularized in the information extraction community, and employed in this thesis as well, defines named entities as "the atomic elements of text" - single or multiword expression - that can be predefined into categories, such as organizations, locations, persons, events, quantity, expression of time, etc. [NS07]. Note that there is a difference between named entities and entities. The word *named* indicates the restriction of categories of text referents, or *rigid designators* [1] (e.g. the term "Michael Jordan" is a term to a person), while *entities* refer to the thing that exists uniquely and have an unambiguous indicator.(e.g. Michael Jordan is the US professional basketball player). As a result, there are two closely-related areas of information extraction: Named entity recognition and entity linking, described in 2.1.2 and 2.1.3.

Knowledge Base. Between entities can exist different semantic relationships. For instance, the entity Berlin is the capital of the entity Germany, or Germany is the type (or belongs to the class) Country. A data structure that stores all entities together with their relations is called a knowledge base [2]. One popular standard used to represent and process knowledge bases is *Resource Description Framework* (RDF [LS99]), which contains a set of vocabularies and specifications guiding how entities and relations should be formatted (e.g., entity should have one unique resource identifier, for example an URL). There exists research work on how to construct a knowledge base automatically from knowledge resources, most notably Wikipedia, resulting in different large-scale ontologies such as YAGO [SKW07], DBpedia [ABK+07], Freebase [BEP+08], etc.

2.1.2 Named Entity Recognition

Named entity recognition (NER for short) is a sub-task in information extraction that aim to identify and classify named entities, or text referents, inferred from the textual phrases (mention) in an unstructured or semi-structured data [NS07]. NER can be considered as one phase of a natural language processing (NLP) pipeline, which in the full form [MSB+14] consists of tokenization (identifying unit of texts or tokens from the raw text), lemmatization (normalization of tokens), tagging (equipping phrases and words with meta-data), chunking (grouping relevant words), dependency parsing (inferring the syntax of the text), and role labelling (detecting semantic arguments across phrases and sentences). In this pipeline, NER corresponds to the tagging phase, which takes into account a block of tokens, and annotates with most applicable categories (person, location, organization, etc.). Much of these phases require high-quality training data, as in typical supervised machine learning approaches. Because of this, performance of NER systems greatly vary depending on the language of

[1] https://en.wikipedia.org/wiki/Rigid_designator

[2] Or more precisely, an ontological knowledge base, to distinguish with general knowledge bases [GG95]

study, nature of the text corpora (e.g., news or scientific domains), etc. In standard English text benchmark, state-of-the-art NER systems achieve near-human performance [3]

Speaking of algorithms, most of NER systems employ some classification methods. What makes an NER system special is the dependency between tokens and phrases in close positions (e.g. the word "Jordan" succeeding "Michael" is more likely about a person than a location). Therefore, most of the NER algorithms are variants of the sequence labelling algorithms, such as Hidden Markov Model (HMM [Edd96]), Conditional Random Fields (CRF [LMP+01], section 2.3.1) or Long Short-Term Memory (LSTM, section 2.3.1).

Early NER systems rely heavily on the syntactic nature of the languages, on the hand-crafted features [RR09], or even on the presence of gazeeters (dictionary of tokens to most probable classes), which makes it difficult to switch from one domain to the other (and in some cases, the system has to be re-developed from scratch [RCE+11]). Recently, there is a new trend in using recurrent neural networks to design a reusable and robust NER system with impressive performance, not only in English but also other languages [CWB+11]. Core of this boost is the use of word embedding techniques [MSC+13a], which represent tokens in a language-independent, continuous vector form.

2.1.3 Entity Linking

In contrast to NER, entity linking, also named entity linking or named entity disambiguation, addresses the problem of ambiguity of referents in named entities, and aims to determine the identity of the entity mentioned in the text. For example, in the sentence "Michael Jordan played three seasons for coach Dean Smith at the University of North Carolina", "Michael Jordan" should be referred to the basketball player and not the professor of the UC Berkeley [4]. Target entities are often registered in an ontological knowledge bases such as YAGO [HYB+11], references to Wikipedia pages [MW08] (sometimes also called *wikification*), or the federated sources of linked data [UNR+14].

Entity linking is an NP-hard problem [RRDA11], and thus all algorithms employ some approximation heuristics. For example, Cucerzan et al. [Cuc07] requires each entity profile should be textually similar to the context of the mentions, AIDA [HYB+11]) or TagMe [FS12] assumes there is a dense connection of entities and coherence in types in the document, or Guo et al. [GB14] hypothesizes that the hyper-links in Wikipedia reveals the contextual coherence among the correctly assigned entities.

In NLP pipeline, entity linking can be one additional phase continuing NER phase, and take advantage of available recognized named entities, which can be used to identify candidates, there also exist end-to-end entity linking systems that integrates the two steps [SY13, DK14, LHLN15], or even integrates entity linking with other tasks such as word sense disambiguation [MRN14].

When applying entity linking to larger scales such as the Web, many techniques have

[3] https://en.wikipedia.org/wiki/Named-entity_recognition
[4] https://people.eecs.berkeley.edu/~jordan/

been proposed to boost the efficiency, such as the expansion of mention-candidate dictionary [HB16], the avoidance of text-heavy reasoning methods in favour of more lightweight sources such as hyperlinks [GGL+16], relying on the crowdsourced platforms [DDCM12], or the use of advanced sampling algorithms [ZKGN14].

Besides traditional text, the entity linking task in other domains such as in search queries [BOM15], or in microblogs (e.g. Twitter [5]) is also of high interest. In Twitter, due to its popularity, entity linking methods have been actively studied, and can be divided into two classes, i.e., content-based and graph-based methods. The content-based methods [MWDR12, GCK13] consider tweets independently when linking mentions to entities, while the graph-based methods [CJR+12, LLW+13] use all related tweets (e.g., posted by a user) together.

When evaluating entity linking, one needs high-quality annotated data sources such as Wikipedia [HRN+13] or CoNLL [HYB+11], etc. GERBIL [CFC13] provides a good set of benchmark for entity linking in standard and in social media domains. Alternatively, crowdsource sites such as CrowdFlower [6] can also be used for evaluation, with special care in controlling quality and designing interface [BDR17].

2.2 Relevant Information Retrieval Background

A large amount of subjects in this thesis lie in the area of Information Retrieval (IR), where the main concern is to obtain information from one or multiple resources relevant to particular information needs, either stated explicitly or implicitly. Since IR is a very broad research area, with more than three decades of active study, in this thesis, we only focus on the three sub topics of IR that are closest to our study: Temporal information retrieval, semantic search, and learning to rank.

2.2.1 Temporal Information Retrieval

Temporal information retrieval is a family of research topics that deal with temporal information in documents, collections or in queries, in order to improve a wide range of tasks from document exploration, similarity search, summarization, clustering, etc. [ASBYG11]. Temporal information retrieval is becoming an important trend in recent development of IR, due to the rapid development of dynamic contents such as news, social media, digitized historical collections [Aye99, Tof07], etc., and the increasing interest in searching information with temporal requirements [BBAW10]. Depending on the source of time dimensions, whether from documents, context or from the user query, research in this field can be classified into different sub-areas. When time stems from document contents, we have temporal information extraction and tagging [CM12, SG13], document dating [KN09]. When time comes from global resources (inter-documents, other collections, global contexts) we have

[5] https://twitter.com
[6] http://www.crowdflower.com

temporal summarization [YWO+11], temporal clustering [AGBY09], query understanding [KTSD11], real-time search [WML10], etc.

In addition to sources of time, temporal qualities such as timeliness and currency of the results are also key aspects in determining the credibility of a retrieval system [7]. Improving these qualities results in a rich body of study in time-aware ranking models, which can be grouped into different types: Recency-based ranking, time-dependent ranking, event- and entity-aware ranking. [KBN+15] gives a detailed survey of these groups of ranking models. Here we briefly discuss the recency-based ranking model, as this relates directly to our study.

Recency-based Ranking. In this approach, retrieval systems reward documents that are published in the more recent time period with respect to the time expressed or interpreted in the query. Early findings by Li et al. [LC03] and later confirmed by other authors (e.g., [DCZ+10, ZCZ+09]) suggest that in time-sensitive queries, favoring such recent documents can improve retrieval performance in general. Most of recency-based ranking models incorporate decay functions to model the distribution of information in the documents, which is inspired by the decaying of information retention in human memory [PDR13]. This "temporal prior" can be incorporated to traditional ranking models such as relevance models [LC03], language models [BBAW10], link analysis [DD10], or machine learning-based models [DSD11], etc. While recency is mostly determined via document publication time, other factors can also contribute to determine the recency of the documents, such as maintenance activities of their snapshots [DD10].

2.2.2 Semantic Search

Another line of IR that gains increasing interest - semantic Search - aims to harness the structured information for search, which is useful for information needs that concern entity or structural information such as attributes, relation of objects, etc. A study by Pound et al. [PMZ10] found that more than 40% of Web search queries are related to entities, suggesting the importance of semantic search techniques in today's search engines. Strictly speaking, semantic search refers to not a single but to a number of different tasks [MBO14]: Understanding intent and contextual meaning from the documents and queries, finding and ranking answers (objects, attributes) instead of documents (sometimes also called entity or object retrieval), and combining free-form contents with structures in all steps of the search system. Compared to traditional, keyword-based search, ranking models in semantic search need to be revisited to accommodate the special structure of search queries and results. For example, when applying BM25 ranking model to entity retrieval, one can build a separate model for each field of the entity [PAAG+10], or if using graph-based ranking models such as PageRank, the propagation from different documents and entity types should be considered separately [SRH08]. For query understanding, some special challenges also need to be

[7] Metzger et al. [Met07] argue that five most important factors for a good search results are: relevance, accuracy, objectivity, coverage, and timeliness.

addressed, such as ambiguity of named entities in the query [HBB15], entity typing (detecting the proper types of the entity in the queries based on the context [SC13]). [MBO14] provides a good list of existing works in this area.

Entity Ranking and Entity Salience. One particular area, entity ranking, has gained significant interest from the semantic search community recently [MWL+12, MMBJ13]. Although a lot of interesting work on entity ranking has been proposed, most of previous works have focused on static collections, thus ignoring the temporal dynamics of queries and documents. The relevant work closest to ours in this respect is by Demartini et al. [DMBZ10], where the task is to identify the entities that best describe the documents for a given query. The entities are identified by analyzing the top-k retrieved documents at the time when the query was issued as well as relevant documents in the past. In IR, salient entities in news articles help improving the performance of retrieval[MMBJ13]. There is a body of work in identifying salient entities in general Web [GYS+13] and in news domain [DG14], but the existing work does not take into account the time dimension, as one of the core contribution in our thesis.

2.2.3 Learning to Rank

An additional line of research in machine learning that is touched in this thesis (particularly in chapter 4) is Learning to Rank (L2R), which aims to learn patterns from ordered data that "explain" their ranks the best. The data is often documents (text, images, digital objects) relevant to an information need, represented by one or multiple queries. The ranks can be explicit such as document scores with respect to the query, or implicit such as via binary relevance labels. The criteria of how one explanation is better than the others specify different L2R approaches, summarized in the three main categories below.

In *pointwise* approach, the goal is to approximate document-query scores using ordinal regression or classification algorithms. Example methods include OPRF [Fuh89], SLR [CGD92].

In *pairwise* approach, the absolute approximation errors are not important, the goal is instead to minimize the mis-ranks, i.e. given two documents of the same query, documents of less relevance should not be scored higher [Joa02, FISS03]. Compared to pointwise approach, the learning performance is often higher and less sensitive to biasedness. The pair can be constructed between binary-labeled or ordinal-scored documents, or aggregated from crowdsourced results [CBCTH13].

In *listwise* approach, the L2R system tries to optimize directly the list in which documents are scored and ordered, based on the lists presented in training queries. Because variables of the optimization functions are sets instead of individual documents, listwise L2R are more difficult to model, and different assumptions must be introduced to simplify the process, such as in Plackett-Luce model [XLW+08, KCW09].

Adaptive Learning to Rank. In traditional L2R, all queries are treated equivalently. In

many real-world scenarios, however, some queries are more informative than the others, such as those which are more recent, if timeliness is of the major concerns of the system. Other examples are queries of different domains of interest, such as informational queries versus navigational queries [Bro02], which should be treated differently during the training time.

A special line of L2R research has addressed these adaptivity issues. The common framework is to design different models for each type of queries, then using ensemble method to learn the final ranking function. One of the earliest work is by Bian et al. [BLL+10], which modifies RankSVM to adapt to each individual training queries. In [DSD11], Dai et al. defined for each documents the quality of being recent (fresh) and relevant, and argue that these qualities contribute differently for different queries. For navigational queries on versioned collections, recent work also suggested that timeliness is a significant factor and should be integrated into relevance [CCS14, NKNZ15]. In social media, adaptive L2R is one of the effective tool to model different aspects of user opinions, and can help finding relevant comments for a product [DSS12].

2.3 Relevant Machine Learning Background

In this thesis, we employ a number of supervised machine learning methods, where the models are learnt from a set of training data and applied to a new set of data to produce the prediction. Since this is a very broad discipline, not all aspects are discussed in this thesis. In the following, we give the overview of the related work in three focused domains, which are relevant the most to the thesis: Learning from sequential data and graph data.

2.3.1 Sequence Learning

The first related area is sequence learning. The main research goal is given a sequence of items, the system is expected to automatically provide certain answers based on the combination of evidences from individual input items, as well as the dependencies between them. In the context of *sequence prediction* for text[8], the research goal is to identify the labels of the next text snippets (tokens, phrases, etc.) given the labels of the observed text. Supervised models often assume the Markovian properties of the data, i.e. the current text snippet is dependent on up to k previous ones. Sequence prediction has a wide variety of applications: Named entity recognition, gene prediction, image classification, etc. While there are numerous models for sequence prediction such as Hidden Markov Models (HMM) [RJ86], Maximum-Entropy Markov Model [MFP00], Dynamic Bayesian Network (DBN) [Mur02], in this chapter, we focus on two most recent models that are the current states of the art:

[8] In general, there are four main problems of sequence learning: Sequence prediction, sequence generation, sequence recognition, and sequence decision making (Source: https://en.wikipedia.org/wiki/Sequence_learning)

Conditional Random Fields

Conditional Random Field (CRF) [LMP+01] is a family of undirected graphical models that try to exploit the dependencies of data in a systematic fashion. Each undirected graphical model considers both observations and desired outcomes as random variables, and factorizes the joint distribution to different factors, each of which involve only a small number of variables. Conceptually, each factor is a non-negative function defined on a subset of random variables, and also called *local function* or *compatibility function*. For sequence models, a popular local function is defined by three parameters: An observation (feature) of the current data item, the hidden variable (e.g. label) of the data item, and the hidden variable of the previous item. This model, called *linear-chain CRF*, corresponds to the following joint distribution:

$$P(\mathbf{y}|\mathbf{x}) \propto \prod_{i=1}^{n} \exp\{\sum_{k=1}^{K} \theta_k f_k(y_{i-1}, y_i, \mathbf{x}_i)\} \tag{2.1}$$

where y_i is the hidden variable of the item x_i, and f_k are factors or local functions depending on the features and the presence of labels. The parameters θ_k's can be trained by *maximum likelihood estimation*, i.e. the parameters are chosen to optimize the probability to observe the training data, assuming each training item is independent. In practice, because the number of parameters is very high, the optimization is often performed via numerical methods such as stochastic gradient descent (SGD) [PJ92] or LBFG [LN89].

Recurrent Neural Networks

The problem with CRFs, as also the case with general probabilistic graphical models, are the computationally expensive training and learning processes, and the difficulties in designing the proper set of features. To reduce the need for directly hand-crafting of high-order and advanced features, recent research efforts have shifted to the deep neural network architectures, where such features are implicitly defined through the hidden layers. In the context of sequence labelling, the suitable architecture is recurrent neural networks (RNN). In RNN, outputs are not just dependent on the input, but also on the previous computation, thereby to create an "internal memory". One of the successful RNN architecture is *Long Short-term Memory* (LSTM), which was proposed by Hochreiter et al. [HS97]. Inspired by the human remembering and forgetting scheme, the author carefully re-designed each hidden cell in the RNN to either accept previous results or discard them. In essence, the hidden cells consist of a forget gate layer (a cell using sigmoid activation function σ), an update layer (two cells using functions σ and $\tanh$), and the output layer (using σ function). The forget gate layer allows the previous result (h_{t-1}) to be completely let through or ignored depending on the current input (x_t). The update layer modifies the input x_t (using $\tanh$) and concatenates it with previous result h_{t-1} ("+" symbol). Finally, the output layer allows x_t to contribute directly to the next result h_t. The weights of forget, update and output layers are determined by the training sequences and the choice of the loss functions. The training of LSTM is done

via back-propagation method, as other standard neural network models. LSTM is currently among the most successful models to handle sequential data. LSTM can also be combined with CRF, for example, in an architecture that encodes a sequence into continuous vectors by LSTM and uses CRF to give the prediction [HXY15].

2.3.2 Graph Learning

Another relevant supervised learning research have emerged recently and attracted much of the attention is *graph learning*, in which documents are not independent but interconnected via different relationships. In graph learning, documents are modelled as nodes in a (heterogeneous) graphs, and the aim is to predict the structure of one graph given the information of other training graphs. Graph learning can be applied to various applications, including link prediction [BL11], document ranking [GLW+11], summarization [DGC11], clustering [AVL10], etc. In most of applications, the graphs are often large enough to make any hard optimization infeasible. Thus, many approximation approaches are employed, among which, the most successful approaches often employ, directly or indirectly, sampling methods using random walks [Pea05].

Random Walk-based Models

Random walk [Pea05] is a powerful mathematical object that finds success in a vast amount of applications. In the context of graph learning, random walk is an efficient way to model the dynamics of information passed through the graph by assuming a stochastic process in which one message is passed from nodes to nodes, either by following the edges, or by jumping to a random node in the graph (called *damping*) or to itself (called *restarting*). When certain requirements of the graph are met, after a finite number of steps, all nodes will reach stationary states, when the probability in which the message stays in each individual node can be interpreted to serve other purposes. In supervised learning settings, the random process can be governed by the training data, and with respect to a some objective function. Below are some popular algorithms that employ this approach.

Supervised Random Walk. (SRW) Backstrom et al. [BL11] were the first to integrate random walk in classification problems. They proposed the SRW algorithm to predict whether two people in a social network will establish a social tie (e.g. friendship). In their setting, the graph nodes are simple identifiers, but the edges are represented by a vector of features to encode the association aspects. The goal is to learn the weights to aggregate edge features to build the transition matrix, on which the random walk with restart will give nodes with established ties higher scores than the others.

Semi-supervised PageRank. (SSP) Gao et al. [GLW+11] extended the ideas from [BL11] to model the graph where both nodes and edges are represented by feature vectors. The aim is to rank the nodes in the graph using pairwise training data (see 2.2.3). When the features are dependent on the query, SSP can be used as a learning to rank framework as well. Zhukovskiy et al. [ZGS14] improve this idea further by exploiting the graph structure both in

feature construction and learning steps, increasing the performance of the L2R significantly.

2.4 Relevant Background on Human Memory

Since some parts of this thesis are inspired by the foundations in human memory, in this Seciton we review some relevant literature in this field. Specifically, we briefly discuss the principles of individual and organizational remembering and forgetting for different types of memory. We also review the ongoing research in the human memory complementing in sociology and knowledge management research.

2.4.1 Human Memory and Forgetting

Human memory is a distinct and crucial part of the humans, for them to function in everyday life. The widely accepted perception of human memory is that it is the set of all ingredients that are associated with preservation and retrieval of information about public and personal events. However, psychologists define a much broader definition of human memory to include different types of memory [LTB+13, Tul86, Rea00]: The memory associated with the acquisition, preservation and retrieval of knowledge and skills (*semantic memory*), the memory associated with events and experiences across a person's lifetime (*episodic memory*), the memory associated with the carry out intended actions (*prospective memory*), and the memory applied to the temporary storage and moment-to-moment updating of information required for a focus on the current task (*working memory*). Among them, episodic and semantic memory attract high attention with a significant number of studies, due to its wide influence in society, and to its potential in human learning support. It also has direct implications in user behaviour analysis and data mining, which inspire some studies in this thesis.

2.4.2 Decay and Interference Theories

Causes and effects of human remembering and forgetting in semantic and episodic memories have long been of high interest in psychology research. One of the first systematic studies of human episodic forgetting was conducted by Ebbinghaus in 1885 [Ebb13], which led to the *decay theory*. Ebbinghaus' research stated that the main cause of memory loss is because of the decay in memory strength when there is no attempt to retain it, illustrated in the famous concept of the *forgetting curve*. In his survey, Schacter argues that the forgetting curve is the special form of transcience-caused failure, which is one of the seven kinds of memory loss [Sch99]. Over a century, there are several mathematical functions have been proposed to model the forgetting curve [Whi01]. In addition to the original Ebbinghaus function, there are also logarithmic, power, exponential, and hyperbolic functions shown to best fit forgetting curves in different scenarios and data sets [RW96, LS85, RHW99].

Forgetting as a decay is also shown to consist self-similar phases: Recall, recognition, and reproduction, each follows similar curves [WS54].

Also decay theory introduces important insights into how memory and forgetting works, it has its limitations by excluding the effects of contexts and other schemes. For example, in his experiments, Ebbinghaus examined with nonsense materials (three letters syllabus with no established schema such as BAZ, FUB, etc.) [Ebb13], which is not a realistic scenario. Other schools of thought called *interference theory* explain the remembering and forgetting mechanism as the interactions of materials, that sometimes bring negative influence into the human ability of recalling information. More specifically, psychologists hypothesized that the main causes of forgetting are either the interference of previously stored details about similar events, or of the stored details of similar subsequent events [RL14]. The former is the subject of *proactive interference* study [Und57], and the later is the subject of *retroactive interference* study [MM31]. Retroactive interference can also explain memory loss in learning language [IM01].

It is notesworthy that decay and interference theories are not contradicting, but complementing each other. Recent studies show that even in the presence of context or interferring events, Ebbinghaus-liked forgetting functions can still be applied with some modifications. In their seminal study, McKenna and Glendon [MG85] showed that contexts significantly decelerate the forgetting speed; however, in one period between two consecutive interference, the shapes of forgetting functions are remarkably similar, and still follow decay theory. This effect, sometimes called memory bumping or forgetting with context support, is confirmed in other studies [Bah84, CCS92].

2.4.3 Complementing Human Memory

With the advent of digital devices and especially the Internet, retrieving previously stored information becomes much easier than before. For example, one does not need to memorize phone numbers of his contacts, for today's cellphones are all able to keep such information with people name instead of raw numbers. This has great impacts on human remembering habits and even cognitive patterns. Experiments by Sparrow et al. [SLW11] revealed what is called "Google effect" in human memory, in which when humans are aware of the easy future access to the information, they tend not to recall the information itself, but where to access it. Other studies such as [Pen09, WJA14] also confirm this effect, and even suggests that some online encyclopedic resources such as Wikipedia are being increasingly used as a global memory place. This change in reconstruction mechanism in human memory in the digital age provide a good foundation for designing new methods to improve the human retrieval process.

Information Re-finding. In information retrieval, an area that has interesting links with human memory concerns seeking information that has been previously seen and remembered. This activity indeed dominates in Web search, accounting for 40% of search queries [TAJP07]. Information re-finding is different from ad-hoc search in that people retain partially the im-

ages of materials in memory, and rely more on the clues at hand [Tee06]. The complementing of human memory such as Google effects suggests that good interfaces for re-finding information should be able to identify good clues for memory re-construction [MRMV08, dJBR+12, BJD04]. In desktop search, such clues are the main sources of relevance provenanec [JLW+10, SSGN07], and can be mined from document relationships and similarity [MSHD11, SG05]. Recent work suggests that this relation-based memory assistance is also helpful in Web revisitation [KPHN11].

3 Temporal Data Enrichment

In this chapter, we study the problem of enriching data collections through various schemes. The result of this chapter builds the basics for the Chapters 4 and 5 of this dissertation, since it provides meta-data of the texts that are useful for further mining tasks.

3.1 Introduction

Temporal text (text consisting temporal contents) are ubiquitous in our everyday life. They can be news, social media, scientific publications, even encyclopedia with temporal facts, etc. Compared to normal text, temporal text contains information that should be comprehended with the concept of time in mind, whether it is the past, present or the future. Examples include news storylines reporting an ongoing topic, scientific papers analyzing a historical phenomenon, or some reports predicting the future political trends. Nowadays, temporal data, especially social media, are generated every day at an unprecedentedly rapid speed, with millions of documents published every second. It is therefore an acute task to assist human to quicker filter and digest the information. One such popular mechanism is data enrichment, which aims to extract meta-data from the contents and use them to ease the text processing, either at document or collection level.

In this chapter, we will discuss two specific tasks to assist the automated document processing: Annotation and indexing. The former is concerned with extracting meta-data at document levels: It identifies relevant entities to describe the main topic in the temporal collection. The latter – temporal collection indexing – is concerned with extracting meta-data at the collection level: It establishes the inverted index to easily filter documents in large collection by the means of full-text search. For annotation, we study the two big datasets with temporal text: Twitter and Wikipedia Revision History. We focus more on the Twitter dataset, because of many unique challenges the dataset imposes for the annotation task. Specifically, we address the problem of *semantic annotation of trending topics* in Twitter, because it exposes many interesting and relevant challenges: Trending topics in Twitter are

both interesting and difficult to analyze because of their terse and noisy nature, and the data is enormous during the trending time period. For indexing, we choose to index the pages describing the entities from Wikipedia Revision History. This choice helps us look at both the temporal and the complex structure of documents (Wikipedia revisions), and because our work heavily rely on Wikipedia-derived knowledge bases, the index is valuable for our work, as described in section 3.2.5 in this chapter, as well as in chapter 4.

The following of this chapter is organized as follows. Section 3.2 discusses the annotation in Twitter dataset, focusing on trending topics. Section 3.3 discusses the annotation and indexing in Wikipedia revision history dataset, and introduces our developed framework called Hedera. Finally, section 3.4 concludes the chapter and discusses the open research directions.

3.2 Annotations of Trending Topics in Microblogs

With the proliferation of microblogging and its wide influence on how information is shared and digested, the studying of microblog sites has gained interest in recent NLP research. Information in Twitter is rarely digested in isolation, but rather in a collective manner, with the adoption of special mechanisms such as hashtags. When put together, the unprecedented adoption of a hashtag in a large number of tweets within a short time period can lead to bursts that often reflect trending social attention. Understanding the meaning of trending hashtags offers a valuable opportunity for various applications and studies, such as viral marketing, social behavior analysis, recommendation, etc. Unfortunately, the task of hashtag annotation has been largely unexplored so far.

In this section, we study the problem of annotating trending hashtags on Twitter by entities derived from Wikipedia. Instead of establishing a static semantic connection between hashtags and entities, we are interested in *dynamically* linking the hashtags to entities that are closest to the underlying events during trending time periods of the hashtags. For instance, while '#sochi' refers to a city in Russia, during February 2014, the hashtag was used to report the *2014 Winter Olympics* (cf. Figure 3.1); therefore, it should be linked more to Wikipedia pages related to the event than to the location.

Compared to traditional domains of text (e.g., news articles), annotating hashtags by entities poses additional challenges. Hashtags surface forms are very ad-hoc, as they are chosen not in favor of the text quality, but by the dynamics in attention of the large crowd. In addition, the rapid evolution of the semantics of hashtags (e.g., in the case of '#sochi') makes them more ambiguous. Furthermore, a hashtag can encode multiple topics at one time period. For example, in March 2014, '#oscar' refers to the *86th Academy Awards*, but at the same time also to the *Trial of Oscar Pistorius*. Sometimes, it is difficult even for humans to understand a trending hashtag without knowledge about what is happening with the entities in the real world.

During this PhD study, we proposed a novel solution to these challenges by leveraging temporal knowledge about entity dynamics derived from Wikipedia. We hypothesize that a

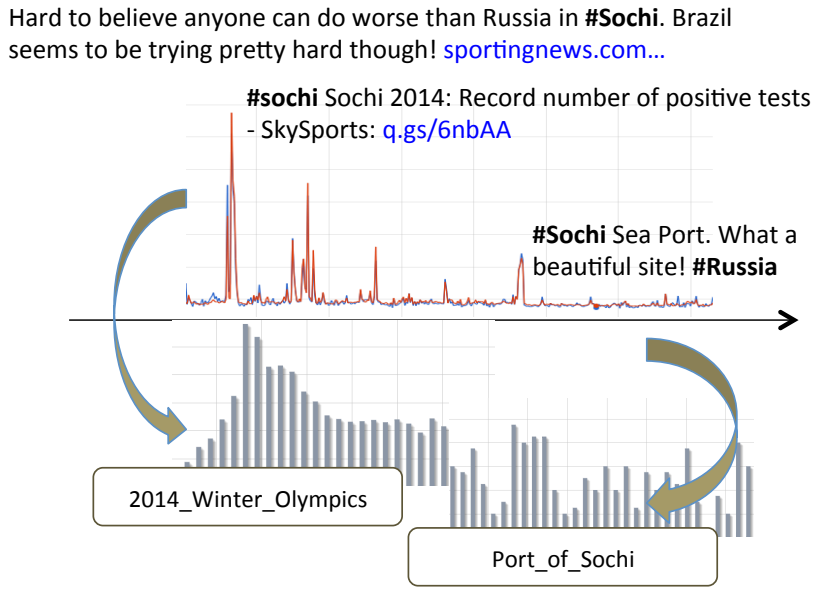


Figure 3.1: Example of trending hashtag annotation. During the *2014 Winter Olympics*, the hashtag '#sochi' had a different meaning.

trending hashtag indicates an increase in public attention to certain entities, and this can also be observed on Wikipedia. As shown in Figure 3.1, we can identify *2014 Winter Olympics* as a prominent entity for '#sochi' during February 2014, by observing the change of user attention to the entity, for instance via the page view statistics of Wikipedia articles.

In literature, few works have paid attention to the semantics of hashtags, i.e., to the underlying topics conveyed in the corresponding tweets. Recently, [BBV15] attempt to segment a hashtag and link each of its tokens to a Wikipedia page. However, the authors only aim to retrieve entities directly mentioned within a hashtag, which are very few in practice. The external information derived from the tweets is largely ignored. In contrast, we exploit both context information from the microblog and Wikipedia resources. We exploit both Wikipedia edits and page views for annotation. We also propose a novel learning method, inspired by the information spreading nature of social media such as Twitter, to suggest the optimal annotations without the need for human labeling. To our knowledge, we are the first to combine the content of the Wikipedia edit history and the magnitude of page views to handle trending topics on Twitter.

3.2.1 Problem Statement

Preliminaries Following the notion of the disambiguated entities in section 2.1.3, we refer to an *entity* (denoted by e) as any object described by a Wikipedia article (ignoring disambiguation, lists, and redirect pages). This includes also articles about events, as shown in above example with Sochi. The number of times an entity's article has been requested is called the *entity view count*. The text content of the article is denoted by $C(e)$. In this work,

we choose to study hashtags at the daily level, i.e., from the timestamps of tweets we only consider their creation day.

A hashtag is called *trending* at a time point if the number of tweets adopting it is significantly higher than on other days. This can be measured in different ways [LAP+09, LGRC12]. Each trending hashtag has one or multiple *burst time periods*, surrounding the trending time points, where the users' interest in the underlying topic remains stronger than in other periods. We denote with $T(h)$ (or T for short) one hashtag burst time period, and with $D_T(h)$ the set of tweets containing the hashtag h created during T.

Task Definition Given a trending hashtag h and the burst time period T of h, identify the top-k most prominent entities for h.

It is worth noting that not all trending hashtags can be mapped to Wikipedia entities, as the coverage of topics in Wikipedia is much lower than on Twitter. This is also a limitation of systems relying on Wikipedia such as entity disambiguation, which can only disambiguate popular entities and not the ones in the long tail. In this study, we focus on the precision and the popular trending hashtag, and leave the improvement of recall as future work.

3.2.2 Methodology Overview

Overview We approach the task in three steps. The first step is to identify all entity candidates by checking surface forms of the constituent tweets of the hashtag. In the second step, we compute different similarities between each candidate and the hashtag, based on different types of contexts, which are derived from either side (Wikipedia or Twitter). Finally, we learn a unified ranking function for each (hashtag, entity) pair and choose the top-k entities with the highest scores. The ranking function is learned through an unsupervised model and needs no human-defined labels.

Candidate Identification

In the first step, we need to identify the set of entity candidates to annotate the topic. The most obvious resource to identify candidate entities for a hashtag is via its tweets. We follow common approaches that use a lexicon to match each textual phrase in a tweet to a potential entity set [SWLW13, FC14]. Our lexicon is constructed from Wikipedia page titles, hyperlink anchors, redirects, and disambiguation pages, which are mapped to the corresponding entities. As for the tweet phrases, we extract all n-grams ($n \leq 5$) from the input tweets within T. We apply the longest-match heuristic [MWDR12]: We start with the longest n-grams and stop as soon as the entity set is found, otherwise we continue with the constituent, smaller n-grams.

Candidate Set Expansion. While lexicon-based linking works well for single tweets, applying it on the hashtag level has subtle implications. Processing a huge amount of text, especially during a hashtag trending time period, incurs expensive computational costs.

Therefore, to maintain a feasible computation, we apply entity linking only on a random sample of the complete tweet set. Then, for each candidate entity e, we include all entities whose Wikipedia article is linked with the article of e by an outgoing or incoming link.

Similarities

To rank the entity by prominence, we measure the similarity between each candidate entity and the hashtag. We evaluate three types of similarities:

Mention Similarity. This measure relies on the explicit mentions of entities in tweets. It assumes that entities directly linked from more prominent anchors are more relevant to the hashtag. It is estimated using both statistics from Wikipedia and tweet phrases, and turns out to be surprisingly effective in practice.

Context Similarity. For entities that are not directly linked to mentions (the mention similarity is zero) we exploit external resources instead. Their prominence is perceived by users via external sources, such as web pages linked from tweets, or entities' home pages. By exploiting the content of entities from these external sources, we can complement the explicit similarity metrics based on mentions.

Temporal Similarity. The prior two metrics rely on the textual representation and are degraded by the linguistic difference between the two platforms. To overcome this drawback, we incorporate the temporal dynamics of hashtags and entities, which serve as a proxy to the change of user interests towards the underlying topics [CN10]. We employ the correlation between the times series of hashtag adoption and the entity view as the third measure.

