INFORMATION EXTRACTION FROM DIGITAL SOCIAL TRACE DATA WITH APPLICATIONS TO SOCIAL MEDIA AND SCHOLARLY COMMUNICATION DATA

BY

SHUBHANSHU MISHRA

DISSERTATION

Submitted in partial fulfillment of the requirements
for the degree of Doctor of Philosophy in Library and Information Science
in the Graduate College of the
University of Illinois at Urbana-Champaign, 2020

Urbana, Illinois

Doctoral Committee:

Associate Professor Jana Diesner, Chair
Associate Professor Vetle I. Torvik
Professor Karrie G. Karahalios
Professor Robert J. Brunner

Abstract

Information extraction (IE) aims at extracting structured data from unstructured or semi-structured data. The thesis starts by identifying social media data and scholarly communication data as a special case of digital social trace data (DSTD). This identification allows us to utilize the graph structure of the data (e.g., user connected to a tweet, author connected to a paper, author connected to authors, etc.) for developing new information extraction tasks. The thesis focuses on information extraction from DSTD, first, using only the text data from tweets and scholarly paper abstracts, and then using the full graph structure of Twitter and scholarly communications datasets. This thesis makes three major contributions.

First, new IE tasks based on DSTD representation of the data are introduced. For scholarly communication data, methods are developed to identify article and author level novelty and expertise. Furthermore, interfaces for examining the extracted information are introduced. A social communication temporal graph (SCTG) is introduced for comparing different communication data like tweets tagged with sentiment, tweets about a search query, and Facebook group posts. For social media, new text classification categories are introduced, with the aim of identifying enthusiastic and supportive users, via their tweets. Additionally, the correlation between sentiment classes and Twitter meta-data in public corpora is analyzed, leading to the development of a better model for sentiment classification.

Second, methods are introduced for extracting information from social media and scholarly data. For scholarly data, a semi-automatic method is introduced for the construction of a large-scale taxonomy of computer science concepts. The method relies on the Wikipedia category tree. The constructed taxonomy is used for identifying key computer science phrases in scholarly papers, and tracking their evolution over time. Similarly, for social media data, machine learning models based on human-in-the-loop learning, semi-supervised learning, and multi-task learning are introduced for identifying sentiment, named entities, part of speech tags, phrase chunks, and super-sense tags. The machine learning models are developed with a focus on leveraging all available data. The multi-task models presented here result in competitive performance against other methods, for most of the tasks, while reducing inference time computational costs.

Finally, this thesis has resulted in the creation of multiple open source tools and public data sets, which

can be utilized by the research community. The thesis aims to act as a bridge between research questions and techniques used in DSTD from different domains. The methods and tools presented here can help advance work in the areas of social media and scholarly data analysis. All resources related to this thesis are available at https://shubhanshu.com/phd_thesis/

Dedicated to three leaders in my life

Mrs. Pushpa Mishra,

Mr. Krishna Shankar Mishra,

and Mrs. Shraddha Pandey.

Acknowledgments

This PhD has been a long journey, and it would not have been possible without the support, encouragement, and guidance of many individuals.

I would like to start this acknowledgement by thanking my chair and advisor, Jana Diesner, who guided me and believed in me throughout this process. Thanks for introducing me to the field of computational social science, as well as to all the exciting projects (many of which are part of this thesis). I am grateful that you always encouraged me to finish projects, learn and apply new ideas, and be mentally and physically fit; all of which made this PhD process very memorable. Because of you, I was able to develop confidence as a researcher. Your focus on research and data ethics has helped me to understand my responsibilities as a scholar. I would also like to thank my co-advisor Vetle I. Torvik, for introducing me to the field of science of science. I learnt a lot from our discussions on academic critique, precise language usage, and analytical rigour. You taught me the importance of critically understanding our research data. I owe tremendous gratitude to both Jana and Vetle, for providing timely feedback with regards to my research as well as career goals. I am grateful to my other committee members — Karrie Karahalios, for always encouraging me since the social visualizations class; and Robert J. Brunner, for providing feedback on the deep learning aspect of my work which gave me confidence to apply this approach.

I would also take this moment to thank my class 12th computer science teacher Mr. Vinay Singh, who taught me the value in learning outside the curriculum, which made me excited to pursue a career in research.

Many parts of this thesis are based on collaborative work with Jinseok Kim, Brent D. Fegley, Amirhossein Aleyasen, Sneha Agarwal, Liang Tao, Jinlong "Kenny" Guo, Kirstin Phelps, Johna Picco, Jingxian Zhang, Hyejin Lee, Jason Byrne, and Elizabeth Surbeck. I would like to thank you for your contributions, and discussions. I am thankful to Daniel Collier for helping me to expand my research to the domain of higher education. I would like to thank my lab-mates Shadi, Ming, Julian, Mihai, Ly, Aseel, Yingjun, Niko, Tiffany, Kanyao, and Pingjing for all the collaborations, help, and enlightening discussions. Additionally, my research experience would have been difficult without people who took care of all the non-research work for me. I would like to extend my heartfelt thanks to whole the iSchool staff, including everyone at the front-desk,

the administrative office, and the help desk. Special thanks to Penny Ames for being patient with my thesis submission and always being there to help with all the academic matters. I am extremely grateful to Linda Smith for providing me support when it was needed the most.

Thanks, Atul, Nikita, Shivangi, Ravi, and Harshala for uncountable adventures, and all the potlucks. Thanks, Kaushik, Mansi, Ramsai, Harpreet, for all the fun times, especially in the first years.

I owe extreme gratitude to my mother Pushpa Mishra, father Krishna Shankar Mishra, and my sister Shraddha, to whom I dedicate this dissertation. Apart from your support, the three of you always collectively taught me perseverance, patience, and the pursuit of happiness. I was fortunate to have my brother, Sudhanshu, who entertained me throughout my whole academic journey as well as took care of things at home. I am grateful to have the support and affection of the rest of my family, you have always had my back. Lastly, this journey would not have been possible without my wife Shivangi. You gave me confidence to apply for a PhD at UIUC, and I couldn't be happier that you chose to stick around with me. Having you by my side, made this journey memorable. You have taught me "*to take a break*" and that "*time fixes everything*". You ensured that I finish this journey healthy and happy.

In the end, I would like to thank all the funding agencies and sponsors who supported me during my PhD: FORD Foundation (Grant no. 0125-6162), Anheuser Busch, National Institute on Aging of the NIH (Award Number P01AG039347), the Directorate for Education and Human Resources of the NSF (Award Number 1348742), Korea Institute of Science and Technology Information (Grant No. C2210). During the PhD I was also funded by the iSchool Graduate College Fellowships. I am also thankful to the iSchool and the Smith Endowment for Student Travel Funds for helping me present my work at multiple venues. I was also awarded the Microsoft Azure Research Award between 2016-2018. Many experiments presented in this thesis were possible because of the free GPU servers provided by the Google Colab platform. The content reported in this thesis is solely my responsibility and does not necessarily represent the official views of any of the funding agencies or sponsors.

Table of Contents

List of Tables

List of Figures

List of Abbreviations

DSTD	Digital Social Trace Data
DNN	Deep Neural Network
HITL	Human In The Loop
IE	Information Extraction
MeSH	Medical Subject Headings
ML	Machine Learning
MTL	Multi Task Learning
NER	Named Entity Recognition
NLP	Natural Language Processing
POS	Part-Of-Speech
SCTG	Social Communication Temporal Graph
SSL	Semi-Supervised Learning
SVM	Support Vector Machine

Chapter 1

Introduction

What does it mean to extract information from data? Does a collection of news articles laying on the floor qualify as data? Is this collections suitable for knowing a) the prevalence of Ebola in news?, b) years since the first diagnosis of Ebola in a human?, or c) which journalists have done the most reporting on Ebola?

In order to answer the questions above we need to distinguish between information and data. Information is a latent property of a data and is not visible unless the data is observed in the right format. Information Extraction (IE) [Sarawagi 2008] deals with the process of converting a raw data into a format which reveals its information (see figure 1.1 for an illustration). For example, in our collection of news articles, an IE system may proceed to identify the salient topics or concepts, and arrange them into some form of knowledge graph. This can help us answer question a). However, unless we know the information about publication date and author of a news article, we cannot answer the two remaining questions solely based on the text of the news reports.

Being able to answers the above set of questions will be an effective way to validate social science theories via data from the domains of social media and scholarly publishing. The current rise in usage of social media platforms like Twitter, Facebook, Reddit, etc. has given scholars a suitable benchmark to test existing social science theories [Kosinski et al. 2015, Wilson et al. 2012, Miller 2011, Lazer et al. 2009, Kwak et al. 2010, Diesner and Chin 2015, Diesner et al. 2014, Kim 2014]. Similarly, scholarly data has long been utilized to evaluate social science theories of collaboration, knowledge evolution, and social expertise.

The approach described above can be facilitated by utilizing text data along with the associated metadata (e.g., date of publication, author identity, etc.) in each domain. However, this will require using tools or techniques which can efficiently and accurately extract information from the text and combine it with the information from the metadata. Unfortunately, existing IE systems have been found to poorly transfer their accuracy to text from new domains, compared to text from news corpora (for which most IE systems were made). Especially for social media text (also known as Noisy User-generated Text (NUT) [Baldwin et al. 2015]), this reduction in accuracy of IE systems is a major bottleneck. A major reason for this poor performance is the usage of non-traditional vocabulary, word forms, and short message length [Eisenstein

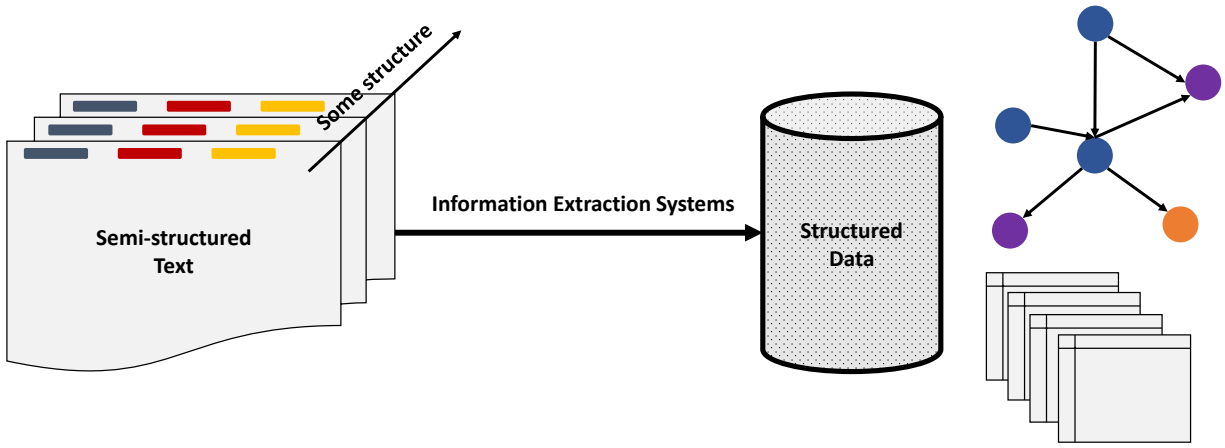


Figure 1.1: Illustration of information extraction. Converting unstructured data to structured data.

2013].

In this thesis I aim to develop techniques that facilitate answering the above mentioned questions (a)-(c) for the domain of social media and scholarly data by utilizing a specific representation of data called Digital Social Trace Data (DSTD). Furthermore, we will show the utility of the DSTD representation by answering novel questions in these domains.

1.1 Digital Social Trace Data (DSTD)

In order to make IE accessible to social scientists and applicable to a broad range of data sets, we introduce an abstraction called Digital Social Trace Data (DSTD). DSTD is based on the concept of digital social trace data [Diesner and Chin 2015] and digital trace data [Howison et al. 2011]. DSTD are digital activity traces generated by individuals as part of a social interactions, such as interactions on social media websites like Twitter, Facebook; or in scientific publications. A DSTD has the following properties:

- Temporal information associated to each item of the data
- Presence of connection between various data items
- Optionally associated metadata for data items.

DSTD are very similar to heterogeneous information networks (HINs) [Sun and Han 2012], and temporal

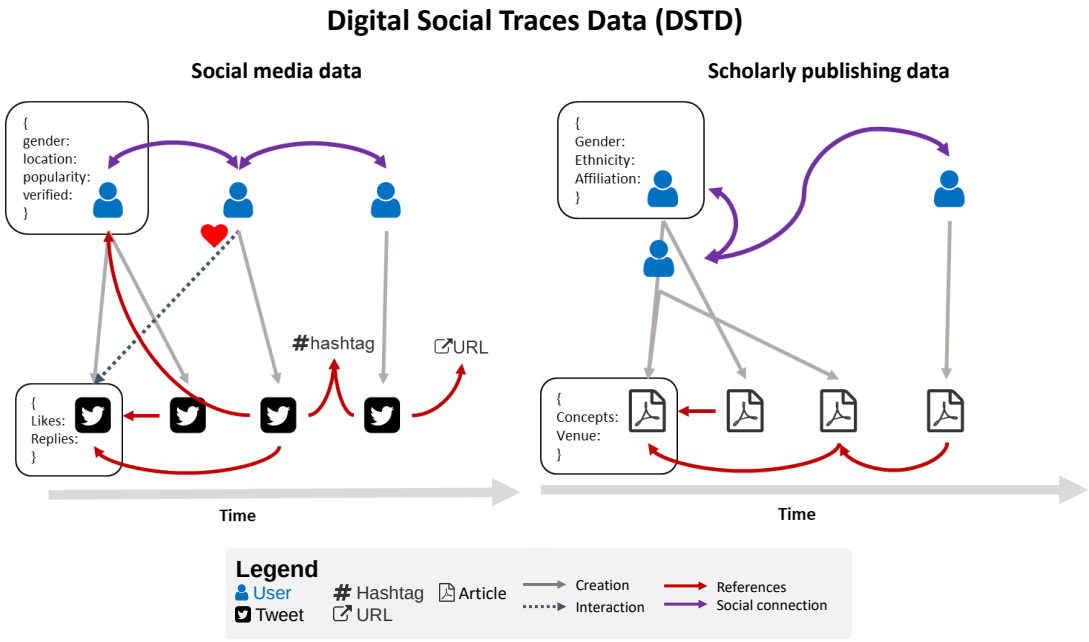


Figure 1.2: Illustration of digital social trace data

networks [Holme and Saramäki 2012] in terms of their representation and properties. HINs represent information network as a "typed semi-structured heterogeneous networks" [Sun and Han 2012], where nodes and edges are typed. On the other hand, DSTD are focused on simplifying this relationship into three core components mentioned above. Similarly, temporal networks only capture the temporal evolution of the network and are commonly discussed in the homogeneous network setting. Again, DSTD expand on this definition by including metadata attributes. Finally, DSTD are described in a social science setting as opposed to the graph theoretic setting. This will ensure effective communication of DSTD to the social science community. Figure 1.2 gives an illustration of DSTD for social media and scholarly publishing data.

1.1.1 DSTD in social media

An example of DSTD will be an individual's Twitter feed, or a collection of all the tweets about *Hurricane Irma*. Both these datasets will have a temporal distribution as the tweets are generated in a temporal order. Furthermore, the interaction of tweets with each other satisfies the connection presence. This can be seen when other users re-tweet a given tweet, or reply to it. Finally, tweets carry a large amount of metadata both about their content as well as about their creator, e.g., tweets may be tagged by a geo-location, or contain a URL. The metadata about the user of the tweet usually contains the user's location, number of followers, friends, and current number of posted-tweets. Furthermore, social dialogues can be identified as

DSTD.

1.1.2 DSTD in scholarly publishing

Similar to social media data, scholarly publishing data can also be considered an instance of DSTD. Often, scholarly publication datasets consists of articles, their associated metadata, and the citation network. However, this dataset can be expanded to include authors and publication data, to convert the dataset into an instance of DST. Figure 1.2 shows how scholarly publishing data is an instance of DST.

1.1.3 DSTD in other domains

Many datasets can be modeled as DSTD by ensuring certain properties of the dataset are available–such as time-stamp of individual activities; connection between individuals and items; and metadata associated with each individual and item. More specific examples are discussed in chapter 4.

1.2 Information extraction from DSTD

What? IE requires the specification of what information to extract. Earlier research in IE has focused on improving search engine results quality, text classification, or question answering systems. However, with IE on DSTD, we can investigate more complex questions, e.g., how do concepts evolve over time, and how do individuals interact with these evolving concepts? In scholarly publishing, publications are temporally ordered, have connections between each other via citations and co-authorship networks, and are tagged with additional metadata such as concepts, publication venue, author's prior publication count, and author's h-index. Using this data, additional information can be extracted such as an overall conceptual novelty [Mishra and Torvik 2016] and expertise of an article or author [Mishra et al. 2018a]. This information can also be utilized to assess patterns of self-citation in authors as shown in Mishra et al. [2018c]. For problems of text classification, the DSTD representation allows us to investigate if $p(label|text, user, time)$ is a better model compared to $p(label|text)$.

How? Additionally, a major challenge in building any IE system is to design evaluation data to assess the quality of the extracted information. Furthermore, with the advancements in supervised learning algorithms, there is a demand for building larger and higher quality training corpora for any IE system [Mishra et al. 2015, Mishra and Diesner 2016]. In the domain of social media DSTD, this training data is very scarce, annotated based on varying guidelines, and can be biased to certain time spans, domains, or geo-locations [Eisenstein 2013]. This challenge promotes an investigation into how to efficiently extract this information.

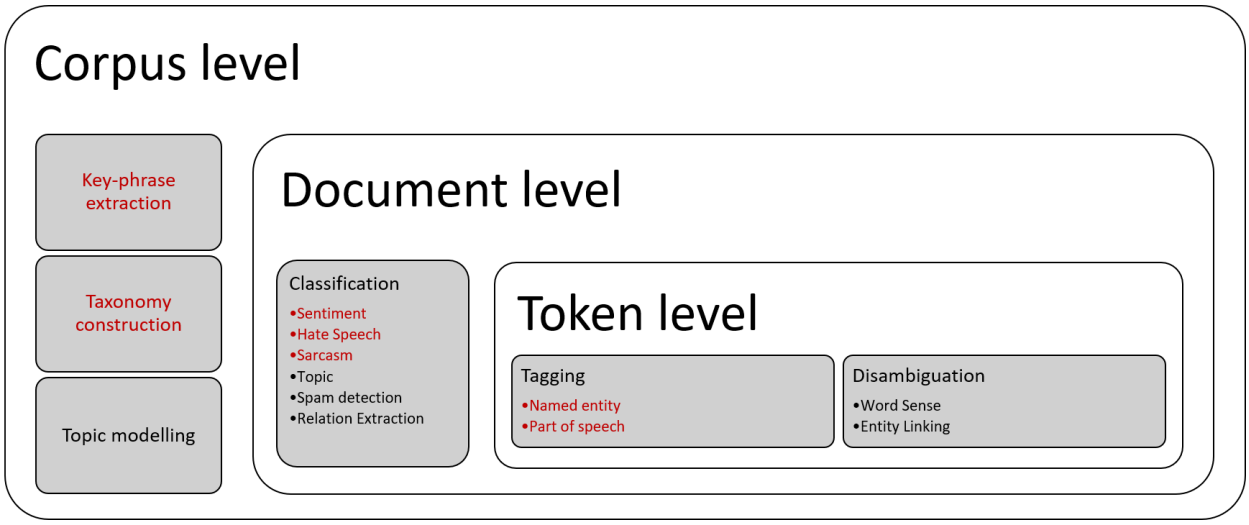


Figure 1.3: A hierarchy of IE tasks for text. Tasks in red are addressed in detail in this thesis.

Specifically, how can the existing annotated sources be best utilized to bring the social media IE systems on par with existing IE systems? Similarly, for scholarly articles in computer science, there is a lack of controlled vocabularies for identifying concepts.

Applications Next, once information is extracted, it needs to be made accessible to the social science community. Specifically, the presentation of the extracted information should follow the structure of the information, i.e. temporal, connected, and metadata enriched. However, existing IE systems, if not focused on these aspects, cause a difficulty in interpreting the extracted information. Here lies the third challenge, how to make the extracted information more accessible.

1.3 Information Extraction from Text

Often DSTD data include some text based constituents. In order to convert this text data into something we can query on, we utilize information extraction tasks defined for text data. Figure 1.3 proposes a task hierarchy of possible IE tasks for text data based on the unit of analysis (e.g., tokens, documents, or corpus) in these tasks. The task hierarchy consists of the following types of tasks:

- **Token Level**: These tasks result in an output for each token (e.g., if a word belongs to a named entity class) in the text. Common examples are named entity extraction, part of speech tagging, chunking, super-sense tagging, and dependency parsing.
- **Document Level**: These tasks result in an output for the whole document (e.g., sentiment for a tweet) in a corpus. Common examples are opinion mining, geo-location prediction of tweets, and

concept identification of scholarly articles.

- **Corpus Level**: These tasks result in an output for the whole corpus (e.g., topics in all tweets about 2016 US elections). Common examples are key-phrase extraction, taxonomy construction, and topic modelling.

Below we will discuss some of the prominent tasks from each type in the task hierarchy and underline their utility towards extracting information from DSTD in social media data.

1.3.1 Named entity recognition, classification and linking

Named entity recognition [Sarawagi 2008] is the identification of named entities in text (see figure 1.4). It is a token level task. Here, named entities are single or multi word units of text, e.g., *Barack Hussein Obama*. Furthermore, named entities are usually tagged with the type of entity, e.g., *Barack Hussein Obama* is an entity of type *person*, it can also be an entity of type *political figure* given the context of the text. This is commonly referred to as named entity classification. A common task in IE is to perform named entity recognition and classification together (NERC). The output of NERC can be further enhanced by linking named entities to existing knowledge bases such as Wikipedia or Wikidata. This task is commonly referred to as named entity linking or disambiguation (NELD). Named entities can be utilized for improving search query results, building better question answering systems, as well as for identifying the target of sentiment in text.

1.3.2 Sentiment classification

Sentiment classification [Pang and Lee 2008] is typically modeled as a text classification problem or a document level task. The fundamental assumption of sentiment classification is that the sentiment $y \in labels$ depends on the text data. It is often modeled as $p(y|X)$, where X are features derived from the text. The most commonly used labels for sentiment are *positive* and *negative*, with the optional inclusion of *neutral*. However, a consistent meaning of these labels is not adopted across datasets. Furthermore, sentiment itself is a highly subjective quantity which can be described using the state of the content author, as well as the state of the receiver of the sentiment. This can be demonstrated using an example. Consider the tweet *"Roger Federer killed this, Nadals sucks."* [Mishra et al. 2014]. In the presented tweet, suppose the author was a fan of Federer, and Federer won the match with Nadal. Here, the author is most likely to show their support for Federer and their opposition of Nadal. However, if it was Nadal who won the match, and the author is a Federer fan, then the author shows their dislike towards Federer (or is not in support of Federer's game in that match), as well as their dislike towards Nadal (overall). Similarly, if the author was a Nadal fan,

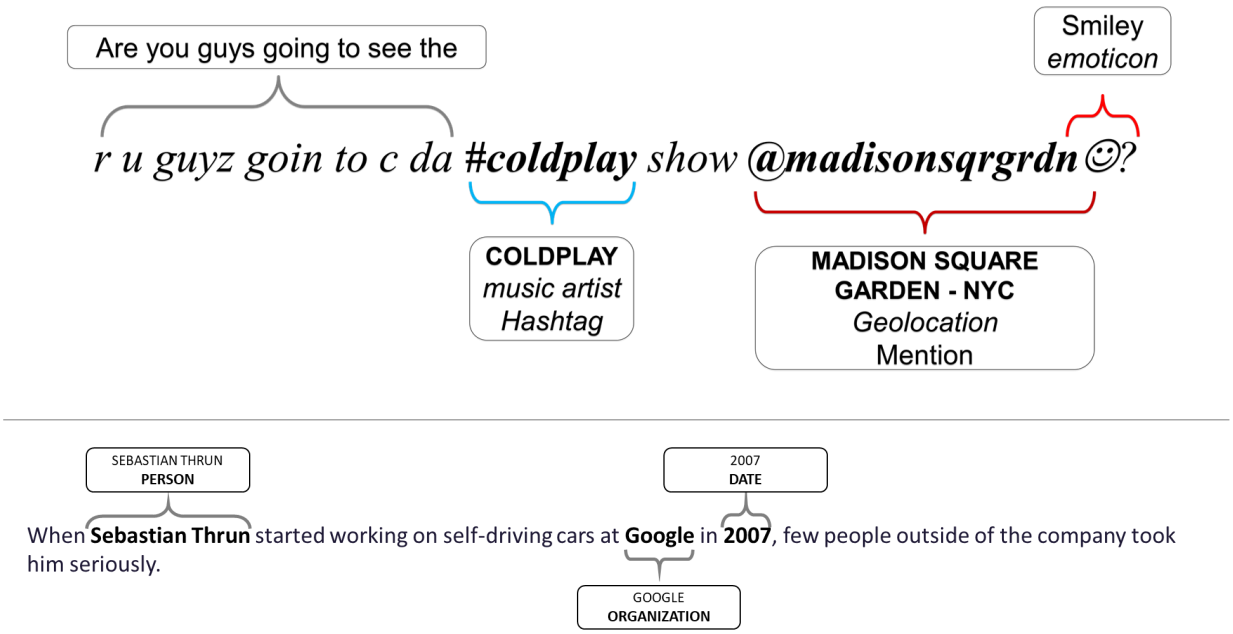


Figure 1.4: Illustration of named entity recognition for tweets (top) and newswire text (bottom).

and the game was won by Federer, they would have surely shown their dislike towards Nadal's game while appreciating Federer. The example serves as a demonstration of the subtle nuances of sentiment analysis which are not incorporated in most application scenarios. These approaches are often studied under aspect based sentiment analysis [Pontiki et al. 2015].

1.3.3 Key-phrase extraction and Taxonomy construction

Key-phrase extraction is a corpus level task aimed at identifying salient multiword concepts in text data. In the domain of scholarly publishing, domain specific concepts are needed. For example, in biomedical research the goal is to extract biomedical concepts from an article, and map them to a domain specific lexicon like Medical Subject Headings (MeSH).

1.3.4 Other tasks

There are many other tasks which are informed by the linguistic structure of the text, and the output of these tasks are often utilized to solve some of the prominent tasks described above. Some examples of such token level tasks are part of speech tagging, chunking, and super-sense tagging. Similarly, examples of such document level tasks are geo-location prediction [Han et al. 2016], rumor classification [Zubiaga et al. 2016a;b], and author-profiling [Rosso et al. 2016].

1.4 Existing challenges in IE research from DSTD

The first step in doing IE on DSTD is constructing the DSTD. There are several challenges in existing approaches for constructing DSTD. In scholarly publishing data, we first need to disambiguate authors before constructing DSTD. Incorrect disambiguation has been shown to introduce a lot of errors in inferences drawn from the resulting DSTD [Kim and Diesner 2016, Fegley and Torvik 2013, Kim and Diesner 2017]. In the social media domain, author identity is unambiguously provided via user IDs, however, existing IE techniques for text data have been shown to have lower accuracy and recall rates compared to IE on text data from newswire corpora [Augenstein et al. 2017, Ritter et al. 2011, Derczynski et al. 2015]. This is often attributed to high out of vocabulary tokens introduced by the diverse community of users who post content in different languages, dialects, and domains [Augenstein et al. 2017, Ritter et al. 2011, Derczynski et al. 2015]. Furthermore, many IE systems suffer from the issue of domain adaptation [Derczynski et al. 2015]. Considering opinion mining or sentiment analysis as an IE task, earlier research has shown that sentiment in social media is more nuanced, and harder to predict compared to newswire or review corpora [Mishra et al. 2014; 2015, Maynard et al. 2012, Aue and Gamon 2005, Rezapour and Diesner 2017].

A more practical issue in performing IE on DSTD from social media is the lack of standardized annotated corpora similar in quality and scale to Penn Tree-Bank, Universal Dependencies [Nivre et al. 2016], or the Movie Review corpus [Pang and Lee 2008]. In recent years, efforts have been made to construct such corpora e.g., for sentiment analysis [Nakov et al. 2016a;b], named entity recognition [Ritter et al. 2011, Derczynski et al. 2016, Baldwin et al. 2015], and part of speech tagging [Derczynski et al. 2013b, Owoputi et al. 2013]. However, many of these datasets suffer from varying tokenization issues, tag annotation discrepancies, and disproportionate tag distributions [Mishra and Diesner 2016]. Furthermore, many of the IE tasks are constructed as a pipeline of tasks, e.g., NER systems usually pre-process the text with tokenization, Part of Speech (PoS) tagging, and noun phrase chunking, before training a NER model. Similarly, many aspect based sentiment analysis systems require the extraction of named entities from text and then assign a sentiment to each entity. However, in the case of social media data, errors in the pre-processing models are very likely to propagate to the training of the final model.