Ranking Entity Prominence

While each similarity measure captures one evidence of the entity prominence, we need to unify all scores to obtain a global ranking function. In this work, we propose to combine the individual similarities using a linear function:

$$f(e,h) = \alpha f_m(e,h) + \beta f_c(e,h) + \gamma f_t(e,h) \tag{3.1}$$

where α, β, γ are model weights and f_m, f_c, f_t are the similarity measures based on mentions, context, and temporal information, respectively, between the entity e and the hashtag h. We further constrain that $\alpha + \beta + \gamma = 1$, so that the ranking scores of entities are normalized between 0 and 1, and that our learning algorithm is more tractable. The algorithm, which automatically learns the parameters without the need of human-labeled data, is explained in detail in section 3.2.4.

3.2.3 Similarity Measures

We now give details on how to compute the similarity measures discussed above.

Link-based Mention Similarity

The similarity of an entity with one individual mention in a tweet can be interpreted as the probabilistic prior in mapping the mention to the entity via the lexicon. One common way to estimate the entity prior exploits the anchor statistics from Wikipedia links, and has been proven to work well in different domains of text. We follow this approach and define $LP(e|m) = \frac{|l_m(e)|}{\sum_{m'}|l_{m'}(e)|}$ as the link prior of the entity e given a mention m, where $l_m(e)$ is the set of links with anchor m that point to e. The mention similarity f_m is measured as the aggregation of link priors of the entity e over all mentions in all tweets with the hashtag h:

$$f_m(e,h) = \sum_m (LP(e|m) \cdot q(m)) \tag{3.2}$$

where $q(m)$ is the frequency of the mention m over all tweets of h.

Context Similarity

To compute f_c, we first construct the contexts for hashtags and entities. The context of a hashtag is built by extracting all words from its tweets. We tokenize and parse the tweets' part-of-speech tags [OOD+13], and remove words of Twitter-specific tags (e.g., @-mentions, URLs, emoticons, etc.). Hashtags are normalized using the word breaking method by [WTH11].

The textual context of an entity is extracted from its Wikipedia article. One subtle aspect is that the Wikipedia articles are not created at once, but are incrementally updated over time in accordance with changing information about entities. Texts added in the same time period of a trending hashtag contribute more to the context similarity between the entity and the hashtag. Based on this observation, we use the Wikipedia revision history – an archive of all revisions of Wikipedia articles – to calculate the entity context. We collect the revisions of articles during the time period T, plus one day to acknowledge possible time lags. We compute the difference between two consecutive revisions, and extract only the added text snippets. These snippets are accumulated to form the *temporal context* of an entity e during T, denoted by $C_T(e)$. The distribution of a word w for the entity e is estimated by a mixture between the probability of generating w from the temporal context and from the general context $C(e)$ of the entity:

$$\hat{P}(w|e) = \lambda \hat{P}(w|M_{C_T(e)}) + (1-\lambda)\hat{P}(w|M_{C(e)})$$

where $M_{C_T(e)}$ and $M_{C(e)}$ are the language models of e based on $C_T(e)$ and $C(e)$, respectively. The probability $\hat{P}(w|M_{C(e)})$ corresponds to the background model, while $\hat{P}(w|M_{C_T(e)})$ corresponds to the foreground model in traditional language modeling settings. Here we use a simple maximum likelihood estimation to estimate these probabilities: $\hat{P}(w|M_{C(e)}) = \frac{tf_{w,c}}{|C(e)|}$ and $\hat{P}(w|M_{C_T(e)}) = \frac{tf_{w,c_T}}{|C_T(e)|}$, where $tf_{w,c}$ and tf_{w,c_T} are the term frequencies of w in the two text sources of $C(e)$ and $C_T(e)$, respectively, and $|C(e)|$ and $|C_T(e)|$ are the lengths of the two texts, respectively. We use the same estimation for tweets: $\hat{P(w|h)} = \frac{tf_{w,D(h)}}{|D(h)|}$,

where $D(h)$ is the concatenated text of all tweets of h in T, after normalization. We use the Kullback-Leibler divergence to compare the distributions over all words appearing both in the Wikipedia contexts and the tweets:

$$KL(e \parallel h) = \sum_{w} \hat{P}(w|e) \cdot \frac{\hat{P}(w|e)}{\hat{P}(w|h)}$$
$$f_c(e,h) = e^{-KL(e \parallel h)} \tag{3.3}$$

Temporal Similarity

The third similarity, f_t, is computed using temporal signals from both sources – Twitter and Wikipedia. For the hashtags, we build the time series based on the volume of tweets adopting the hashtag h on each day in T: $TS_h = [n_1, n_2, \ldots, n_{|T|}]$. Similarly, for the entities, we build the time series of view counts for the entity e in T: $TS_e = [v_1, v_2, \ldots, v_{|T|}]$. A time series similarity metric is then used to compute f_t. Some metrics can be used, however most of them suffer from the time lag and scaling discrepancy, or incur expensive computational costs [RAGM11]. In this work, we employ a simple yet effective metric that is agnostic to the scaling and time lag of time series [YL11]. It measures the distance between two time series by finding optimal shifting and scaling parameters to match the shape of two time series:

$$f_t(e,h) = \min_{q,\delta} \frac{\|TS_h - \delta d_q(TS_e)\|}{\|TS_h\|} \tag{3.4}$$

where $d_q(TS_e)$ is the time series derived from TS_e by shifting q time units, and $\|\cdot\|$ is the L_2 norm. It has been proven that Equation 3.4 has a closed-form solution for δ given fixed q, thus we can design an efficient gradient-based optimization algorithm to compute f_t [YL11].

3.2.4 Annotation Using Influence Learning

Ranking Framework

To unify the individual similarities into one global metric (Equation 3.1), we need a guiding premise of how the prominence of an entity to a hashtag can be reflected or observed. Such a premise can be inferred through manual assessment [MWDR12, GCK13], but it requires human-labeled data and is biased from evaluator to evaluator. Other heuristics assume that entities close to the main topic of a text are also coherent to each other [RRDA11, LLW+13]. Based on this, state-of-the-art methods in traditional disambiguation attempt to find prominent entities by optimizing the overall coherence of the entities' semantic relatedness. However, this coherence does not hold for topics in hashtags: Entities reported in a big topic such as the Olympics vary greatly with different sub-events. They are not always coherent to each other, as they are largely dependent on the users' diverse attention to each sub-event. This heterogeneity of hashtags calls for a different premise, abandoning the idea of coherence.

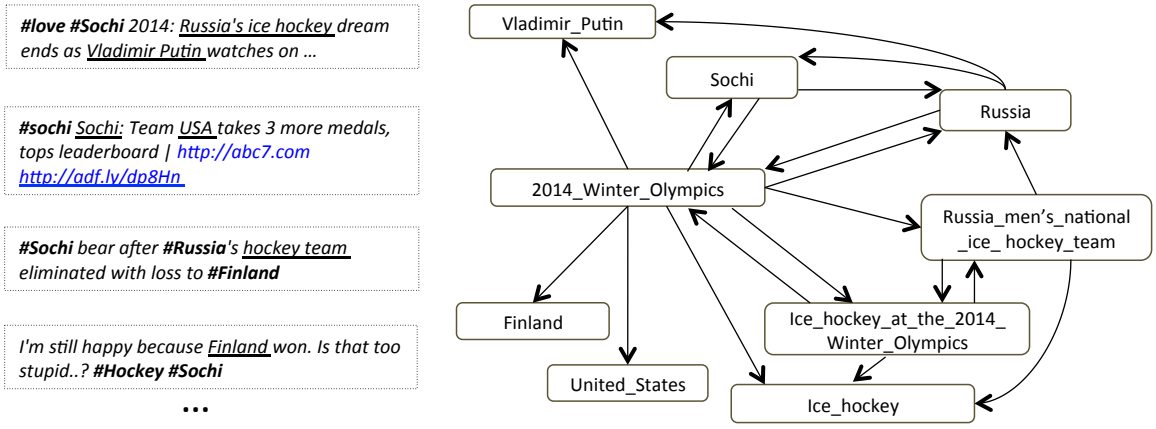


Figure 3.2: Excerpt of tweets about ice hockey results in the *2014 Winter Olympics* (left), and the observed linking process between time-aligned revisions of candidate Wikipedia entities (right). Links come more from prominent entities to marginal ones to provide background, or more context for the topics. Thus, starting from prominent entities, we can reach more candidate entities

Influence Maximization (IM) We propose a new approach to find entities for a hashtag. We use an observed behavioral pattern in creating Wikipedia pages for guiding our approach to entity prominence: Wikipedia articles of entities that are prominent for a topic are quickly created or updated,[1] and subsequently enriched with links to related entities. This linking process signals the dynamics of editor attention and exposure to the event [KGC11]. We argue that the process does not, or to a much lesser degree, happen to more marginal entities or to very general entities. As illustrated in Figure 3.2, the entities closer to the 2014 Olympics get more updates in the revisions of their Wikipedia articles, with subsequent links pointing to articles of more distant entities. The direction of the links influences the shifting attention of users [KGC11] as they follow the structure of articles in Wikipedia.

We assume that, similar to Wikipedia, the entity prominence also influences how users are exposed and spread the hashtag on Twitter. In particular, the initial spreading of a trending hashtag involves more entities in the focus of the topic. Subsequent exposure and spreading of the hashtag then include other related entities (e.g., discussing background or providing context), driven by interests in different parts of the topic. Based on this assumption, we propose to measure the entity prominence as its potential in *maximizing the information spreading* within all entities present in the tweets of the hashtag. In other words, the problem of ranking the most prominent entities becomes identifying the set of entities that lead to the largest number of entities in the candidate set. This problem is known in social network research as *influence maximization* [KKT03].

Iterative Influence-Prominence Learning (IPL) IM itself is an NP-hard problem [KKT03]. We propose an approximation framework to *jointly* learn the influence scores of the entity

[1] [OPM+12] suggested a time lag of 3 hours.

and the entity prominence together. The framework (called IPL) contains several iterations, each consisting of two steps:

- Pick up a model for entity prominence and use it to compute the entity influence score.
- Based on the influence scores update the entity prominence.

In the sequel we detail our learning framework.

Entity Graph

Influence Graph To compute the entity influence scores, we first construct the entity *influence graph* as follows. For each hashtag h, we construct a directed graph $G_h = (E_h, V_h)$, where the nodes $E_h \subseteq E$ consist of all candidate entities (cf. section 3.2.2), and an edge $(e_i, e_j) \in V_h$ indicates that there is a link from e_j's Wikipedia article to e_i's. Note that edges of the influence graph are inversed in direction to links in Wikipedia, as such a link gives an "influence endorsement" from the destination to the source entity.

Entity Relatedness In this work, we assume that an entity endorses its influence score more to its related entities than to others. We use a popular entity relatedness measure [MW08]:

$$MW(e_1, e_2) = 1 - \frac{\log(\max(|I_1|,|I_2|) - \log(|I_1 \cap I_2|))}{\log(|E|) - \log(\min(|I_1|,|I_2|))}$$

where I_1 and I_2 are sets of entities having links to e_1 and e_2, respectively, and E is the set of all entities in Wikipedia. The influence transition from e_i to e_j is defined as:

$$b_{i,j} = \frac{MW(e_i, e_j)}{\sum_{(e_i,e_k)\in V} MW(e_i, e_k)} \tag{3.5}$$

Influence Score Let $\mathbf{r}_h$ be the influence score vector of entities in G_h. We can estimate $\mathbf{r}_h$ efficiently using random walk models, similar to [LXC+14]:

$$\mathbf{r_h} \coloneqq \tau \mathbf{B} \mathbf{r_h} + (1 - \tau)\mathbf{s_h} \tag{3.6}$$

where $\mathbf{B}$ is the influence transition matrix, $\mathbf{s_h}$ are the initial influence scores that are based on the entity prominence model (Step 1 of IPL), and τ is the damping factor.

Learning Algorithm

Now we detail the IPL algorithm. The objective is to learn the model $\omega = (\alpha, \beta, \gamma)$ of the global function (Equation 3.1). The general idea is that we find an optimal ω such that the average error with respect to the top influencing entities is minimized

$$\omega = \arg\min \sum_{E(h,k)} L(f(e,h), r(e,h))$$

Algorithm 1: Entity Influence-Prominence Learning

Input : $h, T, D_T(h), \mathbf{B}, k$, learning rate μ, threshold ϵ
Output: ω, top-k most prominent entities.

Initialize: $\omega := \omega^{(0)}$
Calculate $\mathbf{f}_m, \mathbf{f}_c, \mathbf{f}_t, \mathbf{f}_\omega := \mathbf{f}_{\omega^{(0)}}$ using Eqs. 3.1, 3.2, 3.3, 3.4
while *true* **do**
  $\hat{\mathbf{f}}_\omega :=$ normalize $\mathbf{f}_\omega$
  Set $\mathbf{s_h} := \hat{\mathbf{f}}_\omega$, calculate $\mathbf{r_h}$ using Eq. 3.6
  Sort $\mathbf{r_h}$, get the top-k entities $E(h,k)$
  if $\sum_{e \in E(h,k)} L(f(e,h), r(e,h)) < \epsilon$ **then**
    Stop
  $\omega := \omega - \mu \sum_{e \in E(h,k)} \nabla L(f(e,h), r(e,h))$
return $\omega, E(h,k)$

where $r(e,h)$ is the influence score of e and h, $E(h,k)$ is the set of top-k entities with highest $r(e,h)$, and L is the squared error loss function, $L(x,y) = \frac{(x-y)^2}{2}$.

The main steps are depicted in Algorithm 1. We start with an initial guess for ω, and compute the similarities for the candidate entities. Here $\mathbf{f}_m$, $\mathbf{f}_c$, $\mathbf{f}_t$, and $\mathbf{f}_\omega$ represent the similarity score vectors. We use matrix multiplication to calculate the similarities efficiently. In each iteration, we first normalize $\mathbf{f}_\omega$ such that the entity scores sum up to 1. A random walk is performed to calculate the influence score $\mathbf{r_h}$. Then we update ω using a batch gradient descent method on the top-k influencer entities. To derive the gradient of the loss function L, we first remark that our random walk Equation 3.6 is similar to context-sensitive PageRank [Hav02]. Using the linearity property [FRCS05], we can express $r(e,h)$ as the linear function of influence scores obtained by initializing with the individual similarities f_m, f_c, and f_t instead of f_ω. The derivative thus can be written as:

$$\nabla L(f(e,h), r(e,h)) = \alpha(r_m(e,h) - f_m(e,h)) + \beta(r_c(e,h) - f_c(e,h)) + \gamma(r_t(e,h) - f_t(e,h))$$

where $r_m(e,h), r_c(e,h), r_t(e,h)$ are the components of the three vector solutions of Equation 3.6, each having $\mathbf{s_h}$ replaced by $\mathbf{f}_m$, $\mathbf{f}_c$, $\mathbf{f}_t$ respectively.

Since both $\mathbf{B}$ and $\hat{\mathbf{f}}_\omega$ are normalized such that their column sums are equal to 1, Equation 3.6 is convergent [Hav02]. Also, as discussed above, $\mathbf{r_h}$ is a linear combination of factors that are independent of ω, hence L is a convex function, and the batch gradient descent is also guaranteed to converge. In practice, we can utilize several indexing techniques to significantly speed up the similarity and influence scores calculation.

Total Tweets	500,551,041
Trending Hashtags	2,444
Test Hashtags	30
Test Tweets	352,394
Distinct Mentions	145,941
Test (Entity, Hashtag) pairs	6,965
Candidates per Hashtag (avg.)	50
Extended Candidates (avg.)	182

Table 3.1: Statistics of the dataset.

3.2.5 Experiments

Setup

Dataset There is no standard benchmark for our problem, since available datasets on microblog annotation (such as the Microposts challenge [BRV+14]) often skip global information, such we cannot infer the social statistics of hashtags. Therefore, we created our own dataset. We used the Twitter API to collect from the public stream a sample of $500,551,041$ tweets from January to April 2014. We removed hashtags that were adopted by less than 500 users, having no letters, or having characters repeated more than 4 times (e.g., '#oooommmgg'). We identified trending hashtags by computing the daily time series of hashtag tweet counts, and removing those of which the time series' variance score is less than 900. To identify the hashtag burst time period T, we compute the *outlier fraction* [LGRC12] for each hashtag h and day t: $p_t(h) = \frac{|n_t - n_b|}{\max(n_b, n_{\min})}$, where n_t is the number of tweets containing h, n_b is the median value of n_t over all points in a 2-month time window centered on t, and $n_{\min} = 10$ is the threshold to filter low activity hashtags. The hashtag is skipped if its highest outlier fraction score is less than 15. Finally, we define the *burst time period* of a trending hashtag as the time window of size w, centered at day t_0 with the highest $p_{t_0}(h)$.

For the Wikipedia datasets, we process the dump from 3rd May 2014, so as to cover all events in the Twitter dataset. To process the Wikipedia revision history dataset, we have developed Hedera [TN14], a scalable tool to quickly extract different meta-data of Wikipedia based on the Map-Reduce paradigm. The tool is explained in more details in section 3.3. To get the information about Wikipedia page views, we download the Wikipedia page count dataset that stores how many times a Wikipedia article was requested on an hourly level[2] and also use Hedera to process. We extract the information of Wikipedia pages for the four months of our study and use Hedera to accumulate all view counts of redirects to the actual articles.

Sampling From the trending hashtags, we sample 30 distinct hashtags for evaluation. Since our study focuses on trending hashtags that can be mapped to entities in Wikipedia, the

[2] https://dumps.wikimedia.org/other/pagecounts-ez/

	Tagme	Wikiminer	Meij	Kauri	M	C	T	IPL
P@5	0.284	0.253	0.500	0.305	0.453	0.263	0.474	**0.642**
P@15	0.253	0.147	**0.670**	0.319	0.312	0.245	0.378	0.495
MAP	0.148	0.096	0.375	0.162	0.211	0.140	0.291	**0.439**

Table 3.2: Experimental results on the sampled trending hashtags.

sampling must cover a sufficient number of "popular" topics that are reflected in Wikipedia, and at the same time rare topics in the long tail. To do this, we apply several heuristics in the sampling. First, we only consider hashtags where the lexicon-based linking (section 3.2.2) results in at least 20 different entities. Second, we randomly choose hashtags to cover different types of topics (long-running events, breaking events, endogenous hashtags). Instead of inspecting all hashtags in our corpus, we follow [LGRC12] and calculate the fraction of tweets published before, during and after the peak. The hashtags are then clustered in this 3-dimensional vector space. Each cluster suggests a group of hashtags with a distinct semantics [LGRC12]. We then pick up hashtags randomly from each cluster, resulting in 200 hashtags in total. From this rough sample, three inspectors carefully checked the tweets and chose 30 hashtags where the meanings and hashtag types were certain to the knowledge of the inspectors.

Parameter Settings We initialize the similarity weights to $\frac{1}{3}$, the damping factor to $\tau = 0.85$, the weight for the language model to $\lambda = 0.9$, and the learning rate $\mu = 0.003$.

Baseline We compare IPL with other entity annotation methods. Our first group of baselines includes entity linking systems in domains of general text, Wikiminer [MW08], and short text, Tagme [FS12]. For each method, we use the default parameter settings, apply them for the individual tweets, and take the average of the annotation confidence scores as the prominence ranking function. The second group of baselines includes systems specifically designed for microblogs. For the content-based methods, we compare against [MWDR12], which uses a supervised method to rank entities with respect to tweets. We train the model using the same training data as in the original paper. For the graph-based method, we compare against KAURI [SWLW13], a method which uses user interest propagation to optimize the entity linking scores. To tune the parameters, we pick up four hashtags from different clusters, randomly sample 50 tweets for each, and manually annotate the tweets. We also compare three variants of our method, using only local functions for entity ranking (referred to as M, C, and T for *mention*, *context*, and *time*, respectively).

Evaluation In total, there are $6,965$ entity-hashtag pairs returned by all systems. We employ five volunteers to evaluate the pairs in the range from 0 to 2, where 0 means the entity is noisy or obviously unrelated, 2 means the entity is strongly tied to the topic of the hashtag, and 1 means that although the entity and hashtag might share some common contexts, they are not involved in a direct relationship (for instance, the entity is a too general concept

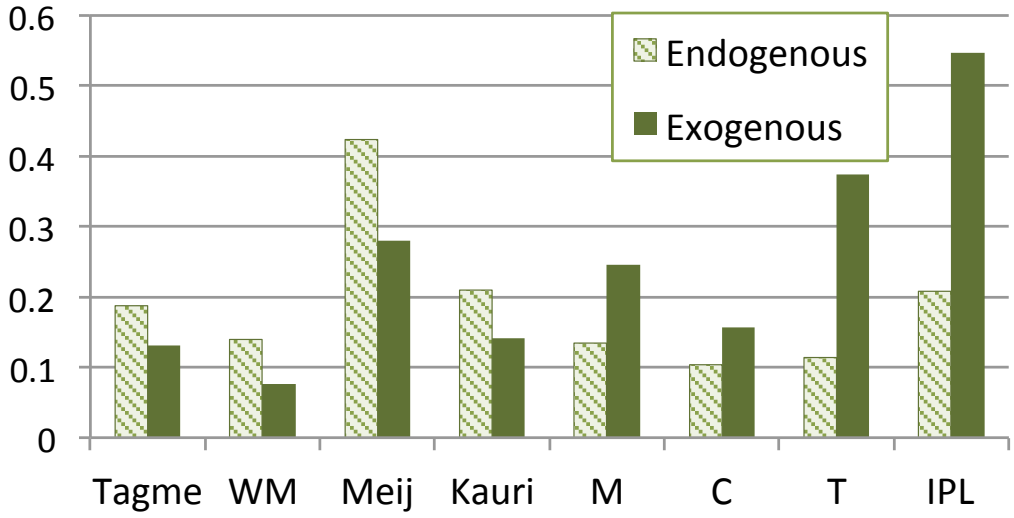


Figure 3.3: Performance of the methods for different types of trending hashtags.

such as *Ice hockey*, as in the case illustrated in Figure 3.2). The annotators were advised to use search engines, the Twitter search box or Wikipedia archives whenever applicable to get more background on the stories. Inter-annotator agreement under Fleiss score is 0.625.

3.2.6 Results and Discussion

Table 3.2 shows the performance comparison of the methods using the standard metrics for a ranking system (precision at 5 and 15 and MAP at 15). In general, all baselines perform worse than reported in the literature, confirming the higher complexity of the hashtag annotation task as compared to traditional tasks. Interestingly enough, using our local similarities already produces better results than Tagme and Wikiminer. The local model f_m significantly outperforms both the baselines in all metrics. Combining the similarities improves the performance even more significantly.[3]

Compared to the baselines, IPL improves the performance by 17-28%. The time similarity achieves the highest result compared to other content-based mention and context similarities. This supports our assumption that lexical matching is not always the best strategy to link entities in tweets. The time series-based metric incurs lower cost than others, yet it produces a considerably good performance. Context similarity based on Wikipedia edits does not yield much improvement. This can be explained in two ways. First, information in Wikipedia is largely biased to popular entities, it fails to capture many entities in the long tail. Second, language models are dependent on direct word representations, which are different between Twitter and Wikipedia. This is another advantage of non-content measures such as f_t.

For the second group of baselines (Kauri and Meij), we also observe the reduction in

[3] All significance tests are done against both Tagme and Wikiminer, with a p-value < 0.01.

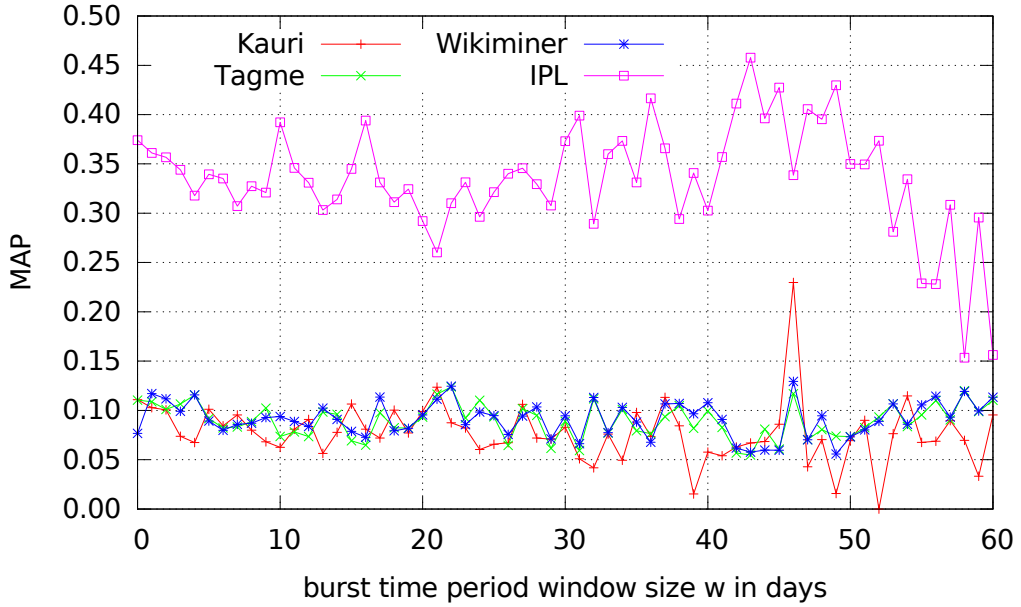


Figure 3.4: IPL compared to other baselines on different sizes of the burst time window T.

precision, especially for Kauri. This is because the method relies on the coherence of user interests within a group of tweets to be able to perform well, which does not hold in the context of hashtags. One astonishing result is that Meij performs better than IPL in terms of P@15. However, it performs worse in terms of MAP and P@5, suggesting that most of the correctly identified entities are ranked lower in the list. This is reasonable, as Meij attempts to optimize (with human supervision effort) the semantic agreement between entities and information found in the tweets, instead of ranking their prominence as in our work. To investigate this case, we re-examined the hashtags and divided them by their semantics, i.e. whether the hashtags are spurious trends of memes inside social media (*endogenous*, e.g., "#stopasian2014"), or whether they reflect external events (*exogenous*, e.g., "#mh370").

The performance of the methods in terms of MAP scores is shown in Figure 3.3. It can be clearly seen that entity linking methods perform well in the endogenous group, but then deteriorate in the exogenous group. The explanation is that for endogenous hashtags, the topical consonance between tweets is very low, thus we can barely annotate further than identifying just individual concepts. In this case, topical annotation is trumped by conceptual annotation. However, whenever the hashtag evolves into a meaningful topic, a deeper annotation method will produce a significant improvement, as seen in Figure 3.3.

Finally, we study the impact of the burst time period on the annotation quality. To this extent, we expand the window size w (cf. section 3.2.5) and examine how different methods perform. The result is depicted in Figure 3.4. It is obvious that within the window of 2 months (where the hashtag time series is constructed and a trending time is identified), our method is stable and always outperforms the baselines by a large margin. Even when the

trending hashtag has been saturated, hence introduced more noise, our method is still able to identify the prominent entities with high quality.

3.3 Hedera: Large-scale Entity Extraction and Indexing for Wikipedia Revisions

In the previous section, we have studied the problem of semantic annotation in one temporal dataset, which is Twitter. In this section, we study another temporal dataset: Wikipedia Revision History, a collection of full snapshots of all Wikipedia pages since the beginning. While section 3.2 discusses how to enrich data at the document level, in this section, we discuss the enrichment of documents in both document and collection level: entities are mapped to their documents to construct an inverted index for retrieval tasks in chapter 4, the incoming and outgoing connections from and to other entities are extracted to build the entity graph, which are useful for methods in chapter 5.

The choice of Wikipedia revision history as the second temporal text collection to enrich has a number of reasons. Over more than one decade of research in NLP, IR and semantic Web communities, Wikipedia has become the leading online resource for various disciplines, e.g. information extraction, resource linking and knowledge management. The exceptional richness of structure and semantic information in Wikipedia, and the public availability of the dataset make them become the major source for building large-scale Knowledge Bases (KBs). Popular KBs such as DBpedia [ABK+07], YAGO [SKW07] are derived from Wikipedia via automated extracting methods. However, such knowledge bases often treat Wikipedia as static, i.e. knowledge rarely increases or changes over time. Facts about entities are extracted from a single snapshot of a Wikipedia corpus, thus any changes can only be reflected in the next version of the knowledge bases (typically extracted fresh from a newer Wikipedia dump). This undesirable quality of KBs has a few negative impacts. For instance, it is unable to capture temporally dynamic relationships that are found among revisions of the encyclopedia (e.g., *participate* together in complex events), which are difficult to detect in one single snapshot. Furthermore, applications relying on obsolete facts might fail to reason under new contexts, because they were not captured in the KBs. In order to complement these temporal aspects, the whole Wikipedia revision history should be well-exploited. However, such longitudinal analytics over enormous size of Wikipedia require huge computation.

As part of this thesis, we develop *Hedera*, a large-scale framework that supports processing, indexing and visualizing Wikipedia revision history. Hedera is an end-to-end system that works directly with the raw dataset, processes them to streaming data, and incrementally indexes and visualizes the information of entities registered in the KBs in a dynamic fashion. In contrast to existing work which handles the dataset in centralized settings [FZG11], Hedera employs the Map-Reduce paradigm to achieve the scalable performance, which is able to transfer raw data of 2.5 year revision history of 1 million entities into full-text index within 12 hours in an 8-node cluster. We open-sourced Hedera to facilitate further

research [4].

3.3.1 Motivation

Here we revisit the existing approaches in processing Wikipedia revision history, their advantages and disadvantages, then state the motivations and requirements of the Hedera framework. Wikipedia revision history data is very big text corpus[5], thus processing them is a non-trivial task. There have been some frameworks developed to address this issue in different ways. We list here the most similar approaches to ours.

JWPL Time Machine. The Time Machine component developed as part of JWPL [FZG11] was among the earliest approach to tackle the large-scale processing problem of Wikipedia Revision History dataset. It relies on relational database technologies and use MySQL as the backend engine to process the dataset. To address the scale issues, JWPL Time Machine performs an aggressive compression: It only keeps the differences between revisions instead of the full snapshots. As a result, to reconstruct information of the Wikipedia page for a given entity, one needs to go back to the first revision of the page, and perform the comparison of Differences on the fly. Such a comparison is expensive [Mye86], and in some cases, unacceptable (For instance, the query to Barack Obama revisions from May 2013 to June 2013 takes 10 minutes). Although the toolkit caches a number of meta-data to facilitate the access, such as edit counts per entity or the contributor lists, it is not flexible enough to satisfy the general computation requirements. In addition, the use of a central MySQL database results in the bottleneck in heavy querying scenarios.

WHAD. Alfonseca et al. [AGDP13] introduced another system called WHAD – Wikipedia History Attributes Data – to process Wikipedia revision history. The system also relies on a central database and performs compression to reduce the size of full revisions. Although employing Map-Reduce style computation to speed the extraction process, WHAD mainly extract information from info boxes, which are the structured tables in the Wikipedia page to summarize the entity facts. However, many entities in the long tail only have texts and will be skipped. The limited schema of WHAD, which consists of only attributes and values of entities, also make it not flexible for various tasks in Wikipedia revision.

Requirements of a new framework. The above drawbacks of existing tools enable us to develop Hedera with a new processing paradigm. We briefly discuss here a number of desiderata influencing the Hedera work flow and architecture:

- *Scalability*: Not only Wikipedia revision history is exceptionally big, it is also growing rapidly, as for all Wikipedia pages, a full snapshot is generated for even a small update. Therefore, the system should be able to scale up with the data growth.

[4]Project documentation and code can be found at: https://github.com/antoine-tran/Hedera

[5]Only the English part of the Wikipedia Revision History corpus dated 2016 November 16 was around 750 Gigabytes of bzip2 compression

- *Flexibility*: Wikipedia research spans a very wide range of topics, from digital humanities to event detection and tracking, from natural language processing to graph mining. Therefore, the system APIs should not be limited to some specific applications, but must be flexible and easy to be extended.

- *Interoperability*: With the rapid development of big data ecosystem, frameworks are evolving constantly, with many languages supported: Java, Scala, Python, R, etc. To be able to interface with other framework, Hedera should support at least common programming languages, and the APIs should be easy to integrated with other frameworks, especially those developed on the Hadoop ecosystem.

Based on these desiderata, we developed Hedera using Hadoop Map-Reduce, and provided interfaces in Java, Pig Latin as well as in Python. Hedera can interact with data from Hadoop HDFS or a NoSQL database such as HBase[6] or Hive[7]. It can also be used as the access APIs in higher-level processing frameworks such as Spark, thanks to its several input readers customized to Wikipedia revision format. In addition, Hedera APIs are designed in a generic way, so that implementations have the freedom to extend to their specific applications. In the following, we describe these aspects of Hedera in more details, in the context of extracting and indexing tasks.

3.3.2 Extracting Entity Information on Wikipedia Revisions

Preprocessing Dataset

We now describe the Hedera architecture and work flow. As shown in Figure 3.5, the core data input of Hedera is a Wikipedia Revision history dump [8]. Hedera currently works with the raw XML dumps, it supports accessing and extracting information directly from compressed files. Hedera makes use of the Hadoop framework. The preprocessor is responsible for re-partitioning the raw files into independent units (a.k.a *InputSplit* in Hadoop) depending on users' need. There are two levels of partitioning: Entity-wise and Document-wise. Entity-wise partitioning guarantees that revisions belonging to the same entity are sent to one computing node, while document-wise partitioning sends content of revisions arbitrarily to any node, and keeps track in each revision the reference to its preceding ones for future usage in the Map-Reduce level. The preprocessor accepts user-defined low-level filters (for instance, only partition articles, or revisions within 2011 and 2012), as well as list of entity identifiers from a knowledge base to limit to. If filtered by the knowledge base, users must provide methods to verify one revision against the map of entities (for instance, using Wikipedia-derived URL of entities). The results are Hadoop file splits, in the XML or JSON formats.