In terms of presenting the information extraction results from DSTD, we often still stick to one of its views, choosing either to present its temporal, networked, or metadata based aspect. Many approaches combine the metadata aspect with the networked or temporal aspects, which leaves an important gap in seeing the DSTD for what it is –a socially connected, temporally ordered, and metadata enhanced representation of the data.

1.5 Existing approaches

In recent years, scholars have proposed using advanced machine learning (ML) methods for improving IE systems. A common objective of these advanced methods is reducing the amount of human effort in annotating the training data. In the field of sentiment analysis, authors have proposed using distant supervision [Mintz et al. 2009] for training models using emoticons present in tweets [Go et al. 2009, Felbo et al. 2017]. Distant supervision utilizes noisy labels to train the model on large unlabeled data. Distant supervision has also been applied to PoS tagging of tweets [Plank et al. 2014].

Similarly, active human-in-the-loop learning, also known as interactive machine learning (iML), enables high quality training data generation by collecting annotations based on the model's uncertainty [Settles 2009]. This is important as usually the costs of annotating data, especially those with structured output spaces are high [Settles and Craven 2008]. Active human-in-the-loop learning has been successfully applied for sentiment analysis [Mishra et al. 2015] and sequence labeling tasks such as named entitiy recognition [Settles et al. 2008].

Another popular approach for efficiently training machine learning systems is using Semi-Supervised Learning (SSL) [Chapelle et al. 2006, Zhu 2008]. This approach uses unlabeled data together with labeled data to learn a model under certain assumptions of the distribution of the data. It has been successfully applied to NER on tweets [Mishra and Diesner 2016].

Finally, crowdsourcing approaches are often used to generate training data or perform annotation for information extraction tasks [Lofi et al. 2012].

1.6 Research Questions

The main research questions of this thesis are as follows:

RQ 1 *How to use all information available to improve the efficiency and accuracy of IE from DSTD* This addresses the importance of using machine learning algorithm which are more suitable to the nature of DSTD.

RQ 2 *What information to extract?* This addresses the need to identify what new information is useful for social science research.

RQ 3 *How can the extracted information be presented and utilized?* This addresses the need for new visualization and presentation interfaces to make the extracted information from DSTD more accessible to the social science research community.

1.7 Proposed methods and solutions

In order to solve each research question we propose the following solutions:

RQ 1 Use active human-in-the-loop learning (chapter 8), semi-supervised learning (chapter 9), and multi-task learning for improving sequence tagging (chapter 10) and text classification in tweets (chapter 11). Also, use Wikipedia information for identifying concepts in scholarly data (chapter 7).

RQ 2 Extract temporal profiles of concepts and authors in scholarly data to quantify conceptual novelty (chapter 2) and conceptual author expertise (chapter 3). Suggest an alternative orthogonal set of labels and annotated data which identifies if a tweet supports or opposes the cause, and if it conveys an author's enthusiasm or passiveness towards the cause (chapter 5). Extract bias towards user and tweet metadata in sentiment annotated corpora (chapter 6).

RQ 3 Present a visualization framework for DSTD which allows presenting temporal, network, and meta-data aspects of the corpus (chapter 4).

The above mentioned approaches can be summarized into the following goals:

- Allow the model to learn over time; i.e., online learning of classifiers using online modeling techniques as well as data augmentation techniques, like adaptable gazetteers, which are effective for NER in tweets.
- Show the benefit of using multi-task learning approaches for tasks where training data is sparse by utilizing training data for similar tasks, e.g., sentiment prediction, PoS, NER.
- Meta data (such as users network and post interactions) can be used for improving the classification accuracy of existing models. How are metadata features correlated with sentiment labels?
- For certain applications, move away from default labels, i.e., positive versus negative, to task specific labels e.g., enthusiastic versus passive, and supportive versus non-supportive. This will help social scientists evaluate the models trained on these datasets by grounding them in prior literature.
- Visualize the social network aspect of the text data visualization.

Finally, the thesis has resulted in the following contributions:

- An annotated set of data for alternative opinion labels.
- A graphical user interface (GUI) to allow online learning of text classification and sequence labeling models with data augmentation.
- Visualizing the network structure of social conversation using a temporal network visualization which can be modified to show user as well as post level attributes.

- List of metadata features which can improve text classification tasks.
- A principled approach and ready to use tool for multi task learning of supervised models which use information from differently annotated corpora.
- Consolidate the existing corpora for learning from social media data, annotate for multiple tasks, and map to universal dependencies data.

Major portions of the thesis are taken from existing publications (after reviewing publisher's thesis reuse guidelines) in which I was the primary author. Following is the list of publications and the chapters based on them:

Chapter 2 Mishra, S. and Torvik, V. I. (2016). Quantifying Conceptual Novelty in the Biomedical Literature. *D-Lib Magazine*, 22(9/10)

Chapter 3 Mishra, S., Fegley, B. D., Diesner, J., and Torvik, V. I. (2018a). Expertise as an aspect of author contributions. In *WORKSHOP ON INFORMETRIC AND SCIENTOMETRIC RESEARCH (SIG/MET)*, Vancouver

Chapter 4 Mishra, S. (2017). SCTG: Social Communications Temporal Graph A novel approach to visualize temporal communication graphs from social data. In *UIUC Data Science Day*

Chapter 5 Mishra, S., Agarwal, S., Guo, J., Phelps, K., Picco, J., and Diesner, J. (2014). Enthusiasm and support. In *Proceedings of the 2014 ACM conference on Web science - WebSci '14*, pages 261–262, New York, New York, USA. ACM Press and Mishra, S. and Diesner, J. (2019). Capturing Signals of Enthusiasm and Support Towards Social Issues from Twitter. In *Proceedings of the 5th International Workshop on Social Media World Sensors - SIdEWayS'19*, pages 19–24, New York, New York, USA. ACM Press

Chapter 6 Mishra, S. and Diesner, J. (2018). Detecting the Correlation between Sentiment and User-level as well as Text-Level Meta-data from Benchmark Corpora. In *Proceedings of the 29th on Hypertext and Social Media - HT '18*, pages 2–10, New York, New York, USA. ACM Press

Chapter 8 Mishra, S., Diesner, J., Byrne, J., and Surbeck, E. (2015). Sentiment Analysis with Incremental Human-in-the-Loop Learning and Lexical Resource Customization. In *Proceedings of the 26th ACM Conference on Hypertext & Social Media - HT '15*, pages 323–325, New York, New York, USA. ACM Press

Chapter 9 Mishra, S. and Diesner, J. (2016). Semi-supervised Named Entity Recognition in noisy-text.

In *Proceedings of the 2nd Workshop on Noisy User-generated Text (WNUT)*, pages 203–212, Osaka, Japan. The COLING 2016 Organizing Committee

Chapter 10 Mishra, S. (2019a). Multi-dataset-multi-task Neural Sequence Tagging for Information Extraction from Tweets. In *Proceedings of the 30th ACM Conference on Hypertext and Social Media - HT '19*, pages 283–284, New York, New York, USA. ACM Press

1.8 Thesis outline

This thesis is divided into two parts, each focusing on various IE techniques for data from social media and scholarly activities. This chapter (chapter 1) gave an overview of IE and the definition of DSTD, it also identified the main goals of this thesis. Part I discuses ways of utilizing the DSTD structure to extract higher order information. Part II discuses approaches for improving IE on text data.

In part I, I focus on utilizing the DSTD structure of the data to extract higher order information. The part starts with examples of DSTD in the scholarly publishing domain. First, I discuss how extracted information can be utilized to build temporal concept profiles and quantify conceptual novelty of scientific articles and authors (chapter 2) and conceptual expertise of authors on a scientific article (chapter 3). In chapter 4, I describe a visualization framework called social temporal communication graph (SCTG), which provides an interactive way to explore this extracted information while preserving the DSTD structure of the data. Then we transition to IE use cases for DSTD in the social media domain. This includes finding users who enthusiastically support a social cause (chapter 5), correlations between tweet metadata and sentiment in existing corpora, and how this metadata can be utilized to improve sentiment prediction accuracy (chapter 6).

In part II, I focus on how DSTD data can be constructed more efficiently and accurately in various domains using different methodological frameworks. In chapter 7, I discuss the process for constructing a hierarchical subject headings for computer science concepts using Wikipedia's category tree. This hierarchical subject heading allows organizing computer science information in terms of relevant keyphrases and concepts. Next, we focus on improving IE from social media text using advanced machine learning techniques like active human-in-the-loop learning (chapter 8), semi-supervised learning (chapter 9), and multi-task learning (chapters 10 and 11).

Finally, chapter 12 describes the limitations and reiterates thesis contributions.

Part I

Information extraction from DSTD

Chapter 2

Quantifying conceptual novelty in scholarly data

Content in this chapter is based on our paper Mishra, S. and Torvik, V. I. (2016). Quantifying Conceptual Novelty in the Biomedical Literature. *D-Lib Magazine*, 22(9/10).

2.1 Introduction

In principle every published paper should have one or more novel aspects to it. However, not all papers are equally novel. Some papers introduce new, never before described ideas, while others increment or confirm established ones. Here we propose several measures of novelty guided by a literal interpretation of novelty as a function of time isolated from value, impact, or creativity. This allows us to study the intrinsic contribution of time as an aspect of novelty. We argue that novelty should be measurable at the time of publication while the value and impact of any idea reflects what happens (long) after publication. A quantifiable measure of novelty of an article can help scholars in tracing the origin of concepts in science. Previously, scholars have used various methods for measuring novelty of an article and studied its correlation with its impact as well as correlation with collaboration patterns [Uzzi et al. 2013, Packalen and Bhattacharya 2015b;a, Trapido 2015]. Scholar's have also argued that novelty plays an important role in the evolution of science [Uzzi et al. 2013], while maintaining that novel articles are rare [Uzzi et al. 2013] and might not get enough attention early on [Nicholson and Ioannidis 2012, Stephan et al. 2015]. Additionally, novelty has also been identified as through new ideas [Packalen and Bhattacharya 2015b;a, Trapido 2015] or unconventional pairing of exiting ones [Uzzi et al. 2013]. However, many of these methods suffer from issues related to noisy identification of relevant concepts [Packalen and Bhattacharya 2015b;a, Trapido 2015] or don't identify what concepts make an article novel [Uzzi et al. 2013]. We try to overcome this issue by using concepts identified by domain experts in a large corpus of biomedical articles, and quantifying how novel the individual and pairs of concepts are on every article.

We measure the scientific novelty of articles using a data set of 22.3 million articles published in MED-

LINE. Each article is tagged with set of Medical Subject Headings (MeSH)[1], where each term is part of a larger hierarchy called MeSH tree. We propose the usage of MeSH terms for identifying a standardized set of important biomedical concepts mentioned in an article. Furthermore, for each article its publication year is also recorded. Using the complete MEDLINE corpus we created a temporal profile for every individual and pairs of concepts ever mentioned. Additionally, we propose a logistic growth curve for modelling four common temporal phases of a concept, namely, initial burn in characterized by slow growth in the number of articles on the concept; followed by a rapid growth in publications related to the concept; leading to a phase where growth starts to slow down; and finally leading to a near constant growth phase. Our model captures the various temporal trends for a majority of the MeSH terms in MEDLINE. These temporal profiles of individual and pairs of concepts are then utilized for computing various novelty scores for every MEDLINE article. Our analysis of these article level novelty scores reveal that individual topic novelty is rare in biomedicine while combinatorial pairing of concepts is the norm. Aggregation of the novelty scores for 150,000 prolific authors (> 50 papers) reveal a complex trend in how the novelty of articles published by the author changes across their career. For a majority of the authors the average novelty of their articles goes down during their career span. However, there is an even split between authors whose average combinatorial novelty of articles goes either up or down, as they age. Additionally, there is no consistent pattern indicating at what professional age of their career authors publish their most novel work. Our article level novelty scores have a significant but weak predictive correlation with scientific impact (measured using citations received). We released a data set containing temporal profiles of all individual and pairs of MeSH terms in MEDLINE as well as pre-computed novelty scores for all articles, via an interactive user interface, *Gimli*[2].

2.2 Related work

Novelty, originality, and priority are three important concepts related to scientific publishing [Morgan 1985]. Some scholars have discussed novelty being related to newness while others relate it to interestingness. Scholars who study the growth of science directly or indirectly have commented on the aspect of novelty, originality or innovation in science. Thomas Kuhn suggests that science moves forward through innovation and work on novel concepts [Kuhn 1970]. Dirk uses a self-reported survey of the authors of *Citation Classic* papers in the biomedical field to adjudge the novelty of a paper using the topology of scientific originality based on the structural analysis of the paper [Drik 1999]. The author defines novelty as a permutation of the novelty of the following three sections of the paper: Hypothesis, Methods and Results. Using a small sample

[1] https://www.nlm.nih.gov/mesh/
[2] http://abel.lis.illinois.edu/gimli/novelty

of 209 articles the author reports that most papers have the following topology: *new hypothesis/previously-reported methods/new results*. Even though the study is focused on a small subset of articles, it establishes the need to understand novelty of a paper and identify its novel elements. However, a survey based approach is not feasible for a corpus as large as MEDLINE. Hence, there is a need for a more computational approach towards identifying novelty of a paper. Trapido [Trapido 2015] discusses the effects of novelty on the recognition of authors. Their work shows that highly novel articles are at a higher risk of rejection. However, authors who have a history of publishing novel articles generally receive positive feedback for their future novel papers. Additionally, disciples of authors with a history of novel publications tend to publish novel work. Recent work has shown that bibliometric indicators like early citation counts are biased against novel articles [Stephan et al. 2015], and novel articles are more likely to be published in low impact factor journals. Uzzi et al. [Uzzi et al. 2013] define novelty of an article as a measure of its combination of cited journals, where every journal signifies a respective domain of science. Using a corpus of 17.9 million web of science articles from multiple fields, the authors argue that most articles have a high propensity for citing conventional combination of journals while few papers cite novel combinations. Their work claims that highest impact articles demonstrate a mix of high conventional combination and introduce some novel combinations. Additionally, teams are more likely to work on novel combinations compared to solo authors. Packalen et al. [Packalen and Bhattacharya 2015b;a], define novelty as *newness of the concepts* mentioned in an article. The authors describe a word/phrase-based analysis for identifying novel ideas in the MEDLINE database [Packalen and Bhattacharya 2015b]. They manually remove synonyms of popular ideas and replace them with its more canonical meaning. Their methods are limited to finding novel articles on popular ideas and only consider novelty of single ideas in a paper. Correlation of novelty with authors' professional age shows that younger authors co-author most of the novel work. The presence of experienced co-authors is correlated with higher novelty of that article. Methods for measuring novelty can be effectively applied to patent corpora as their structure is similar to bibliometric corpora of research articles, making these methods useful for a broader community of researchers. In [Packalen and Bhattacharya 2015a], Packalen et al. extend their approach from [Packalen and Bhattacharya 2015b] for finding novel ideas in the USPTO patents data set [Packalen and Bhattacharya 2015a]. Youn et al. [Youn et al. 2015] discuss the combinatorial aspect of invention in patents, and describe how process exhibits a pattern of exploration and exploitation of new technologies. Schoenmakers et al. [Schoenmakers and Duysters 2010] discuss patterns of novelty in patents using an empirical examination of 157 patents. The authors find that radical inventions are based more on existing knowledge than non-radical inventions. They further explain that radical innovation is mostly a result of a combination of different knowledge domains. Evans et al. [Evans 2010, Evans and Foster

2015] also present a detailed review of the various methods used for quantifying novelty in various fields. The preceding works have motivated us to devise a simple yet consistent method of quantifying novelty of articles. We build upon some preliminary work presented in [Mishra et al. 2014, Mishra and Torvik 2016] and present the details of our methods and the corresponding results in the sections which follow.