[6] https://hbase.apache.org
[7] https://hive.apache.org/
[8] http://dumps.wikimedia.org

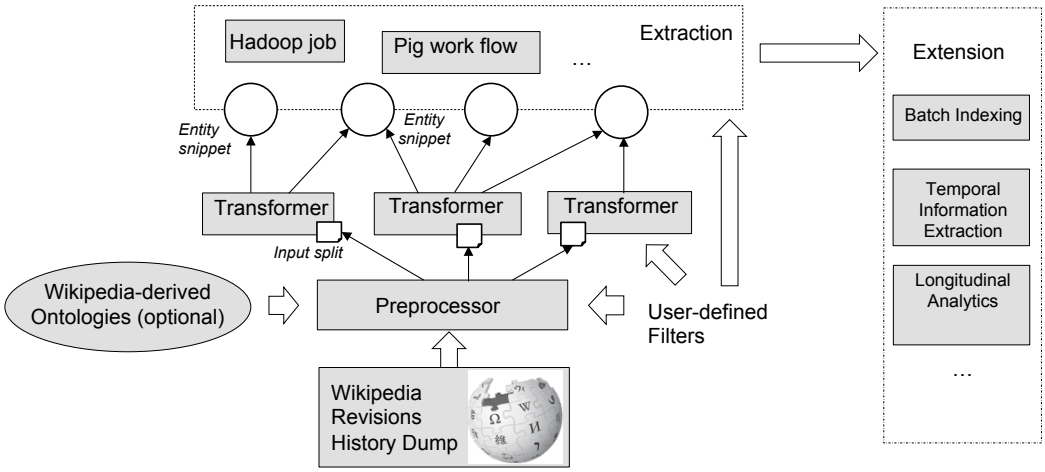


Figure 3.5: Overview of the Hedera Architecture

Extracting Information

Before extracted in the Map-Reduce phase (Extraction component in Figure 3.5), file splits outputed from the preprocessor are streamed into a *Transformer*. The main goal of the transformer is to consume the files and emits (*key*, *value*) pairs suitable for inputting into one Map function. Hedera provides several classes of transformer, each of which implements one operator specified in the extraction layer. Pushing down these operators into transformers reduces the volume of text sent around the network. The extraction layer enables users to write extraction logic in high-level programming languages such as Java or Pig [9], which can be used in other applications. The extraction layer also accepts user-defined filters, allowing user to extract and index different portions of the same partitionsat different time. For instance, the user can choose to first filter and partition Wikipedia articles published in 2012; and later she can sample, from one partition, the revisions about people published in May 2012. This flexibility facilitates rapid development of research-style prototypes in Wikipedia revision dataset, which is one of our major contributions.

Programming Models

To support the scalability, flexibility and interoperability, in Hedera we decided to develop APIs in two levels. The low levels interact directly with raw data and either extract information from them, or transform the format to the next step (Transformer). The high levels provide the APIs to handle the transformed data in different languages. In Java languages, it is supplied in the form of Mapper and Reducer jobs to support standard Map-Reduce work [Whi12]. In Pig language, it provides PageFunc, an operator built on Pig EvalFunc interface [ORS+08]. In essence, a PageFunc implementation defines a declarative function

[9] http://pig.apache.org

to be operated on a Wikipedia page, and output various information either directly to the end users, or dispatched to the next PageFunc operators. Finally, to support Python language, Hedera relies on MRJob[10], a pure Python framework for writing multi-step Map-Reduce jobs in a Hadoop or Amazon Elastic (EMR[11]) cluster.

Entity Temporal Meta-data

Hedera enriches the Wikipedia revision history dataset by extracting the different meta-data about entities, both at document (Wikipedia page) level, as well as at collection level. To be able to extract entity meta-data both efficiently and flexibly, Hedera uses the following principles to process the data:

1. Each entity is identified by a URL
2. List of desired entities is "virtually" loaded into the main memory and mirrored in each Mapper.
3. A dedicated Map-Reduce job is performed on the whole dataset to filter the revisions containing annotations or anchors to the entities in the list.
4. Each user-defined Map-Reduce job only process the filtered revisions.
5. The output is an inverted index keyed by entity URL, followed by information about the meta-data. Each metadata can have optionally time values.

The entity URL can be as simple as a Wikipedia page URL address, or can be identifiers in knowledge bases such as DBpedia and YAGO. Each task of Hedera only works on a predefined list of entities. In different applications in our thesis, we see that this is sufficient to support various mining tasks, for instance the entity re-ranking task where the top retrieved entities are pruned before. Note that the list of entities can be very large, and it still works well in our framework, because Hedera supports a off-heap storage to load partially a portion of the list on-demand ("virtual" in-memory map)[12].

The dedicated annotation Map-Reduce job scans the Wikipedia pages in parallel and detects the annotation to the entity in the desired list. In the simplest variant, the annotation job is a simpler Media-Wikie parser that identifies the internal link markups pointing to other Wikipedi pages. Hedera also supports the end-to-end annotation on plain text using Stanford CoreNLP pipeline [MSB+14]. Finally, the user-defined Map-Reduce jobs are implemented by extending the Hedera programming models in the respective languages, described above. The time value is an epoch value associated with the meta-data to represent revisions from which the meta-data is extracted. Some examples of Entity meta-data extracted by Hedera built-in Map-Reduce jobs include (but not limited to):

[10] http://pythonhosted.org/mrjob/
[11] https://aws.amazon.com/emr/
[12] We make use of MapDB http://www.mapdb.org/ to load the list

Entity Anchors: For each entity, Hedera outputs the list of anchors from all Wikipedia revisions pointing to the entity.

Entity Temporal Graph: For each entity, Hedera outputs all other entities in the desired list that are linked from the input entity. Each link is equipped with the time values indicating which revisions the links appear.

Entity References: For each entity, Hedera outputs the list of web URLs referenced in its Wikipedia page content.

Entity Inverted Index: Build the normal entity inverted index, where the terms are entities and payloads are the words extracted from its revisions, together with the time value indicating the revision timestamp.

3.3.3 Entity Temporal Indexing

Indexing large-scale longitudinal data collections i.e., the Wikipedia history is a straightforward problem. Challenges in finding a scalable data structure and distributed storage that can most exploit data along the time dimension are still not fully addressed. In this section, we present a distributed approach in which the collection is processed by Hedera and thereafter we parallelize the indexing using the Map-Reduce paradigm. This approach (that is based on the document-based data structure of ElasticSearch) can be considered as a baseline for further optimizations. The index schema is loosely structured, which allows flexible update and incremental indexing of new revisions (that is of necessity for the evolving Wikipedia history collection). Our preliminary evaluation showed that this approach outperformed the well-known centralized indexing method provided by [FZG11]. The time processing (indexing) gap is exponentially magnified along with the increase of data volume. In addition, we also evaluated the querying time (and experienced the similar result) of the system. We describe how the temporal index facilitate large-scale analytics on the semantic-ness of Wikipedia with some case studies. The detail of the experiment is described below.

We extract 933,837 entities registered in DBpedia, each of which correspond to one Wikipedia article. The time interval spans from 1 Jan 2011 to 13 July 2013, containing 26,067,419 revisions, amounting for 601 GBytes of text in uncompressed format. The data is processed and re-partitioned using Hedera before being passed out and indexed into ElasticSearch [13] (a distributed real-time indexing framework that supports data at large scale) using Map-Reduce.

Case Study: Entity Timelines Exploration. Figure 3.6 illustrates one toy example of analyzing the temporal dynamics of entities in Wikipedia. Here we aggregate the results for three distinct entity queries, i.e., *obama*, *euro* and *olympic* on the temporal *anchor-text* (a visible text on a hyperlink between two Wikipedia revision) index. The left-most table shows the top terms appear in the returned results, whereas the two timeline graphs illustrate

[13] http://www.elasticsearch.org

Figure 3.6: Exploring Entity Structure Dynamics Over Time

the evolving of the entities over the studied time period (with 1-week and 1-day granuality, from left to right respectively). As easily observed, the three entities peak at the time where a related event happens (Euro 2012 for *euro*, US Presidential Election for *obama* and the Summer and Winter Olympics for *olympic*). This further shows the value of temporal anchor text in mining the Wikipedia entity dynamics. We analogously experimented on the Wikipedia *full-text* index. Here we brought up a case study of the entity co-occurrance (or temporal relationship) (i.e., between Usain Bolt and Mo Farah), where the two co-peak in the time of Summer Olympics 2012, one big tournament where the two athletes together participated. These examples demonstrate the value of our temporal Wikipedia indexes for temporal semantic research challenges.

3.4 Chapter Summary

In this chapter, we address the problem of enriching temporal text collections by the means of semantic annotation and indexing. We study on two big temporal datasets: Twitter and Wikipedia Revision History, each with Terabytes in size. For Twitter data, we address the problem of annotating the trending topics. For indexing, we demonstrate Hedera, our ongoing work in supporting indexing and exploring entity dynamics over time at large scale. Hedera can work directly with Wikipedia revision history dataset in the low-level, it uses Map-Reduce to achieve the high-performance computation.

Our main contributions in this chapter can be summarized as:

- We address the problem of topical annotation of trending hashtags, which goes beyond the traditional annotation of individual tweets. Topical annotations have the

benefit of giving more high-level semantics of the topics, making it more user intuitive.

- We are the first to combine the Wikipedia edit history and page view statistics to overcome the temporal ambiguity of Twitter hashtags.
- We propose a novel and efficient learning algorithm based on influence maximization to automatically annotate hashtags. The idea is generalizable to other social media sites that have a similar information spreading nature.
- We propose an efficient algorithm to learn the unified similarity for entity ranking in Twiter annotation, without the need for human-labeled data.
- We conduct thorough experiments on a real-world dataset and show that our system can outperform competitive baselines by 17-28% for the task of topical annotation in Twitter.
- We developed a large-scale Wikipedia processing tool using Map-Reduce paradigm. Our tool is open-sourced to enable further research in the similar line.

Future Directions. There are many directions for future work. As for the Twitter topic annotation, we aim to improve the efficiency of our entire work flow, such that the annotation can become an end-to-end service. We also aim to improve the context similarity between entities and the topic, for example by using a deeper distributional semantics-based method, instead of language models as in our current work. In addition, we plan to extend the annotation framework to other types of trending topics, by including the type of out-of-knowledge entities. Finally, we are investigating how to apply more advanced influence maximization (IM) methods. We believe that IM has a great potential in NLP research, beyond the scope of microblogging topics. For Hedera, we aim to extend the tool with deeper integration with knowledge bases, provide more API and services to access the extraction layer more flexibly. Another direction is to adopt the recent development of Apache Spark [14] to replace the Map-Reduce low level processing APIs to ease the tool usage.

[14] https://spark.apache.org/

4 Text Summarization Using Timeline and Neural Networks

In the previous chapter, we study different schemes to enrich temporal text collections with meta-data about entities. In this chapter, we discuss another popular task in text mining: *Summarization*. Due to the high volume of text data and the requirements for humans to quickly digest the contents, summarization has attracted more and more attention in text mining community. We study two different summarization applications: Timeline summarization of news, and the summarization of meeting transcripts using artificial neural network models. These two studies focus on the two aspects of the thesis, temporal and cognitive models. Moreover, in both cases, we leverage the information of the semantic annotation to improve the quality accordingly.

4.1 Introduction

Text summarization is the process of automatically reducing contents of one or multiple textual documents to retain only representative information that humans can quickly comprehend [Wik17]. Due to the explosive volumes of text data in all types of digital environments (public media, organization data, personal collections, etc.), summarization has become an important task and attracts increasing attention from the text mining community. A high-quality summarization does not only assists humans in quickly grasping the main messages from the vast amounts of documents, but also enables them to discover the hidden connections between contents and extract useful knowledge [SG10a].

In general, summarization research can be categorized into two types: *Extractive* and *Abstractive*. Extractive summarization aims to identify the key pieces of text from the document(s) that deem to be the most representative to construct the summary, while abstractive summarization generates a human-friendly summary based on the original contents, either by paraphrasing and generating sentences or clauses [DM07]. Research on extractive summarization has achieved significant success with the advances of text processing

techniques [ER04, WPF+99, WRC+13], most of which attempt to identify sentences or key phrases for the summary. Our research falls into this category, but we address more specific requirements of temporal text, and also leverage the ideas of human memory to build cognitive models to increase the summarization quality. More specifically, we conduct two independent research studies:

In the first research, we study the problem of temporal summarization, i.e. generating summaries where texts are available at different points in time, typically in chronological order. One popular setting is the news summarization with respect to an event of interest, either in a timeline fashion [YWO+11, TTT+13a], in a topic hierarchy [AHE+11], or through a graph of information flow [SGH12a]. Such settings often have additional requirements for the summary. For example, in timeline summarization, a text snippet (e.g. sentences) should not only be representative, but also novel compared to other snippets in the summary along the temporal developments of the news topic. We discuss this in more details in section 4.1.

In the second research, we turn to cognitive aspects and study the problem of decision summarization from conversational text. Specifically, we analyze multi-party dialogues in business meetings, and identify the main phrases that summarize the process in which participants make decisions in a collaborative manner. Such a summarization relies heavily on the understanding of human cognitive processes, and also requires advanced methods to go beyond the shallow analysis of spoken contents. Section 4.3 describes this study in more details.

4.2 Timeline Summarization of News Events

In the first part of our study, we address the timeline summarization task for high-impact and long-running events such as the Boston Marathon Bombing [1] or the crash of Germanwings Flight 9525 [2]. Such types of events are interesting case studies because they often develop through many stages and involve a large number of entities in their unfolding. This is more pronounced for long-running events, when full information about the event and its development becomes available only in the course of days after the happening as in the case of the Germanwings airplane crash in March 2015. We present a novel method which shows key entities at different time points of an event thus capturing this dynamic event unfolding. In contrast to other work in event summarization [YWO+11, MMO14], our *entity timelines* use entities instead of sentences as main units of summarization as depicted in the case of the 2015 Germanwings plane crash (Figure 4.1). Such summaries can be easily digested and used both as starting points for personalized exploration of event details, and for retrospective revisiting. The latter can be triggered by a new similar event, or by a new twist in the story. For example, the testimonial of the captain in the Costa Concordia trial in late 2014 triggered a revisiting of the disaster in 2012.

From a cognitive perspective, for event revisiting, we rather create "memory cues" to

[1] https://en.wikipedia.org/wiki/Boston_Marathon_bombing
[2] https://en.wikipedia.org/wiki/Germanwings_Flight_9525

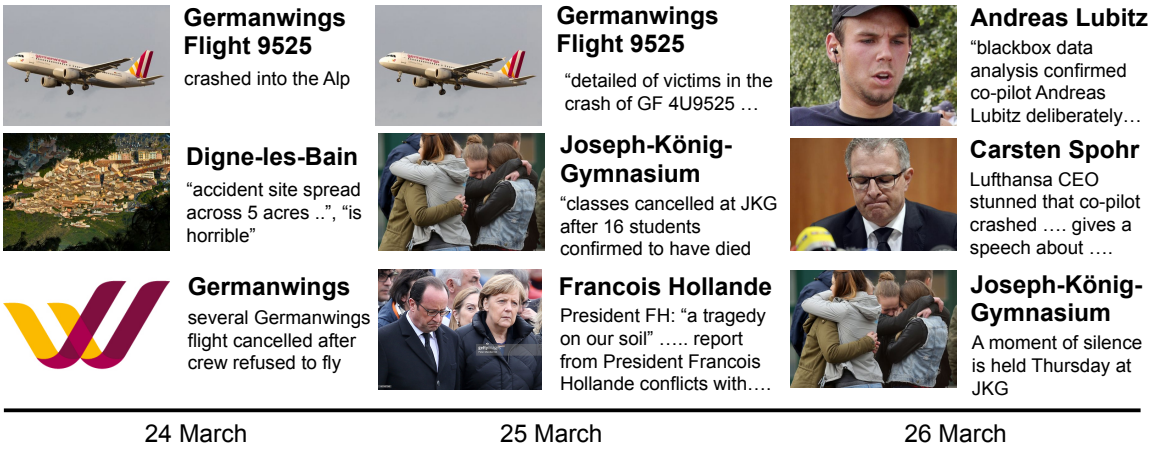


Figure 4.1: Entity Timeline for the 2015 Germanwings plane crash event.

support remembering the unfolded events than summaries for rehearsing the details. In fact, memory cues can be regarded as "a circumstance or piece of information which aids the memory in retrieving details not re-called spontaneously" (Oxford online dictionary, 2015). In this sense, our work is related to the idea of designing or creating memory cues for real-life remembering [vdHE14]. Entities such as persons and locations have been identified as very effective external memory cues [Ber09]. In addition, the importance of entities in event summarization has also been shown in recent work [MMBJ13].

For creating an entity timeline, the entities to be used in the summary have to be chosen carefully. They should 1) be characteristic for a respective time point of the event, 2) not be repetitive (if nothing new happened with respect to the entities), 3) be associated to relevant event information, and 4) be interesting to the reader. For this purpose, we propose an approach to entity summarization, which dynamically combines *entity salience* with the *informativeness* of entities at a considered point in time. Entity salience, on the one hand, considers the property of being in the focus of attention in a document has been studied in previous work [BK99, GYS+13, DG14]. In [BK99], Boguraev and Kennedy use salient text phrases for the creation of so-called *capsule overviews*, whereas recently methods for the identification of salient entities, e.g., in Web pages [GYS+13] and news articles [DG14], have been developed. Informativeness, on the other hand, assesses the level of new information associated with an entity in a text and can be computationally measured using features derived from statistical and linguistic information [WG13].

The task of news summarization has been already studied in various contexts, which range from focusing on multi-document summarization [BK99, ER04] to generating a timeline summary for a specific news story [YWO+11, MMO14, TTT+13a, ZGY+13]. News stories can be complex, having a non-linear structure and associated to multiple aspects. Shahaf et al. [SGH12b] propose a novel method for summarizing complex stories using metro maps that explicitly capture the relations among different aspects and display a temporal development of the stories. Instead of using documents or sentences as a information units, we provide a set of entities, as *memory cues*, for supporting event exploration and

digestion at each individual point in time.

To the best of our knowledge, none of the previous work provides a trade-off solution that balances between content-based and collective attention-based approaches, in supporting the entity-centric summarization. In contrast, we aim at optimizing a trade-off between the in-document salience of entities and the informativeness of entities across documents describing the unfolding of an event.

4.2.1 Approach Overview

Preliminaries

In this section, we introduce the concepts used in our summarization framework. We also briefly describe our framework from the computational perspective (section 4.2.2).

Long-running Event. Following [MMO14], we define a long-running event as "a newsworthy happening in the world that is significant enough to be reported on over multiple days". Each event is represented by a short textual description or a set of keywords q, where we will use q to denote the event in the rest of this chapter. For example, the bombing incident during Boston Marathon in April 2013 can be described by the terms "boston marathon bombing". We assume that the relevant time frame is split into a sequence of consecutive non-overlapping, equal-sized time intervals $T = \{t_1, \ldots, t_n\}$ in a chronological order, in our case individual days. Furthermore, for a given event q, there is a set of timestamped documents D_q (each with a publication date) reporting on the event. We define a *reporting timeline* $T_q = \{t_{k_1}, \ldots, t_{k_j}\}$ as an ordered list of (not necessarily consecutive) those time periods t_{k_i} in T, which contain the publication date of at least document in D_q. Finally, we denote the set of all documents about q published within a time period t_{k_i} as $D_{q,i}$.

Entity. We are interested in named entities mentioned in documents, namely, persons, organizations, locations. An entity e can be identified by a canonical name, and can have multiple terms or phrases associated with e, called **labels**, which refer to the entity. We call an appearance of an entity label in a document, a **mention** of e, denoted by m. We define E_q as the set of all entities mentioned in D_q. Furthermore, we define the text snippet surrounding m (e.g. sentence or phrases) as the **mention context**, denoted $c(m)$.

Entity Salience and Informativeness. Similar to [DG14], we define the entity salience as the quality of "being in the focus of attention" in the corresponding documents. Another relevant aspect considered for selecting entities to be included in an event timeline is *informativeness* [GYS+13], which imposes that selected entities in an evolving event should also deliver novel information when compared to the past information. For example, although the airline "Germanwings" stays relevant for many articles reporting on the plane crash, it will only be considered as informative, if new information about the airline becomes avail-

able.

Problem Statement. Given a long-running event q, a time interval t_i in its reporting timeline T_q, and the set of entities E_q, we aim to identify the top-k salient and informative entities for supporting the exploration and digestion of q at t_i.

Framework

Figure 4.2 gives an overview of our entity ranking framework covering both the training and the application/testing phase.

Given one event q, its reporting timeline, and the set of documents D_q (in practice, D_q can be given a priori, or can be retrieved using different retrieval models) we identify the entity set E_q using our entity extraction, which consists of named entity recognition, co-reference and context extraction (section 4.2.1).

When the event is used for training (training phase), we link a subset of E_q to Wikipedia concepts, which comprises the popular and emerging entities of the event. To facilitate the learning process, these entities are softly labeled using view statistics from Wikipedia (section 4.2.3), serving as training instances. Although we use popular entities for training, we design the features such that it can be generalized to arbitrary entities, independent from Wikipedia.

The next component in our framework is the adaptive learning that jointly learns the salience and informativeness models, taking into account the diverse nature of events and their evolution. (section 4.2.2).

In the application phase, entity and feature extraction are applied the same as in the training phase. First, the input event and time interval is examined against the joint models to return the adaptive scores (details in section 4.2.2). Then, entities are put into an ensemble ranking, using the adapted models, to produce the final ranks for the summary.

Entity Extraction

As mentioned in the beginning of this chapter, named entities are exploited to improve our summarization framework and as such, the availability of the entity annotations in the documents plays a crucial role. While applying methods developed in chapter 3 will give high-quality entities for the retrieved collection, in order to identify the salient and informative entities in a machine learning manner, we need to extract features of the entities, and map the entities with their contexts. We rely on different sources of text to extract the features:

First, for each entity of which mentions are available, we include the containing sentences as the context. Mentions that do not contain any alphabetical characters or only stop words are removed.

Second, we additionally use intra- and cross-document co-reference to track mentions

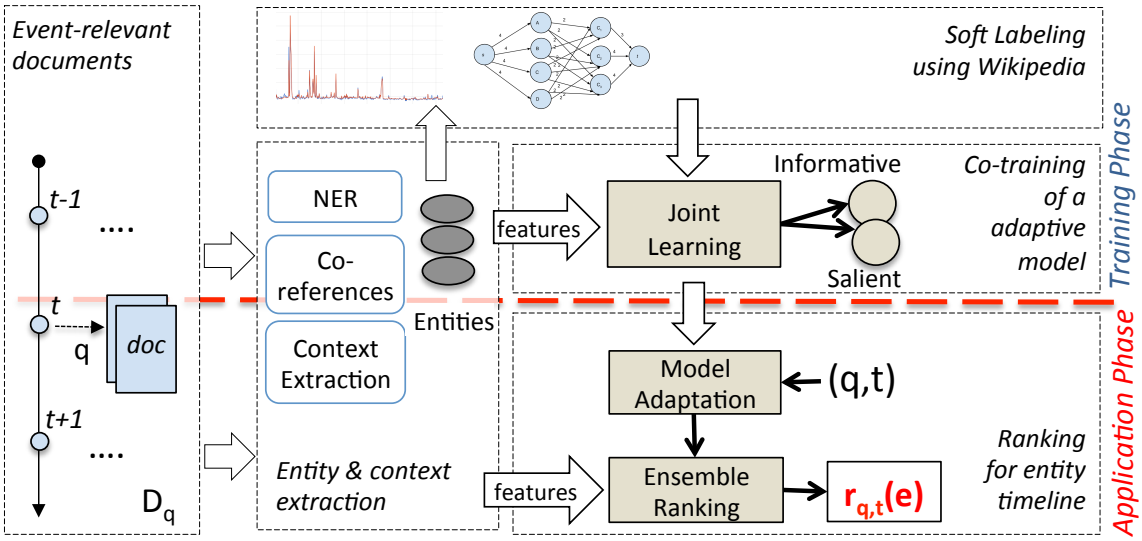


Figure 4.2: Overview of the Entity-centric Summarization Framework

pertaining to the same entity. Here, an intra-document co-reference system is employed to identify all co-reference chains for entity mentions within a document. We include each reference in the chain together with its sentence to the set of mention contexts of the entity. In addition, to identify mentions that potentially refer to the same real-world entity across documents, we adapt the state-of-the-art cross-document co-reference method proposed by Lee et al.[LRC+12]. This method first clusters documents based on their content using an Expectation Maximization (EM) algorithm, then iteratively merges references (verbal and nominal) to entities and events in each cluster.

Speeding up co-reference resolution: To speed up the computation, we do not use EM clustering as [LRC+12], but employ a set of heuristics which have proven to be effective in practice. First, we only consider cross-document co-references from documents of the same day. Second, instead of clustering an entire document set, we use mentions with their contextual sentences (kept in the order of their appearance in the original documents) as "pseudo-documents" for the clustering. Third, we assume that mentions to the same entity have similar labels. Hence, we represent entity mention labels as vectors using two-grams (for instance, "Obama" becomes "ob", "ba", "am", "ma") and apply LSH clustering [GIM+99] to group similar mentions. The use of LSH has been proven to perform well in entity disambiguation tasks [HSN+12], and it is much faster than the standard EM-based clustering. We train the regression model for text reference merging using the ECB+ corpus [BH10].

Finally, for each entity mention, we merge its contextual sentences with those of all other references of the same co-reference chain to obtain the event-level context for an entity , which will be used as inputs for constructing the entity features. We note that while we are aware of other methods to increase the quality of entity extraction by linking them to a knowledge base such as YAGO or DBpedia, we choose not to limit our entities to such

knowledge bases, to be able to identify and rank more entities in the "long tail".

4.2.2 Adaptive Summarization

Optimization Goal

Given the document equipped with entities, we tackle the summarization problem by learning a function for ranking entities. In essence, the aim is to optimize the trade-off between in-document entity salience and the informativeness of entities across documents:

$$y_t^{(q)} = f(\mathbf{E}, \omega_\mathrm{s}, \omega_\mathrm{i}), f \in \mathcal{F} \tag{4.1}$$

where $y_t^{(q)}$ is the vector of ranking scores for entities in E_q at time interval t, $\mathbf{E}$ is a matrix composed from feature vectors of entities in E_q extracted from their mention contexts. $\omega_\mathrm{s}, \omega_\mathrm{i}$ are the unknown parameter vectors for ranking entities based on salience and informativeness, respectively.

In our work, a ranking function is based on a learning-to-rank technique [Joa02]. A general approach for learning-to-rank is to optimize a defined loss function L given manual annotation or judgments $\mathbf{y}_j^{(q)}$ of entities for a set of training events $\mathcal{Q}$ within the time intervals T_q:

$$\hat{f} = \arg\min_{f \in \mathcal{F}} \sum_{q \in \mathcal{Q}} \sum_{t_j \in T_q} L(f(\mathbf{E}_j^{(q)}, \omega_\mathrm{s}, \omega_\mathrm{i}), \mathbf{y}_j^{(q)}) \tag{4.2}$$

Two major challenges must be taken into account when learning a ranking function defined in Equation 4.2. First, we need a reliable source for building judgments (ground truths) for annotating entities by considering their salience with respect to a given event. In addition, the judgments must be dynamically adapted to the evolving of entities along the unfolding event, i.e., bearing of novel information. Second, the models of our two aspects $\omega_\mathrm{s}, \omega_\mathrm{i}$ must be unified to produce a joint learned function for ranking entities. In the following, we will explain our proposed method for these challenges in more detail.

Unified Ranking Function

We now turn our attention to defining the ranking function in Equation 4.1. The intuition is that for each event q and time t_i, we rank an entity e higher than others if: (1) e is more relevant to the central parts of documents in $D_{q,i}$ where it appears (salience); and (2) the context of e is more diverse to other contexts (of e or other entities) at $D_{q,i-1}$. Moreover, these two criteria should be unified in an adaptive way that depends on the query, that is, event and time. For example, users interested in a festival might wish to know more about salient entities of the event, while those that follow a breaking story prefer entities with more fresh information. Even for one event, the importance of salience and informativeness might vary over time. For instance, informativeness is more important at the beginning when the event is updated frequently. Based on this intuiton, we propose the following

ranking function:

$$\mathbf{y_t^{(q)}} = S(q,t)\,\omega_{\mathrm{s}}^T\mathbf{E}_{\mathrm{s}} \;+\; \gamma(t)\,I(q,t)\omega_{\mathrm{i}}^T\mathbf{E}_{\mathrm{i}} \tag{4.3}$$

where $\mathbf{E}_s$ is the $|E_q| \times M$ matrix representing the M dimensional feature vectors of entities used to learn the salience score (M is the number of salience features), and $\mathbf{E}_i$ is the $|E| \times N$ matrix of N dimensional informativeness feature vectors (N is the number of informativeness features). $S(q,t)$ and $I(q,t)$ represent the scores of salience and informativeness tendency for an event q at t. Here we introduce another factor, $\gamma(t)$, which is the decay function of time t, controlling how much the informativeness should have impact on the overall ranking. The rationale of γ is that when the distance between two time intervals t_i and t_{i-1}[3] is long, informativeness has less impact on the overall ranking. For example, if there are only reports about the news after one year (anniversary of a past event), the changes of entities in that long time period should not contribute much to the informativeness criterion.

Multi-criteria Learning Model

We now discuss how to learn the above ranking function using Equation 4.2. A straightforward way is to learn the two models ω_{s} and ω_{i} separately, and assign values to S, I in a predefined manner, then aggregate into Equation 4.3. However, this is not desirable, since it requires building two sets of training data for salience and informativeness at the same time, which is expensive. Secondly, previous work has pointed out that a "hard" classification of a query based on intent itself is a difficult problem, and can harm the ranking performance [GLQ+08]. In this work, we exploit the *divide and conquer* (DAC) learning framework in [BLL+10] as follows. We define $\mathbf{E}^*$ as the $|E| \times (M+N)$ matrix of $(M+N)$ dimensional *extension vectors* from the corresponding vectors of $\mathbf{E}_{\mathrm{s}}$ and $\mathbf{E}_{\mathrm{i}}$ matrices. Similarly, we define ω_{s}^* as the $(M+N)$ extension vectors of zero vector $\mathbf{0}$, and the vectors ω_{s}, and ω_{i}^* as the $(N+M)$ extension vector of ω_{i} and $\mathbf{0}$. With this transformation, Equation 4.3 becomes:

$$y_t^{(q)} = S(q,t)\,g(\mathbf{E}^*,\omega_{\mathrm{s}}^*) \;+\; \gamma(t)\;I(q,t)g(\mathbf{E}^*,\omega_{\mathrm{i}}^*) \tag{4.4}$$

where $g(\mathbf{E}^*,\omega) = \omega^T\mathbf{E}^*$ is a linear function. Incorporating Equations 4.2 and 4.4, we can co-learn the models $\omega_{\mathrm{s}}^*, \omega_{\mathrm{i}}^*$ (and thus $\omega_{\mathrm{s}}, \omega_{\mathrm{s}}$) simultaneously, using any loss functions. For instance, if we use hinge loss as in [BLL+10], we can then adapt RankSVM [Joa02], an algorithm that seeks to learn the linear function g in (4.4) by minimizing the number of misordered document pairs. For completeness, we describe here the traditional objective function of RankSVM:

$$\begin{gathered}\min_{\omega,\xi_{q,t,a,b}} \frac{1}{2}\|\omega\|^2 + c\sum_{q,t,a,b}\xi_{q,t,a,b}\;,\; s.t.\\ \omega^T\mathbf{E}^{*(q)}_{t,a} \geq \omega^T\mathbf{E}^{*(q)}_{t,b} + 1 - \xi_{q,t,a,b} \forall \mathbf{E}^{*(q)}_{t,a} \succ \mathbf{E}^{*(q)}_{t,b}, \xi_{q,t,a,b} \geq 0\end{gathered} \tag{4.5}$$

where $\mathbf{E}^{*(q)}_{t,a} \succ \mathbf{E}^{*(q)}_{t,b}$ implies that entity a is ranked higher than entity b for the event q at time t, $\xi_{q,t,a,b}$ denotes slack variables, and c sets the trade-off between the training error and

[3] note that news events are not always reported on consecutive time intervals

model complexity. If we change the linear function g to f, we can adapt (4.2) into (4.5) to obtain the following objective function:

$$\begin{aligned} \min_{\omega,\xi_{q,t,a,b}} \frac{\|\omega_{\mathrm{s}}\|^2 + \|\omega_{\mathrm{i}}\|^2}{2} + c \sum_{q,t,a,b} \xi_{q,t,a,b} \quad \text{s.t.} \\ S(q,t)g(\mathbf{E}^{*(q)}_{t,a}, \omega^*_{\mathrm{s}}) + \gamma(t)I(q,t)g(\mathbf{E}^{*(q)}_{t,a}, \omega^*_{\mathrm{i}}) \geq \\ S(q,t)g(\mathbf{E}^{*(q)}_{t,b}, \omega^*_{\mathrm{s}}) + \gamma(t)I(q,t)g(\mathbf{E}^{*(q)}_{t,b}, \omega^*_{\mathrm{i}}) + 1 - \xi_{q,t,a,b}, \\ \forall \mathbf{E}^{*(q)}_{t,a} \succ \mathbf{E}^{*(q)}_{t,b}, \xi_{q,t,a,b} \geq 0 \end{aligned} \tag{4.6}$$

Event-based Models Adaptation

The adaptive scores $S(q,t), I(q,t)$ and the decay function $\gamma(t)$ is critical to adapting the salience and informativeness models. A naïve supervised approach to pre-define the categories for event (S, I) is impractical and detrimental to ranking performance if the training data is biased. Instead, previous work on query-dependent ranking [GLQ+08, BLL+10] often exploit the "locality property" of query spaces, i.e., features of queries of the same category are more similar than those of different categories. Bian et al.[BLL+10] constructed query features using top-retrieved documents, and clustered them via a mixture model. However, the feature setting is the same for all clusters, making it hard to infer the semantics of the query categories.

In this work, we inherit and adjust the approach in [BLL+10] as follows. For each event q and time t, we obtain all entities appearing in $D_{q,t}$ to build the "pseudo-feedback" for the query (q,t). We then build the query features from the pseudo-feedback as follows. From each matrix $\mathbf{E}_{\mathrm{s}}, \mathbf{E}_{\mathrm{i}}$, we take the *mean* and *variance* of the feature values of all entities in the pseudo feedback. As a result, each pair (q,t) is mapped into two feature vectors (with $2M$ and $2N$ dimensions) corresponding to the salience and informativeness spaces. In each space, we use Gaussian Mixture model to calculate the centroid of the training queries, and use the distance of the query feature vector to the centroid as its adaptive score:

$$C(q,t) = 1 - \frac{\|\mathbf{x}^{q,t}_C - \mathbf{x}^C\|^2}{\max_{q' \in \mathcal{Q}, t' \in T_{q'}} \|\mathbf{x}^{q',t'}_C - \mathbf{x}^C\|^2} \tag{4.7}$$

where $C \in \{I, S\}$ indicates the event categories, $\mathbf{x}^{q,t}_C$ is the query feature in the feature space of C, and $\mathbf{x}^C$ is the centroid of feature vectors in training set $\mathcal{Q}$ in the corresponding space. The scores are scaled between 0 and 1.

Decayed Informativeness. The decay function $\gamma(t_i)$ adjusts the contribution of informativeness into the adaptive model and is defined by:

$$\gamma(t_i) = \alpha^{\lambda \frac{|t_i - t_{i-1}|}{\mu}} \tag{4.8}$$

where λ, α are parameters ($0 < \alpha < 1, \lambda > 0$), and μ is the interval unit distance. Equation 4.8 represents the time impact onto the informativeness of entities: When the time lag

between two intervals is high, the difference in contexts of entities between them is less likely to correlate with the informativeness quality of entities.