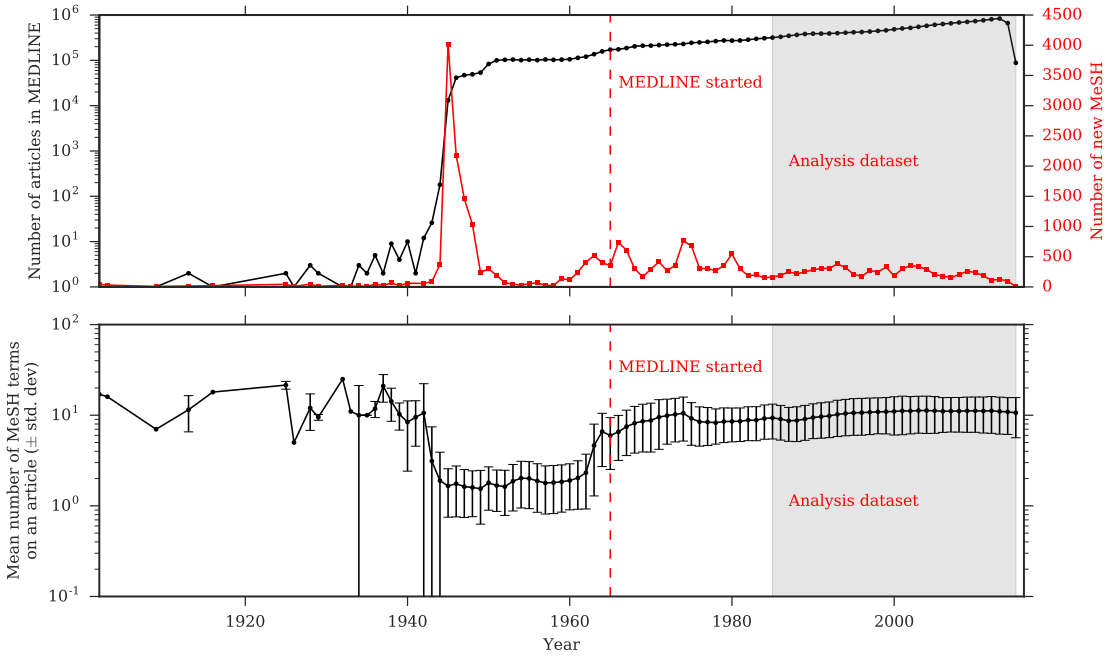


Figure 2.1: **TOP:** Growth of MEDLINE across the years and number of new MeSH terms added each year. **BOTTOM:** Mean number of MeSH terms used to index articles in MEDLINE across years. MEDLINE was started in 1965 and a lot of noise – with regards to many MeSH terms being wrongly spelled or older articles being index by too few MeSH terms – is present in our corpus around those years. The data after 1985 (shaded grey and marked as Analysis data) has a stable growth and is used for most of the results presented in the Results section.

2.3 Temporal profile of a concept

Novelty scores capture the age of a concept (or pairs of concepts) as measured in years (or number of prior articles) since its first appearance in a corpus, henceforth also referred to as *empirical novelty scores*. It is important to note that a **more novel article** will have a **lower empirical novelty score** and vice versa. We discuss the following two types of novelty in this paper:

- **Time novelty** is based on the number of years since the first appearance of a concept in a corpus
- **Volume novelty** is based on the number of articles since the first appearance of a concept in a corpus

Absolute value of number of articles on a concept (C) in an year (y) ($N_{C,y}$), is a noisy indicator of the

growth patterns of that concept in MEDLINE as MEDLINE itself has grown exponentially through time. This automatically results in more papers on a concept in later years, which do not signify the growth of this concept relative to the growth of MEDLINE. To account for this effect, we define a log normalized count of articles ($N^e_{C,y}$) on concept C in year y as the log of $N_{C,y}$ multiplied by a normalizing factor (Z_y), such that Z_y is the ratio of mean number of articles in MEDLINE and total articles in MEDLINE in year y. Hence,

$$N^e_{C,y} = ln(1 + N_{C,y} * Z_y) \tag{2.1}$$

.

On empirical observation of temporal profile of concepts in MEDLINE, we found that most concepts' growth in MEDLINE is defined by the following four phases:

- **Burn-In Phase**: Topic is new, publication rate is small, and growth is marginal.
- **Accelerating Growth Phase**: Topic is bursting, publication rate is rapidly increasing.
- **Decelerating Growth Phase**: Publication rate is still increasing, but is starting to stabilize.
- **Constant Growth Phase**: Growth is marginal and publication rate has stabilized.

In order to capture the above phases for each concept, we model $f(t) = N^e_{C,t}$ as a function of age (t) of the concept measured in years, such that

$$N^e_{C,t} = f(t) \sim \frac{N_o}{1 + exp(-\frac{(t-t_o)}{s})} \tag{2.2}$$

where,

$N^e_{C,t}$ is $N^e_{C,y}$ such that $t = y - y_o$,

y_o is the first year after 1965[3], when the concept was used, N_o captures asymptotic max value the concept can attain,

t_o captures the age when the concept goes from accelerating to decelerating growth phase,

s captures temporal spread of both the growth phases.

Using the model described above, we also define the velocity and acceleration of a concept as $f'(t)$ and $f''(t)$, respectively.

[3]MeSH terms were introduced in 1965, hence data is very sparse for years before 1966 and many articles are indexed with few MeSH terms. This is important for proper curve fitting using least squares.

2.4 Novelty of articles in MEDLINE

This section explains how we assign a novelty score to each article in MEDLINE.

2.4.1 Data

For generating the novelty scores we consider 22.3 million articles published in MEDLINE between 1902 and 2015. Our study uses 27,249 MeSH terms[4] as a basis for identifying the concepts on a MEDLINE article. From figure 2.1, the rapid growth of MEDLINE after 1945 is quite evident. We also observed that the number of MeSH terms first indexed in a year saw a sharp spike in 1945. After 1985 this trend has been stable. Similarly, the mean number of MeSH terms on an article has a steady trend of an average of 10 MeSH terms per year, since 1985.

MeSH terms are assigned by experts from the National Institute of Health (NIH), resulting in a consistent identification of concepts on the article. This is a major reason for our choice of using a controlled vocabulary like MeSH for the identification of concepts on an article. Another reason is that only well recognized concepts are included in the MeSH vocabulary, ensuring articles which coin new terms, not recognized by the National Library of Medicine (NLM), are not marked as novel by us. In similar research described in [Packalen and Bhattacharya 2015b;a], the authors use words and phrases extracted from title and abstract of an article. We argue that these features suffer from several limitations of NLP based information extraction systems, mainly disaggregation of name variants and spelling errors for words and phrases describing the same concept. In [Uzzi et al. 2013], the scholars use co-citation of articles in defining the novelty of an article across four categories. This approach does not allow us to identify the novelty of an article across multiple fields. An article (on average) is indexed using 9.6 ($\pm$5.0) MeSH terms with the maximum MeSH terms on an article being 103 while the minimum is 1. Furthermore, each article (on average) contributes towards the novelty scores of 45.5 ($\pm$22.9) MeSH terms, after each of the article's terms are exploded, i.e., also assign all parents of each MeSH term to the article.

Generating temporal profiles for MeSH terms

Articles in MEDLINE are indexed using MeSH terms that are organized in a hierarchy although it should be noted that some terms have multiple parents. E.g., Breast Neoplasms has two MeSH IDs C04.588.180 and C17.800.090.500, which point to their positions in the hierarchy:

- Neoplasms [C04] $\rightarrow$ Neoplasms by Site [C04.588] $\rightarrow$ **Breast Neoplasms [C04.588.180]**

[4]We use the 2015 MeSH tree available from https://www.nlm.nih.gov/mesh/

- Skin and Connective Tissue Diseases [C17] → Skin Diseases [C17.800] → Breast Diseases [C17.800.090] → **Breast Neoplasms [C17.800.090.500]**

PubMed search[5] for articles in MEDLINE, on any MeSH term, also lists the articles on any of its children MeSH terms. This means that for counting the number of articles published on a MeSH term, we can use the total number of articles index by that MeSH term or any of its children in the MeSH tree. E.g. in the case of Neoplasms (Cancer), a growth in the number of publications about Neoplasms should not be gauged by a growth in the number of papers mentioning the exact MeSH. A growth in the number of publications involving any of the child terms should also be considered towards the growth of usage of the MeSH term *Neoplasms*. We count an article mentioning any children of a MeSH term as an article in the parent MeSH. This results in having a better estimation of how research has progressed in rarer MeSH categories. The list of MeSH in an article are considered as a list of all exploded MeSH terms. So if an article has a MeSH term list as *Breast Neoplasms*, then the exploded MeSH list will be [*Breast Neoplasms, Neoplasms by Site, Neoplasms, Breast Neoplasms, Breast Diseases, Skin Diseases, Skin and Connective Tissue Diseases*]. This exploded MeSH term list is used as the list of all concepts related to the article.

Some simple pre-processing steps were carried out. Misspelled terms were merged with the correct term in the MeSH tree by removing spaces and lower casing the terms. A preliminary investigation into the effect of incorrect assignment of MeSH terms by annotators at NLM was carried out by removing the spelling correction pre-processing step. But the resulting affect on the novelty scores was marginal. Additionally, the trends described in our results were not affected. We believe that errors in data such as indexing artifacts, and incompleteness of the MeSH tree with respect to all the important terms in biomedical community; might also have a marginal effect on the presented results. Articles published before 1966 had multiple irregularities in assignment of MeSH terms. Hence, we only use the MeSH counts after 1965 for fitting the model to these MeSH terms. Model based scores were calculated for all MeSH terms. However, the model fitting algorithm failed to converge for 316 (1.2%) terms, most often due to sparse data or recent terms (116 were introduced after 2010). Of the MeSH terms which had more than 5K articles, only three were introduced after 1965: *Autophagy; Protein Multimerization; and Influenza A Virus, H1N1 Subtype*, while the other 97 terms were introduced before 1965 with a majority (64) of them being first used in the year 1945 (see Figure 2.1). A closer inspection of the temporal profile of the MeSH terms, for which our model failed to converge, revealed two major growth periods from which the algorithm was unable to pick one. Empirical novelty scores, velocity and acceleration of all MeSH terms for each year were computed using the complete MEDLINE corpus and the methods described in section 2.3. Empirical novelty scores were also

[5]http://www.ncbi.nlm.nih.gov/pubmed/

computed for every pairs of MeSH terms for each year.

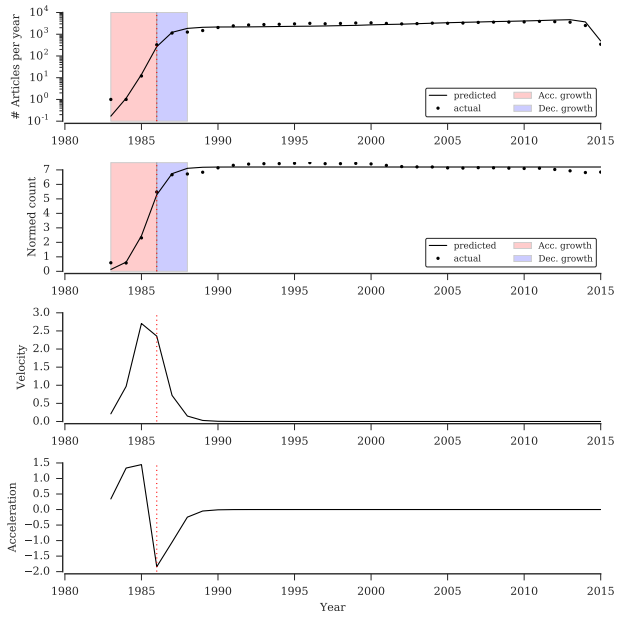


Figure 2.2: Temporal profile of MeSH term HIV including describing the empirical as well as predicted trends using our model.The figure describes the rapid growth in publications on HIV around 1985 marking a 4 year period of accelerated growth followed by a 3 year period of decelerated growth leading into a final phase of constant growth. The model was fitted on the normed count and the predicted values were rescaled to predict the actual number of articles on the concept.

Figure 2.2 shows the profile of the HIV MeSH term in our dataset. We can observe that the data of HIV fits perfectly with our model, and we observe the four distinct phases of growth of this MeSH term. Specifically, year 1986 is the year the term enters a *Decelerated growth phase* and soon after that it enters a *Constant Growth Phase*. The observations are interesting because AIDS was first clinically discovered in 1981 and HIV was discovered in 1983 under two different names: LAV [Gallo et al. 1983] and HLTV-III [Barre-Sinoussi et al. 1983], which matches with the accelerated growth in research on this topic, the terms were renamed to HIV in the year 1986. We found that there were no articles mentioning HIV directly before the year 1986, which also proves why our method is robust in identifying the initial phases of a concept by using an exploded MeSH tree, merging name variants and fixing common spelling issues.