4.2.3 Mining Collective Memory in Wikipedia

The above learning method requires the availability of a high-quality training data. Specifically, given an event q, an interval t and an entity e, we need the score label $y_j^{(q)}$ such that $y_j^{(q)}(e) > y_j^{(q)}(e')$ if the entity e is more prominent than the entity e' with respect to the event q at time t. This score is used to learn the ranking functions mentioned in Equation 4.2.

Unfortunately, manual acquiring such labels are tedious and expensive tasks, and can be easily biased by the annotators' opinion and knowledges. Instead, in this work, we aim to mine the user behaviour in viewing Wikipedia as the proxy of their feedback to particular entities, and define methods to generate the "soft labels" (i.e., labels that asymptotically represent the ground truth) for the entities. Indeed, the use of soft labeling for entities' salience has already been proposed in [GYS+13], where user click behaviour in query logs is used as an indicator for entity salience scores. Dunnietz et al. [DG14] proposed treating entities in news headlines as salient, and propagate those salience scores to other entities via the PageRank algorithm. The limitation of these measures is that they restrict the assessment of salience to the scope of individual documents, and do not consider the temporal dimension. In contrast, our soft labels are evolving, i.e., an entity can have different labels for one event at different time intervals.

Mining Wikipedia View Behaviour. The soft labeling is based on the assumption that for globally trending news event, prominence of related entities can be observed by the collective attention paid to resources representing the entities. For instance, during the Boston marathon bombing, Wikipedia pages about the bomber *Tsarnaev* were created and viewed 15,000 times after one day, indicating their strong salience driven by the event. For soft labeling, we exploit the page view statistic of Wikipedia articles, which reflects the interest of users in an entity: Most obviously, Wikipedia articles are viewed for currently popular entities indicating entity salience. However, taking the encyclopedic character of Wikipedia into account, Wikipedia articles are also viewed in expectation of new information about an entity indicating its (expected) informativeness, especially in the context of an ongoing event. Actually, Wikipedia has gained attention in recent years as a source of temporal event information [GKK+13]. Event-triggered bursts in page views, as they are for example used in [CN10] for event detection, are thus a good proxy for the interestingness of an event-related entity at a considered point in time, which is influenced both by the salience and the informativeness of the event.

Therefore, we propose a new metric called *View Outlier Ratio* or VOR to approximate the soft labels for a combined measure of entity salience and informativeness as follows. For each entity e and for a given time interval t_i, we first construct the time series of view

count from the corresponding Wikipedia page of e in the window of size w:

$$T_e = [v_{e,i-w}, v_{e,i-w+1}, \dots v_{e,i}, v_{e,i+1}, \dots v_{i+w}]$$

where each $v_{e,j}$ is the view count of the Wikipedia page of e at t_j. From T_e we calculate the median $m_{e,i}$ and define VOR as follows.

Definition 4.1 *The View Outlier Ratio is the ratio of difference between the entity view and the median:*

$$vor(e_i) = \frac{|v_{e,i} - m_{e,i}|}{\max(m_{e,i}, m_{min})} \tag{4.9}$$

where m_{min} is a minimum threshold to regularize entities with too low view activity.

4.2.4 Entity Features

We now discuss the salience and informativeness features for entity ranking. Ranking features are extracted from event documents where the entity appears as follows. These features, called *individual features*, are extracted on two different levels. First, on **context** level, features are extracted independently from each mention and its contexts. Features of this level include mention word offset, context length, or importance scores of the context within the document using summarization algorithms (SumBasic or SumFocus features). Second, on **label** level, features are extracted from all mentions, for instance aggregated term (document) frequencies of mentions.

Based on the individual features, the entity features are constructed as follows. For each entity and feature dimension, we have the list of feature values $(z_1, z_2, \dots, z_n)$, where z_i is the individual feature of label or mention categories. For label level, we simply take the average of z_i's over all entity labels. For mention level, each z_i is weighted by the the confidence score of the document containing the corresponding mention and context. Such confidence score can be calculated by several ways, for instance by a reference model (e.g. BM25) when retrieving the document, or by calculating the authority score of the document in the collection (e.g. using PageRank algorithm). For all features, we apply quantile normalization, such that all individual features (and thus entity aggregated features) are scaled between $[0, 1]$. Below we describe the most important features.

Salience Features

Context importance features. One important evidence of entity salience is the context of the entity mentions. It is well-known that text at the beginning of a document contains more salient information [GYS+13, DG14]. Besides the position, the content of sentences, per se or in relations with other sentences, also indicates the salience of entities. We apply

Salience features		*Informativeness features*	
Feature(s)	**Description**	**Feature(s)**	**Description**
Tf / Df (M)	Term / Doc. frequency of mention in $D_{q,i}$	PTf / Pdf (M)	Term / Doc. frequency of mention in $D_{q,i-1}$
WO / SO (C)	Word / sent. offset of mention in context	CoEntE (M)	Number of co-occurring entities in $D_{q,i-1}$
SentLen (C)	Context length with/without stopwords	CTI(C)	CTI score of mention given its context in $D_{q,i}$ and $D_{q,i-1}$ [WG13]
Sent-5 /-10 (C)	Context length with/without stopwords $>$ 5/10 ?		
1- / 2- / 3-Sent (C)	Is context among 1/3/5 first sentences ?	TDiv (C,M)	Topic diversity of context in $D_{q,i}, D_{q,i-1}$
TITLE (M)	Is the mention in titles of any $d \in D_{q,i}$?	CosSim (C)	Cosine similarity of context in $D_{q,i}, D_{q,i-1}$
CoEntM (M)	No. of entities in same context as mention	DisSim (C)	Distributional similarity of context in $D_{q,i}, D_{q,i-1}$
Sum-B / -F (C)	SumBasic / SumFocus score of context		
Uni / Bi / Bi4 (C)	Uni-/Bi-/skip-Bigram overlap with query	EntDif (M)	Entity difference of context in $D_{q,i}, D_{q,i-1}$
Att / Sent (C)	Attitude / Sentimentality score of context	TITLEP (M)	Is the mention in titles of any $d \in D_{q,i-1}$?
Read-1 / -2 / -3 (C)	Fleisch / Fog / Kincaid readability scores	NewsFracP (M)	Frac. of news-specific terms in $D_{q,i-1}$
PR (C)	Sentence centrality score (PageRank)		
POS (C)	POS-tag of mention in context		
NewsFrac (M)	Fraction of news-specific terms co-occurring with mention		

Table 4.1: Selected Informativeness and Salience Features for Entity Ranking at Label (M) and Context (C) Level

SumBasic [NP04] and LexRank [ER04] summarization algorithms to obtain the scores of contexts (features Sum-B and PR, respectively).

Human-perceived salience features. Entity salience can be assessed by the reader's intentions or interests [GYS+13], and recent studies suggest that user interest in entities can be attracted via serendipity in texts [BML13]. We follow this direction and apply features presented in[BML13], namely sentimentality, attitudes, and readability scores of each mention context. For readability, we also include two other standard metrics, Gunning Fog index [Gun52] and Fleisch-Kincaid index [KFJRC75].

Query-related features. Another class of salience features involve ones that are dependent on the event queries. Following [WRC+13], we use the overlap between contexts and the event query at the unigram, bigram levels, and at bigram where tokens are tolerable to have 4 words in between (Bi4 feature). We also compute the query-focused SumFocus [VSBN07]

score of contexts as another feature. It is worth noting that for all of these features, the event queries used are the ones that have been extended using the query expansion (see section 4.2.5).

Informativeness Features

Informativeness for words has been studied extensively in linguistics community. We adapt the state-of-the-art measure [WG13], which incorporates the context of mentions. Given a mention m of entity e, the context-aware informativeness score of m for event q at t_i is defined by:

$$inf(m,i) = \begin{cases} 1 & U_{q,i-1} = \emptyset \\ \frac{\sum_{c' \in U_{q,i-1}} \kappa(c', c(m)) s(d_{c'}, q, i-1)}{|U_{q,i-1}|} & U_{q,i-1} \neq \emptyset \end{cases}$$

where $U_{q,i-1}$ is the set of mention contexts of e in $D_{q,i-1}$, $d_{c'}$ is the document consisting the context c', $\kappa(c', c(m))$ is the sentence dis-similarity, and $s(d_{c'}, q, i-1)$ is the retrieval score of $d_{c'}$. We normalize the metric to $[0, 1]$, and set it to 1, when the entity first appears in $D_{q,i}$. To measure $\kappa(c', c(m))$, we can employ different strategies, leading to different features:

CTI. The most straightforward strategy is to employ a lexicon for the sentence similarity. We use the NESim method [DRS+09] for this strategy.

Topic Diversity. Besides lexicon-based, we also calculate the informativeness on a higher level by representing contexts by latent topics, using latent Dirichlet Allocation model [BNJ03]. Topic diversity of a context c w.r.t other context c' of the same entity on previous time interval is defined as

$$\kappa(c', c(m)) = \sqrt{\sum_{k=1}^{\tau} (p(\psi_k|c(m)) - p(\psi_k|c'))^2}$$

where τ is the number of topics and ψ_k is the topic index. For label-wised topic diversity, we use the same metric, but using concatenated contexts instead of individual ones.

Distributional similarity. Another strategy uses Kullback-Leibler divergence for the dissimilarity:

$$DisSim(c', c(m)) = -KL(\theta_{c(m)}, \theta_{c'}) = \tag{4.10}$$

$$-\sum_{w_i} P(w_i|\theta_{c(m)}) \log \frac{P(w_i|\theta'_c))}{P(w_i|\theta_c(m))} \tag{4.11}$$

where θ_d and θ_c are the language models for contexts c', c. We use Dirichlet smoothing method for scarce words.

Label-level. Besides context feature class, we also calculate a number of label-level fea-

tures, by aggregating the mention features over $D_{q,i-1}$ for event q at time t_i (Table 4.1). Additionally, we also calculate the entity difference between two context sets of the same entity in $D_{q,i-1}$ and $D_{q,i}$:

$$EntDif(e) = \|CoEnt_i(e) \cap \overline{CoEnt_{i-1}(e)}\|$$

where $CoEnt_i(e)$ is set of co-occurring entities of e in $D_{q,i}$

4.2.5 Experimental Setups

Datasets. For our experiments, we work with a real-world, large-scale news dataset. Specifically, we use KBA 2014 Filtered Stream Corpus (SC14) dataset, which is used for TREC 2014 Temporal Summarization track[4]. We extract news and mainstream articles from the dataset, consisting of 7,592,062 documents. The dataset covers 15 long-running news events from December 2012 to April 2013. All events are high-impact, discussed largely in news media, and have their own Wikipedia pages. Each event has a predefined time span (ranging from 4 to 18 days), and is represented by one textual phrase that is used as the initial event query. Based on the event type (available in the dataset), we group the events into 4 categories: Accident, riot and protest, natural disaster, crime (shooting, bombing).

Event Documents. To construct the event document set for the study, we firstly group the documents into individual days by their publication timestamps, and index documents for each day. In total, this results in 126 different indices. For each index, we remove boilerplate texts from document using Boilerpipe [KFN10], skip stop words, and lemmatize the terms. Then we use the pre-defined textual phrases of the events, issue it as a query to the corresponding indices (indices of days within the event period), retrieving from each the top 10 documents using BM25 weighting model. We improve the results using Kullback-Leibler query expansion [IS10], and add top 30 expanded terms to construct the event query q used for query-related features computation.

Timelines. To build the reporting timeline T_q, for each event, we manually go through all the days of the event period, check the content of the top-retrieved document, and remove the day from the timeline if this top-ranked documents is not about the event. In total, we have 153 pairs (*event*,*day*) for all event reporting timelines.

Entities. We use Stanford parser to recognize named entities from the boilerplated content of the documents, and match them with the entities detected by BBN's Serif tool (provided in SC14 corpus) to reduce noise. For the matching entities, we use the in-document coreference chains, which is already provided in SC14, and apply the cross-document coreference (section 4.2.1) to group mentions to entities. We use the sentences as the mention contexts. In total, we detect 72,267 named entities from the corpus, with an average of 5.04

[4] http://s3.amazonaws.com/aws-publicdatasets/trec/kba/index.html

contexts per each.

Training Data. From 153 (*event*,*day*) pairs, we randomly choose 4 events belonging to 4 different categories mentioned above as a training data, resulting in 39 pairs. To build training entities (i.e. to identify subset of entities to Wikipedia concepts, see section 4.2.1), we apply two named entity disambiguation softwares, WikipediaMiner and Tagme. These are the supervised machine learning tools to identify named entities from natural language texts and link them to Wikipedia. The tools both use the models trained from a Wikipedia dump downloaded in 2014 July, so as to cover all possible entities in the SC14 corpus. We only use entities co-detected by both the tools, resulting in 402 distinct entities and 665 training tuples (*entity*,*event*,*day*). We use the Wikipedia page view dataset, which is publicly available, to build the soft labels for these entities.

Parameter Settings. We modify RankSVM for our joint learning tasks. Features are normalized using the Standard scaling. We tune parameters via grid search with 5-fold cross validation and set trade-off parameter $c = 20$. WikipediaMiner and Tagme tools are used with default parameter settings. For the decay parameters, given the rather small time range of events in our dataset, we empirically set $\lambda = 2, \alpha = 0.5$, $\mu = 1$day, and leave more advance tuning for future work. For soft labeling, we set the window size w to 10 days, which is the average length of reporting timelines. The threshold m_{min} is tuned as followed. For each training pair, a human expert knowing the 4 training events well is presented with the entities, their mention contexts and content of the corresponding document. The expert is asked to put the labels on the entity from "falsely detected", "non-salient entity", "salient but not informative" to "salient and informative". Based on this judgment, we compute VOR scores with m_{min} from 1 to 100, and optimize the rank of entities based on VOR scores using NDCG metric. We find that $m_{min} = 12$ yields the best performance.

4.2.6 Evaluation

Baselines. We compare our approach with the following competitive baselines.

TAER: Dermatini et al. [DMBZ10] proposed a learning framework to retrieve the most salient entities from the news, taking into consideration information from documents previously published. This approach can be considered as "salience-pro", since the entity salience is measured within a document, although it implicitly complements the informativeness via history documents. We train the model on the same annotated data provided by the author.

IUS[MMO14]: This work represents the "informativeness-pro" approach, it attempts to build update summaries for events by incrementally selecting sentences, maximizing the gain and coverage with respect to summaries on previous days. Since we are not interested in adaptively determining the cutoff values for the summary, we implement only the learning-to-rank method reported in [MMO14] to score the sentences. We adapt IUS into entity summarization by extracting named entities from each sentence. Then, the ranking score of the entity is calculated as the average of scores of all of its sentences across all

documents.

In addition, we evaluate three other variants of our approach. The first two variants involve only salience and informativeness features for learning. We denote these as *SAL* and *INF*. The third variant linearly combines all salience and informativeness features, denoted as *No-Adapt*. All are trained and predicted using the traditional RankSVM.

Evaluation Metrics. We consider the traditional information retrieval performance metrics: precisions, NDCG and MAP. Besides, we also aim to evaluate the performance of ranking in timeline summarization context, where effective systems do not just introduce relevant, but also novel and interesting results compared to the past. This was inspired by recent work in user experience of entity retrieval [GDBJ10, BML13]. One popular metric widely adopted in existing work to measure such user-perceived quality is the *serendipity*, which measures degree to which results are "not highly relevant but interesting" to user taste [ATD09]. Traditional serendipity metric was proposed to contrast the retrieval results with some obvious baseline [BML13]. In our case, we propose to use serendipity to measure the informativeness and salience by contrasting the results of one day to previous day of the same event:

$$SRDP = \frac{\sum_{e \in UNEXP} rel(e)}{|UNEXP|} \tag{4.12}$$

where $UNEXP$ is the set of entities not appearing on the previous day, and rel is the human relevance judgment of the entity. The relevance part ensures the salience of the entity, while the $UNEXP$ part ensures the informativeness of the entity over the event reporting timeline.

Assessment Setup We exclude the 39 training pairs from the overall 153 (*event*,*day*) pairs to obtain 114 pairs for testing. For each of these pairs, we pooled the top-10 entities returned by all methods. In total, this results in 3,336 tuples (*entity*, *event*, *day*) to be assessed. To accommodate the assessment, we contextualize the tuples as follows. For each tuple, we extract one sentence containing the entity from the document with the highest retrieval score (BM25), using the event as the query, and on the index corresponding to the day. If there are several sentences, we extract the longest one. Next, we describe our two assessments setups, expert-based and crowdsource-based.

Expert Assessment. To evaluate the quality of the systems, we employ an expert-based evaluation as follows. 5 volunteers who are IT experts and work on temporal and event analysis were asked to assess on one or several events of their interest. For each event, the assessors were encouraged to check the corresponding Wikipedia page beforehand to gain sufficient knowledge. Then, for each tuple, we add one more contextualizing sentence, extracted from the previous date of the event. If there is no such sentence, a "NIL" string will be presented. We asked the assessors to check the tuple and the two sentences, and optionally, to use search engines to look for more event information on the questioned date.

Then, the assessors were asked to assess the importance of the entity with respect to the event and date, in four following scales. **1**: Entity is obviously not relevant to the event; **2**: Entity is relevant to the event, but it has no new information compared to the previous day; **3**: Entity is relevant to the event and linked to new information, but it does not play a salient role in the sentence; **4**: Entity is relevant to the event, has new information, and is salient in the presented sentence. The inter-assessor agreement score for this task is $\kappa = 0.4$ under the Cohen's Kappa score.

Crowdsourced Assessment. In addition, we also set up a larger-scale assessment based on crowdsourcing. We use Crowdflower.com platform to deploy the evaluation tasks. Each tuple presented to workers consists of date, short description of the event, entity, and the sentence. Instead of directly asking for salience and informativeness of entities to event and date we decide for a simpler task: Asking the workers to assess entities in two steps: (1) Assessing whether the sentence is obviously relevant to the event (worker assess on a 3-point Likert scale, from "1-Not Relavant" to "3-Obviously Relevant"); and (2) assessing whether the entity is important in the sentence (by virtue of being a subject or object of the sentence, binary feedback).

Tasks are delivered such that tuples of the same *(event, day)* pair go into one Crowdflower job, thus the worker has a chance to gain knowledge about the event on the day and respond faster and more reliably. We pay USD 0.03 for each tuple. To maintain the quality, we follow state-of-the-art guidelines and recommendations, and receive 5 independent responses for each tuple. We create a gold standard for 311 tuples, and discard responses from workers who fail to maintain an agreement of above 70% against the gold standard. In total, we received $20,760$ responses, $8,940$ from which were qualified. The inter-worker agreement was 98.67% under Pairwise Percent Agreement, with average variance of 42%, indicating a reasonably good quality given the fairly high complexity of the task.

4.2.7 Results and Discussion

The upper part of Table 4.2 summarizes the main results of our experiments from the expert evaluation. The results show the performance of the two baselines (*TAER* and *IUS*) and of the consideration of Salience and Informativeness features in isolation with respect to precision. In general, all performances are low, indicating the relative complexity of this new task. In addition, as can be seen from this part of the table, even the approach relying on our salience features or informativeness features in isolation already outperforms the two baselines. This is due to the fact that our approach does not consider documents in isolation as the baselines do. Rather, we take a more comprehensive view considering event level instead of document level features via feature aggregation. In more details, the first baseline (*TAER*) employs a quite restricted feature set for entity ranking (e.g. document frequency), and thus fails to identify important entities event-wise.

Furthermore, the results also show the performance of the non-adaptive combination of salience and informativeness (*No-Adapt*) as well as our approach (*AdaptER*), which uses an

Method	P@1	P@3	P@10	MAP	SRDP@1	SRDP@3	SRDP@10
Ranking performance from expert assessment							
TAER	0.436	0.315	0.182	0.109	0.315	0.210	0.121
IUS	0.395	0.325	0.236	0.141	0.335	0.217	0.176
SAL	0.493▲	0.423	0.338▲	0.217▲	0.421	0.320	0.240▲
INF	0.480▲	0.436	0.354▲	0.227▲	0.441▲	0.340	0.256▲
MAX(S,I)	0.493	0.436	0.354	0.227	0.441	0.340	0.256
No-Adapt	0.503	0.461	0.320	0.225	0.396	0.338	0.215
AdaptER	**0.546**	**0.485**	**0.368**	**0.264**	**0.507**▲	**0.440**▲	**0.275**
Ranking performance from crowdsourced assessment							
TAER	0.229	0.183	0.106	0.066	0.201	0.146	0.079
IUS	0.258	0.202	0.154	0.092	0.197	0.165	0.119
SAL	0.320	0.279	0.207	0.139	0.279	0.218	0.154
INF	0.313	0.283	0.214	0.146	0.306	0.229▲	0.160
MAX(S,I)	0.320	0.283	0.214	0.146	0.306	0.229	0.160
No-Adapt	0.271	0.252	0.181	0.123	0.236	0.208	0.144
AdaptER	**0.388**▲	**0.340**	**0.237**	**0.178**	**0.361**	**0.361**▲	**0.315**▲

Table 4.2: Entity-ranking performance using different assessment settings. In each setting, significance is tested against line 1, TAER (within the first group), and line 5, MAX(S,I) (within the second group). Symbol ▲ indicates cases with confirmed significant increase

adaptive combination of informativeness and saliency. It becomes clear that an improvement by combining the salience and the informativeness features over the use of the isolated features can only be achieved by fusing the two features in a more sophisticated way: *No-Adapt* does not perform better than the maximum of *SAL* and *INF* ($MAX(S, I)$), it even performs worse in under some metrics such as P@10. In contrast, *AdaptER* clearly outperforms the maximum of *SAL* and *INF* as well as its non-adaptive version for most metrics. For instance, we achieve 16% improvement of MAP scores as compared with the $MAX(S, I)$.

Besides precision, we also consider serendipity (SRDP) as a complementary meassure in our experiments, as discussed above. This metric measures how likely the approach brings unseen and interesting results to the user. Under SRDP, our approach outperforms significantly both the baseline and the maximum of *SAL* and *INF*. We achieve 14% improvement of serendipity at top-1 entities, and 29% at top-3 entities. Thus, our top-retrieved entities do not only cover relevance, but are also more interesting, often unseen on the previous day (contributing to more informative results)

The lower part of the table 4.2 shows the same results for the crowdsourced assessment. The same trends of performance can be observed, where our approach outperforms in all the metrics. In comparison to the expert evaluation, the results are overall lower. A possible

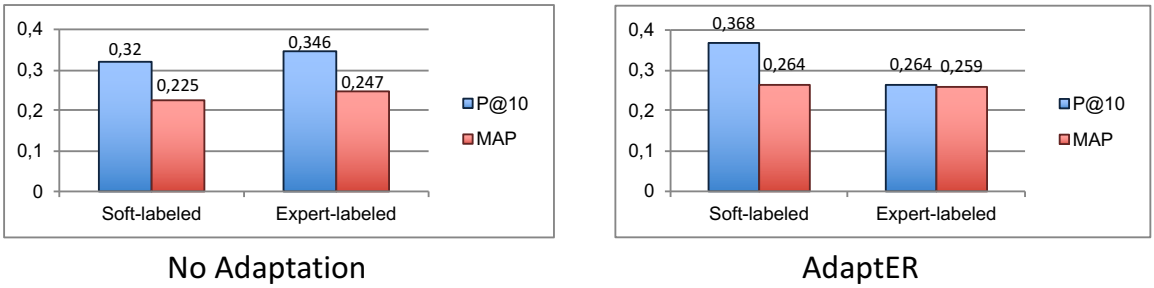


Figure 4.3: Performance of systems with(-out) soft labeling

reason for this is the complexity of the crowdsourcing task, which requires knowledge about the considered event in order to give high quality feedback (see expert assessment setup, section 4.2.5). Nevertheless, the adaptive model is still able to achieve significant gain, especially under the serendipity measurement.

Next, we evaluate the effectiveness of soft labeling in covering salience and informativeness. For this purpose, we manually annotate entities obtained from the reporting timeline of 4 training events, with respect to the salience and informativeness (see parameter settings, section 4.2.5). We then re-train both non-adaptive and adaptive models using this annotated data (supervised approach). Figure 4.3 shows the precision and MAP scores of the supervised approach in comparison to our soft labelling-based approach. The similarity in performance between them, regardless of the models they have been used for, confirms that our soft labelling properly captures both salience and informativeness.

Feature Analysis. Analyzing the influence of different feature groups (see section 4.2.4) can give insights into what factors contribute to the entity ranking performance. To study the feature impacts, we do an ablation study and remove incrementally a group of features, and re-evaluate the performance using the expert assessment. Since reducing the feature dimensions directly affects the query features and its adaptive scores, to ensure the fair comparison, we perform the study for the non-adaptive setting. Figure 4.4 shows the MAP scores of ablated models. The yellow arrows indicate a group of feature with significant decrease in comparison to *No-Adapt* model (with full features), and thus implies the high influence of the corresponding feature group. From the table, we can see that the most influential feature groups include context importance feature (salience featurs), and informativeness feature group of context level.

Anecdotic Example. In table 4.3, we show one example of top-selected entities for the event "Boston marathon bombing 2013". Additionally, we show some selected sentences covering the entities, to enable the understanding of the entities' roles within the event on the presented days. As can be seen, the timeline corresponding to *TAER* approach (upper part) gives more salience credits to entities frequently mentioned throughout the news (such as Boston marathon), keeping them in high ranks throughout the timeline. The approach is not responsive to less salient but interesting entities (such as Pope Francis, a rather unrelated entity to the event, but get involved via his condolences and activities to victims of the

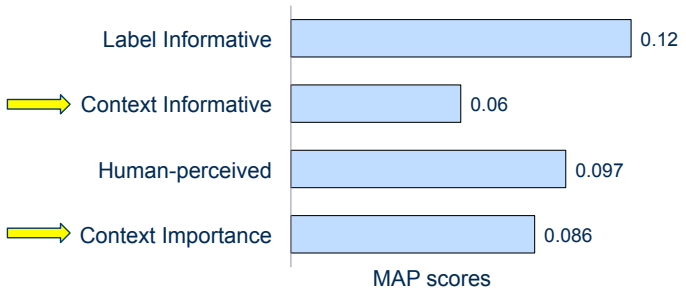


Figure 4.4: MAP of No-Adapt when feature groups are ablated (full-feature MAP score: 0.225)

April 15	*April 16*	*April 17*
Boston Marathon **Mass General Hospital** **Boston.com**	**Boston** **Boston Marathon** **Vatican**	**Boston Marathon** **Boston** **Boston University**
- Two bombs exploded near the finish of the Boston Marathon on Monday, killing two people, injuring 22 others - At least four people are in the emergency room at Mass General Hospital	Deeply grieved by news of the loss of life and grave injuries caused by the act of violence perpetrated last evening in Boston, His Holiness Pope Francis wishes me to assure you of his sympathy . . .	- FBI confirmed that pressure cookers may have been used as explosive devices at the Boston Marathon. - The third victim was identified Wednesday as Boston University graduate student Lingzi Lu.
Boston Marathon **Marathon Bruins** **New York City**	**Pope Francis** **Vatican** **Boston Marathon**	**FBI** **Boston University** **Lingzi Lu**
- Two bombs exploded near the finish of the Boston Marathon on Monday, killing two people, injuring 22 others - The NHL postponed the Boston Bruins' Monday hockey game due to the bombing	The Vatican sent a telegram to Boston Cardinal on Tuesday, in which Pope Francis expresses sympathy for the victims of the marathon bombings. . .	- FBI confirmed that pressure cookers may have been used as explosive devices at the Boston Marathon. - The third victim was identified Wednesday as Boston University graduate student Lingzi Lu.

Table 4.3: Examples of top-3 entities on Boston Marathon Bombing 2013 using TAER (top) and AdaptER(bottom) for April 15-17

bombing). On the other hand, using an adaptive ranking with informativeness incorporated, the resulting entities are not just more diverse (including related events such as Marathon Bruins), but also expose more new and emerging information.

4.3 Decision Summarization by Recurrent Neural Network

In the previous section, we discussed the text summarization in timeline fashion for news events. The main criteria for a high-quality summary is the new and salient information present in the news documents, which are encoded via different hand-crafted features. In this section, we turn to different type of summary: Decision summary, in which the major steps in the human decision making processing are identified and extracted to build a summary. Compared to the previous task, which is somewhat traditional, this task requires different approach, both the building a useful corpus and in developing a deeper model that does not rely on the hand-crafted features. The choice of decision summarization study is due to the fact that decisions are an important part of our daily lives, and the ability as well as process to make decisions are unique for humankind [HA15]. Summarizing the essential parts of the decision making thus enables insightful understanding of the dynamics of such cognitive process. In order to study the decision making process, we develop a new annotation scheme for decision elements based on the concepts from the field of Decision Analysis [5]. Specifically, we consider spoken conversations among humans, and seek to identify mentions of alternatives considered and criteria expressed within. We release the annotated corpus to enable further research in this line.

4.3.1 Introduction

From online discussions to corporate meetings, multiparty dialogues represent an essential communication channel to exchange ideas and for collaboratively making decisions. The demand for automatic methods, that process, understand and summarize information contained in audio and video recordings of meetings, has been growing rapidly, as shown by the projects focused on this goal, (e.g., [CAB+05, JAB+04]), among others.

User studies [BRR05, LPBA04] show how decisions are one of the essential pieces of information that users expect to retrieve from a meeting summary or meeting notes. Recently, many studies concentrated on meeting summarization [MH03, JZCF07, MH06, Gal06, XLL08, MH05], and some on decision extraction from meetings [FFE+08, FHBP09, BP10]. However, to the best of our knowledge, little effort has been done on the automatic analysis of the decision making process as formalized in Decision Analysis.

In multi-party dialogues, while some decisions (often very strategic ones) are framed through controlled and structured approaches (e.g. facilitated by consultants), many other are informally discussed among relevant people, whether full stakeholders in the decisions or advisers to the main decision maker. Our research objective is to leverage transcripts (or chat exchanges) from those discussions about decisions to extract the following **decision elements**: (i) the decision topics, (ii) the proposed alternatives, (iii) the preferences discussed, (iv) the constraints mentioned. In addition, we are interested in keeping track of the source of each mention (who said what). Altogether, this enables a structured summary of

[5] https://en.wikipedia.org/wiki/Decision_analysis

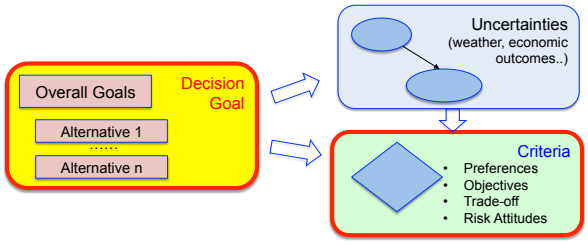


Figure 4.5: Decision Analysis Major Components

the decision discussion (textually or graphically). We call this summarization process *Decision Gisting*. One of the benefit of such a summary would be to remind a decision maker of the arguments that were raised to help him/her make a choice at a later time. A longer term application of the decision gisting process is to make decision analysis more broadly accessible by making use of the decision gisting output to facilitate a formal mathematical model of the decisions.

This section reports our first efforts in extracting from natural language conversations the decision gists (i.e. a chunk of text pertaining to a specific decision element). We describe here the development of a novel annotation scheme, the annotation process and preliminary results using supervised learning methods. While some studies have focused on decision extraction from meetings [FFE+08, FHBP09, BP10], to the best of our knowledge, no effort has been done towards a formal representation of Decision Analysis Concepts in multiple-party, human-human form of communications.

Our contributions

Our contributions are two-folds. Firstly, to have a sound annotation scheme for decision gist, we collaborated with Decision Analysis experts in annotating the AMI corpus[CAB+05], a standard multimodal meeting dataset. We developed a novel annotation scheme to cover the different decision elements related to Decision Analysis concepts. The annotation process has been done in both expert-based and crowdsourced fashion, and the resulting annotated meetings will be made available to the public.

In addition, we solve the decision identification problem as a binary sequence prediction task, where given a sequence of words, a classification model outputs the likely class label for each word, which is either a 'decision' or not, depending on whether the word is a part of a manually identified decision or otherwise. To train a supervised classifier for this task, we employ the recurrent neural network (RNN) framework. A limitation of the RNN is that the context provided by the word vectors [MSC+13b] themselves is too fine-grained. While this fine-grained context information proves useful for tasks, such as POS tagging and NER, in the case of decision analysis, the context rather depends on the semantics of larger units of text, e.g. sentences, that precede the constituent words of the current sentence. Our key idea is to make use of pre-trained document vector embeddings [LM14] and feed them as inputs to an RNN for learning the prediction model.

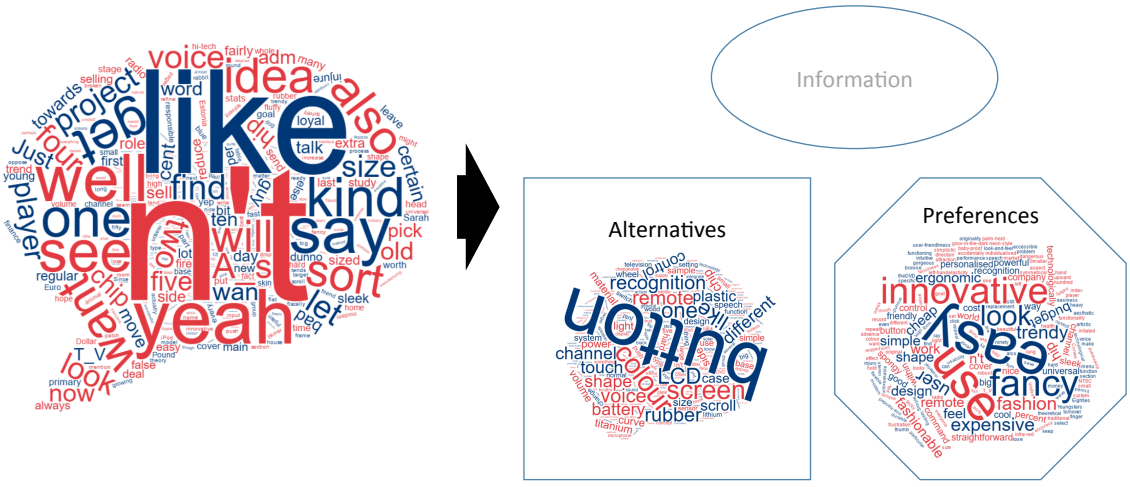


Figure 4.6: Decision Analysis Major Components

4.3.2 Decision Analysis Concepts

Before discussing our method, we give the brief introduction to the theory of Decision Analysis. Further details can be found at work by Ron Howard et al. [HA15]. In general terms, Decision Analysis is a termed coined by Ron Howard to describe research and consulting efforts related to helping decision makers make rational decisions under uncertainty. Decision analysis builds upon Expected-utility decision theory [6] which is a normative framework for making decisions in the face of uncertainty. It is based on a set of axioms that lead to the maximization of the expected utility. An important characteristic of expected-utility decision analysis is that it is designed for a single decision maker (or a group of decision maker with homogeneous beliefs and preferences). One of the focus of decision analysis is on practical approaches and challenges to rigorously making decisions. In particular, greater attention is given to the modeling and framing of the decision (definition of the utility function including criteria, risk attitudes and trade-offs, definition of the relevant uncertainties, investigation into the benefits of gathering additional information) rather than on the mathematical task of solving a well-defined problem. In order to achieve this, the decision analysis process needs certain inputs. Such inputs are divided into three categories:

- **Alternatives**: options available to the decision maker, in other words *what could be* (represented in a rectangle in Fig. 4.6);
- **Preferences**: values, risk preferences, and time preferences of the decision maker, in other words *what should be* (represented in octagon in Fig. 4.6) and
- **Information**: models and probability distributions of the relevant uncertainties (usually represented by oval nodes)[HA15].