Assigning novelty scores to an article

The following empirical novelty scores are assigned to each article:

- **Minimum concept age (years) or Individual time novelty:** Minimum concept age in years among all concepts on the article

- **Minimum concept-pair age (years) or Pairwise time novelty:** Minimum concept-pair age in years among all concepts-pairs on the article
- **Minimum concept age (papers) or Individual volume novelty:** Minimum concept age in papers among all concepts on the article
- **Minimum concept-pair age (papers) or Pairwise volume novelty:** Minimum concept-pair age in papers among all concepts-pairs on the article

Using the velocity and acceleration of every concept on an article, the following growth scores are also assigned to every article:

- **Accelerated minimum growth:** Minimum velocity among all concepts in the accelerated phase
- **Accelerated maximum growth:** Maximum velocity among all concepts in the accelerated phase
- **Decelerated minimum growth:** Minimum velocity among all concepts in the decelerated phase
- **Decelerated maximum growth:** Maximum velocity among all concepts in the decelerated phase

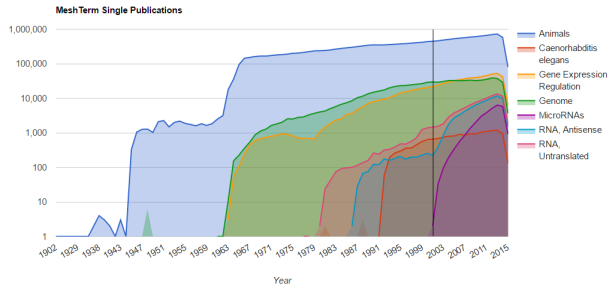


Figure 2.3: Temporal profile of all MeSH terms listed on PubMed ID 11779458 published in the year 2001, one of the first few papers listed on the MeSH term microRNAs. microRNAs saw a rapid growth after their introduction and MeSH terms like RNA Untranslated and RNA Antisense also experienced a phase of rapid growth in research output during the same years. MeSH terms like Animals, Genomes and Caenorhabditis elegans appear to be in their constant growth phase.

Figure 2.3 show the profile of an article published in MEDLINE in the year 2001. This was one of the first articles on microRNAs. As is evident from the figure, the most novel MeSH term on the article in microRNAs is resulting in a *Individual time novelty* score of 0 and *Individual volume novelty* score of 2. We also observe that the term has multiple pairs of MeSH terms occurring for the first time because of the novel nature of this paper, resulting in a *Pairwise time novelty* score of 0 and *Pairwise volume novelty* score of 1.

2.5 Results and Discussion

In this section we describe how our novelty scores can be utilized to study the distribution of novelty in biomedical literature, careers of authors, and impact of an article.

2.5.1 Distribution of novelty scores in MEDLINE

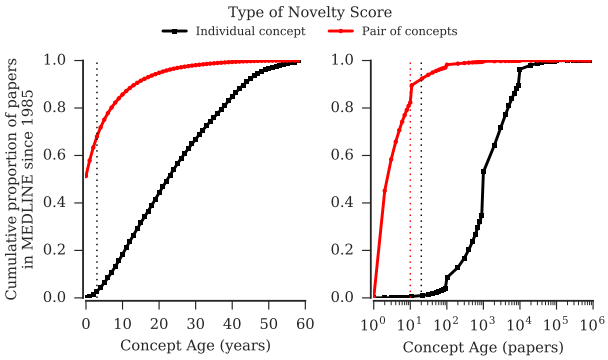


Figure 2.4: Cumulative distribution of novelty scores for 15.72M (15.71M with at-least a pairs of MeSH terms) articles published in MEDLINE since 1985. Lower is more novel. Colored dotted lines represent the respective cutoff for marking an article novel.

Novelty scores were generated for all the articles in MEDLINE. A subset of 15.1 million articles published in MEDLINE after 1985 (as shown in Figure 2.1) was considered for the analysis of the distribution of novelty scores. Figure 2.4 shows the cumulative distribution of the various novelty scores. Using a cut-off of 3 years for time novelty scores, and 20 years and 10 years for individual and pairwise volume novelty scores, respectively, we find that individual concept novelty is rare in MEDLINE. The distribution of novel article with respect to our cutoff values are shown in Table 2.1. Further, it appears that pairwise novelty scores are better at capturing the prevalent kind of novelty in articles.

An interesting trend depicted in figure 2.4 is that for a majority of articles, their most novel individual concept is more than 1000 prior papers. This might reflect a trend in the biomedical community of working on well established concepts, but at the same time feature quick adoption of new pairing of terms. From that figure it is also evident that the volume novelty scores are capturing a more granular level of novelty compared to time novelty scores. The model based novelty scores identify 61.1% of the articles, which have at least one concept fitting our model specification, have a concept in an accelerated growth phase, whereas 38.9% articles have no concept on such phase. This might reveal that a majority of scientific articles are published on at least one topic which is hot at that time.

Table 2.1: Proportion of novel articles identified using specified cutoff for different novelty scores

	Age	
Novelty type	**(years)**	**(papers)**
Individual concept	2.73% (< 3)	1.0% (< 20)
Pairs of concepts	68.0% (< 3)	89.6% (< 10)

2.5.2 Modeling change in novelty across an author's career

How does the novelty of articles, published by an author, change across their career? In order to answer this question, we consider a subset of the Author-ity data set [Torvik and Smalheiser 2009] of all authors who have published at least 50 papers and have started their careers after 1965. This results in a data set of 150K authors. For each author, a linear model was fitted for the log value of novelty scores (y) of each of their articles versus the professional age of the author x, measured as the years since the first publication by the author in MEDLINE. For every author, the model $log_{10}(1 + y) \sim mx + c$ is fitted using the least squares method, and the slope m is recorded. If $m \geq 0$, we infer that the average novelty of papers across an author's career is decreasing (higher novelty score means less novel paper). Similarly, if $m < 0$, the author is considered, on average, to publish more novel papers later in their career. Similarly, we also fit a model only on the minimum novelty scores per year of an author's career. Table 2.2 presents the proportion of authors who have a decreasing novelty across their career. We observe that the majority of authors (85%) have a decreasing individual concept novelty across their careers. The proportion of authors who have decreasing concept-pair novelty (60%) is only slightly different from those who have an increasing concept pair novelty (40%), across their careers.

Table 2.2: Proportion of authors with increasing average novelty of articles over careers

	Novelty	
Type of novelty	**Average**	**Minimum**
Concept Age (years)	84%	59%
Concept Age (papers)	85%	58%
Concept-pair Age (years)	64%	67%
Concept-pair Age (papers)	56%	68%

These results indicate that the average novelty of articles published by most of the prolific authors (> 50 papers in MEDLINE) decreases as their career progresses. However, exactly when these authors published their most novel article, does not show any specific trend. Further investigation revealed that an author's most novel articles based on either of the novelty scores can occur any time in their career, but are less likely to occur at the beginning of their careers. The relation of an author's mean article novelty with an author age also varied based on when that author started their career. These patterns indicate that the relationship of an author's age with the novelty of their article is rather complex.

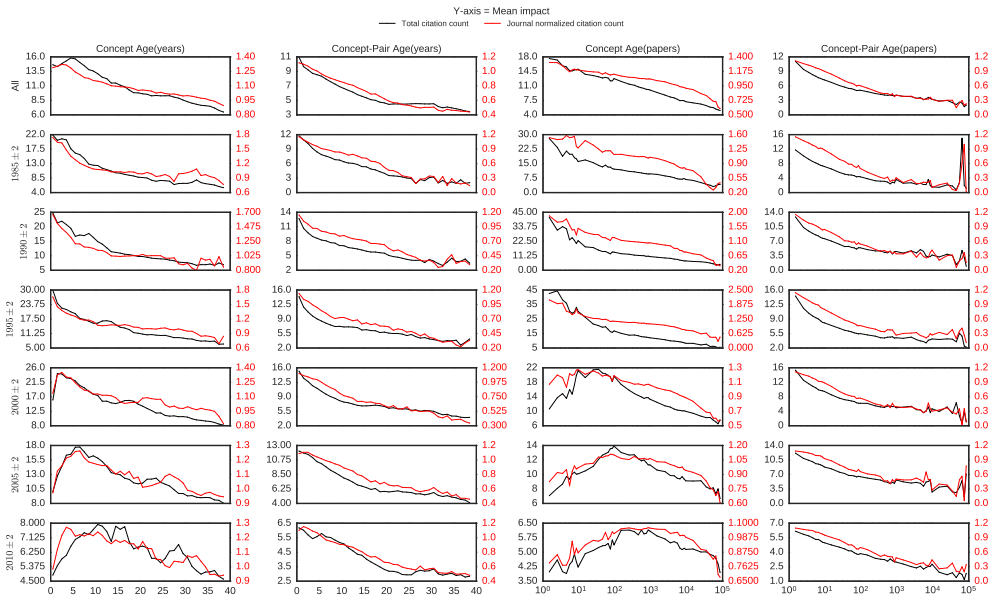


Figure 2.5: Novelty scores correlated with mean impact as measured using total and journal normalized citation count. Y axis in each sub plot represents the mean impact (black - total cites, red - journal normalized). X axis in each subplot column measures the concept age as denoted in the column title.

2.5.3 Correlation of novelty with impact

One of the most common evaluation of any bibliometric measure is its correlation with the impact measured in terms of citation count. We use the citation information available from a collection of Web of Science PubMed citations, Microsoft Academic Graph[Sinha et al. 2015] and PubMed Central. The impact of an article is measured in the following ways:

- **Total citation count**
- **Journal normalized citation score**, which normalizes the citations received by an article using the average citations received by all articles published in that journal in the same year.

Analyzing the correlation of our novelty scores with the journal normalized citation score allows for comparing correlation between novelty and citation, after removal of journal specific citation impact (e.g. articles published in high impact journals on average receive more citations compared to those published in low impact journals). The journal normalized citation score of an article published in a given journal is the ratio of the total citation count divided by the average citation counts of all articles published in the same journal in the same year.

Figure 2.5 reveals the relation between the mean impact scores versus the various novelty scores of articles aggregated in time windows of 5 years. The figure depicts a positive correlation between more novel articles and higher impact scores. However, the trends are not consistent across the years, and we observed that

the articles on novel individual concepts that were published in more recent years have lower impact scores. No such trend was visible in articles on novel pairs of concepts in the same years. This might suggest that articles which introduce completely new topics take some time to gain their potential impact as has been discussed in earlier literature [Stephan et al. 2015]. However, articles which are among the first few to merge existing topics require less time to reach their potential impact. A possible reason for the observation of this trend might be that there is a slow adoption of research in new concepts in the biomedical community, resulting in a low impact of articles published on these concepts in their earlier years, but as the concepts age, a larger number of papers refer to these concepts, leading to higher impact later on.

A series of regression tests revealed significant ($p < 1e-3$) negative correlation between novelty scores (lower is more novel) of an article and its impact measured through total citation counts and journal normalized citation score. However, our models were not very predictive ($R^2 \sim 0.1$) of the impact of the article. Furthermore, inclusion of factors like mean journal citation, year of publication and number of MeSH terms improved the model's predictive ability ($R^2 \sim 0.47$). This indicates that novelty of an article is not the most significant factor that contributes towards its high impact, although it helps to be novel. Additionally, other factors such as impact of the publishing journal, reputation of listed authors, year of publication, and topic of the article might play a greater role in determining its impact.

2.5.4 Public data set and user interface for exploration of novelty scores

In addition to our analysis and the results presented in this paper, we also make available all the novelty scores and the code generated by our model as well as a web based user interface *Gimli* [6] for interactively exploring the novelty profiles of each article as well as the temporal trends of each MeSH term. Our interface also supports a feature for tracking the change in novelty across an author's career. Figure 2.3 is an example of the temporal profiles of all the MeSH terms presented on an article. Figure 2.6 shows the screen shot of our web page displaying the most novel terms across various top level categories in the MeSH tree for a given article. The interface also allows a user to explore the profile for each individual MeSH term in the MeSH dictionary, as well as a comparative view of the temporal profiles of all individual, and pairs of MeSH terms on a given article. We believe this resource can be useful for researchers working on studying innovation in biomedical literature as well as those who want to study trends in the growth of concepts in bio-medicine.

[6] http://abel.lis.illinois.edu/gimli/novelty

2.6 Conclusion

We propose several measures of quantifying conceptual novelty of a MEDLINE article. From our experiments we conclude that using individual and pairs of MeSH terms to measure novelty uncovers allow us to assess novelty distribution among papers and author careers in the biomedical domain. Pairwise scores are more resonant with our general idea of novelty, where an article published on a combination of topics can be considered novel even if it is not novel in any of its individual topics. By modelling the change in novelty across an author's career, we discover that for the majority of the authors, the average individual concept novelty of their published articles goes down as the authors age. However, authors might publish their most novel work at any time during their careers. Our measures suggest a complex relation between novelty and impact of an article. The methods presented here for quantifying the novelty of an article can be applied to any corpus, given that the articles are annotated with a consistent hierarchical set of categories, e.g., the ACM Computing System Classification for ACM articles [Coulter 1997] or Microsoft Academic Graph's hierarchy of fields of study and paper keywords [Sinha et al. 2015]. We complement our study with an online resource which allows a user to visually inspect the temporal profiles of all the MeSH terms assigned to a given MEDLINE article. Our system uses pre-computed novelty scores to present most novel concepts on an article, aggregated into relevant biomedical categories such as Drug, Disease and Organism. Other scholars can utilize our data set of novelty scores in studying the evolution of concepts in biomedical literature.

2.6.1 Connection with DSTD

The modeling of novelty in scholarly domain can be considered an application of using DSTD. A scholarly corpus utilized in this chapter can be considered an example of a DSTD (as described in chapter 1) where the papers, authors, and MeSH terms are nodes; an authorship, and a term mention are edges, and the time ordering is based on date of publication of the paper. The DSTD framework allows us to perform an operation on these nodes by doing accumulating counts of a MeSH term for each year and then doing a cumulative sum over these counts to get the MeSH profile. The novelty score can then be identified by using the lowest cumulative sum value of all the MeSH terms assigned to the paper. These novelty scores can then be grouped at author level and their mean, maximum, or minimum values per year, can then be used to construct an author's novelty profile. Overall, this temporal profiling approach allows us an abstract way to answer questions about novelty on any DSTD and defines a new IE task which the DSTD facilitates.