[6] https://en.wikipedia.org/wiki/Expected_utility_hypothesis

.

Our research aims at automatically extracting Alternatives and Preferences, hereafter *Dec-A* and *Dec-P*, from text conversations where decisions are discussed.

4.3.3 Corpus and Annotation

The AMI (Augmented Multi-party Interaction) Meeting Corpus is a English multi-modal data set consisting of 100 hours of meeting recordings [CAB+05]. The dataset is derived from real meetings, as well as scenario-driven meetings, designed to elicit several realistic human behaviour. We selected 17 meetings, having some form of decision making in them, as in [FFE+08] (statistics in Table 4.4). We use the manually annotated transcripts for this study.

# Meetings:	17
#Tokens:	234, 607
#Sentences:	22, 903
#Utterances:	1, 193
# Median of tokens per utterance:	65

Table 4.4: AMI dataset general statistics.

4.3.4 Hybrid Annotation Process Overview

Our aim is to annotate *Dec-A* and *Dec-P* in the AMI corpus transcriptions, according to the definitions of the Decision Analysis theories. Fig 4.7 shows examples of *Dec-A* and *Dec-P* chunks in a meeting excerpt. While previous works label the entire utterances or sentences [FFE+08, SRW07], we target the phrase level, i.e., we label sequences of tokens

So first thing is we need power source for the remote control.
So I was of the idea that we can have two kind of power supplies, one is the usual batteries which are there, they could be chargeable batteries if there's a basis station kind of thing and on top of that we can have solar cells, when the lighting conditions are good they can be used so it'll be pretty uh innovative kind […]
Then uh we need plastic with some elasticity so that if your if the remote control falls it's not broken directly into pieces, there should be some flexibility in t I guess that fits in with the spongy kind of design philosophy[…]
So there should we should think of something like that and then it should be double curve.
The s science for the ease of handling and there are some other issues why we need double curve.

Figure 4.7: Example *Dec-A* in blue and *Dec-P* in purple

within one sentence. This approach gives more fine-grained annotations, but it is challenging for labellers, in particular if using crowdsourced annotations. One option is consulting decision analysis experts for the entire data; which guarantees high-quality annotations but is costly.

Therefore, we employ a *hybrid* annotation process that exploits both domain expert-knowledge and crowd-sourcing. We first develop an in-house annotation system for domain experts to annotate a subset of the data (**Phase 1**). Then, we design crowdsourced annotation tasks for *Dec-A* and *Dec-P* for the entire dataset, using domain-expert annotations as quality control (**Phase 2**). Finally, annotations made by the crowd are reviewed by domain experts (**Phase 3**).

Data pre-processing

We started with segmentation of the conversations. Segmenting long documents into smaller chunks is crucial in annotation tasks, especially in crowdsourcing setup, because crowd-sourced workers are normally engaged with a sentence or small paragraph [SBDS14]. On the other hand, the decision annotation is sensitive to anaphora in the document, and to the context in general, so the segmentation can result in "breaking the context", or cross-chunk anaphora, thus reducing the quality of annotation. To address this issue, we relied on the following heuristics: We conduct some pilot annotations to estimate the size of text chunk of each meeting. Then we process the DAs of the meeting transcripts (available in AMI) and segment the meeting according to the estimated size, and some rules to include the correlated DAs (e.g. a response).

Phase 1. We randomly selected 75 segments of transcripts for domain-expert annotation. Three domain experts annotated the segments for *Dec-A* and *Dec-P* and we obtained 3081 annotated chunks in total (including overlaps between experts). The inter-annotator agreement using Fleiss' Kappa was k=0.41[Fle71]. Those annotations were used for quality control in Phase 2. Fig. 4.8 shows the interface developed for the domain expert annotation.

Phase 2. We used the crowdsourced annotation platform CrowdFlower[7], CF. We designed two separate tasks, one for *Dec-A* and one for *Dec-P* annotation. Separation has been shown to reduce the crowd-workers' cognitive load and enhance both engagement and response quality [BDR17]. Figure 4.9 shows the crowdsourced annotation system. The annotators were presented with the conversations in natural language format, and were asked to freely highlight any phrases that were considered to be *Dec-A* and *Dec-P* following detailed guidelines and examples summarized in Table 4.5. Annotators in CrowdFlower are ranked according to a score (level 1 being the more expert and level 3 being the newbies). We selected annotators of level 2, from English speaking countries, and we have at least 3 annotators annotating each segment. As quality control setting, we forced any annotator to stay at least 10 seconds on each segment. In addition, CrowdFlower provides a quality control system based on test questions. Test questions are gold annotation hidden among the data to label.

[7] www.crowdflower.com

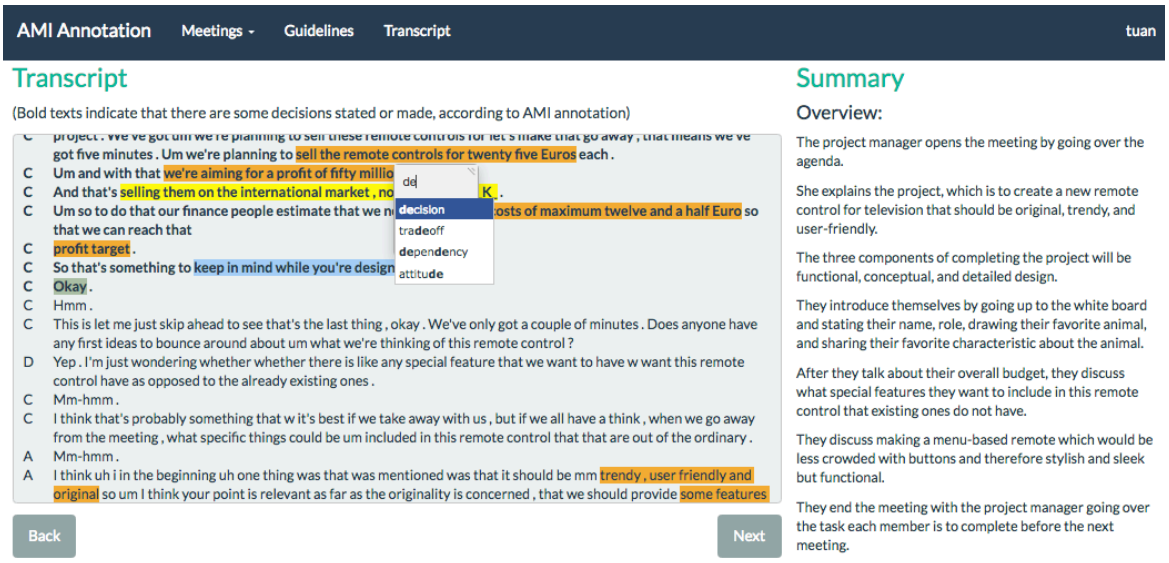


Figure 4.8: Domain-expert annotation interface

Alternatives	"...Select chunks of contiguous text where the speaker expresses what he/she wants. or could do. Eg. *A power in wood or plastic ...*"
Preferences	"...Select chunks of contiguous text where the speaker expresses how he/she evaluates. the options. Eg. *The plastic one is cheaper ...*"

Table 4.5: Excerpt of the annotation guidelines

Each annotator has to maintain a certain level of accuracy (70% in our case) on the test questions throughout the task, in order to continue annotating. We selected the test questions from the annotations obtained in Phase 1. At the end of this phase, we obtained 558 responses for *Dec-A* and 749 responses for *Dec-P*, with a IAA of 0.546 Fleiss' Kappa for *Dec-A* annotation task and 0.315 for *Dec-P* annotation task.

Phase 3. Finally, a domain expert revised all the annotations that passed Phase 2 quality controls. The expert confirmed the correct annotations, rejected incorrect annotations and gave suggestions on the phrase windows. Since only the positive-labeled annotations were revised (which are much smaller than the total number initial of phrases), we reduced the domain expert time needed for examining all annotations. From the responses in phase 3, 335 are confirmed as *Dec-A* and 249 are confirmed as *Dec-P*.

These steps allow for a double level of quality control: only the high quality annotations are retained in phase 2, and they are further revised in phase 3. This process allows us to exploit the crowd and reduce the time and the cost of the domain expert consultancy.

4.3.5 Summarization Methodology

Having defined the training data, we now move to discuss our methodology for the decision summarization. Given a meeting transcript, our summary is extracted by identifying the phrases that deem relevant to different decision elements described in Seciton 4.3.2. As discussed above, we model this task as the binary sequence prediction task, where for each type of decision elements, we analyze the transcripts in sequence of utterance presence, and to predict whether each utterance contains the information of that type. It is not trivial to manually design features to capture the information of the utterances with respect to the decision elements, since this would require deep expert knowledge of both linguistics and decision analysis, which is very hard to achieve in practice. Instead, we turn to the neural network-based techniques that can automatically learn the high-order features from the raw inputs via *hidden layers* of the networks. Furthermore, such feature design of one utterance depends on the presence of the previous utterances, since in multi-party dialogues such as meetings, participants tend to respond to previous points made by other participants, and decisions are often made through an iterative and collaborative manner. To capture this dependency, we design a *recurrent neural network* architecture, described below.

A recurrent neural network (RNN) architecture [RHW+88] consists of a standard neural network where the current output not only depends on the current input but also on the sequence of previous inputs to the network, referred to as the *memory*. This ability of the RNN to take into account the contextual information enables it to effectively model tasks where this contextual information plays a crucial role, such as POS tagging [WQS+15], named entity recognition [LBS+16, MH16], predicting the next query in a search session [SBV+15] etc. More concretely, given a sequence of inputs $(x_1, \ldots, x_T)$, an RNN computes a sequence of outputs $(y_1, \ldots, y_T)$ by Equation 4.13, where W^{hx} is the weight for feeding the input x_ts into the recurrent layer and W_{hh} is the weight associated with the previous

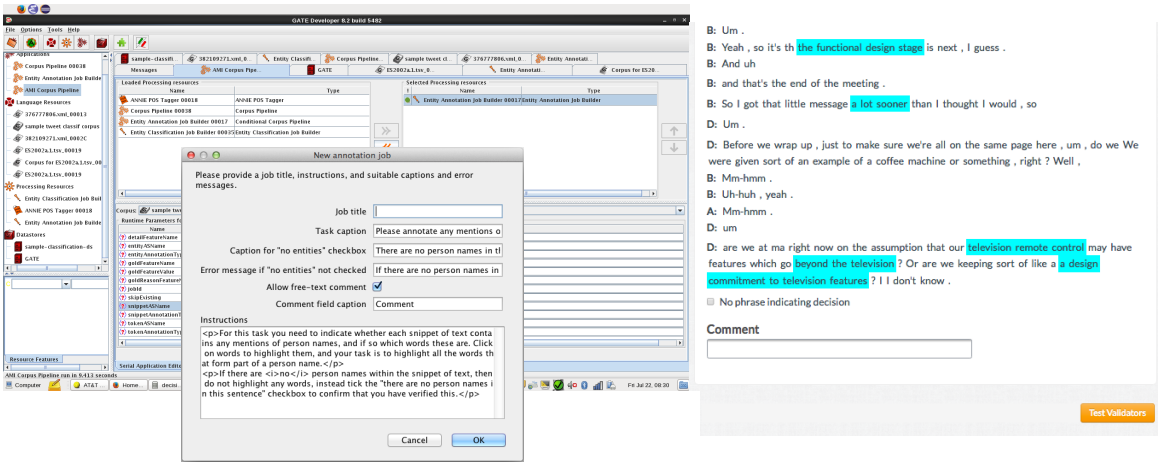


Figure 4.9: Snapshot of our extension to GATE Crowdsourcing Plugin (left) and an annotation task (right)

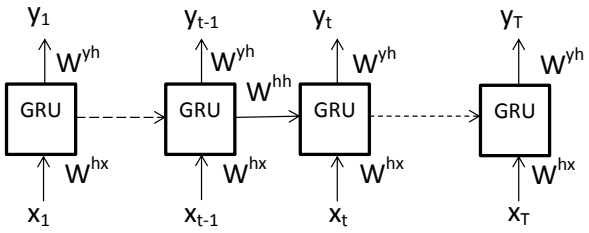


Figure 4.10: Time unfolded view of an RNN with GRU units.

input x_{t-1}, as shown in the schematic diagram of Figure 4.10.

$$h_t = \sigma(W^{hx}x_t + W^{hh}x_{t-1}), \quad y_t = W^{yh}h_t \tag{4.13}$$

Each cell in Figure 4.10 is a gated recurrence unit (GRU). GRUs enhance the modeling power of RNNs by controlling the amount of memory content exposure [CGCB14].

In fact, the architecture above resembles the Long shor-term Memory models, except that each hidden cell is trained using GRU instead of the memory and adaptation gates. We now describe how we apply the architecture as shown in Figure 4.10 for solving our problem of predicting decisions. A standard approach is to feed in the individual word vectors, pre-trained with an unsupervised approach such as word2vec [MSC+13b] into the network as the x_t vectors, so that during learning the model is able to take into account the context at the level of word vectors. We hypothesize that the context at the level of words does not provide enough semantic information to train the model effectively. In fact, as already observed in [HM07, FFE+08], utterances from previous speakers and speaker roles provide important cues to solve this problem.

To address the requirement of a broader context from a deep learning perspective, we propose to use pre-trained document vectors (called paragraph vectors [LM14]), instead of word vectors, to input into the RNN. The key idea of the paragraph vector, as proposed in [LM14] is to encode the contents of a higher unit of text, say a sentence, as an additional vector in the wor2dvec RNN such that the vector representation of the sentence is similar to those of its constituent words.

4.3.6 Experiments

Next, we discuss the experiments to evaluate the effectiveness of our decision summarization method. In our baseline approach, we feed in word vectors (pre-trained by 'word2vec' [MSC+13b]) to the RNN architecture of Figure 4.10. The labels in this case are associated to each word as obtained from the annotations. Evaluation of the baseline approach, dubbed as 'WVEC-RNN', is carried out at the granularity level of words, which is the standard way to evaluate sequence prediction tasks, such as NER [MH16, WQS+15]. Since the input unit for our proposed approach of sentence vector based RNN (which we call SVEC-RNN) is a sentence, the evaluation had to be carried out at the level of sentences. To ensure a fair comparison between WVEC-RNN and SVEC-RNN, we also train and test WVEC-RNN

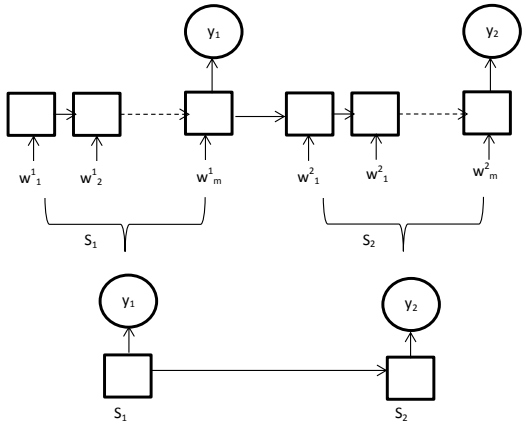


Figure 4.11: Masking WVEC-RNN with sentence labels (top). SVEC-RNN is shown in the bottom.

Embedding Units		Evaluation measures			
Approach	Evaluation	Acc.	Prec.	Recall	F-score
WVEC-RNN	Words	0.3600	0.3314	0.3961	0.3609
WVEC-RNN	Sentences	0.5533	0.5614	0.5422	0.5516
SVEC-RNN	Sentences	**0.6875**	**0.6321**	**0.7578**	**0.6893**

Table 4.6: Comparative evaluation of decision alternative prediction.

using the labels at the granularity level of sentences, i.e. by applying *masking* to enable only one output label for each sentence. This is schematically shown in Figure 4.11. Note from Figure 4.11 that the labeling scheme for SVEC-RNN is similar to that of WVEC-RNN trained on sentence labels, the only difference being in the number of vectors that are used as inputs to the RNN during training and testing.

The document and the word embeddings were obtained with identical settings, i.e. with distributed memory model [LM14] (similar to the continuous bag-of-words model for word2vec), which takes into account the word ordering, the number of dimensions being 200 and window size being 5. To capture domain specific semantic relations corresponding to the meeting scenarios, the word and document vectors were trained on the AMI corpus, instead of using an external corpus to obtain the embeddings. The dimensionality of the hidden layer in the GRU cell was set to 5. We used stochastic gradient descent as the learning algorithm with L2 regularization parameter set to 0.001 and a softmax function at the output layer of the GRU cell. The models were trained and evaluated with a 4:1 train-test split on the corresponding datasets, i.e. at the level of words and sentences.

Results. Table 4.6 presents the results of our experiments. It can be seen from the first row of Table 4.6 that the standard word vector-based approach to sequence label learning produces poor results (F-score of 0.3609), which demonstrates that the problem of identification of decision alternatives from the text is a challenging one. The second row of Table

4.6 describes the scenario when the word-vector based approach is trained using masked sentence labels It can be seen that the model performance improves with respect to the more fine-grained word labels.

The third row of Table 4.6 shows the results with the SVEC-RNN approach. It can be seen that the results produced with the SVEC-RNN approach are significantly better than the WVEC-RNN approach, making use of sentence based labels, in terms of all standard evaluation metrics. In particular, the increase in recall is higher than that of precision, which suggests that the approach is able to identify a higher number of decision alternatives from meetings.

4.4 Chapter Summary

In this chapter, we have studied the problems of text summarization, as one of the application types that benefit from temporal as well as cognitive models. We have conducted two independent studies; each of which focus on one model category. For temporal models, we studied the problem of timeline summarization of news, while for cognitive models, we studied the decision summarization problem using deep learning techniques.

For timeline summarization of high-impact news events, we have proposed a novel method that use entities as the main unit of summary. We propose to dynamically adapt between entity salience and informativeness for improving the user experience of the summary. Furthermore, we introduce an adaptive learning to rank framework that jointly learns the salience and informativeness features in an unified manner. To scale the learning, we exploit Wikipedia page views as an implicit signal of user interest towards entities related to high-impact events. Our experiments have shown that the introduced methods considerably improve the entity selection performance, using both small-scale, expert-based and large-scale crowdsourced assessments. The evaluation also confirms that integrating salience and informativeness can significantly improve the user experience of finding surprisingly interesting results.

For decisions summarization, we address the problem of identifying decision alternatives from meeting transcripts by treating it as a sequence labeling task. In contrast to previous approaches which rely on manually designed features, we propose a deep learning based solution to the problem with the application of GRU based RNN. We hypothesize that it may potentially be beneficial to model the decision alternative identification problem by capturing broader semantic contexts of sentences rather than that of the fine-grained context of words. Therefore, contrary to the standard approach of inputting pre-trained word vectors (as in POS tagging and NER), we train RNNs with pre-trained sentence vectors. Experiments on crowd-source AMI corpus shows that the coarse-grained context of sentences significantly improves the effectiveness of decision alternative identification in comparison to standard word vector based approaches.

Future Direction. As the approaches discussed in the chapter are novel, there are several promising directions to explore for future work:

For the timeline summarization, we aim to investigate further the impact of adaptation in different types of events, using larger and more diverse sets of events. Furthermore, we plan to study more advanced ways to mine Wikipedia temporal information as signals of collective attention towards public events. We are planning to use this for further improving our VOR measure for the soft labeling approach. Other direction includes investigating a deeper model to improve the performance of current entity timeline summarization systems.

For the decision summarization work, we would like to extract other decision analysis related concepts, such as extracting the decision alternatives (what it could be) and criteria (what it should be). This would eventually be useful to automatically construct relevant decision summaries, as what the Decision Analysis theorists envisioned.

5 Recommendation Using Temporal Graph Learning

In the previous chapter, we show text summarization can benefit from timeline-based summarization methods, as well as memory-based deep learning methods. In this chapter, we study another area of application that also profits from modelling of temporal topics: Information re-finding in semantic space, consisting of entities, semantic relationships and activity logs. We study several datasets to examine the effects of time and cognitive models into the ranking, combining semantic data and cognitive sources does not limit to text data, but can be seen in other types of data as well. We frame our study to a particular application: How to reduce information overload to workers by automatically focusing on important documents, adaptive to the business tasks at hand.

5.1 Introduction

Recommendation in computing systems refers to techniques, software tools, methodologies that provide suggestions for the items to be of interest to the user [RV97, RRS11]. The output is typically a list of suggested items, optionally ranked. Recommender systems can be developed in two ways, collaborative or content-based filtering. In collaborative filtering, the users' past behavior is modelled for the recommendation decision [KB15]; while in content-based filtering, characteristics of the items are used as the main source [PB07]. These two approaches can also be combined to create hybrid recommender systems. Due to the explosive rate of digital contents generated every day, recommendation has become one of the major solutions to help users deal with information overload. Thus, there has been significant amount of research work in recommender systems in recent years, benefiting several successful online platforms as Amazon, Netflix, Youtube, etc.

Traditional recommendation methods tend to see problems through the behavioral perspective, as well as the computational perspective, where user experience or feedback are observed as response to stimulus from the context or environment. Recently, there is also a trend in studying recommendation via the cognitive perspective, which does not observe

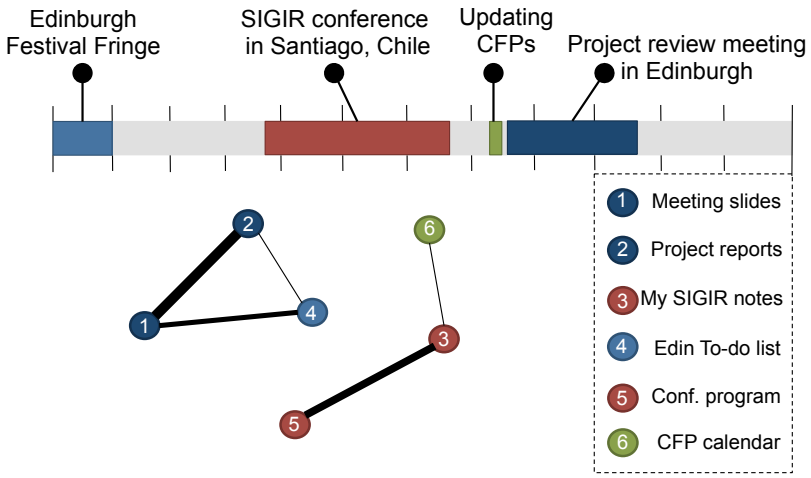


Figure 5.1: Recommended documents in a personal information management system: The upper bar corresponds to the timeline of tasks. Bottom right is the recommended documents. User accessed the documents when conducting the tasks (displayed with the same color). Graph on bottom left represents the document relations. Width of lines indicates the strength of the connections. The forgotten document 4 resurges due to its relevance to the current task.

but also attempt to explain how the way people perceive, learn and make decision can help computer make suggestions [TKS+16, JWF+15]. Under this new direction, this thesis aims to study, in the temporal dimension, how insights in human remembering and forgetting information can help improve the recommender system performance. We choose a special kind of applications: recommendation in personal desktop, where the human remembering and forgetting can be modelled with fewer effects from external sources and events. We also target semantic desktop environment [SvED07], where structured information about the documents are well maintained by humans, thereby less demand for the content analysis methods and more focus can be made to the user modelling part.

5.2 Task-Aware Document Rankings in Digital Workplaces Using Forgetting Models

The technology-driven increased ease of content creation and sharing and the reluctance to delete content leads to very large and continuously growing information spaces in the personal as well as in the organizational realm. Thus, finding or re-finding a resource important for a task at hand in an information space such as a desktop computer, a digital camera, a mobile device or an intranet portal becomes increasingly difficult and tedious. In such non-public sources, navigation is a preferred mechanism to find a document for a task at hand, due to its lower cognitive load, its consistency and the strength of the location metaphor [BBMN+08]. However, the task becomes increasingly challenging, due to the advance of ubiquitous technologies that support the idea of massive digital content cre-

ation and sharing, leading to a growing number of data, and making finding and re-finding a particular resource increasingly difficult. The mission is more frustrating in large heterogeneous information ecosystems such as in different devices, or in systems that get cluttered after long-time working on different tasks.

Graph Ranking using Managed Forgetting. For decluttering such information spaces and supporting the finding or re-finding of resources according to the user's short-term interest, we propose a graph ranking approach to ease the navigation, based on the idea of *managed forgetting*. The idea is inspired by science of forgetting and remembering [Log11, NKGL15]: The human brain is very effective in focusing on important things, while forgetting irrelevant details. This trait is reflected in human practices of organizing their collections, i.e., they often create shortcuts to easily navigate to relevant resources in the desktop environment or mobile home screens, or bookmark important Web pages. Managed forgetting aims to relieve such manual efforts by automatically computing the *short-term* importance of a document with respect to the user attention. It replaces the binary decision on importance by a gradually changing value: Information sinks away from the user with a decreasing value, which we call *memory buoyancy*. This value can be used for ranking important resources for a task at hand, thereby decluttering the information spaces adaptively.

As in human forgetting, memory buoyancy is driven by resource usage, importance decay, and semantic associations [AS91]. In Information Retrieval, different algorithms and systems have been proposed to identify important documents in an information space [DCC+03, PKHN11]. Many existing approaches rely on activity logs and assumes that recently accessed documents are also more likely to be accessed in the future [WBD10]. As a result, many proposed algorithms assess documents based on recency and frequency evidences of access. They ignore a wide variety of other factors, which equally influence the importance of a document for a task at hand. For instance, according to the associative character of the brain, a document might be important because of its relatedness to the current task, even if the document has not been touched for a long time [AS91, RCD+03]. In the example shown in Figure 5.1, while preparing for a business trip to Edinburgh, the user might recall or might have forgotten useful notes from her private holiday in Edinburgh some years ago. Ideally, a system should bring this information up again, but it is infeasible when only relying on the activity history alone. This example demonstrates the need for a more comprehensive ranking method, taking into account the intrinsic relatedness of documents to current tasks.

Main Ideas. Our method combines evidences from activity logs with semantic associations between documents to devise a unified document ranking framework. The idea is that a document is important to the user's current task, if it either has been frequently accessed by the user, or is highly related to other important documents. This is illustrated in Figure 5.1 for six documents. The user's intensive accesses to "Meeting slides" and "Project reports" during the preparation for the project meeting in Edinburgh (dark-blue part of the upper horizontal bar) give the documents higher ranks. Meanwhile, the connections between these documents and the "To-do list" from a past trip to Edinburgh endorses this list to the current

task, bringing it back to the user's attention.

The importance of considering document relationships to rank documents, e.g, in personal information management (PIM) systems, has been shown in [SG05, CGWW09]. The common idea is to propagate the "importance score" of a document to other related documents in a graph of different document relationship. However, most of existing work studies the document relations in isolation, assumes they are equally important, or puts arbitrary weights to the relations in an ad-hoc manner [SG05]. In contrast, we propose a unified framework based on machine-learning methods. We conduct studies in different settings, and systematically evaluate the effectiveness of our framework from the quantitative to qualitative perspectives.

5.2.1 Methodology

In this section, we describe our approach to managed forgetting, which exploits document's access information, and subsequently applies different forgetting (or decay) functions as well as propagates the importance of related documents via semantic relationships.

Preliminaries

A semantic information space in a PIM system is a collection of documents or resources, which is denoted as D. A document or a resource d can be of different types (e.g., photo, office document, folder, web page, etc.) and has different attributes (e.g., title, authors, and creation time). Between any two documents d_1 and d_2 can exist multiple relations with different semantics. For instance, d_1 and d_2 are both created by the same author, or d_1 is the containing folder of the file d_2. Relations can be associated with some scores indicating the strength of their relation, for instance, the cosine score for content similarities. Let R denote a set of all semantic relations. For each pair (d_i, d_j), we have an $|R|$-dimensional vector $X_{ij} = (x_{ij1}, x_{ij2}, \ldots, x_{ij|R|})^T$, where $x_{ijk} \geq 0$ represents the score of the k-th relation between d_i and d_j, $x_{ijk} = 0$ if the d_i and d_j are not connected by the relation. Usually, the number of relations is small compared to the number of all documents in the information space. The collection of relation scores $X = \{X_{ij}\}$ forms the weights of edges in a multigraph, where nodes are documents in D, and each edge represents a semantic relation.

In our work, we model time as the sequence of equal time intervals and denote by $T = (t_1, t_2, ..., t_m)$, where the time point t_i is the index of i-th interval from the beginning of the PIM system. In one interval, a document can be accessed and used by one or multiple users. Each access is represented by a triple $a = (d, u, t)$, indicating that the user u performed an action on the document d at time t. Given a user u (or a group of users $U = \{u_i\}$), a document d and a time point of interest t, the sequence of actions on all documents of D, performed by u (or in the case of U, by at least one user u_i), happened before t and in chronological order, forms an activity history of u (or U) in the information space, and is denoted by $L_t = (a_1, a_2, ..., a_n)$. Given a document d and time t, we refer to as a document access history, denoted by $L_{d,t}$, as those actions performed on d. A sequence of time points

Method name	Function	Parameters
Most Recently Used	*MRU*(d,t)=$\frac{1}{t-t_d+1}$	None
Polynomial Decay	PD(d,t)=$\frac{1}{(t-t_d)^{\alpha}+1}$	α: Decay rate
Ebbinghaus Curve	Ebb(d,t)=$e^{\frac{(t_d-t)}{S}}$	S: Relative memory strength
Weibull Distribution	Wei(d,t)=$e^{-\frac{\alpha(t-t_d)^s}{s}}$	s: Forgetting steepness, α: Volume of what can be remembered

Table 5.1: List of Activity-based ranking functions

t_i for actions in $L_{d,t}$ (can be repeated because of multiple accesses to d within one interval) constitutes the access times of d, denoted by $T_{d,t}$. The most recent access time to d before t (last time point in $T_{d,t}$) is denoted by t_d.

Problem. Given a collection D, a set of relation scores X, time of interest t, and an activity history L_t corresponding to a user u, or to a group of users U, identify documents with the highest importance with respect to u's or U's tasks at time t, as inferred from L_t.

We tackle the aforementioned problem in two steps. In the first step, we mine the activity history and devise a memory buoyancy scoring function based on the recency and frequency (see section 5.2.1), so that more recently and frequently accessed documents get higher memory buoyancy scores. In the second step, we employ a *propagation* method that identifies highly connected documents, to transfer the activity-based scores of documents along the connection. This is similar to the layered approach by Kawase et al. [KPHN11]. However, while the authors merely identify connections from sessions of the activity history, we devise a generalized framework that works with different heterogeneous relations.

Memory Buoyancy: Learning from User Activities

In order to compute the memory buoyancy scores, we use the access times of the document from the access history. We estimate the score of a document through the distances of previous access time points and the time of interest.

Definition 1 *An activity-based memory buoyancy scoring function is a function that takes as input the time t and document d, and outputs a value $v(d,t) \in [0,1]$ (memory buoyancy score) such that:*

1. $v(d,t) = 0$ *if* $T_{d,t} = \emptyset$
2. $v(d,t_{i+1}) < v(d,t_i)\ \forall t_i, t_{i+1} \in T_{d,t}$
3. $v(d_1,t) < v(d_2,t)$ *if* $|T_{d_1,t}| < |T_{d_2,t}|$ *or* $t_{d_1} < t_{d_2}$

The above conditions ensure that the memory buoyancy scores, if no other evidences present, is driven by the decay effect. In Table 5.1, we present different activity-based scoring functions studied in this work, each corresponds to one decay function. Each of these functions only considers the most recent time t_d, and can be considered as a basic

recency-based method.

Frequency. In [AS91], Anderson et al. suggest that the frequency of interactions also play an important role in the human's recalling process of a resource, as by the re-learning effect. Hence, for each of the functions in Table 5.1, we introduce a "frequency"-based variant, which aggregates the effect of decays in different time points:

$$v_f(d,t) = \sum_{t_i \in W} v_r(d,t_i) \tag{5.1}$$

where $v_r(d,t_i)$ can be any of recency-based functions in Table 5.1. The sequence $W \subseteq T_{d,t}$ represents the time window in which all time points are taken into consideration for the ranking. For instance, if $W = T_{d,t}$ and $v_r = MRU$, we have the well-established *most frequently used* method in cache replacement policies. If $W = T_{d,t}$ and $v_r = PD$, we have the decay ranking model in [PKHN11]. The *Frequency* algorithm used in Mozilla Firefox [CS15], on the other hand, constructs W from only the last ten items of $T_{d,t}$, in order to avoid the convolution of too old accesses into the current rank. In this work, we follow this idea, and only aggregate from the last ten time points of accesses for each document.

Propagating Memory Buoyancy over Data Graph

The drawback of recency-based and frequency-based scoring functions is that they consider each document in isolation. In practice, however, humans tend to recall and find documents together within some contexts, e.g., they can follow some cues and associate a document with other related documents which are easier to recall and navigate. This exploitation of document relations is inspired by the cognitive science of associative memory [CGWW09], and is studied in a rich body of work [SG05, CGWW09, WKY13, KPHN11]. Most of the related work employs a propagation method in the document relations graph, which "transfer" the ranking score of each individual document to other related documents along the edges of the graph. However, these methods are non-learning and largely based on heuristics to combine relations weight, which requires extensive tuning, as mentioned in [SG05].

In our work, we develop a propagation method that combines different relations into a unified framework, and learns the weighting for the combination automatically. We model the process that the user finds an important document as a Markov process, where she recalls and searches for important documents via the related resources. For each pair of connected documents (d_i, d_j), we define the transition probability from document d_i to document d_j as:

$$p_{ij}(w) = \begin{cases} \frac{\sum_{k=1}^{|R|} w_k x_{ijk}}{\sum_{l=1}^{|D|} \sum_{k=1}^{|R|} w_k x_{ilk}} & \text{if } X_{ij} \neq \emptyset \text{ and } L_{d_j,t} \neq \emptyset \\ 0 & \text{otherwise} \end{cases} \tag{5.2}$$

where w is the weighting vector for the semantic relations in R. The condition $L_{d_j,t} \neq \emptyset$ ensures that the propagation has no effect on the documents that have not been created

before the time t, i.e., no propagation to the future. Similarly, the indices l's run only over the documents d_l with $L_{d_l,t} \neq \emptyset$. Consequently, we have $\sum_j p_{ij} = 1$ for all documents d_i. In practice, to avoid rank sink when performing the propagation process, if a document has no relations at all, we assume a dummy edge from it to all other documents with zero probability.