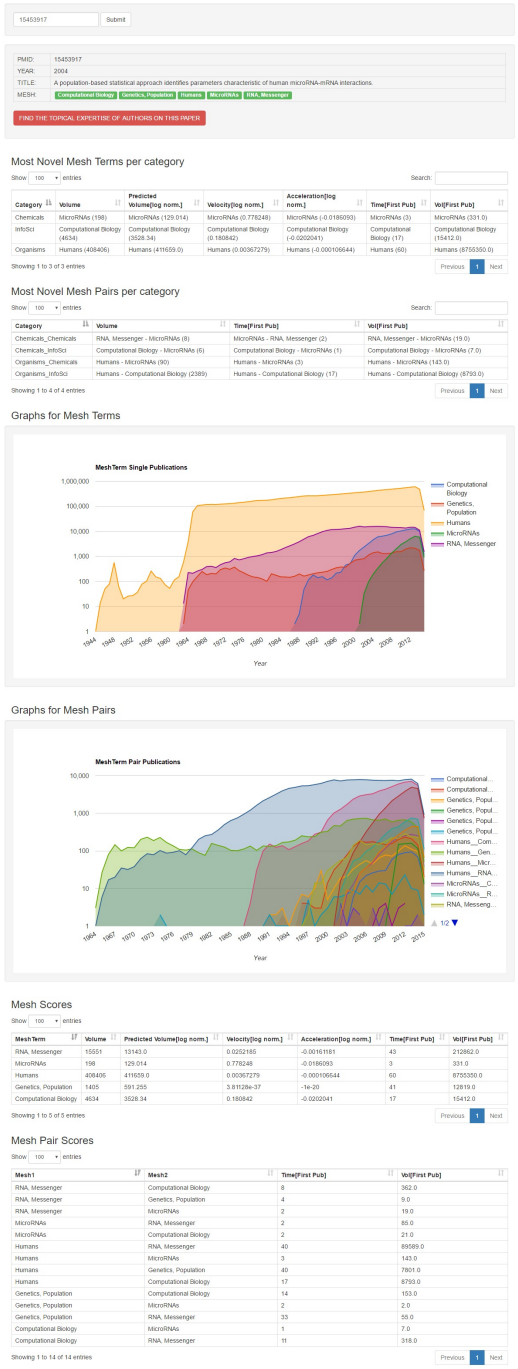


Figure 2.6: Screen shot of the web interface *Gimli* for exploring novelty profiles of articles in MEDLINE, temporally modelled growth of individual MeSH terms, and change in novelty across an author's career. The above figure shows the novelty profile of an article indexed by the listed MeSH terms. The profile of every individual and pairs of MeSH terms is displayed along with their empirical and model based profile scores. The interface also mentions the most novel individual and pairs of concepts in top level MeSH categories like Drug, Disease and Organisms.

Chapter 3

Conceptual expertise in scholarly data

Content in this chapter is based on our talk abstract Mishra, S., Fegley, B. D., Diesner, J., and Torvik, V. I. (2018a). Expertise as an aspect of author contributions. In *WORKSHOP ON INFORMETRIC AND SCIENTOMETRIC RESEARCH (SIG/MET)*, Vancouver.

3.1 Introduction

Authorship credit allocation is a widely studied issue in the field of bibliometric analysis [Merton 1968, Rennie et al. 1997, Shen and Barabási 2014, Yank and Rennie 1999, Zuckerman 1987; 1968, Kim 2014]. The International Committee of Medical Journal Editors (ICMJE) has a specific set of guidelines regarding authorship and credit allocation on papers; indicating the importance of this process [International Committee of Medical Journal Editors 2018]. Bibliometric scholars usually employ methods like self-reported author contributions on journals [Bates et al. 2004, Rennie et al. 1997, Yank and Rennie 1999], citation patterns of authors [Shen and Barabási 2014], or some other heuristic methods [Clement 2014] for estimating or defining credit per author on a paper. There is wide consensus in the academic community that authorship implies responsibility and significant contribution [International Committee of Medical Journal Editors 2018, Rennie and Flanagin 1994]. Even though many journals have recently started to require authors to report their contribution level on the article, the resulting data are usually sparse and subject to an author's interpretation of what is meant by "contribution". Further issues with the reliability of this kind of self-reported data [Bates et al. 2004, Rennie and Flanagin 1994, Yank and Rennie 1999] are intransparency over the type of contribution, e.g. intellectual, technical or editorial. Assessing author contribution allows for the analysis of mechanisms that drive scholarly collaborations, such as incentive schemas [Leimu and Koricheva 2005]. Work in this area has shown that getting full instead of partial credit for a publications may be perceived of greater value to scholars [Katz and Martin 1997]. Other scholars have considered author contribution as a measure of collaboration strength [Newman 2001; 2004].

Our work is focused on quantifying conceptual expertise based on overlapping concepts of a paper and

its author's prior papers. This approach helps quantify complementary expertise added by each author on a given paper. It is related to previous studies on identifying topics in scientific articles and their association with authors [Blei et al. 2010, Rosen-Zvi et al. 2004, Steyvers et al. 2004].

3.2 Data

The Author-ity 2009 [Torvik and Smalheiser 2009; 2018] dataset consists of disambiguated author names for MEDLINE articles published through mid 2009. Each article in MEDLINE has been tagged with at least one Medical Subject Heading (MeSH) term. Between 1980-2009 the Author-ity MEDLINE subset consists of 10.2M papers authored by 7M authors of which around 90% had their first paper after 1980.

3.3 Methods

In this section we describe the methodology for computing relative conceptual expertise of each author on a given paper, and the age at which an author gains independence, i.e., is consistently a top expert on majority of their papers.

3.3.1 Computing author expertise on a paper

For each author their temporal author concept profile was constructed. This profile facilitates computing the number of prior papers of an author on a given concept in every year of their scholarly career. The expertise of an author on a concept is based on the number of prior articles by that author on that concept (we denote this by x). The expertise scores are scaled as $y = log_{10}(x + 1)$ to capture expertise difference based on order of number of prior papers. The expertise of an author for a concept on a given paper is then normalized by the max expertise on that concept on the paper. This is called the weighted expertise of that author. The conceptual expertise by an author on an article is defined as the iterative cumulative weighted measure of their expertise on each topic on an article. The resulting ranking represents the order in which each person has contributed their expertise to an article. The resulting value of conceptual expertise identify each author's complimentary expertise on an article after removing all expertise contributed by all the other authors who have a higher value of conceptual expertise. A demonstration of the calculations at: http://abel.ischool.illinois.edu/legolas/coverage?pmid=15922829.

3.3.2 Identifying author independence

As an author's career progresses, they change roles, venture into new areas of research, and publish with different collaborators. All of these factors can affect the expertise of an author in their future articles. Collaborating with authors from different domains may initially lead to lower expertise for an author on these publications, but also gives them the opportunity to broaden their expertise profile. This evolution eventually leads to higher complimentary or top expertise contributions by the author in their future papers. On the other hand, sustained collaborations with partners who have higher expertise on an author's areas will keep the author's contribution to a paper continuously low. This can be considered an example of "living under the shadow" of a senior author. The temporal change in conceptual expertise of an author over their career can be operationalized to study when an author becomes an independent contributor. An author can be characterized by two types of profiles: a) **maximum expertise profile:** the proportion of their articles in which they have the maximum expertise, i.e., they have a conceptual expertise > 0 and are a top expert on that paper, b) **significant expertise profile:** the proportion of their articles in which they have the significant expertise, i.e., they have conceptual expertise > 0.

The career profile of an author is identified via a polynomial logistic model that predicts the ratio of papers per the two above mentioned type of expertise over the author's professional age in years (years since first publication).

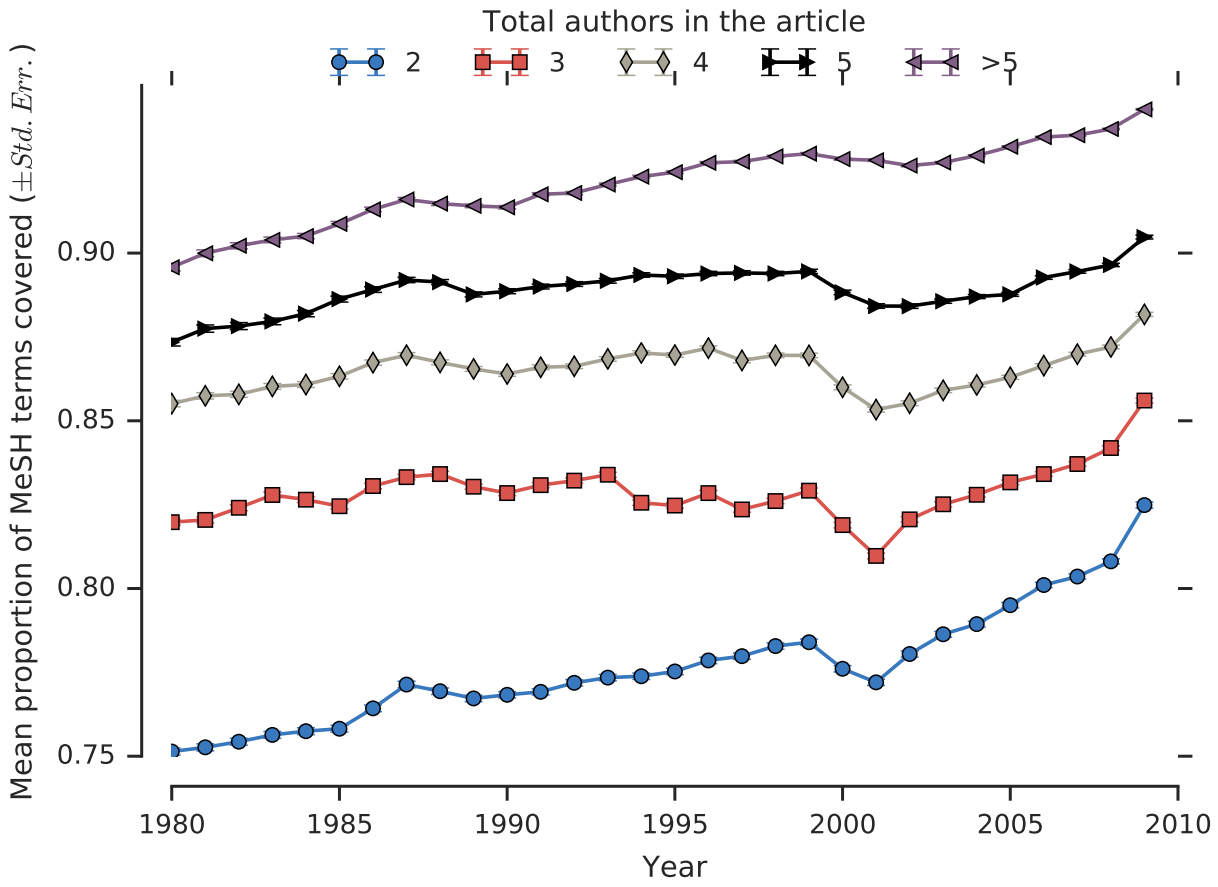
3.4 Results

Figure 3.1 shows the collective contribution of all authors to the conceptual coverage, on average. The great majority of concepts are typically covered, and the coverage has been going up over time. Also, each additional author adds complementary expertise by several percentage points. This might help explain the widely known phenomenon that the number of authors on papers has been steadily increasing over time.

Figure 3.2 shows the distribution of the position of the dominant author (the author with the highest coverage). It is not surprising that the last author is most often the one who contributes the most expertise. However, the last author is dominant in less than 50% of the papers with four or more authors. In other words, it cannot be taken for granted that the last author contributes the most expertise. The second to last author is dominant in more than 20% of the papers, regardless of the number of authors. All the other authors are equally likely to be dominant, in their future. It is also clear that the role of the first author has dramatically reduced over the short time period studied here.

Figure 3.3 shows the temporal profile of an author's contributing expertise. The author published their

Figure 3.1: **Mean proportion of MeSH terms covered in Medline articles.** Articles published between 1980 and 2009, with given number of authors in the byline. Only articles with 2 or more authors are considered.



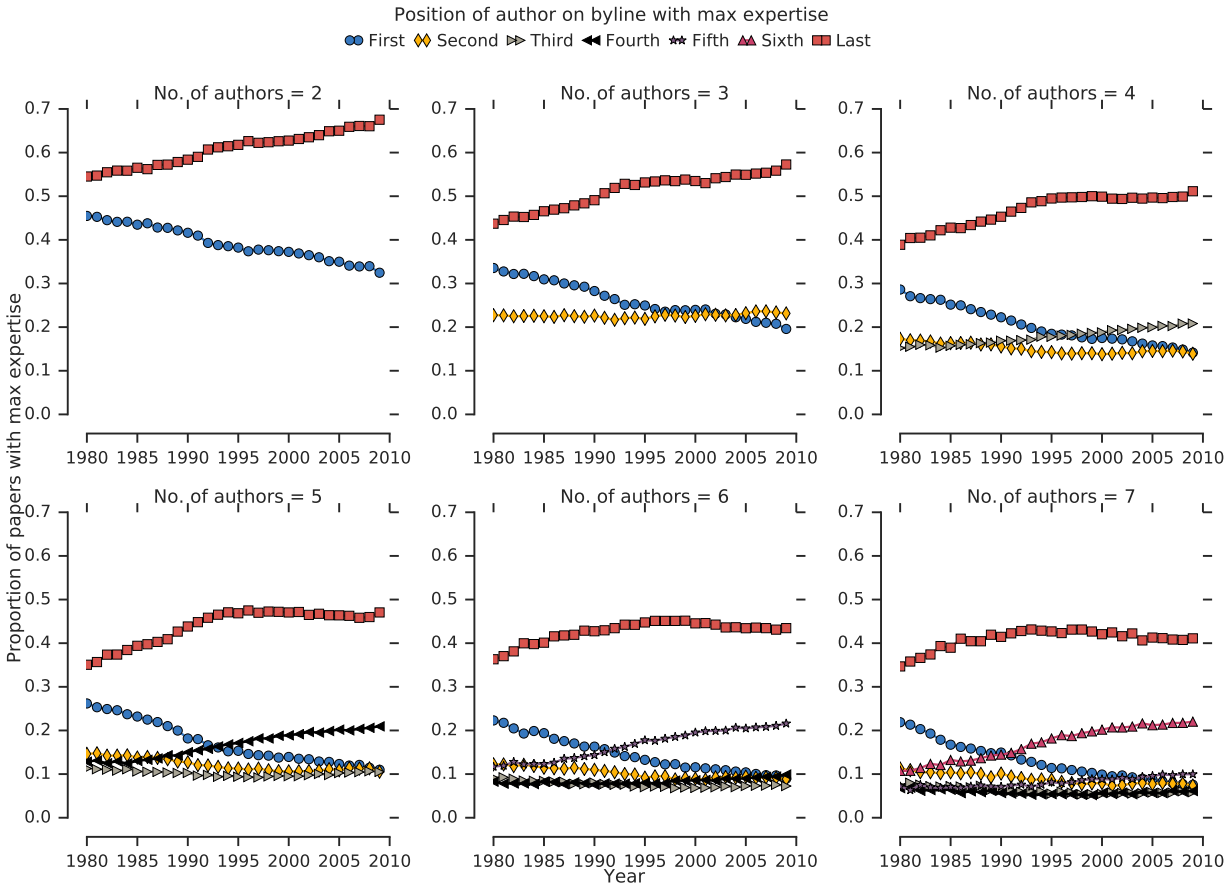


Figure 3.2: **Proportion of papers with maximum expertise by authors at various position in the bylines.** Papers having 2, 3, 4, 5, 6 and 7 authors.