Next, we describe our propagation framework. Let P denote the transition matrix of documents in D, we follow PageRank model and define the propagation as the iterative process, where in each iteration, the memory buoyancy values of documents are updated by the following equation,

$$\mathbf{s}^{(n+1)} = \lambda P^T \mathbf{s}^{(n)} + (1-\lambda)\mathbf{v} \tag{5.3}$$

where $\mathbf{s}^{(n)} = (s(d_1,t), s(d_2,t), ..., s(d_m,t))$ is the vector of documents' memory buoyancy values at iteration n, (m is the number of documents appearing in L_t), $\mathbf{v}$ is the vector of values obtained by an activity-based scoring method, and λ is the damping factor. In Equation 5.3, we need to learn the model of weighting parameters w in order to complete the transition matrix P, described in the following.

Propagation Learning Framework

The aim of the learning is to identify the weights $w_1, .., w_{|R|}$ of the semantic relations with which we obtain the best prediction of document rankings. In this work, we propose to exploit the activity history to learn the optimal w. In particular, we simulate the navigation of the user at each time points t' in the past, and compare the computed ranks of the documents with the ranks based on the frequency of access in the time point $t'+1$. The idea is to learn w so as to minimizes the number of mis-ranked pairs, i.e., pairs (d_1, d_2) with $s(d_1,t') > s(d_2,t')$ but d_1 has been accessed less frequently than d_2 until $t'+1$.

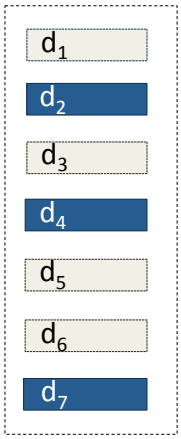
Formally, we define the label $y_{ij} = s(d_i,t') - s(d_j,t')$ and the groundtruth $\hat{y}$, $\hat{y}_{ij} = -1$ if d_i has less access than d_j at $t'+1$ and $\hat{y}_{ij} = 1$ otherwise. We learn w by the following optimization problem:

$$\min_w F(w) = \|w\|^2 + \theta \sum_{(d_i,d_j)\in A} h(y_{ij}) \tag{5.4}$$

where A is the training data, θ is the regularization parameter that controls the complexity of the model (i.e., $\|w\|^2$) while minimizes the mis-ranked pairs in A via the loss function h. In this work, we apply the simple hinge loss function: $h(y) = \max(0, 1-\hat{y}.y)$. Next, we detail how to solve Equation 5.4 and to how we collected the training data A.

Optimization Solution. Following [BL11], we solve the Equation 5.4 by a gradient descent method. The partial derivative of $F(w)$ with respect to each relation weight w_k is:

$$\frac{\partial F}{\partial w_k} = 2w_k + \theta \sum_{(d_i,d_j)\in A} \frac{\partial h(y_{ij})}{\partial y_{ij}} \left(\frac{\partial s(d_i,t')}{\partial w_k} - \frac{\partial s(d_j,t')}{\partial w_k}\right) \tag{5.5}$$



(d_1, d_2)	-1	(d_3, d_4)	-1
(d_1, d_4)	-1	(d_3, d_7)	-1
(d_1, d_7)	-1	(d_4, d_5)	1
(d_2, d_3)	1	(d_4, d_6)	1
(d_2, d_5)	1	(d_5, d_7)	-1
(d_2, d_6)	1	(d_6, d_7)	-1

Figure 5.2: Example training data: Left-hand side is a baseline document ranks. Documents in dark blue are accessed in the next time point. All documents from the first rank to the lowest rank of the accessed documents (d_7) are used for the training. Table in the right-hand side consists of training pairs, together with the labels.

The derivative of the hinge function is trivial. For $s(d,t)$, we have from Equation 5.3:

$$\frac{\partial s(d_i,t')}{\partial w_k} = \lambda \sum_j (p_{ji} \frac{\partial s(d_j,t')}{\partial w_k} + s(d_j,t') \frac{\partial p_{ji}}{\partial w_k}) \tag{5.6}$$

and:

$$\frac{\partial p_{ji}}{\partial w_k} = \frac{x_{jik} \sum_i \sum_k w_k x_{jik} - (\sum_k w_k x_{jik})(\sum_i x_{jik})}{(\sum_i \sum_k w_k x_{jik})^2} \tag{5.7}$$

From Equations 5.6 and 5.7, we can easily calculate the gradients $\frac{\partial s}{w_k}$ by a power-iteration algorithm such as in [BL11], and then apply a gradient descent-based learning method, for instance L-BFGS method [LN89], into Equation 5.5 to learn w.

Soft Labeling. We start with identifying the training time points t' to observe the subsequent accesses. In principle, any time points before the time of interest t can be chosen. In practice, however, we observe that time points during the burst periods, i.e., the period where there is a significantly higher number of access than usual, are more "interesting" to observe both correctly and falsely ranked pairs. The burst can also indicate an implicit event or task happening [TCG+14]. To this extent, we apply the Kleinberg algorithm [Kle02] to identify the burst periods from the times series of the document accesses. Then, for each period, we pick up the time point with the highest number of access for the training.

Next, we build a balanced set of positive and negative pairs for the training data as follows (Figure 5.2). For each training time point t', we extract the set of all documents accessed by the user at $t'+1$, denoted by $D_{t'+1}$. Then, we apply a baseline activity-based scoring method with respect to t for documents in D, sort them in descending order of the memory buoyancy values. From this sorted list, we get the top-scored documents until all

documents in $D_{t'+1}$ are included. We call this set S. The set $E = S \setminus D_{t'+1}$ consists of documents with high memory buoyancy scores but not accessed in the next time points, i.e., the false positives. Finally, we construct the training pairs (d_i, d_j) by picking d_i from the sets $D_{t'+1}$ or E, such that d_i, d_j are not in the same set, and that the estimated memory buoyancy score of d_i is higher than of d_j. As an example from Figure 5.2, we have (d_1, d_2) is a training pair, but (d_2, d_1) is not, because the estimated score of the document d_2 is smaller than d_1's. Similarly, $(d_1, d_3) \notin A$ because $d_1, d_3 \in E$.

To assign the training labels, if $d_i \in D_{t'+1}$ and $d_j \in E$, then we assign $\hat{y}_{ij} = 1$. Otherwise, if $d_i \in E$ and $d_j \in D_{t'+1}$, we set $\hat{y}_{ij} = -1$ (Figure 5.2).

5.2.2 Semantic Graphs

The semantic relations R play an important role in the propagation method. Depending on specific domains of applications and scenarios, we can have different types of semantic relations, which can fall in two categories:

Explicit Relations: This category consists of relations that are observable from the structures of the documents and the information space, e.g., references or hyperlinks from one document to others, the containment relations between folders and files, etc. The relations can be specified by users with the help of some software components and interfaces, such as aforementioned NEPOMUK Semantic Desktop. Recently, with the proliferation of Semantic Web and RDF technologies, some semantic relations are standardized as predicates between documents, such as hasPartOf, hasAttachment, etc.

Implicit Relations: This category includes relations that are inferred from the contents of documents, or from user activity patterns; for instance, content similarities, the correlation of two documents being accessed frequently in the same or close sessions or time. The advantage of this type of relations as compared to the explicit ones is that it can be constructed automatically without much human effort. In the following, we will focus on this type of relations. Note that while many explicit relations are asymmetric, the implicit relations discussed below are symmetric. To unify them in the same relation space R, for each implicit relation between documents d_i and d_j, we fill the score into the corresponding dimension in both X_{ij} and X_{ji}. In this work, we consider the following types of implicit relations:

Text-based Relation: This type of relation relies on the similarity of document contents. More specifically, for each document, we build the bag of words from its main content body, and calculate the Cosine similarity of the two documents to measure the strength of their relation.

Atribute-based Relation: The text-based relation is only applicable for rich-text types such as e-mails, web pages, office documents, etc. For other types of documents, we assume the Cosine similarity is zero, as no text can be extracted. For these documents, we rely on metadata specified by the users or software components. These attributes are often represented in form of <attribute, set of values>. For instance, tags of a photo, list recipients of an email, etc. We define one relation for each specific attribute, and measure

the relation strength by calculating the Jaccard similarity over the two corresponding sets of values.

Time-based Relation: This relation is derived from the activity history. It is based on the assumption that two documents are highly related with respect some latent tasks, if the user accessed to both of them in many sessions. To identify the "good" sessions for the two documents d_1, d_2, we apply the same heuristic as for building the training data: We extract all time points from the burst time periods of $T_{d_1,t}$ and $T_{d_2,t}$ to create the two sub-sequences for d_1, d_2 respectively. We then calculate Jaccard similarity between these two sequence and use as the time-based relation strength.

5.2.3 Experiments on Digital Workplaces

In this work, we conduct experiments on two real-world datasets with different characteristics. Table 5.2 summarizes the statistics of these datasets. In the following, we give more detailed information for each dataset.

Dataset 1: Semantic Desktop

The first dataset (named Person) consists of personal collections obtained via a Semantic Desktop infrastructure described in [MSD13] and deployed at the DFKI. The resulting knowledge base consists of resources, their semantic representations and relations with concepts spanning a semantic graph based on the PIMO (Personal Information Model), a state-of-the-art ontology for PIM [SvED07]. At the time of evaluation, the Semantic Desktop infrastructure has been used for over 3 years in the Knowledge Management team at DFKI on daily basis by 17 users, who are employees and students in DFKI. Among these users, 7 are active with usage of 4 to 8 hours per day, others are occasional users such as interns or assistant students. The PIMO data is stored in a knowledge base on a central server and is related to professional or research activities, e.g., business meetings, project proposals, tasks, and notes, etc. The knowledge base is accessed via the Semantic Desktop infrastructure consisting of components such as a plug-in embedded into the Windows File Explorer, a Firefox add-on, a plug-in to an Email client, and a web-based stand-alone application.[1] These are installed on each individual's computer at work, and are used on a daily basis. It enables the user to easily annotate documents when they conduct their regular tasks: Browsing the Web, reading emails, managing files on hard disks or creating calendar events. The user is also encouraged to create and use semantic concepts, such as topics, locations, persons, tasks, events, or documents. To this extent, a semantic layer is built over the "physical" information objects, e.g., in the file system, mapping each document or concept to a *resource*. In our experiments, we apply our methods to provide ranks for these semantic PIMO concepts instead of the actual documents.

[1] For a detailed explanation and videos see our ForgetIT Pilot documentation at https://pimo.opendfki.de/wp9-pilot/

Figures	Person	Collaboration
No. of documents	20363	1437
Time span	Sep 2011 - Sep 2014	Oct 2008 - Sep 2014
Users	17	268
Relations	155539	126326
Activity log entries	337528	217588

Table 5.2: Statistics of the Datasets

As for the activity history, a monitoring tool[2] was used to capture each event on the user's computer in a centralized database (only the owner can explore the log in raw format). To guarantee the privacy of personal data, real data resides in the knowledge base, only encoded information of document metadata and action logs (no content and physical files) are sent to our system for the experiments.

Relationship. All explicit semantic relations are represented in the form of RDF predicates (Table 5.3). Some explicit relations are symmetric, while others have inverse relations (displayed in Table 5.3 in parentheses). Concerning the implicit relations, as we could not obtain the contents due to privacy, we only construct a number of attribute-based and time-based relations (Table 5.3).

Dataset 2: Collaborative Wiki

The second dataset (named Collaboration) is a intranet portal used within L3S Research Center or communicating daily research activities. The portal has been continuously used in the course of 6 years and includes research information such as collaboration projects with external partners, internal research activities, as well as administration information of the lab. Documents are mostly in hypertext format, but also include digital files uploaded to the portal. In contrast to the Person dataset, the Collaboration dataset has no full-fledged ontologies, and the documents are not associated with abstract concepts. Nevertheless, the portal runs on top of the DokuWiki platform[3], and support a few annotations via different plugin: Tagging with words, showing document author and contributors, etc. For the activity history, it uses Squid cache[4] to log HTTP access requests to the portal resources. In addition, an archiving tool is developed to log all revisions of the portal documents, together with their edit activities. The dataset are obtained in the form of an archive with all raw data content.

Relationship. Compared with the dataset Person, the documents in this dataset have much less explicit semantic relations, and all come from the structure of the portal (Table 5.3). However, as we have access to contents of the documents, we can build more implicit relations, listed in Table 5.3. The tagged-token-based (TTB) relations are constructed as follows. First, we extracted the content of all documents from the portal developing a Dokuwiki parser and using a MIME paser. Next, we sample the documents related to 4 different collaborative research projects, tokenize the corresponding texts, and remove over-

[2] http://usercontext.opendfki.de/
[3] https://www.dokuwiki.org/dokuwiki
[4] http://www.squid-cache.org

Dataset Person		Dataset Collaboration	
Relation(s)	*Description*	*Relation(s)*	*Description*
hasPart (isPartOf)	Relations between container document (folder, albums, etc.) and individual files	page_namespace	relations between a page and a dokuwiki namespace
hasNewerVersion (isNewerVersionOf)	Two revisions of a document	hyperlink	a webpage is linked to other page
hasLocation (isLocationOf)	A document is tagged with a location	attachment	a webpage is attached with a file
hasRecipient (isRecipientOf)	Relations between emails and the recipient		
hasSender (isSenderOf)	Relations between emails and the recipient		
creates (isCreatedBy)	Relations between documents and owners		
isRelatedTo	Two document contents are related		
hasTopic (isTopicOf)	A resource has a topic, which is another resource		
Attribute(s)	*Description*	*Attribute(s)*	*Description*
member	Relations based on shared number of members annotated with the documents	contributors	relations between shared number of contributors to the page
containedThing	Relations based on related Thing instances	TTB	Tagged token-based relations
task	Relations based on tasks tagged to the documents	tag	relations between tags of each dokuwiki page

Table 5.3: Selected semantic relations used in two datasets Person and Collaboration. The upper part corresponds to the explicit relations, the lower-part corresponds to the attribute-based implicit relations

popular words and stop words. An experienced colleague working in numerous projects is asked to annotate the tokens with respect to 6 different classes: 1) Person or Person role; 2) Location; 3) Organization; 4) Technical word (e.g., middleware); 5) Professional domain (e.g., meeting); 6) Project-specific terms. Each class is then treated as one attribute of the documents, with their tagged tokens treated as values, for calculating the corresponding attribute-based relation.

5.2.4 Empirical Experiments

Baselines

We evaluate our system against the following baselines:

Recency-Frequency: This set of baselines use values of the activity-based scoring functions to provide the final ranking, without using propagation. This includes the two recency-based methods MRU and Ebb, and their frequency-based variants, denoted by FMRU and FEbb (Table 5.1). For polynomial and Weibull functions, we evaluate only the frequency-based methods, denoted by FPD and FWei, as they are shown to outperform the recency versions [PDR13].

PageRank: This baseline ranks the documents by their authority scores, estimated in a

graph of documents relations. The scores of documents are initialized equally. It can be thought of as the propagation method without the activity-based rankings and the semantics of relation. PageRank is shown to be effective in file retrieval tasks in non-semantic systems [SG05]. In our case, we adapt the PageRank algorithm by aggregating all relations between two documents into one single relation, with the weighting score obtained by averaging out all the individual relation weights.

SUPRA: Papadakis et al. [PKHN11] proposed combining the activity-based ranking results with a one-step propagation in a layered framework. The relations are constructed simply by identifying documents accessed in the same sessions. In our scenarios, we define the "sessions" to be one unit time step, which is one hour. We only study the MRU decay prior and simple connectivity transition matrix for this baseline, as it is among the best performing variants and requires no parameter tuning. We use the implementation provided by the authors[5].

Parameter Tuning

In all experiments, we set the granularity of the time intervals to be one day. We use MRU as the baseline scoring method for building the trading data. The parameters of the activity-based scoring functions are chosen empirically via grid search with respect to the success rate at 1 for the access prediction task (see section 5.2.4). The best performing for each function is as follows. For FEbb, $S = 90$; for FPD, $\alpha = 1.5$; for FWei, $s = 0.9, \alpha = 0.3$. The damping factor is set to $\lambda = 0.25$, as for the standard PageRank as well as for our propagation method. As for the regularization parameter θ, we experiment with a number of different values and see no significant changes in the performance, possibly due to the small number of dimensions of our relation weight vectors X. We empirically set $\theta = 1$.

Experiments on Revisit Prediction

The first experiment aims to evaluate how well the system performs in the revisit prediction task, i.e., predicting the likelihood that a document will be accessed by the user in the subsequent time point. This is the well-established task in research on web recommendation [CS15], personal file retrieval [FC12], etc. We evaluate the correlation between the predicted rank of a document at a time point t and the real document accesses at the time point $t + 1$. Inspired by [KPHN11], we employ the following evaluation metrics:

1. *Success at 1* (**S@1**): It quantifies the fraction of time points t (from all time points of study) at which the first-ranked documents according to a ranking method is truly accessed at $t+1$. This resembles the Precision at 1 (P@1) metric in traditional retrieval tasks.

[5] http://sourceforge.net/projects/supraproject

2. *Success at 10* (**S@10**): It quantifies the fraction of documents truly accessed in the next time point, from all documents ranked at top 10, averaging over all time points of study in the micro-average manner (i.e., per-document average).

3. *Average Ranking Position* (**ARP**): This metric starts from the subsequent document access backwards. It computes the average ranking position of accessed documents as produced by a ranking method. The lower the value is, the better the performance of the corresponding ranking system.

For each dataset, we run the burst detection algorithm to identify the "interesting" time points (section 5.2.1), resulting in 122 points in the dataset Person and 203 points in Collaboration. We partition each set of time points into the training and testing sets using 5-fold cross validation.

Results. The average results over the two datasets are summarized in Table 5.4. Among the ranking methods, PageRank has the worst predictive performance. This is because it ignores the recency and frequency signals of the documents. Other interesting observation is that for activity-based ranking methods, adding frequency into the ranking function did not really help in revisit prediction: FMRU performs worse than MRU and FEbb performs worse than Ebb in all metrics, although the differences are not significant. At the first look, this contradicts somewhat to previous findings on the influence of frequency in document ranking [PDR13]. However, analyzing deeper, we believe that the cause stems from the fact that a revisiting action typically involves very recent documents, as also argued in [KPHN11]. Aggregating recency scores over a time span (10 day-window as in our case) can introduce some documents belonging to different tasks and thus bring more noise to the ranking results. One possible way to solve this is to design a more flexible time window size which adapt to the user's task. We leave this direction to be explored in the future.

Compared to the sole activity-based ranking methods, adding propagation shows clear improvements in prediction, starting from the baseline SUPRA. Bringing semantic relations into the propagation improves even further, producing significantly higher performance for all case of temporal priors. The best performing method, propagation with polynomial decay prior, improves the results by 60% as compared to SUPRA. In addition, in contrast to the observed trend in the activity-based ranking, here the combination of frequency and recency with the propagation actually produces better results than the only combination between recency and the propagation. This is because using frequency makes the scores of all documents higher (Equation 5.2.1), thus enhance the contribution in the propagation point, as there will be more documents with non-zero scores than in the case of using recency only. This effect is similar to smoothing in standard information retrieval.

User-perceived Evaluation

We next aim to evaluate the effectiveness of our proposed system with respect to the user perception and appreciation. We do this by simulating the way users re-access and re-assess

Method	S@1	S@10	ARP
MRU	0.162	0.310	76
FMRU	0.131	0.291	87
Ebb	0.213	0.357	65
FEbb	0.193	0.328	70
FPD	0.195	0.331	68
FWei	0.220	0.378	60
PageRank	0.120	0.231	112
SUPRA	$0.320^{\triangle}$	$0.671^{\triangle}$	39
MRU+Prop	$0.353^{\triangle}$	$0.710^{\blacktriangle}$	34
FMRU+Prop	$0.402^{\triangle}$	$0.762^{\blacktriangle}$	30
Ebb+Prop	$0.416^{\triangle}$	$0.733^{\triangle}$	42
FEbb+Prop	$0.452^{\blacktriangle}$	$0.780^{\blacktriangle}$	25
FPD+Prop	$\mathbf{0.512}^{\blacktriangle}$	$\mathbf{0.818}^{\blacktriangle}$	**20**
FWei+Prop	$0.430^{\triangle}$	$0.750^{\blacktriangle}$	40

Table 5.4: Results on the revisit prediction task. The upper part reports baseline results, the lower part reports results of the proposed system. Symbol $^{\triangle}$ confirms significance against the baseline MRU. Symbol $^{\blacktriangle}$ confirms both significance against the baselines MRU and SUPRA

	Dataset Person				**Dataset Collaboration**			
Method	P@1	P@10	NDCG@10	MAP	P@1	P@10	NDCG@10	MAP
MRU	0.365	0.283	0.219	0.207	0.461	0.375	0.285	0.267
FMRU	0.329	0.307	0.221	0.213	0.457	0.346	0.271	0.258
Ebb	0.407	0.350	0.258	0.218	0.507	0.392	0.287	0.256
FEbb	0.391	0.292	0.217	0.213	0.493	0.357	0.275	0.260
FPD	0.382	0.290	0.214	0.220	0.480	0.400	0.301	0.288
FWei	0.443	0.402	0.324	0.293	0.552	0.424	0.319	0.290
PageRank	0.318	0.251	0.195	0.164	0.388	0.325	0.195	0.204
SUPRA$^{\triangle}$	0.547	0.502	0.426	0.389	0.590	0.469	0.345	0.333
MRU+Prop$^{\triangle}$	0.518	0.456	0.358	0.333	0.561	0.448	0.334	0.340
FMRU+Prop$^{\triangle}$	0.592	0.511	0.431	0.366	0.630	0.493	0.400	0.361
Ebb+Prop$^{\triangle}$	0.615	0.529	0.503	0.481	0.752	0.642	0.501	0.476
FEbb+Prop$^{\blacktriangle}$	**0.728**	0.621	**0.556**	**0.540**	**0.821**	**0.679**	**0.528**	**0.519**
FPD+Prop$^{\blacktriangle}$	0.710	**0.635**	0.523	0.510	0.780	0.667	0.500	0.482
FWei+Prop	0.678	0.575	0.521	0.478	0.715	0.634	0.479	0.460

Table 5.5: Performances of ranking methods in the user study. Symbols $^{\triangle}$,$^{\blacktriangle}$ indicate the significance test in all scores of the method against MRU and SUPRA respectively.

the documents in their collections. The experiment is set up as follows. We first ask the user to pick up different time periods of one week length from the past, such that each week covers some prominent events or tasks, and thus manifests considerable amount of user activities on many documents. For example, the user can choose the week when she conducts intensive work on a scientific publication, or on a project review. Within each week of study, we extract the set of documents that draw high attention from the user back in the time. These documents can be chosen manually from the user (e.g., dataset Person), or from the set of highest-frequently accessed document (e.g., dataset Collaboration). Then, the user is presented with the document information and contents[6], and is asked the question "What do you prefer to do with this document as for now ?". The options are:

1. *Pin*: The document is needed for now, I would keep it as short-cut or highlight.
2. *Show*: I would keep the document, but not in the highlight.
3. *Fade*: I would not keep the document.
4. *Trash*: I would delete the document *now*.

Each option corresponds to the user's perception of the current importance of the document, from the highest score (*Pin*) to the lowest one (*Trash*). From the perspective of information retrieval, these can be treated as the relevance feedback, and to that extent, we can use standard IR metrics to evaluate the ranking system.

In the dataset Person, each assessor chose 4 weeks to evaluate. For the dataset Collaboration, 2 assessors are asked to choose 3 weeks per each, all are related to joint events they participated in. The activity history is constructed according to this pair of users. The ranking methods are configured to provide the ranks of documents with respect to the same time step of the user's evaluations. For the Person, we cannot calculate the inter-agreement as the documents to be ranked are usually private. For the Collaboration, the inter-agreement under the Cohen's Kappa is 0.6, suggesting the shared perception of the raters on the evaluated documents.

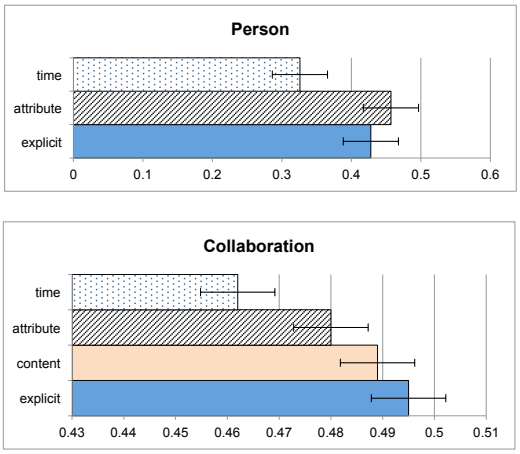


Figure 5.3: Performance of propagation (in MAP score) in ablated relation sets

[6]For the Person dataset, the user study is conducted in each computer of the assessor, thereby all information and contents are accessible

Result. The results are summarized in Table 5.5 for each dataset, as measured as precision, NDCG and MAP scores. The same trend as the prediction task can be observed here: The activity-based ranking methods perform better than PageRank but worse than SUPRA and our propagation variants. Similarly, the frequency-based functions perform worse than the recency ones as isolated methods, but improve the results when combining with the propagation. All propagation methods except the MRU prior-based give higher results than SUPRA. In addition, compared to the prediction task, the performance of all methods in the user-perceived study are slightly higher. This suggests that many documents, although not accessed subsequently, are still deemed "important" to the user. Of the two datasets, methods produce higher performance in the Collaboration than in Person. This can be explained in two ways. Firstly, data in Collaboration is more homogeneous, and the model is learnt with respect to the group of users. In contrast, in Person, the data are highly diverse and the model is learnt over each user, resulting in higher variance level. Secondly, when assessing documents in the collaboration environment, users tend to be more skeptical, and will not likely to assess one document as "Trash" unless completely certain. This results in a higher number of relevant documents than in the case of the dataset Person.

Influence of Semantic Relations Next, we study the influence of different types of semantic relations. We use *FEbb+Prop* method as it performs the best in the user study. The evaluation is done via an ablation study: We repeatedly remove from the semantic graph the relations of a certain group (see Section 5.2.2), then re-execute the framework and re-evaluate using the user study. Finally, we observe the reduction in the performance of the system measured by MAP score (Figure 5.3). In both datasets, removing time-based relations cause the biggest loss in the performance, suggesting the highest influence of this relation type. This also agrees with the existing findings [KPHN11, WKY13]. In the dataset Person, removal of the explicit relations affects more than removal of attribute-based relations[7]. This suggests the higher contributions of human-defined relations in the dataset. On the other hand, in Collaboration, the attribute-based relations has higher influence than the explicit one. This reflects the characteristics of the dataset, as the tagged tokens are Dokuwiki tags are more representative than the other evidences.

5.2.5 Qualitative Experiment: Reducing Information Overload in PIM

In addition to the empirical experiments, we also demonstrate the effectiveness of our system in real-world usage. We integrated memory buoyancy into the Semantic Desktop infrastructure (section 5.2.3) used for *decluttering* information in the HTML5 UI of the Semantic Desktop designed to be used on desktop and mobile devices (see Figure 5.4).

For instance, after days of working in the Semantic Desktop accomplishing various tasks, many things were created in the knowledge base such as new tasks and sub-tasks, notes, new topics, persons, events, annotated web-pages, documents in different versions or

[7] Recall that we could not construct the content-based relations in this dataset

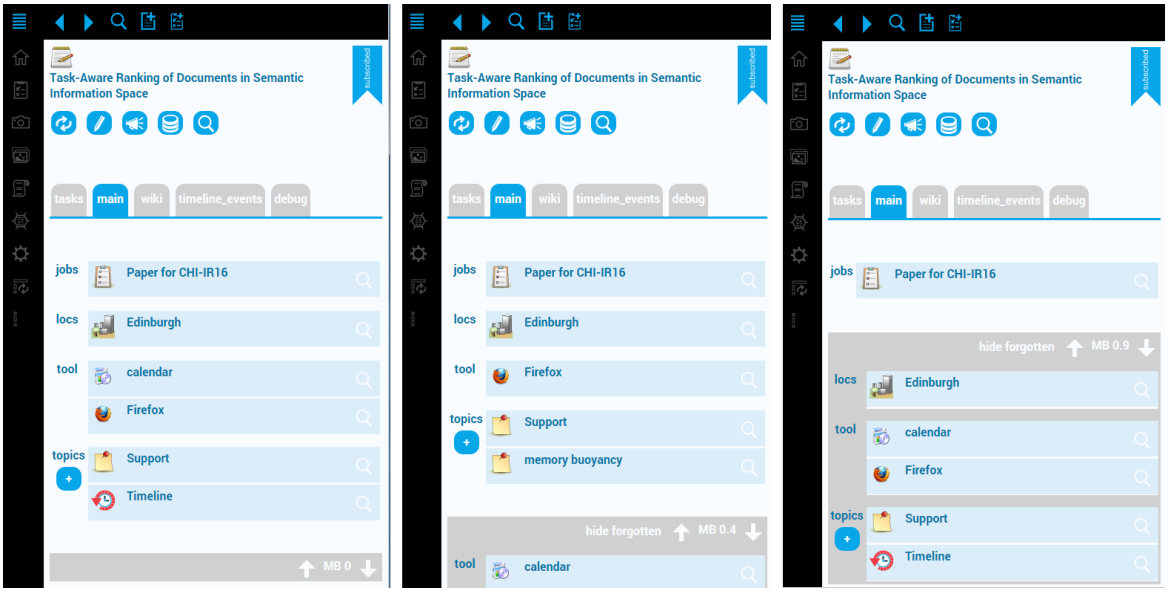


Figure 5.4: Illustration of decluttering functionality in the Semantic Desktop. From left to right: The list of recommended documents at different cut-off (MB) thresholds, 0 (left), 0.4 (middle), 0.9 (right)

only temporary relevant. That means, the user's desktop as well the UI's of the Semantic Desktop start to clutter. Imagine this over several years, lots of once relevant and now irrelevant materials are still shown and pile up, i.e., the PIM application is on the verge to build an information overload to its users.

Therefore, we apply the approach presented in this chapter to enable the user to focus on the main concepts and resources such as documents of the current attention, thereby to "declutter" the PIM application without manual reorganization efforts.[8] Figure 5.4 illustrates this functionality: a note is shown containing a text of this chapter which was used to prepare the writing. Now choosing the "Main" tab, the related things are shown categorized according to their supertype (e.g., Location (e.g., City, Building, Room) or Job (Process, Task). Now to focus on the currently most relevant things, only those things are shown above a certain threshold (displayed in the interface as MB or "Memory Buoyancy" in the gray bar; per default this bar is closed, but can be expanded to show "forgotten" things). The user is able to change the cut-off thresholds to show more forgotten things or to focus on most relevant from user interaction. The default threshold is set differently on the desktop (currently 0.5) vs. being on a mobile (currently 0.8) to account for a more focused information provision on a mobile (to reduce cognitive load and data consumption). A second functionality is to propose files to be forgotten (e.g., removing from the desktop and keeping them just in the Semantic Desktop cloud) if they drop below a certain threshold, thus also decluttering the user's desktop. All these things are associated in the semantic graph and the activity history logs. This shows the generalizability of our method: It can also be applied

[8] see also: https://pimo.opendfki.de/wp9-pilot/#9

to any concepts, given the availability of their activity history logs.

Short discussion on privacy. To integrate and improve the machine learning models for the application mentioned above, one need to cope with the critical privacy issues. On the one hand, the inspection of data is unavoidable for the evaluation and continuous improvement of the models. On the other hand, input data of the framework is highly personal, very private data together with the detailed logs of user on daily basis. Exchange such kind of data over a network is undesirable, since it is both subject to potential intrusion from third-party and to privacy violation. To cope with this issue, we follow a pragmatic approach as follows. The components for background computation, together with the strategy configuration are bundled to client digest systems (PIMO-based desktop). The components can then access to raw data locally and performs the necessary computation. In addition, a component of the evaluation user interface (EUI) is deployed in the client's environment as well. This component displays the information of user's resources, together with their estimated MB values, in a Web-based screen for users to evaluate. Finally, identifier of the resources (encoded), together with their computed MB scores and the feedback given by the users are sent back to the managed forgettor and are used in the quantitative evaluation, as well as in tuning and improvements of the models.

5.3 Chapter Summary

In this chapter we have presented an adaptive ranking approach for identifying documents that are important to the current focus and task of the user. This contributes to helping the user in navigating and decluttering the growing information spaces. Based on the idea of managed forgetting, our framework unifies evidences from activity logs and semantic relations in a principled way for computing the memory buoyancy of resources. In our method we employ machine learning techniques that automatically learn from the user access history without manual supervision efforts.

Our experiments with two real-world datasets have shown that incorporating the importance propagation via semantic relations between resource significantly improves the performance of the method. As a proof of concept, we have also developed a prototypical system for decluttering personal information space using an existing Semantic Desktop.

For future work, we plan to extend our approach to better tailor to particular scenarios (.e.g navigation on desktop is different from on mobile phones). We also plan a more in-depth user study in order to better understand the user expectation in several dimensions such as interactions and injecting human preferences (For instance, in which scenarios or domains the activity-based system works better than the activation, etc.) Other direction is the consideration of more complex user tasks in the learning model, for instance, investigating cross-device tasks or recurrent tasks.

6 Discussion and Future Work

6.1 Summary of Contributions

Time plays an increasingly important role in data mining. This role pronounces even more in modern digital contents, especially in the Web, where there exhibits the rapid shift of topics over time, making isolated analysis unstable and prone to spurious conclusions. Existing work addresses the contextual importance of temporal information in complementing the content analysis, but does not properly address the cognitive perspective of the temporal information. In this thesis, we have shown, through a number of studies, that time does not only affect the changes in document topics and contents, it also influences the way users perceive and remember information. Our work contributes to bridging the gap between cognitive science and the temporal data mining by borrowing many ideas in human remembering and forgetting into developing temporal models. We address three main tasks and show the improvement of our temporal models over the traditional approaches:

Data Enrichment. In data enrichment, we target the text collections where the underlying theme or topic change over time, making conventional annotation techniques that consider only individual documents unsuitable. We study the problem of annotating the trending topics in Twitter, formed as hashtags, and make several contributions. Firstly, we propose the novel machine learning algorithm that considers not only textual contents, but also signal features of the Twitter and Wikipedia platforms as the main sources of reasoning the cognitive dynamics of human sharing in Twitter. Secondly, to our knowledge, we are the first to integrate the analysis of Wikipedia page view and edit history in annotating Twitter topics. Thirdly, we develop the efficient learning algorithms that can scale up to millions of tweets without the need for human-involved training data. Again, our algorithm borrows the idea of activation and propagation in human memory from cognitive science. In architectural aspect, we develop a large-scale indexing framework, Hedera, and study its effectiveness in indexing and facilitating temporal topic analysis in Wikipedia edit history. Our system is

open-sourced to foster future research in similar direction.

Text Summarization. In text summarization, we also show that the ideas of collective memories can be employed to design the effective and intuive summarization system. While much of other work also considers Wikipedia use as the global, collective memory place [Pen09, WJA14], to our knowledge, our work goes beyond the descriptive analysis study to actually propose and design the predictive models based on adaptive learning, resulting in a novel summarization method. In addition, we introduce the idea of using entities as pivots for timeline news summarization, which can be helpful to picture the high-level story.

Apart from the news domains, we also study text summarization in the professional domain. We address the problem of decision summarization from meeting transcripts. Our work relies on long short-term memory, a neural network-based architecture that is inspired from how information is passed out or forgotten in human brain, to devise a novel, dialoge-aware method of summarization. Our approach that relies on pretrained sentence vectors gives considerable improvement over the traditional sequence learning-based methods.

Recommendation. We study the task of finding old documents in professional scenarios with time-sensitive tasks. Our main contribution is bringing the insights of human memory into building a novel temporal models for recommending documents based on user activity logs. We design the graph learning algorithms that exploit the different semantic relationships in a unified framework. We also conduct several studies on different datasets, where textual contents are available with different degrees. Our method brings the gap between work of different disciplines: Decay forgetting and associative memories, information management system design, and information retrieval and complex task search. The models were developed in a prototype systems that have been used in German Research Center for Artificial Intelligence (DFKI) with thorough evaluation studies, and the results suggest its high potential in semantic desktop, as well as in other applications in the future.

6.2 Discussions

6.2.1 Lessons Learned

During the course of this PhD study, we have conducted different research work. After each work, we have obtained a number of findings and lessons, both quantitatively and qualitatively. These lessons have been discussed in sections 3.4, 4.4 and 5.3. In this section, we summarize the overall lessons we learned after finising this PhD. The following list does not cover all lessons we have had during six years of the studies, but rather highlights the major lessons spanning across the work.

The first lesson is that it is crucial to understand the characteristics of data collections before any further processing. We started this thesis aiming to bridge the gap between the two seemingly disconnected communities, the cognitive science (human memory) and

the computer system (information retrieval and machine learning) communities. There has been work in literature suggesting that better understanding of human behavior will improve many information retrieval and machine learning tasks [PDR13, HS97], and we aimed to investigate the effect in time-aware access of text collections. As we continued our work on different datasets, we quickly realized that it is impossible to have a general framework to incorporate cognitive findings into a computer component. In most cases, it is the best practice that we observe how users interact with each particular collection, either via questionaires, interviews, or by relying on logging information. While we tried our best to draw a common trail in building an ideal human memory-inspired search and summarization system (chapters 4 and 5), the methods employed greatly vary from domain to domain and to datasets, and cannot be generalized to other domains and data.

The second lesson is that standization of metadata brings significant benefits to automated processing of collections, and produces much more intuitive results. While the analysis of human behaviour varies from domain to domain and data to data, it is possible to record and exchange the findings in standard formats such as RDF or database schemas. This does not only help unifying the processing work flow, but also makes the results, e.g. with respect to the summarization or recommendation tasks, much more intuitive. In chapter 4, for example, our initial idea was not to use entities as pivots for a timeline summary, however after discussions and collecting feedback from people in different fields and expertise, it become clear to us that humans are more sensitive to entities in drawing a picture of the story they read. Once we developed the entity extraction component based on standard NLP pipelines, the results were much better than existing work. In chapter 5, we set up a standardized semantic desktop based on RDF to capture humans' activities, and the inference of collaborative memory become much easier, with more understandable recommendation results. We also see from here an interesting research question: To which extent we can standardize the knowledge cognition science and bring it into the computer systems. This can be a topic of future work.

The third lesson is that when we go from individual documents to collections, the connections between items play a crucial role. Much of valuable information about human memory lies in analysing these connections. When we built the semantic annotation system for social media topics (chapter 3), the connections established when posts are disseminated across social networks turned out to be one of the most important resources, and this inspired us to develop the influence learning algorithm (section 3.2.4). In chapter 5, when we seeked the way to get over the difficulties in analysing heterogeneous data from a semantic desktop, the contents are not always available to the system due to confidentiality, or due to technical difficulties to process ah-hoc document format. We realized that only by looking at the connections between the files (how the files are organized and grouped by topics, how one file refers to other via meta-data, etc.), we already had invaluable information about the human memory process. It gives us an idea to design the graph learning method, detailed in section 5.2.1.

6.2.2 Weaknesses

Although we tried our best to deliver solid research methods, this thesis still has several weaknesses. Firstly, many of our methods rely heavily on handcrafted features, and they are not generalizable to other domains and datasets. Feature engineering is a tedious task, and in our work, the design of a good feature relies even more on the domain expert, since the research question relates closely to humans cognitive understanding. It is still a long path towards developing a flexible methodology that can exploit this expertise in a fully automated manner.

Secondly, many of our machine learning tasks require special human supervised labels, and thus can only target the limited set of topics. For example, in building the timeline summary of news events (chapter 4), or in annotating social media posts (chapter 3), our methods only works with globally popular topics, when enough labels can be obtained. It cannot be employed directly to local topics, or to personalize the timeline summarization to each user interest without developing a sensible labelling method.

Thirdly, despite some efforts in developing the applications which directly benefit from our methods, the frameworks we design are still relatively complicated, and require considerable engineering effort to be transferred to industry-ready applications. We consider this as the goal for our next step.

6.3 Open Research Directions

There is a recent trend in the convergence of these disciplines in artificial intelligence, especially in machine learning area. The recent advent of cognitive computing [MAE+11] and the wide success of deep learning [LBH15] have confirmed this trend. This thesis contributes to this direction in the context of temporal data mining, and we can see even more directions to explore in the future. They are not only limited to the applications touched in this thesis, but also extended to many other applications as well. In addition to the work extension mentioned in chapters 3, 4 and 5, we discuss here the open research questions and directions in general. The questions we raise here are high-level and do not go into the technical aspects of the problems.

As the first idea, work in information extraction will continue to benefit from the advances and insights in temporal data analysis. Much of recent work realizes that given the high dynamic of digital contents, static knowledge base will become obsolete sooner than before. Many recent and ongoing projects are conducted in building temporal and dynamic knowledge bases [WZQ+10, ÁRM13, TWM12], in expanding knowledge bases with fresh information from new entities and events [KW14, HMW+16], in designing semantic relatedness between entities that are time and context-aware [TTN17, ZRZ16], etc. Principally, semantic annotations and structured data are still the effective ways to enhance automated processing of digital contents. To accommodate the temporal dynamics of knowledge, current standards and frameworks such as RDF need to be revisited [GHV07]. In the future,

we will see similar trends in expanding taxonomies to facilitate events, time, topics, as well as many other dynamic dimensions.

Another idea is to exploit more social signals from social media to reduce the cost of having high-quality training data. While the active development of crowdsourcing has been useful, it is still largely limited to simple and sometimes mundane tasks. Much of the study requires higher human cognitive load, which was illustrated in our study of decision summarization (section 4.3). One future direction will be to integrate crowdsourcing techniques into social media platforms, to be able to infer user behavior with minimum intervention from the system. There is active development in characterizing user perception when being exposed and sharing social media posts [LL16, LGRC12]. Some models are inspired from epidemics and cascading effects [ML14, RMK11], but still limited to simple tasks. How to extend these social effects for advanced learning systems such as summarization of a collection still remains an open research question.

Future research in cognitive science and human memory will continue to foster research in text mining and temporal data mining. Many new memory models have been introduced in the past few years in deep learning community [KIO+16, HA14, BP06, SWF+15], and we can expect to witness more growth in this field, where the data are seen in time dimensions and not static as present. This will lead to time- and context-aware deep learning, an area that is promising yet still largely unexplored. This has many potential applications. For example, a memory network can integrate its prediction with temporal reasoning about the current events and global contexts to give timely suggestion to e-commerce customers (e.g., using event and geo-spatial knowledge bases). Similarly, a robotic system can make use of common-sense knowledge bases, enhanced with temporal and contextual information to better assist elderly people. While such combination between machine learning and knowledge reasoning has long been of interest in machine learning community (one example is Markov Logic Networks [RD06]), general and environment-aware machine intelligence is still the major challenge in artificial intelligence, and will inspire many research work in the future.

A

Curriculum Vitae

Tuan Tran, born on 24 April 1985, in Hanoi, Vietnam.

Studies

09/2011 - 08/2017	Ph.D. study. Gottfried Wilhelm Leibniz Universität Hannover, Germany. Ph.D. thesis: "Temporal Models in Data Mining: Enrichment, Summarization and Recommendation' Advisor: Prof. Dr. Wolfgang Nejdl
04/2009 - 04/2011	M.Sc. in Computer Science. Universität des Saarlandes, Saarbrücken, Germany. Master thesis: "Context-Aware Timeline for Entity Exploration". Advisor: Prof. Dr. Gerhard Weikum Grade: 1.3 (Honors's Degree)
09/2003 - 06/2008	B.Eng in Information Technology. Hanoi University of Science and Technology, Vietnam. Diploma thesis: "Data Mining Approach towards Automatic Categorization of Query Results". Advisor: Assoc. Prof. Dr. Kim Anh Nguyen Grade: 8.18 / 10.00 (Excellent Degree)

Professional Experience

09/2017 - Present	Research and Development Engineer. Robert Bosch GmbH, Abstatt, Germany Develop scalable AI methods for processing data at Chassis System Controls deparments at Bosch.
01/2011 - 8/2017	Researcher and Software Engineer. L3S Research Center, Gottfried Wilhelm Leibniz Universität Hannover, Germany.

Research:	Worked in different EU R&D projects, designed machine learning models and developed software prototypes.
Teaching:	Summer Semester 2015: Web Technologies Laboratory (teaching assistant).
06/2016 - 09/2016	Research Intern. IBM Research, Dublin, Ireland Worked in project "Decision Gisting", developed natural language processing methods to process multi-party dialogues.
10/2010 - 03/2011	Research assistant. Max-Planck Institut für Informatik, Saarbrücken, Germany. Participated in the Living Knowledge project: Studying the diversity and bias of knowledge on the Web.
10/2010 - 12/2010	Research assistant. German Research Center for Artificial Intelligence (DFKI), Saarbrücken, Germany. Developed method to automatically align ontologies from YAGO and Question Answering Systems in DFKI.
04/2008 - 10/2008	Software Engineer. Viet Intelligences Group, Hanoi, Vietnam. Developed e-commerce Websites for Japanese customers.
03/2007 - 10/2007	Software Engineering Intern. Panasonic R&D Singapore. Built an automated C/C++ legacy code analyser for management of code repositories.

Research Projects

2016-2017	Qualimaster (www.qualimaster.eu) EU FP7, Project No.: 619525
2013-2016	ForgetIT (www.forgetit-project.eu) EU FP7, Project No.: 600826
2011-2013	GLOCAL (www.glocal-project.eu) EU FP7, Project No.: 248984
2010-2011	Living Knowledge (livingknowledge.europarchive.org) EU FP7, Project No.: 231126

PC Memberships & Reviews

2016	ACM International Conference on Web Search and Data Mining (WSDM), Sub-Reviewer
2013	ACM Conference on Recommender System (RecSys), Sub-Reviewer

Bibliography

[ABK+07] Sören Auer, Christian Bizer, Georgi Kobilarov, Jens Lehmann, Richard Cyganiak, and Zachary Ives. DBpedia: A nucleus for a web of open data. *The semantic web*, pages 722–735, 2007.

[AGBY09] Omar Alonso, Michael Gertz, and Ricardo Baeza-Yates. Clustering and exploring search results using timeline constructions. In *Proceedings of the 18th ACM conference on Information and knowledge management*, pages 97–106. ACM, 2009.

[AGDP13] Enrique Alfonseca, Guillermo Garrido, Jean-Yves Delort, and Anselmo Peñas. WHAD: Wikipedia historical attributes data. *Language resources and evaluation*, 47(4):1163–1190, 2013.

[AHE+11] Amr Ahmed, Qirong Ho, Jacob Eisenstein, Eric Xing, Alexander J Smola, and Choon Hui Teo. Unified analysis of streaming news. In *Proceedings of the 20th international conference on World wide web*, pages 267–276. ACM, 2011.

[ÁRM13] Alfonso Ávila-Robinson and Kumiko Miyazaki. Dynamics of scientific knowledge bases as proxies for discerning technological emergence - the case of MEMS/NEMS technologies. *Technological Forecasting and Social Change*, 80(6):1071–1084, 2013.

[AS91] J. R. Anderson and L. J. Schooler. Reflections of the environment in memory. *Psychological Science*, 6(2):396–408, 1991.

[ASBYG11] Omar Alonso, Jannik Strötgen, Ricardo A Baeza-Yates, and Michael Gertz. Temporal information retrieval: Challenges and opportunities. *Temporal Web Analytics Workshop at International Conference on World Wide Web*, 11:1–8, 2011.

[ATD09] Paul André, Jaime Teevan, and Susan T Dumais. From x-rays to silly putty via uranus: serendipity and its role in web search. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 2033–2036. ACM, 2009.

[ATT13] Khaled Hossain Ansary, Anh Tuan Tran, and Nam Khanh Tran. A pipeline tweet contextualization system at inex 2013. In *Working Notes of Conference and Labs of the Evaluation Forum (CLEF)*, 2013.

[AVL10] Morteza Alamgir and Ulrike Von Luxburg. Multi-agent random walks for local clustering on graphs. In *10th IEEE International Conference on Data Mining (ICDM)*, pages 18–27. IEEE, 2010.

[Aye99] Edward L Ayers. The pasts and futures of digital history. *unpublished paper delivered at the Organization of American Historians, Toronto*, 1999.

[AYJ11] Ching-man Au Yeung and Adam Jatowt. Studying how the past is remembered: Towards computational history through large scale text mining. In *Proceedings of the 20th ACM international conference on Information and knowledge management*, pages 1231–1240. ACM, 2011.

[Bah84] Harry P Bahrick. Semantic memory content in permastore: Fifty years of memory for spanish learned in school. *Journal of experimental psychology: General*, 113(1):1, 1984.

[BBAW10] Klaus Berberich, Srikanta J Bedathur, Omar Alonso, and Gerhard Weikum. A language modeling approach for temporal information needs. In *Proceedings of European Conference in Information Retrieval*, volume 10, pages 13–25. Springer, 2010.

[BBMN+08] Ofer Bergman, Ruth Beyth-Marom, Rafi Nachmias, Noa Gradovitch, and Steve Whittaker. Improved search engines and navigation preference in personal information management. *ACM TOIS*, 26(4):20, 2008.

[BBNW07] Klaus Berberich, Srikanta Bedathur, Thomas Neumann, and Gerhard Weikum. A time machine for text search. In *Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 519–526. ACM, 2007.

[BBV15] P. Bansal, R. Bansal, and V. Varma. Towards deep semantic analysis of hashtags. In *Proceedings of European Conference in Information Retrieval*, pages 453–464, 2015.

[BC92] Nicholas J Belkin and W Bruce Croft. Information filtering and information retrieval: Two sides of the same coin? *Communications of the ACM*, 35(12):29–38, 1992.

[BDR17] Kalina Bontcheva, Leon Derczynski, and Ian Roberts. *Crowdsourcing Named Entity Recognition and Entity Linking Corpora*, pages 875–892. Springer Netherlands, Dordrecht, 2017.

[BEP+08] Kurt Bollacker, Colin Evans, Praveen Paritosh, Tim Sturge, and Jamie Taylor. Freebase: a collaboratively created graph database for structuring human knowledge. In *Proceedings of the 2008 ACM SIGMOD international conference on Management of data*, pages 1247–1250. ACM, 2008.

[Ber09] Dorthe Berntsen. *Involuntary autobiographical memories: An introduction to the unbidden past*. Cambridge University Press, 2009.

[BH10] Cosmin Adrian Bejan and Sanda Harabagiu. Unsupervised event coreference resolution with rich linguistic features. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 1412–1422. Association for Computational Linguistics, 2010.

[BJD04] Harry Bruce, William Jones, and Susan Dumais. Keeping and re-finding information on the web: What do people do and what do they need? *Proceedings of the Association for Information Science and Technology*, 41(1):129–137, 2004.

[BK99] Branimir Boguraev and Christopher Kennedy. Salience-based content characterisation of text documents. *Advances in automatic text summarization*, pages 99–110, 1999.

[BL11] Lars Backstrom and Jure Leskovec. Supervised random walks: predicting and recommending links in social networks. In *Proceedings of the fourth ACM international conference on Web search and data mining*, pages 635–644. ACM, 2011.

[BLL+10] Jiang Bian, Xin Li, Fan Li, Zhaohui Zheng, and Hongyuan Zha. Ranking specialization for web search: a divide-and-conquer approach by using topical ranksvm. In *Proceedings of the 19th international conference on World wide web*, pages 131–140. ACM, 2010.

[BML13] Ilaria Bordino, Yelena Mejova, and Mounia Lalmas. Penguins in sweaters, or serendipitous entity search on user-generated content. In *Proceedings of the 22nd ACM international conference on Information & Knowledge Management*, pages 109–118. ACM, 2013.

[BNJ03] David M. Blei, Andrew Y. Ng, and Michael I. Jordan. Latent dirichlet allocation. *J. Mach. Learn. Res.*, 2003.

[BOM15] Roi Blanco, Giuseppe Ottaviano, and Edgar Meij. Fast and space-efficient entity linking for queries. In *Proceedings of the Eighth ACM International Conference on Web Search and Data Mining*, pages 179–188. ACM, 2015.

[BP06] Matthew M Botvinick and David C Plaut. Short-term memory for serial order: a recurrent neural network model. *Psychological review*, 113(2):201, 2006.

[BP10] Trung H Bui and Stanley Peters. Decision detection using hierarchical graphical models. In *Proceedings of the ACL 2010 Conference Short Papers*, pages 307–312. Association for Computational Linguistics, 2010.

[Bro02] Andrei Broder. A taxonomy of web search. In *ACM Sigir forum*, volume 36, pages 3–10. ACM, 2002.

[BRR05] Satanjeev Banerjee, Carolyn Rose, and Alexander I Rudnicky. The necessity of a meeting recording and playback system, and the benefit of topic–level annotations to meeting browsing. In *IFIP Conference on Human-Computer Interaction*, pages 643–656. Springer, 2005.

[BRV+14] A. E. Cano Basave, G. Rizzo, A. Varga, M. Rowe, M. Stankovic, and A. Dadzie. Making sense of microposts (#microposts2014) named entity extraction & linking challenge. In *4th Workshop on Making Sense of Microposts*, 2014.

[CAB+05] Jean Carletta, Simone Ashby, Sebastien Bourban, Mike Flynn, Mael Guillemot, Thomas Hain, Jaroslav Kadlec, Vasilis Karaiskos, Wessel Kraaij, Melissa Kronenthal, et al. The AMI meeting corpus: A pre-announcement. In *International Workshop on Machine Learning for Multimodal Interaction*, pages 28–39. Springer, 2005.

[CBCTH13] Xi Chen, Paul N Bennett, Kevyn Collins-Thompson, and Eric Horvitz. Pairwise ranking aggregation in a crowdsourced setting. In *Proceedings of the sixth ACM international conference on Web search and data mining*, pages 193–202. ACM, 2013.

[CCS92] Martin A Conway, Gillian Cohen, and Nicola Stanhope. Very long-term memory for knowledge acquired at school and university. *Applied cognitive psychology*, 6(6):467–482, 1992.

[CCS14] Miguel Costa, Francisco Couto, and Mário Silva. Learning temporal-dependent ranking models. In *SIGIR*, pages 757–766, 2014.

[CFC13] Marco Cornolti, Paolo Ferragina, and Massimiliano Ciaramita. A framework for benchmarking entity-annotation systems. In *Proceedings of the 22nd international conference on World Wide Web*, pages 249–260. ACM, 2013.

[CGCB14] Junyoung Chung, Çaglar Gülçehre, KyungHyun Cho, and Yoshua Bengio. Empirical evaluation of gated recurrent neural networks on sequence modeling. *CoRR*, abs/1412.3555, 2014.

[CGD92] William S. Cooper, Fredric C. Gey, and Daniel P. Dabney. Probabilistic retrieval based on staged logistic regression. In *Proceedings of the 15th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '92, pages 198–210, New York, NY, USA, 1992. ACM.

[CGWW09] Jidong Chen, Hang Guo, Wentao Wu, and Wei Wang. imecho: an associative memory based desktop search system. In *Proceedings of the 18th ACM conference on Information and knowledge management*, pages 731–740. ACM, 2009.

[CJR+12] T. Cassidy, H. Ji, L.-A. Ratinov, A. Zubiaga, and H. Huang. Analysis and enhancement of wikification for microblogs with context expansion. In *COLING*, pages 441–456, 2012.

[CM12] Angel X Chang and Christopher D Manning. Sutime: A library for recognizing and normalizing time expressions. In *LREC*, volume 2012, pages 3735–3740, 2012.

[CN10] Marek Ciglan and Kjetil Nørvåg. WikiPop: personalized event detection system based on Wikipedia page view statistics. In *Proceedings of the 19th ACM international conference on Information and knowledge management*, pages 1931–1932. ACM, 2010.

[CS15] Mike Connor and Seth Spitzer. The places frecency algorithm. https://developer.mozilla.org/en-US/docs/Mozilla/Tech/Places/Frecency_algorithm, (Accessed Aug 31, 2015).

[Cuc07] Silviu Cucerzan. Large-scale named entity disambiguation based on wikipedia data. *EMNLP-CoNLL 2007*, pages 708–717, 2007.

[CWB+11] Ronan Collobert, Jason Weston, Léon Bottou, Michael Karlen, Koray Kavukcuoglu, and Pavel Kuksa. Natural language processing (almost) from scratch. *Journal of Machine Learning Research*, 12(Aug):2493–2537, 2011.

[DCC+03] Susan Dumais, Edward Cutrell, Jonathan J Cadiz, Gavin Jancke, Raman Sarin, and Daniel C Robbins. Stuff i've seen: a system for personal information retrieval and re-use. In *Proceedings of the 26th Annual International ACM SIGIR Conference on Research and Development in Informaion Retrieval*, pages 72–79. ACM, 2003.

[DCZ+10] Anlei Dong, Yi Chang, Zhaohui Zheng, Gilad Mishne, Jing Bai, Ruiqiang Zhang, Karolina Buchner, Ciya Liao, and Fernando Diaz. Towards recency ranking in web search. In *Proceedings of the third ACM international conference on Web search and data mining*, pages 11–20. ACM, 2010.

[DD10] Na Dai and Brian D Davison. Freshness matters: in flowers, food, and web authority. In *Proceedings of the 33rd international ACM SIGIR conference on Research and development in information retrieval*, pages 114–121. ACM, 2010.

[DDCM12] Gianluca Demartini, Djellel Eddine Difallah, and Philippe Cudré-Mauroux. Zencrowd: leveraging probabilistic reasoning and crowdsourcing techniques for large-scale entity linking. In *Proceedings of the 21st international conference on World Wide Web*, pages 469–478. ACM, 2012.

[DG14] Jesse Dunietz and Daniel Gillick. A new entity salience task with millions of training examples. In *Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics*, volume 14, pages 205–209, 2014.

[DGC11] Pan Du, Jiafeng Guo, and Xueqi Cheng. Supervised lazy random walk for topic-focused multi-document summarization. In *Proceedings of 2011 IEEE 11th International Conference on Data Mining (ICDM)*, pages 1026–1031. IEEE, 2011.

[dJBR+12] Martin de Jode, Ralph Barthel, Jon Rogers, Angelina Karpovich, Andrew Hudson-Smith, Michael Quigley, and Chris Speed. Enhancing the 'second-hand' retail experience with digital object memories. In *Proceedings of the 2012 ACM Conference on Ubiquitous Computing - UbiComp '12*, page 451, New York, New York, USA, September 2012. ACM Press.

[DK14] Greg Durrett and Dan Klein. A joint model for entity analysis: Coreference, typing, and linking. *Transactions of the Association for Computational Linguistics*, 2:477–490, 2014.

[DM07] Dipanjan Das and André FT Martins. A survey on automatic text summarization. *Literature Survey for the Language and Statistics II course at CMU*, 4:192–195, 2007.

[DMBZ10] Gianluca Demartini, Malik Muhammad Saad Missen, Roi Blanco, and Hugo Zaragoza. Taer: time-aware entity retrieval-exploiting the past to find relevant entities in news articles. In *Proceedings of the 19th ACM international conference on Information and knowledge management*, pages 1517–1520. ACM, 2010.

[DRS+09] Quang Do, Dan Roth, Mark Sammons, Yuancheng Tu, and V Vydiswaran. Robust, light-weight approaches to compute lexical similarity. *Computer Science Research and Technical Reports, University of Illinois*, 2009.

[DSD11] Na Dai, Milad Shokouhi, and Brian D Davison. Learning to rank for freshness and relevance. In *Proceedings of the 34th international ACM SIGIR conference on Research and development in Information Retrieval*, pages 95–104. ACM, 2011.

[DSS12] Onkar Dalal, Srinivasan H. Sengemedu, and Subhajit Sanyal. Multi-objective ranking of comments on web. In *Proceedings of the 21st International Conference on World Wide Web*, WWW '12, pages 419–428, New York, NY, USA, 2012. ACM.

[Ebb13] H. Ebbinghaus. *Memory: A contribution to experimental psychology*. Teachers college, Columbia university, 1913.

[Edd96] Sean R Eddy. Hidden markov models. *Current opinion in structural biology*, 6(3):361–365, 1996.

[END+16] Holger Eichelberger, Claudia Niederee, Apostolos Dollas, Ekaterini Ioannou, Cui Qin, Grigorios Chrysos, Christoph Hube, Tuan Tran, Apostolos Nydriotis, Pavlos Malakonakis, Stefan Burkhard, Tobias Becker, and Minos Garofalakis. Configure, generate, run: Model-based development for big data processing. In *European Project Space on Intelligent Technologies, Software engineering, Computer Vision, Graphics, Optics and Photonics*. SCITEPRESS, 2016.

[ER04] Günes Erkan and Dragomir R Radev. Lexrank: Graph-based lexical centrality as salience in text summarization. *J. Artif. Intell. Res.(JAIR)*, 22(1):457–479, 2004.

[FC12] Stephen Fitchett and Andy Cockburn. Accessrank: predicting what users will do next. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 2239–2242. ACM, 2012.

[FC14] Y. Fang and M.-W. Chang. Entity linking on microblogs with spatial and temporal signals. *Trans. of the Assoc. for Comp. Linguistics*, 2:259–272, 2014.

[FFE+08] Raquel Fernández, Matthew Frampton, Patrick Ehlen, Matthew Purver, and Stanley Peters. Modelling and detecting decisions in multi-party dialogue. In *Proceedings of the 9th SIGdial Workshop on Discourse and Dialogue*, pages 156–163. ACL, 2008.

[FHBP09] Matthew Frampton, Jia Huang, Trung Huu Bui, and Stanley Peters. Real-time decision detection in multi-party dialogue. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 3 - Volume 3*, EMNLP '09, pages 1133–1141. Association for Computational Linguistics, 2009.

[FISS03] Yoav Freund, Raj Iyer, Robert E Schapire, and Yoram Singer. An efficient boosting algorithm for combining preferences. *Journal of machine learning research*, 4(Nov):933–969, 2003.

[Fle71] Joseph L Fleiss. Measuring nominal scale agreement among many raters. *Psychological bulletin*, 76(5):378, 1971.

[FRCS05] D. Fogaras, B. Rácz, K. Csalogány, and T. Sarlós. Towards scaling fully personalized PageRank: Algorithms, lower bounds, and experiments. *Internet Mathematics*, 2(3):333–358, 2005.

[FS12] P. Ferragina and U. Scaiella. Fast and accurate annotation of short texts with Wikipedia pages. *IEEE Softw.*, 29(1):70–75, 2012.

[Fuh89] Norbert Fuhr. Optimum polynomial retrieval functions based on the probability ranking principle. *ACM Trans. Inf. Syst.*, 7(3):183–204, July 1989.

[FZG11] Oliver Ferschke, Torsten Zesch, and Iryna Gurevych. Wikipedia revision toolkit: efficiently accessing wikipedia's edit history. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies: Systems Demonstrations*, pages 97–102. Association for Computational Linguistics, 2011.

[Gal06] Michel Galley. A skip-chain conditional random field for ranking meeting utterances by importance. In *Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing*, EMNLP '06, pages 364–372, Stroudsburg, PA, USA, 2006. Association for Computational Linguistics.

[GB14] Zhaochen Guo and Denilson Barbosa. Robust entity linking via random walks. In *Proceedings of the 23rd ACM International Conference on Conference on Information and Knowledge Management*, pages 499–508. ACM, 2014.

[GCK13] S. Guo, M.-W. Chang, and E. Kıcıman. To link or not to link? A study on end-to-end tweet entity linking. In *NAACL-HLT*, pages 1020–1030, 2013.

[GDBJ10] Mouzhi Ge, Carla Delgado-Battenfeld, and Dietmar Jannach. Beyond accuracy: evaluating recommender systems by coverage and serendipity. In *Proceedings of the fourth ACM conference on Recommender systems*, pages 257–260. ACM, 2010.

[GG95] Pierdaniele Giaretta and N Guarino. Ontologies and knowledge bases towards a terminological clarification. *Towards very large knowledge bases: knowledge building & knowledge sharing*, 25:32, 1995.

[GGL+16] Octavian-Eugen Ganea, Marina Ganea, Aurelien Lucchi, Carsten Eickhoff, and Thomas Hofmann. Probabilistic bag-of-hyperlinks model for entity linking. In *Proceedings of the 25th International Conference on World Wide Web*, pages 927–938. International World Wide Web Conferences Steering Committee, 2016.

[GHV07] Claudio Gutierrez, Carlos A Hurtado, and Alejandro Vaisman. Introducing time into rdf. *IEEE Transactions on Knowledge and Data Engineering*, 19(2), 2007.

[GIM+99] Aristides Gionis, Piotr Indyk, Rajeev Motwani, et al. Similarity search in high dimensions via hashing. In *Proceedings of the 25th Very Large Database Conference (VLDB)*, volume 99, pages 518–529, 1999.

[GKK+13] Mihai Georgescu, Nattiya Kanhabua, Daniel Krause, Wolfgang Nejdl, and Stefan Siersdorfer. Extracting event-related information from article updates in wikipedia. In *Proceedings of Proceedings of European Conference in Information Retrieval '13*, 2013.

[GLQ+08] Xiubo Geng, Tie-Yan Liu, Tao Qin, Andrew Arnold, Hang Li, and Heung-Yeung Shum. Query dependent ranking using k-nearest neighbor. In *Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval*, pages 115–122. ACM, 2008.

[GLW+11] Bin Gao, Tie-Yan Liu, Wei Wei, Taifeng Wang, and Hang Li. Semi-supervised ranking on very large graphs with rich metadata. In *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 96–104. ACM, 2011.

[Gun52] Robert Gunning. Judges scold lawyers for bad writing, 1952.

[GYS+13] Michael Gamon, Tae Yano, Xinying Song, Johnson Apacible, and Patrick Pantel. Identifying salient entities in web pages. In *Proceedings of the 22nd ACM international conference on Information & Knowledge Management*, pages 2375–2380. ACM, 2013.

[HA14] Geoffrey E Hinton and James A Anderson. *Parallel models of associative memory: updated edition*. Psychology press, 2014.

[HA15] Ronald A Howard and Ali E Abbas. *Foundations of decision analysis*. Pearson, 2015.

[Hav02] Taher H Haveliwala. Topic-sensitive pagerank. In *Proceedings of the 11th international conference on World Wide Web*, pages 517–526. ACM, 2002.

[HB16] Ravindra Harige and Paul Buitelaar. Generating a large-scale entity linking dictionary from wikipedia link structure and article text. In *LREC*, 2016.

[HBB15] Faegheh Hasibi, Krisztian Balog, and Svein Erik Bratsberg. Entity linking in queries: Tasks and evaluation. In *Proceedings of the 2015 International Conference on The Theory of Information Retrieval*, pages 171–180. ACM, 2015.

[HM07] Pei-Yun Hsueh and Johanna D Moore. Automatic decision detection in meeting speech. In *International Workshop on Machine Learning for Multimodal Interaction*, pages 168–179. Springer, 2007.

[HMW+16] Johannes Hoffart, Dragan Milchevski, Gerhard Weikum, Avishek Anand, and Jaspreet Singh. The knowledge awakens: Keeping knowledge bases fresh with emerging entities. In *Proceedings of the 25th International Conference Companion on World Wide Web*, pages 203–206. International World Wide Web Conferences Steering Committee, 2016.

[HNA16] Helge Holzmann, Wolfgang Nejdl, and Avishek Anand. The dawn of today's popular domains: A study of the archived german web over 18 years. In *Digital Libraries (JCDL), 2016 IEEE/ACM Joint Conference on*, pages 73–82. IEEE, 2016.

[HRN+13] Ben Hachey, Will Radford, Joel Nothman, Matthew Honnibal, and James R Curran. Evaluating entity linking with wikipedia. *Artificial intelligence*, 194:130–150, 2013.

[HS97] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

[HSN+12] Johannes Hoffart, Stephan Seufert, Dat Ba Nguyen, Martin Theobald, and Gerhard Weikum. KORE: keyphrase overlap relatedness for entity disambiguation. In *Proceedings of the 21st ACM international conference on Information and knowledge management*, pages 545–554. ACM, 2012.

[HXY15] Zhiheng Huang, Wei Xu, and Kai Yu. Bidirectional lstm-crf models for sequence tagging. *arXiv preprint arXiv:1508.01991*, 2015.

[HYB+11] Johannes Hoffart, Mohamed Amir Yosef, Ilaria Bordino, Hagen Fürstenau, Manfred Pinkal, Marc Spaniol, Bilyana Taneva, Stefan Thater, and Gerhard Weikum. Robust disambiguation of named entities in text. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 782–792. Association for Computational Linguistics, 2011.

[IM01] Ludmila Isurin and Janet L Mcdonald. Retroactive interference from translation equivalents: Implications for first language forgetting. *Memory & cognition*, 29(2):312–319, 2001.

[IS10] Hazra Imran and Aditi Sharan. Improving effectiveness of query expansion using information theoretic approach. *Trends in Applied Intelligent Systems*, pages 1–11, 2010.