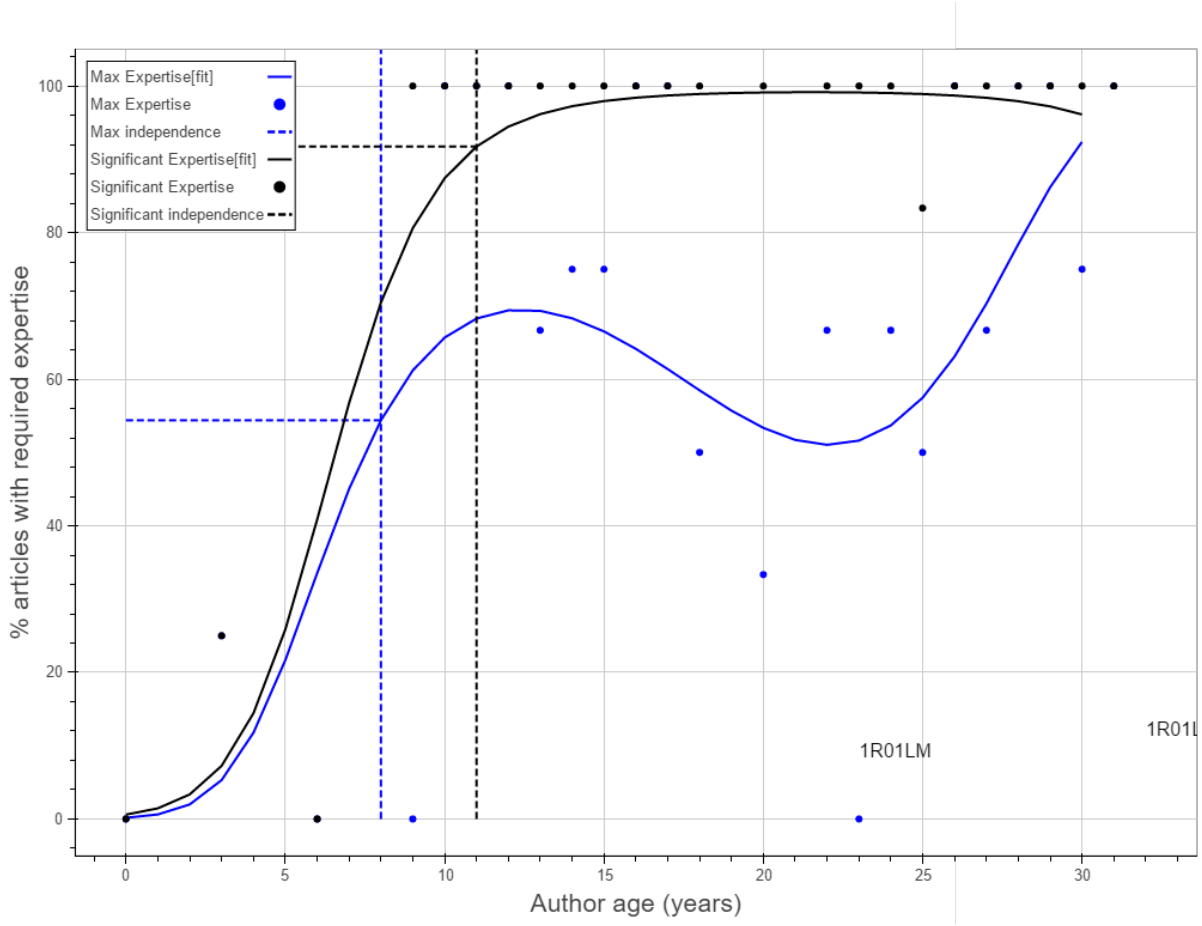


Figure 3.3: **Career profile of an author in PubMed using the Legolas interface.** The scatter points represent the actual proportion of articles in which the author had maximum expertise (blue) and significant expertise (black). For each type of expertise, the fit of the model profile is shown. More details at: http://abel.ischool.illinois.edu/legolas/profile?auid=207390_1

first paper in 1978 and had 77 papers by mid-2009 in MEDLINE. After 8 years, the author is dominant in the majority of their articles, and after 11 years, the author contributes significant expertise (dominantly or in complement to another dominant author) in more than 90% of their articles. These trends reflect a sense of independence. Similar profiles of other authors in Author-ity dataset can also be viewed at: http://abel.ischool.illinois.edu/legolas/profile.

3.4.1 Connection with DSTD

The computation of expertise for authors in scholarly corpus is similar to the computation of conceptual of novelty with DSTD as described in chapter 2. For the case of expertise, we accumulate the counts of MeSH terms not just for year, but for a pair of author and year. This allows us to construct an expertise profile of an author for each MeSH term based on the cumulative counts. Finally, for computing the relative expertise of authors on a paper, we select the current expertise scores of all the authors on all the MeSH terms and perform computation on these scores. This expertise computation framework is applicable to any DSTD, and can be considered a new IE task facilitated by a DSTD.

Chapter 4

Visualizing DSTDs using Social Communications Temporal Graph

4.1 Introduction

Communication on social channels such as social media websites, email, forums, and groups follows an inherent temporal network structure. Herein, each communication, e.g., a post, occurs at a specific point in time, which can be extracted from the post's metadata. Furthermore, each communication is also linked to a creator, e.g., a user, organization, topic, or another communication (like a retweet or quoted tweet, or a citation in scholarly data). Finally, the communication items can be tagged with additional numeric metadata, which can be used to score some attributes about the communication, e.g., number of comments, retweets, or shares. Existing timeline or network visualizations are not able to do justice to the temporal network structure of such communications. As described in chapter 1, we have defined this kind of data as DSTD.

We present Social Communications Temporal Graph (SCTG) [Mishra 2017] [1], which is a framework for visualizing DSTD. An example of such kind of visualization is presented in figure 4.1, which shows how the temporal evolution of conversations on a Facebook course group can be visualized. SCTG is a web based visualization which builds on the visualization principles of overview, zoom, filter, details-on-demand, relate, history, and extracts [Shneiderman 1996]. SCTG is aimed at highlighting the inherent time ordered generation of data in social communication channels while allowing various meta-data attributes to be shown alongside.

4.2 Background

This visualization framework is largely inspired from the overview, zoom, filter, details-on-demand, relate, history, and extracts – theory presented in [Shneiderman 1996]. However, there exists a vast literature on visualizing dynamic and temporal graphs. One particular instance of this is the TimeArcs [Dang et al. 2016],

[1] https://shubhanshu.com/social-comm-temporal-graph/

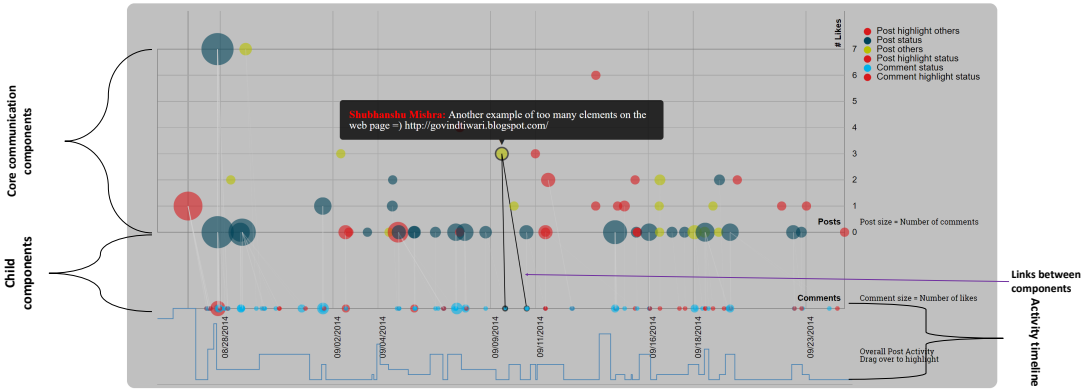


Figure 4.1: Visualization of conversation growth in a Facebook course group http://shubhanshu.com/FacebookGroupVisual/

interface which is most similar to our visualization. It provides an interface for visualizing dynamic network of entities. However, the network has no-meta data information about the entities. The framework is aimed at exploring the evolving relationship between entities, e.g., actors in the IMDB network, or named entities in blog corpora. [Beck et al. 2017] provide a comprehensive overview of dynamic graph visualization techniques in existing literature. Our framework is likely to fall under the **Timeline** $\rightarrow$ **Nodelink** $\rightarrow$ **Integrated layout** as per the taxonomy presented in [Beck et al. 2017]. Our framework is also very similar that of [Reitz 2010], which uses node scaling and coloring properties to visualize the ego network of an author. However, our approach differs by allowing the user to visualize networks of each entity in the complete network by using the hover option. The work of [Shi et al. 2015] is also related. However, they use a common timeline distribution and connect the network on top of it, but they do not account for links between different kind of entities. Finally, most of the aforementioned work is focused on presenting the visualization of ego-networks with examples from scholarly data. Our work enables extending these approaches to generic social network data.

Similarly, prior work on discourse visualization have utilized networks for presenting the discourse structure. For example, the NEREx framework [El-Assady et al. 2017] allows for a comprehensive visualization of debate transcripts using named entity relation graphs. They provide multiple views to explore the data, which allows for close as well as distant readings of the corpus. The utility of this visualization can be evaluated by studying the correlation between the values of various visualization components and a ground truth utility, e.g., prominent node sizes and their status in the social network.

4.3 Components

The general SCTG visualization framework is divided into the following components:

- **Core communication components**: This can be a user in a feed or a specific post
- **Child components**: This can be associated posts by a user or comments to a post
- **Component links**: Core communication is liked to its children
- **Activity timeline**: This quantifies the temporal activity measurement
- **Tool tips**: They provide additional data about each component
- **Component heights, scaling, and color**: Visualize additional metadata.

These components are not tied to a domain, but to any data which can be represented as DSTD. One limitation of the current approach is that the core and child component allows only one to one relationship between parent and child. This is the approach also utilized in the current implementation[2]. However, it can be easily extended to many to many relationships between core and child components. This extension will allow visualizing multi-label datasets, e.g., multiple named entities in each tweet.

4.4 Applications

In this section we discuss some of the applications of SCTG visualization on different DSTD data.

4.4.1 Facebook group data

Each post in the group feed is a core component, each comment is its children. Posts are colored based on content type (e.g. links, text, videos, etc.), scaling based on number of likes. See figure 4.1. The example in the figure 4.1 is for a course discussion group which was used by the students, instructors, and teaching assistants to discuss the assignment and share interesting findings related to each lecture. SCTG allows us to study how this social communication evolved over the course of a semester. Initial posts around syllabus and grading saw a lot of interaction, which was short lived. Later posts saw less interactions in terms of comments but often received a lot of likes. Most importantly, the communication was highest for days closer to the day of instruction.

[2]https://shubhanshu.com/social-comm-temporal-graph/

4.4.2 Twitter data with sentiment

Each user is a core component, each tweet are its children. Tweets can be colored based on sentiment labels, scaled based on retweet counts, and users scaled based on number of followers. See figure 4.2. The data used in this figure is around 400 tweets returned by the search query *Donald Trump*. Each tweet has been tagged with thier sentiment label. The sentiment labels have been aggregated at user level, and each user is colored based on the sentiments expressed in their tweets. This allows us to identify users who express both positive and negative sentiment in their tweets.

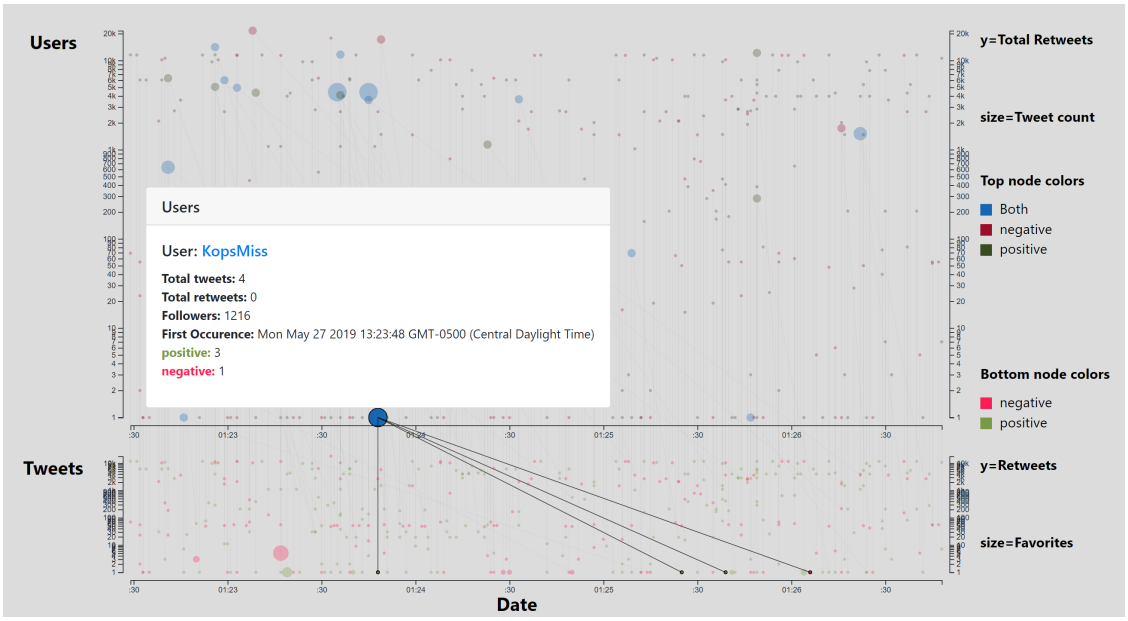


Figure 4.2: Visualization of tweet sentiment in Twitter corpora for the query Donald Trump. https://shubhanshu.com/social-comm-temporal-graph/tweet-sentiment

4.4.3 Wikipedia revision history

Each author is core component, revisions are children. Scaling based on number of revision size. See figure 4.3. This is another example of SCTG being applied to visualize a DSTD. The data is page revisions for the Wikipedia pages of *Game of Thrones (Season 8)* and *Avengers Engdame*. Users who revise both pages are highlighted. The scaling of revisions based on revision size, it shows that the revisions for *Avengers Endgame* are longer, and revision frequency for Endgame was higher few months back.