[JAB+04] Adam Janin, Jeremy Ang, Sonali Bhagat, Rajdip Dhillon, Jane Edwards, Javier Macï£¡as-guarasa, Nelson Morgan, Barbara Peskin, Elizabeth Shriberg, Andreas Stolcke, Chuck Wooters, and Britta Wrede. The icsi meeting project: Resources and research. In *in Proc. of ICASSP 2004 Meeting Recognition Workshop*. Prentice Hall, 2004.

[JLW+10] Carlos Jensen, Heather Lonsdale, Eleanor Wynn, Jill Cao, Michael Slater, and Thomas G. Dietterich. The life and times of files and information: A study of desktop provenance. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '10, pages 767–776, 2010.

[Joa02] Thorsten Joachims. Optimizing search engines using clickthrough data. In *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 133–142. ACM, 2002.

[JWF+15] Anthony Jameson, Martijn C Willemsen, Alexander Felfernig, Marco de Gemmis, Pasquale Lops, Giovanni Semeraro, and Li Chen. Human decision making and recommender systems. In *Recommender Systems Handbook*, pages 611–648. Springer, 2015.

[JZCF07] J Jian Zhang, Ho Yin Chan, and Pascale Fung. Improving lecture speech summarization using rhetorical information. In *Automatic Speech Recognition & Understanding, 2007. ASRU. IEEE Workshop on*, pages 195–200. IEEE, 2007.

[KB15] Yehuda Koren and Robert Bell. Advances in collaborative filtering. In *Recommender systems handbook*, pages 77–118. Springer, 2015.

[KBN+15] Nattiya Kanhabua, Roi Blanco, Kjetil Nørvåg, et al. Temporal information retrieval. *Foundations and Trends® in Information Retrieval*, 9(2):91–208, 2015.

[KCW09] Jen-Wei Kuo, Pu-Jen Cheng, and Hsin-Min Wang. Learning to rank from bayesian decision inference. In *Proceedings of the 18th ACM conference on Information and knowledge management*, pages 827–836. ACM, 2009.

[KFJRC75] J Peter Kincaid, Robert P Fishburne Jr, Richard L Rogers, and Brad S Chissom. Derivation of new readability formulas (automated readability index, fog count and flesch reading ease formula) for navy enlisted personnel. Technical report, DTIC Document, 1975.

[KFN10] Christian Kohlschütter, Peter Fankhauser, and Wolfgang Nejdl. Boilerplate detection using shallow text features. In *Proceedings of the third ACM international conference on Web search and data mining*, pages 441–450. ACM, 2010.

[KGC11] Brian Keegan, Darren Gergle, and Noshir Contractor. Hot off the wiki: Dynamics, practices, and structures in wikipedia's coverage of the tohoku catastrophes. In *WikiSym*, pages 105–113, 2011.

[KIO+16] Ankit Kumar, Ozan Irsoy, Peter Ondruska, Mohit Iyyer, James Bradbury, Ishaan Gulrajani, Victor Zhong, Romain Paulus, and Richard Socher. Ask me anything: Dynamic memory networks for natural language processing. In *International Conference on Machine Learning*, pages 1378–1387, 2016.

[KKT03] David Kempe, Jon Kleinberg, and Éva Tardos. Maximizing the spread of influence through a social network. In *Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 137–146. ACM, 2003.

[Kle02] Jon Kleinberg. Bursty and hierarchical structure in streams. In *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 91–101. ACM, 2002.

[KLPM10] Haewoon Kwak, Changhyun Lee, Hosung Park, and Sue Moon. What is twitter, a social network or a news media? In *Proceedings of the 19th international conference on World wide web*, pages 591–600. ACM, 2010.

[KN09] Nattiya Kanhabua and Kjetil Nørvåg. Using temporal language models for document dating. *Machine Learning and Knowledge Discovery in Databases*, pages 738–741, 2009.

[KPHN11] Ricardo Kawase, George Papadakis, Eelco Herder, and Wolfgang Nejdl. Beyond the usual suspects: context-aware revisitation support. In *Proceedings of the 22nd ACM conference on Hypertext and hypermedia*, pages 27–36. ACM, 2011.

[KTSD11] Anagha Kulkarni, Jaime Teevan, Krysta M Svore, and Susan T Dumais. Understanding temporal query dynamics. In *Proceedings of the fourth ACM international conference on Web search and data mining*, pages 167–176. ACM, 2011.

[KW14] Erdal Kuzey and Gerhard Weikum. Evin: Building a knowledge base of events. In *Proceedings of the 23rd International Conference on World Wide Web*, pages 103–106. ACM, 2014.

[LAP+09] Theodoros Lappas, Benjamin Arai, Manolis Platakis, Dimitrios Kotsakos, and Dimitrios Gunopulos. On burstiness-aware search for document sequences. In *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 477–486. ACM, 2009.

[LBH15] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *Nature*, 521(7553):436–444, 2015.

[LBS+16] Guillaume Lample, Miguel Ballesteros, Sandeep Subramanian, Kazuya Kawakami, and Chris Dyer. Neural architectures for named entity recognition. *CoRR*, abs/1603.01360, 2016.

[LC03] Xiaoyan Li and W Bruce Croft. Time-based language models. In *Proceedings of the twelfth international conference on Information and knowledge management*, pages 469–475. ACM, 2003.

[LGRC12] Janette Lehmann, Bruno Gonçalves, José J Ramasco, and Ciro Cattuto. Dynamical classes of collective attention in twitter. In *Proceedings of the 21st international conference on World Wide Web*, pages 251–260. ACM, 2012.

[LHLN15] Gang Luo, Xiaojiang Huang, Chin-Yew Lin, and Zaiqing Nie. Joint named entity recognition and disambiguation. In *Proc. EMNLP*, pages 879–880, 2015.

[LL16] Dmitry Lagun and Mounia Lalmas. Understanding user attention and engagement in online news reading. In *Proceedings of the Ninth ACM International Conference on Web Search and Data Mining*, pages 113–122. ACM, 2016.

[LLW+13] Xiaohua Liu, Yitong Li, Haocheng Wu, Ming Zhou, Furu Wei, and Yi Lu. Entity linking for tweets. In *ACL (1)*, pages 1304–1311, 2013.

[LM14] Quoc Le and Tomas Mikolov. Distributed representations of sentences and documents. In *Proceedings of the 31st International Conference on Machine Learning (ICML)*, pages 1188–1196, 2014.

[LMP+01] John Lafferty, Andrew McCallum, Fernando Pereira, et al. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of the 18th international conference on machine learning (ICML)*, volume 1, pages 282–289, 2001.

[LN89] Dong C Liu and Jorge Nocedal. On the limited memory BFG-S method for large scale optimization. *Mathematical programming*, 45(1-3):503–528, 1989.

[Log11] Robert H Logie. The functional organization and capacity limits of working memory. *Current Directions in Psychological Science*, 20(4):240–245, 2011.

[LPBA04] Agnès Lisowska, Andréi Popescu-Belis, and Susan Armstrong. *User Query Analysis for the Specification and Evaluation of a Dialogue Processing and Retrieval System*, pages 993–996. LREC 2004 (Fourth International Conference on Language Resources and Evaluation). ELRA - European Language Ressources Association, 2004.

[LRC+12] Heeyoung Lee, Marta Recasens, Angel Chang, Mihai Surdeanu, and Dan Jurafsky. Joint entity and event coreference resolution across documents. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 489–500. Association for Computational Linguistics, 2012.

[LS85] Donald Laming and Peter Scheiwiller. Retention in perceptual memory: A review of models and data. *Attention, Perception, & Psychophysics*, 37(3):189–197, 1985.

[LS99] Ora Lassila and Ralph R Swick. Resource description framework (rdf) model and syntax specification. *W3C*, 1999.

[LTB+13] Anna S Law, Steven L Trawley, Louise A Brown, Amanda N Stephens, and Robert H Logie. The impact of working memory load on task execution and online plan adjustment during multitasking in a virtual environment. *The Quarterly Journal of Experimental Psychology*, 66(6):1241–1258, 2013.

[LXC+14] Qi Liu, Biao Xiang, Enhong Chen, Hui Xiong, Fangshuang Tang, and Jeffrey Xu Yu. Influence maximization over large-scale social networks: A bounded linear approach. In *Proceedings of the 23rd ACM International Conference on Conference on Information and Knowledge Management*, pages 171–180. ACM, 2014.

[MAE+11] Dharmendra S Modha, Rajagopal Ananthanarayanan, Steven K Esser, Anthony Ndirango, Anthony J Sherbondy, and Raghavendra Singh. Cognitive computing. *Communications of the ACM*, 54(8):62–71, 2011.

[MBO14] Edgar Meij, Krisztian Balog, and Daan Odijk. Entity linking and retrieval for semantic search. In *Seventh ACM International Conference on Web Search and Data Mining, WSDM 2014, New York, NY, USA, February 24-28, 2014*, pages 683–684, 2014.

[Met07] Miriam J Metzger. Making sense of credibility on the web: Models for evaluating online information and recommendations for future research. *Journal of the American Society for Information Science and Technology*, 58(13):2078–2091, 2007.

[MFP00] Andrew McCallum, Dayne Freitag, and Fernando CN Pereira. Maximum entropy markov models for information extraction and segmentation. In *Icml*, volume 17, pages 591–598, 2000.

[MG85] STEPHEN P McKENNA and AL Glendon. Occupational first aid training: Decay in cardiopulmonary resuscitation (cpr) skills. *Journal of Occupational and Organizational Psychology*, 58(2):109–117, 1985.

[MH03] Sameer Maskey and Julia Hirschberg. Automatic summarization of broadcast news using structural features. In *INTERSPEECH*, 2003.

[MH05] Sameer Maskey and Julia Hirschberg. Comparing lexical, acoustic/prosodic, structural and discourse features for speech summarization. In *INTERSPEECH*, pages 621–624, 2005.

[MH06] Sameer Maskey and Julia Hirschberg. Summarizing speech without text using hidden markov models. In *Proceedings of the Human Language Technology Conference of the NAACL, Companion Volume: Short Papers*, pages 89–92. Association for Computational Linguistics, 2006.

[MH16] Xuezhe Ma and Eduard H. Hovy. End-to-end sequence labeling via bi-directional lstm-cnns-crf. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, ACL 2016, August 7-12, 2016, Berlin, Germany, Volume 1: Long Papers*, 2016.

[ML14] Seth A Myers and Jure Leskovec. The bursty dynamics of the twitter information network. In *Proceedings of the 23rd international conference on World wide web*, pages 913–924. ACM, 2014.

[MM31] John A McGeoch and William T McDonald. Meaningful relation and retroactive inhibition. *The American Journal of Psychology*, 43(4):579–588, 1931.

[MMBJ13] Yashar Moshfeghi, Michael Matthews, Roi Blanco, and Joemon M Jose. Influence of timeline and named-entity components on user engagement. In *European Conference on Information Retrieval*, pages 305–317. Springer, 2013.

[MMJ05] Martijn Meeter, Jaap M. J. Murre, and Steve M. J. Janssen. Remembering the news: Modeling retention data from a study with 14,000 participants. *Memory & Cognition*, 33(5):793–810, 2005.

[MMO14] Richard McCreadie, Craig Macdonald, and Iadh Ounis. Incremental update summarization: Adaptive sentence selection based on prevalence and novelty. In *Proceedings of the sixteenth ACM conference on Conference on information and knowledge management*, pages 301–310. ACM, 2014.

[MRMV08] Dan Morris, Meredith Ringel Morris, and Gina Venolia. Searchbar: a search-centric web history for task resumption and information re-finding. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 1207–1216. ACM, 2008.

[MRN14] Andrea Moro, Alessandro Raganato, and Roberto Navigli. Entity linking meets word sense disambiguation: a unified approach. *Transactions of the Association for Computational Linguistics*, 2:231–244, 2014.

[MSB+14] Christopher D Manning, Mihai Surdeanu, John Bauer, Jenny Rose Finkel, Steven Bethard, and David McClosky. The stanford corenlp natural language processing toolkit. In *ACL (System Demonstrations)*, pages 55–60, 2014.

[MSC+13a] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119, 2013.

[MSC+13b] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119, 2013.

[MSD13] Heiko Maus, Sven Schwarz, and Andreas Dengel. Weaving personal knowledge spaces into office applications. In *Integration of Practice-Oriented Knowledge Technology: Trends and Prospectives*, pages 71–82. Springer, 2013.

[MSHD11] Heiko Maus, Sven Schwarz, Jan Haas, and Andreas Dengel. CONTASK: Context-Sensitive Task Assistance in the Semantic Desktop. In *Enterprise Information Systems*, volume 73, pages 177–192, 2011.

[Mur02] Kevin Patrick Murphy. *Dynamic bayesian networks: representation, inference and learning*. PhD thesis, University of California, Berkeley, 2002.

[MW00] Inderjeet Mani and George Wilson. Robust temporal processing of news. In *Proceedings of the 38th annual meeting on Association for Computational Linguistics*, pages 69–76. Association for Computational Linguistics, 2000.

[MW08] David Milne and Ian H Witten. Learning to link with Wikipedia. In *Proceedings of the 17th ACM conference on Information and knowledge management*, pages 509–518. ACM, 2008.

[MWDR12] Edgar Meij, Wouter Weerkamp, and Maarten De Rijke. Adding semantics to microblog posts. In *Proceedings of the fifth ACM international conference on Web search and data mining*, pages 563–572. ACM, 2012.

[MWL+12] Xinfan Meng, Furu Wei, Xiaohua Liu, Ming Zhou, Sujian Li, and Houfeng Wang. Entity-centric topic-oriented opinion summarization in twitter. In *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 379–387. ACM, 2012.

[Mye86] Eugene W Myers. Ano (nd) difference algorithm and its variations. *Algorithmica*, 1(1-4):251–266, 1986.

[NKGL15] Claudia Niederée, Nattiya Kanhabua, Francesco Gallo, and Robert H Logie. Forgetful digital memory: Towards brain-inspired long-term data and information management. *ACM SIGMOD Record*, 44(2):41–46, 2015.

[NKNZ15] Tu Ngoc Nguyen, Nattiya Kanhabua, Claudia Niederée, and Xiaofei Zhu. A time-aware random walk model for finding important documents in web archives. In *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 915–918. ACM, 2015.

[NKTN18] Claudia Niederée, Nattiya Kanhabua, Tuan Tran, and Kaweh Djafari Naini. *Preservation Value and Managed Forgetting*, pages 101–129. Springer International Publishing, Cham, 2018.

[NP04] Ani Nenkova and Rebecca Passonneau. Evaluating content selection in summarization: The pyramid method. In *NAACL-HLT*, 2004.

[NS07] David Nadeau and Satoshi Sekine. A survey of named entity recognition and classification. *Lingvisticae Investigationes*, 30(1):3–26, 2007.

[OOD+13] O. Owoputi, B. O'Connor, C. Dyer, K. Gimpel, N. Schneider, and N. A. Smith. Improved part-of-speech tagging for online conversational text with word clusters. In *NAACL-HLT*, pages 380–390, 2013.

[OPM+12] M. Osborne, S. Petrovic, R. McCreadie, C. Macdonald, and I. Ounis. Bieber no more: First story detection using Twitter and Wikipedia. In *Workshop on Time-aware Information Access*, 2012.

[ORS+08] Christopher Olston, Benjamin Reed, Utkarsh Srivastava, Ravi Kumar, and Andrew Tomkins. Pig latin: a not-so-foreign language for data processing. In *Proceedings of the 2008 ACM SIGMOD international conference on Management of data*, pages 1099–1110. ACM, 2008.

[PAAG+10] José R Pérez-Agüera, Javier Arroyo, Jane Greenberg, Joaquin Perez Iglesias, and Victor Fresno. Using bm25f for semantic search. In *Proceedings of the 3rd international semantic search workshop*, page 2. ACM, 2010.

[PB07] Michael Pazzani and Daniel Billsus. Content-based recommendation systems. *The adaptive web*, pages 325–341, 2007.

[PDR13] Maria-Hendrike Peetz and Maarten De Rijke. Cognitive temporal document priors. In *Proceedings of European Conference in Information Retrieval*, pages 318–330. Springer, 2013.

[Pea05] Karl Pearson. The problem of the random walk. *Nature*, 72(1865):294, 1905.

[Pen09] Christian Pentzold. Fixing the floating gap: The online encyclopaedia wikipedia as a global memory place. *Memory Studies*, 2(2):255–272, 2009.

[PJ92] Boris T Polyak and Anatoli B Juditsky. Acceleration of stochastic approximation by averaging. *SIAM Journal on Control and Optimization*, 30(4):838–855, 1992.

[PKHN11] George Papadakis, Ricardo Kawase, Eelco Herder, and Claudia Niederée. A layered approach to revisitation prediction. In *ICWE*, volume 6757, pages 258–273, 2011.

[PMZ10] Jeffrey Pound, Peter Mika, and Hugo Zaragoza. Ad-hoc object retrieval in the web of data. In *Proceedings of the 19th international conference on World wide web*, pages 771–780. ACM, 2010.

[RAGM11] Kira Radinsky, Eugene Agichtein, Evgeniy Gabrilovich, and Shaul Markovitch. A word at a time: computing word relatedness using temporal semantic analysis. In *Proceedings of the 20th international conference on World wide web*, pages 337–346. ACM, 2011.

[RCD+03] Meredith Ringel, Edward Cutrell, Susan Dumais, Eric Horvitz, et al. Milestones in time: The value of landmarks in retrieving information from personal stores. In *Proceedings of INTERACT conference*, volume 2003, pages 184–191, 2003.

[RCE+11] Alan Ritter, Sam Clark, Oren Etzioni, et al. Named entity recognition in tweets: an experimental study. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 1524–1534. Association for Computational Linguistics, 2011.

[RD06] Matthew Richardson and Pedro Domingos. Markov logic networks. *Machine learning*, 62(1):107–136, 2006.

[Rea00] James Reason. Human error: models and management. *Western Journal of Medicine*, 172(6):393, 2000.

[RHW+88] David E Rumelhart, Geoffrey E Hinton, Ronald J Williams, et al. Learning representations by back-propagating errors. *Cognitive modeling*, 5(3):1, 1988.

[RHW99] David C Rubin, Sean Hinton, and Amy Wenzel. The precise time course of retention. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, 25(5):1161, 1999.

[RJ86] Lawrence Rabiner and B Juang. An introduction to hidden markov models. *ieee assp magazine*, 3(1):4–16, 1986.

[RL14] Maria Wolters Robert Logie, Elaine Niven. D2.2: Foundations of forgetting and remembering - preliminary report. Deliverable, ForgetIT consortium, March 2014.

[RMK11] Daniel M Romero, Brendan Meeder, and Jon Kleinberg. Differences in the mechanics of information diffusion across topics: idioms, political hashtags, and complex contagion on twitter. In *Proceedings of the 20th international conference on World wide web*, pages 695–704. ACM, 2011.

[RR09] Lev Ratinov and Dan Roth. Design challenges and misconceptions in named entity recognition. In *Proceedings of the Thirteenth Conference on Computational Natural Language Learning*, pages 147–155. Association for Computational Linguistics, 2009.

[RRDA11] Lev Ratinov, Dan Roth, Doug Downey, and Mike Anderson. Local and global algorithms for disambiguation to wikipedia. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1*, pages 1375–1384. Association for Computational Linguistics, 2011.

[RRS11] Francesco Ricci, Lior Rokach, and Bracha Shapira. Introduction to recommender systems handbook. In *Recommender systems handbook*, pages 1–35. Springer, 2011.

[RV97] Paul Resnick and Hal R Varian. Recommender systems. *Communications of the ACM*, 40(3):56–58, 1997.

[RW96] David C Rubin and Amy E Wenzel. One hundred years of forgetting: A quantitative description of retention. *Psychological review*, 103(4):734, 1996.

[SBDS14] Marta Sabou, Kalina Bontcheva, Leon Derczynski, and Arno Scharl. Corpus annotation through crowdsourcing: Towards best practice guidelines. In *LREC*, pages 859–866, 2014.

[SBV+15] Alessandro Sordoni, Yoshua Bengio, Hossein Vahabi, Christina Lioma, Jakob Grue Simonsen, and Jian-Yun Nie. A hierarchical recurrent encoder-decoder for generative context-aware query suggestion. In *Proceedings of the 24th ACM International Conference on Information and Knowledge Management, CIKM 2015, Melbourne, VIC, Australia, October 19 - 23, 2015*, pages 553–562, 2015.

[SC13] Uma Sawant and Soumen Chakrabarti. Learning joint query interpretation and response ranking. In *Proceedings of the 22nd international conference on World Wide Web*, pages 1099–1110. ACM, 2013.

[Sch99] Daniel L Schacter. The seven sins of memory: Insights from psychology and cognitive neuroscience. *American psychologist*, 54(3):182, 1999.

[SG05] Craig Soules and Gregory Ganger. Connections: Using context to enhance file search. In *SOSP*, pages 119–132, 2005.

[SG10a] Dafna Shahaf and Carlos Guestrin. Connecting the dots between news articles. In *Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 623–632. ACM, 2010.

[SG10b] Jannik Strötgen and Michael Gertz. Heideltime: High quality rule-based extraction and normalization of temporal expressions. In *Proceedings of the 5th International Workshop on Semantic Evaluation*, pages 321–324. Association for Computational Linguistics, 2010.

[SG13] Jannik Strötgen and Michael Gertz. Multilingual and cross-domain temporal tagging. *Language Resources and Evaluation*, 47(2):269–298, 2013.

[SGH12a] Dafna Shahaf, Carlos Guestrin, and Eric Horvitz. Trains of thought: Generating information maps. In *Proceedings of the 21st International Conference on World Wide Web*, WWW '12, pages 899–908, 2012.

[SGH12b] Dafna Shahaf, Carlos Guestrin, and Eric Horvitz. Trains of thought: Generating information maps. In *Proceedings of the 21st international conference on World Wide Web*, pages 899–908. ACM, 2012.

[SH10] Gabor Szabo and Bernardo A Huberman. Predicting the popularity of online content. *Communications of the ACM*, 53(8):80–88, 2010.

[SKW07] Fabian M Suchanek, Gjergji Kasneci, and Gerhard Weikum. YAGO: a core of semantic knowledge. In *WWW*, pages 697–706, 2007.

[SLW11] B. Sparrow, J. Liu, and D. M. Wegner. Google effects on memory: Cognitive consequences of having information at our fingertips. *Science*, 333:776–778, 2011.

[SRH08] Pavel Serdyukov, Henning Rode, and Djoerd Hiemstra. Modeling multi-step relevance propagation for expert finding. In *Proceedings of the 17th ACM conference on Information and knowledge management*, pages 1133–1142. ACM, 2008.

[SRW07] Swapna Somasundaran, Josef Ruppenhofer, and Janyce Wiebe. Detecting arguing and sentiment in meetings. In *Proceedings of the SIGdial Workshop on Discourse and Dialogue*, volume 6, 2007.

[SSGN07] Sam Shah, Craig A. N. Soules, Gregory R. Ganger, and Brian D. Noble. Using provenance to aid in personal file search. In *2007 USENIX Annual Technical Conference on Proceedings of the USENIX Annual Technical Conference*, ATC'07, pages 13:1–13:14, 2007.

[SvED07] Leo Sauermann, Ludger van Elst, and Andreas Dengel. PIMO – A Framework for Representing Personal Information Models. In *I-SEMANTICS*, J.UCS, pages 270–277, 2007.

[SWF+15] Sainbayar Sukhbaatar, Jason Weston, Rob Fergus, et al. End-to-end memory networks. In *Advances in neural information processing systems*, pages 2440–2448, 2015.

[SWLW13] Wei Shen, Jianyong Wang, Ping Luo, and Min Wang. Linking named entities in tweets with knowledge base via user interest modeling. In *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 68–76. ACM, 2013.

[SY13] Avirup Sil and Alexander Yates. Re-ranking for joint named-entity recognition and linking. In *Proceedings of the 22nd ACM international conference on Conference on information & knowledge management*, pages 2369–2374. ACM, 2013.

[TAJP07] Jaime Teevan, Eytan Adar, Rosie Jones, and Michael AS Potts. Information re-retrieval: repeat queries in yahoo's logs. In *Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 151–158. ACM, 2007.

[TCG+14] Tuan A. Tran, Andrea Ceroni, Mihai Georgescu, Kaweh Djafari Naini, and Marco Fisichella. Wikipevent: Leveraging wikipedia edit history for event detection. In *Web Information Systems Engineering - WISE 2014 - 15th International Conference, Thessaloniki, Greece, October 12-14, 2014, Proceedings, Part II*, pages 90–108, 2014.

[Tee06] Jaime Teevan. How people recall search result lists. In *CHI'06 Extended Abstracts on Human Factors in Computing Systems*, pages 1415–1420. ACM, 2006.

[TGK+12] Nina Tahmasebi, Gerhard Gossen, Nattiya Kanhabua, Helge Holzmann, and Thomas Risse. NEER: An unsupervised method for named entity evolution recognition. In *COLING*, 2012.

[TGZK14] Tuan Tran, Mihai Georgescu, Xiaofei Zhu, and Nattiya Kanhabua. Analysing the duration of trending topics in twitter using wikipedia. In *Proceedings of the 2014 ACM Conference on Web Science*, WebSci '14, pages 251–252, New York, NY, USA, 2014. ACM.

[TKS+16] Christoph Trattner, Dominik Kowald, Paul Seitlinger, Tobias Ley, Simone Kopeinik, et al. Modeling activation processes in human memory to predict the use of tags in social bookmarking systems. *J. Web Science*, 2(1):1–16, 2016.

[TN14] Tuan A. Tran and Tu Ngoc Nguyen. Hedera: Scalable indexing, exploring entities in wikipedia revision history. In *Proceedings of the ISWC 2014 Posters & Demonstrations Track a track within the 13th International Semantic Web Conference, ISWC 2014, Riva del Garda, Italy, October 21, 2014.*, pages 297–300, 2014.

[TNK+15] Tuan A Tran, Claudia Niederee, Nattiya Kanhabua, Ujwal Gadiraju, and Avishek Anand. Balancing novelty and salience: Adaptive learning to rank entities for timeline summarization of high-impact events. In *Proceedings of the 24th ACM International Conference on Information and Knowledge Management*, pages 1201–1210. ACM, 2015.

[Tof07] Brad Tofel. Wayback for accessing web archives. In *Proceedings of the 7th International Web Archiving Workshop*, pages 27–37, 2007.

[Tra12] Tuan A. Tran. Exploiting temporal topic models in social media retrieval. In *Proceedings of the 35th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '12, pages 999–999, New York, NY, USA, 2012. ACM.

[TSN+16] Tuan A. Tran, Sven Schwarz, Claudia Niederée, Heiko Maus, and Nattiya Kanhabua. The forgotten needle in my collections: Task-aware ranking of documents in semantic information space. In *Proceedings of the 2016 ACM on Conference on Human Information Interaction and Retrieval*, CHIIR '16, pages 13–22, New York, NY, USA, 2016. ACM.

[TTN17] Nam Khanh Tran, Tuan Tran, and Claudia Niederée. Beyond time: Dynamic context-aware entity recommendation. In *European Semantic Web Conference*, pages 353–368. Springer, 2017.

[TTT+13a] Giang Binh Tran, Tuan Tran, Nam-Khanh Tran, Mohammad Alrifai, and Nattiya Kanhabua. Leverage learning to rank in an optimization framework for timeline summarization. In *TAIA Workshop at SIGIR*, 2013.

[TTT+13b] Giang Binh Tran, Tuan A Tran, Nam-Khanh Tran, Mohammad Alrifai, and Nattiya Kanhabua. Leveraging learning to rank in an optimization framework for timeline summarization. In *SIGIR 2013 Workshop on Time-aware Information Access*. ACM, 2013.

[TTTHJ15] Tuan Tran, Nam Khanh Tran, Asmelash Teka Hadgu, and Robert Jäschke. Semantic annotation for microblog topics using wikipedia temporal information. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 97–106, Lisbon, Portugal, September 2015. Association for Computational Linguistics.

[Tul86] Endel Tulving. Episodic and semantic memory: Where should we go from here? *Behavioral and Brain Sciences*, 9(03):573–577, 1986.

[TWM12] Partha Pratim Talukdar, Derry Wijaya, and Tom Mitchell. Coupled temporal scoping of relational facts. In *Proceedings of the fifth ACM international conference on Web search and data mining*, pages 73–82. ACM, 2012.

[Und57] Benton J Underwood. Interference and forgetting. *Psychological review*, 64(1):49, 1957.

[UNR+14] Ricardo Usbeck, Axel-Cyrille Ngonga Ngomo, Michael Röder, Daniel Gerber, Sandro Athaide Coelho, Sören Auer, and Andreas Both. AGDISTIS-graph-based disambiguation of named entities using linked data. In *International Semantic Web Conference*, pages 457–471. Springer, 2014.

[vdHE14] Elise van den Hoven and Berry Egge. The cue is key - design for real-life remembering. *Zeitschrift für Psychologie.*, 222(2):110–117, 2014.

[VSBN07] Lucy Vanderwende, Hisami Suzuki, Chris Brockett, and Ani Nenkova. Beyond sumbasic: Task-focused summarization with sentence simplification and lexical expansion. *Information Processing & Management*, 43(6), 2007.

[VTN+16] Khoi Duy Vo, Tuan Tran, Tu Ngoc Nguyen, Xiaofei Zhu, and Wolfgang Nejdl. Can we find documents in web archives without knowing their contents? In *Proceedings of the 8th ACM Conference on Web Science*, WebSci '16, pages 173–182, New York, NY, USA, 2016. ACM.

[WBD10] Ryen W. White, Paul N. Bennett, and Susan T. Dumais. Predicting short-term interests using activity-based search context. In *Proceedings of the 19th ACM international conference on Information and knowledge management*, CIKM '10, pages 1009–1018, 2010.

[WG13] Zhaohui Wu and C Lee Giles. Measuring term informativeness in context. In *NAACL-HLT*, 2013.

[Whi01] K Geoffrey White. Forgetting functions. *Learning & Behavior*, 29(3):193–207, 2001.

[Whi12] Tom White. *Hadoop: The definitive guide*. " O'Reilly Media, Inc.", 2012.

[Wik17] Wikipedia. Automatic Summarization. https://en.wikipedia.org/wiki/Automatic_summarization, 2017. Accessed: 2017-03-08.

[WJA14] Stewart Whiting, Joemon Jose, and Omar Alonso. Wikipedia as a time machine. In *Proceedings of the 23rd International Conference on World Wide Web*, pages 857–862. ACM, 2014.

[WKY13] Tetsutaro Watanabe, Takashi Kobayashi, and Haruo Yokota. Fridal: A desktop search system based on latent interfile relationships. In *Software and Data Technologies*, pages 220–234, 2013.

[WML10] Lidan Wang, Donald Metzler, and Jimmy Lin. Ranking under temporal constraints. In *Proceedings of the 19th ACM international conference on Information and knowledge management*, pages 79–88. ACM, 2010.

[WPF+99] Ian H Witten, Gordon W Paynter, Eibe Frank, Carl Gutwin, and Craig G Nevill-Manning. Kea: Practical automatic keyphrase extraction. In *Proceedings of the fourth ACM conference on Digital libraries*, pages 254–255. ACM, 1999.

[WQS+15] Peilu Wang, Yao Qian, Frank K. Soong, Lei He, and Hai Zhao. Part-of-speech tagging with bidirectional long short-term memory recurrent neural network. *CoRR*, abs/1510.06168, 2015.

[WRC+13] Lu Wang, Hema Raghavan, Vittorio Castelli, Radu Florian, and Claire Cardie. A sentence compression based framework to query-focused multi-document summarization. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics*, pages 1384–1394, 2013.

[WS54] Robert Sessions Woodworth and Harold Schlosberg. *Experimental psychology*. Oxford and IBH Publishing, 1954.

[WTH11] Kuansan Wang, Christopher Thrasher, and Bo-June Paul Hsu. Web scale NLP: a case study on url word breaking. In *Proceedings of the 20th international conference on World wide web*, pages 357–366. ACM, 2011.

[WZQ+10] Yafang Wang, Mingjie Zhu, Lizhen Qu, Marc Spaniol, and Gerhard Weikum. Timely yago: harvesting, querying, and visualizing temporal knowledge from wikipedia. In *Proceedings of the 13th International Conference on Extending Database Technology*, pages 697–700. ACM, 2010.

[XLL08] Shasha Xie, Yang Liu, and Hui Lin. Evaluating the effectiveness of features and sampling in extractive meeting summarization. In *Spoken Language Technology Workshop, 2008. SLT 2008. IEEE*, pages 157–160. IEEE, 2008.

[XLW+08] Fen Xia, Tie-Yan Liu, Jue Wang, Wensheng Zhang, and Hang Li. Listwise approach to learning to rank: Theory and algorithm. In *Proceedings of the 25th International Conference on Machine Learning*, ICML '08, pages 1192–1199, New York, NY, USA, 2008. ACM.

[YL11] Jaewon Yang and Jure Leskovec. Patterns of temporal variation in online media. In *Proceedings of the fourth ACM international conference on Web search and data mining*, pages 177–186. ACM, 2011.

[YWO+11] Rui Yan, Xiaojun Wan, Jahna Otterbacher, Liang Kong, Xiaoming Li, and Yan Zhang. Evolutionary timeline summarization: a balanced optimization framework via iterative substitution. In *Proceedings of the 34th international ACM SIGIR conference on Research and development in Information Retrieval*, pages 745–754. ACM, 2011.

[ZCZ+09] Ruiqiang Zhang, Yi Chang, Zhaohui Zheng, Donald Metzler, and Jian-yun Nie. Search result re-ranking by feedback control adjustment for time-sensitive query. In *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics, Companion Volume: Short Papers*, pages 165–168. Association for Computational Linguistics, 2009.

[ZGS14] Maxim Zhukovskiy, Gleb Gusev, and Pavel Serdyukov. Supervised nested pagerank. In *Proceedings of the 23rd ACM International Conference on Conference on Information and Knowledge Management*, pages 1059–1068. ACM, 2014.

[ZGY+13] Xin Wayne Zhao, Yanwei Guo, Rui Yan, Yulan He, and Xiaoming Li. Timeline generation with social attention. In *Proceedings of the 36th international ACM SIGIR conference on Research and development in information retrieval*, pages 1061–1064. ACM, 2013.

[ZKGN14] Zhe Zuo, Gjergji Kasneci, Toni Gruetze, and Felix Naumann. Bel: Bagging for entity linking. In *COLING*, pages 2075–2086, 2014.

[ZRZ16] Lei Zhang, Achim Rettinger, and Ji Zhang. A probabilistic model for time-aware entity recommendation. In *International Semantic Web Conference (1)*, pages 598–614, 2016.