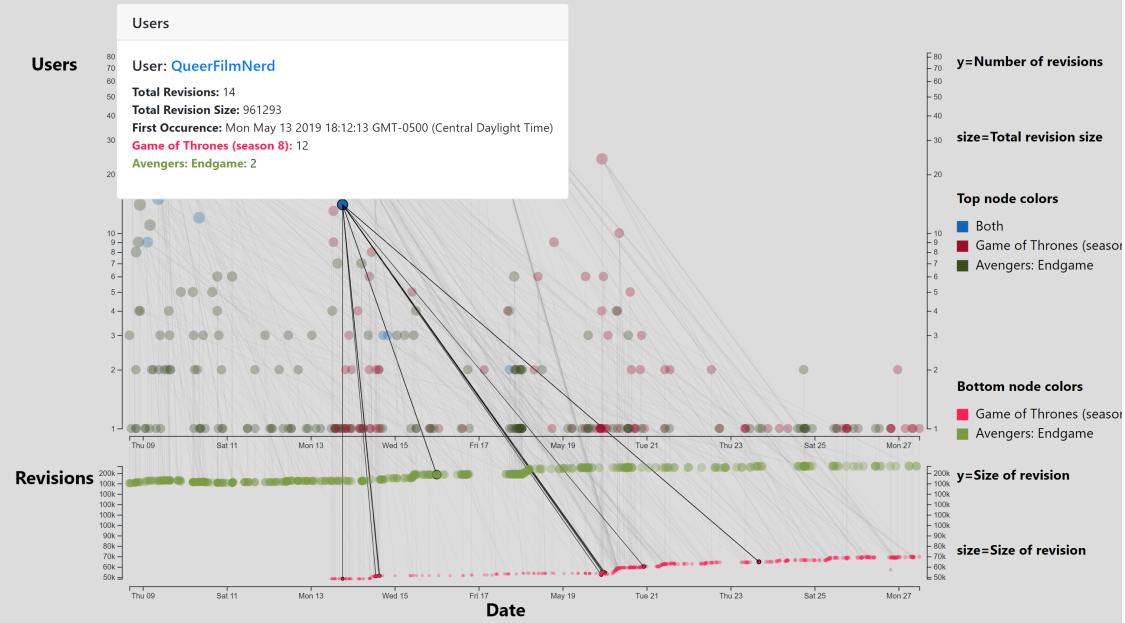


Figure 4.3: Comparison of revisions on Game of Thrones (Season 8) and Avengers: Endgame Wikipedia pages. https://shubhanshu.com/social-comm-temporal-graph/wikipedia-revisions

4.5 Conclusion

We presented a new visualization framework called social communications temporal graph or SCTG. SCTG allows visualizing DSTD data by highlighting parallel temporal trends between core and child components. Overall, SCTG provides the user with five degrees of freedom for visualizing the data. The visualization encourages interactive exploration of the temporal social graph.

Chapter 5

Socially relevant sentiment labels

Content in this chapter is based on our papers Mishra, S., Agarwal, S., Guo, J., Phelps, K., Picco, J., and Diesner, J. (2014). Enthusiasm and support. In *Proceedings of the 2014 ACM conference on Web science - WebSci '14*, pages 261–262, New York, New York, USA. ACM Press and Mishra, S. and Diesner, J. (2019). Capturing Signals of Enthusiasm and Support Towards Social Issues from Twitter. In *Proceedings of the 5th International Workshop on Social Media World Sensors - SIdEWayS'19*, pages 19–24, New York, New York, USA. ACM Press.

5.1 Introduction

Social media has been effective in allowing brands to identify and engage with their potential audiences. It has also allowed users to express their opinions towards social and political topics. A common theme in analyzing social media data has been to identify the opinion expressed in a given post [Kiritchenko et al. 2014, Kouloumpis et al. 2011]. Most opinion classification systems aim at classifying a post as either positive, negative, or neutral. Recently, there has been a focus on identifying the stance in a tweet towards a given topic [Sobhani et al. 2016, Mohammad et al. 2017]. Another common research topic has been identifying influential users in social networks, commonly known as *influencers* [Bakshy et al. 2011]. Identification of influencers allows brands to target specific users in the network, i.e., people who can lead to high visibility of their products.

In this work, we study the extension of the two approaches using the framework proposed in [Mishra et al. 2014] for tagging tweets across the two dimensions of enthusiasm and support. We start by investigating the quality of the annotated data introduced in [Mishra et al. 2014], followed by assessing its quality and robustness for the task of annotating tweets for enthusiasm and support. Next, an algorithm based on a weighted version of personalized page rank [Page et al. 1998, Brin and Page 1998, Xing and Ghorbani 2004] is introduced for combining the annotated tweets with user and hashtag mention networks. The algorithm allows for the identification of top users across the dimensions of enthusiasm and support. Our analysis

is conducted on tweets collected on three topics, namely, *Cyberbullying*, *LGBT community*, and *Chronic Traumatic Encephalopathy (CTE) in the National Football League (NFL)*.

We identify that the annotated datasets are of high quality using a combination of evaluation experiments. Furthermore, classifiers built using these datasets result in a high F1-score within and across subsections of datasets. This claim is studied by conducting three different experiments for assessing the robustness of classifiers trained within a specific dataset, across annotations by different annotators, and across different datasets. The identification of top users and hashtags based on our personalized page rank approach results in identifying users not found using the general page rank approach. We also contribute a unified classifier trained on the three datasets, along with an open source tool for classifying tweets and identifying top users and hashtags.

The chapter is organized as follows:

1. Section 5.2 introduces the labeling schema and emphasizes its importance for user classification for social causes. We also compare the labelling schema with existing labeling schema for tweet and user level.
2. Section 5.3 explains various aspects of the data.
3. Section 5.4 is aimed at thoroughly evaluating the robustness of the data, going beyond inter-annotator agreement scores, and introduces model based approaches for robustness assessment.
4. Section 5.5 compares our label schema with features generated from an emotion and sentiment lexicon.
5. Section 5.6 introduces a network based approach for aggregating enthusiasm and support labels, and user and hashtag levels.
6. Section 5.7 draws theoretical parallels between our approach and the net promoter's score.

5.2 Schema for tweet classification

We utilize the *enthusiastic, passive, supportive, and non-supportive (EPSNS)* orthogonal classification schema described in [Mishra et al. 2014] as the basis of tweet classification see in figure 5.1. This classification system allows us to capture the level of enthusiasm towards a topic, along with the level of support towards the topic. This methodology is more suitable for identifying enthusiastic and supportive users compared to the positive, negative, and neutral classification schema commonly utilized in sentiment classification literature [Pang et al. 2002, Wilson et al. 2005, Liu 2012]. Figure 5.2 presents a few examples from the data, which allows for comparing the EPSNS classification schema against the positive, negative, and neutral sentiment classification schema. We observe that there is no equivalent of enthusiastic and passive labels in sentiment

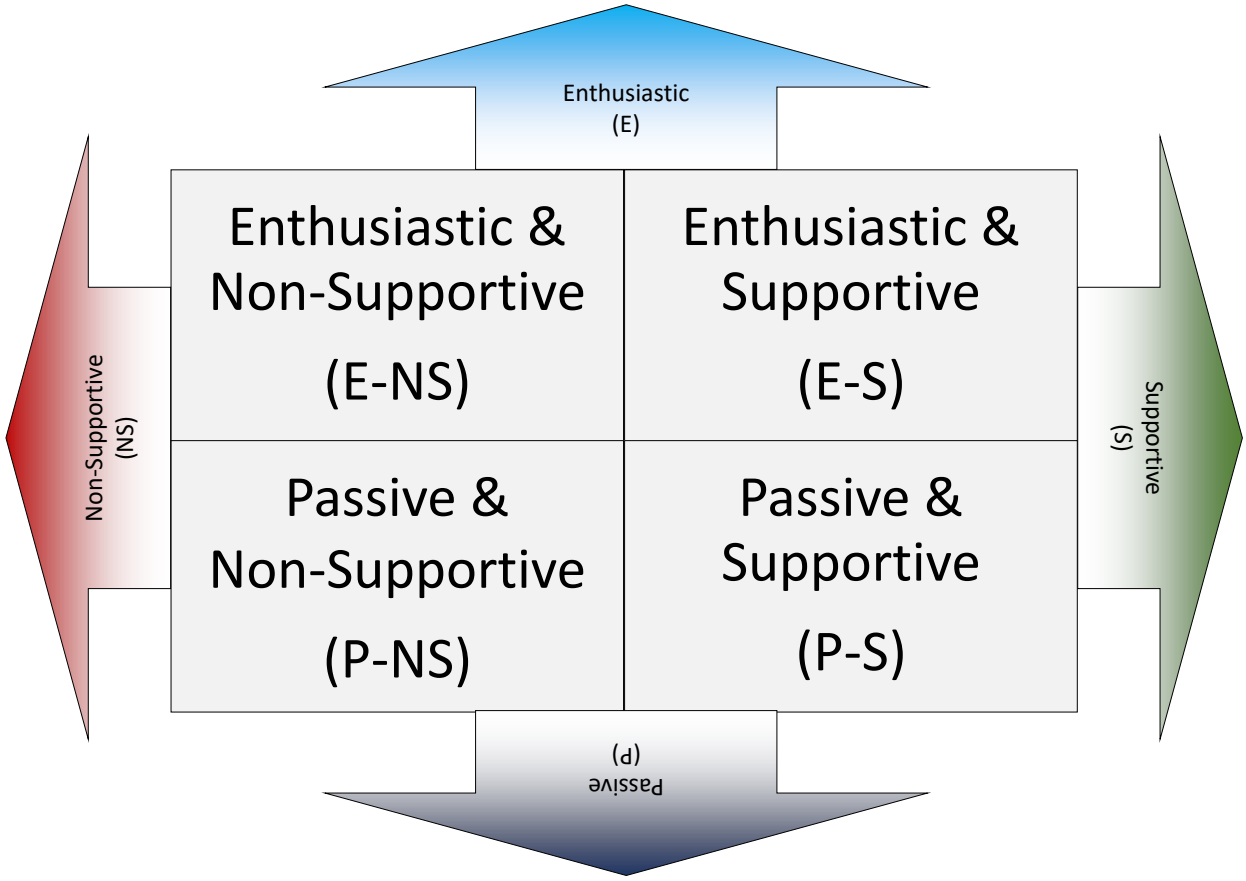


Figure 5.1: Enthusiastic, passive, supportive, and non-supportive (EPSNS), orthogonal classification schema for identifying enthusiasm and support.

classification schemas, and the labels postive, negative, and neutral are less important for identifying users authoring the tweets or users mentioned in them.

This classification schema is also different from valence or sentiment classification, where the goal is to assign sentiment strength to each word or subject in the text. The closest approach similar to our classification schema is the stance detection task, where given a topic, the model tries to identify the stance in the text towards a given topic. However, stance detection attempts to determine if the stance is in favor of or against topic [Mohammad et al. 2017]. In this case, the stance dimension can be considered similar to the supportive or non-supportive dimension. Furthermore, the enthusiasm and support classification schema is tailored to identifying user reactions to social causes as opposed to general topics, as not all topics elicit opinion of support or non-support. The focus on social causes makes the classification schema more focused and appropriate for the use case mentioned in this chapter.

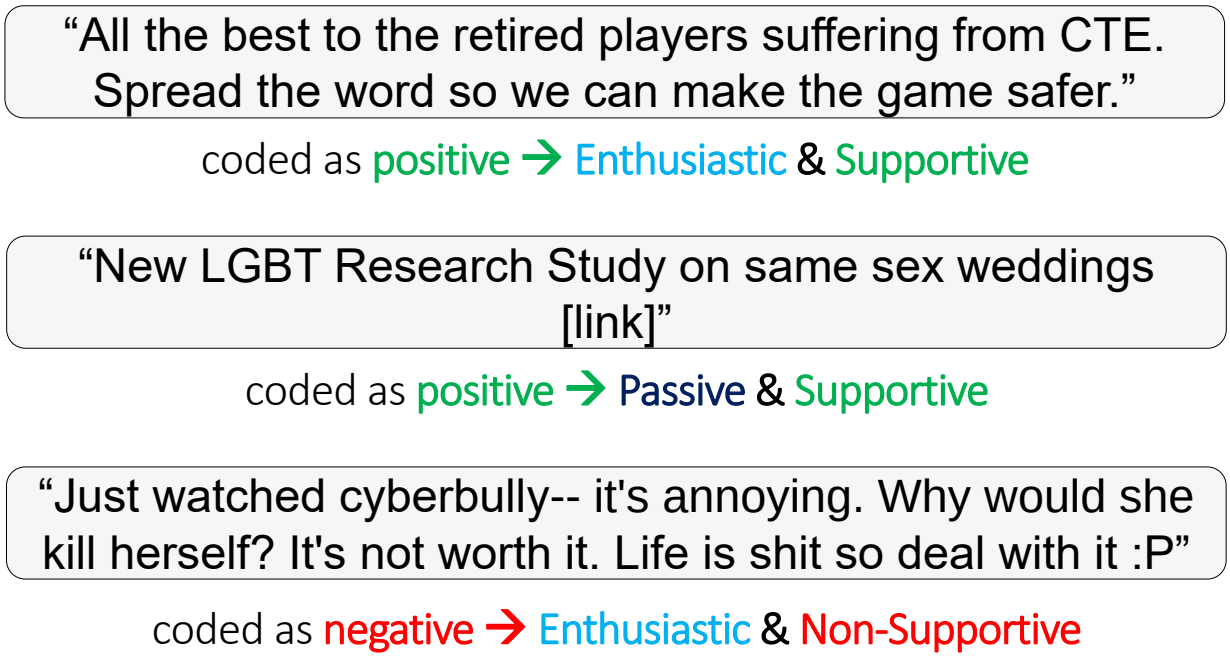


Figure 5.2: Example of using enthusiasm and support dimension instead of sentiment dimension.

5.2.1 Comparison with existing label schema for tweets

The most closely related labeling schema to the proposed schema is affect identification [Mohammad et al. 2018]. However, the affect identification task is context-free and does not account for the topic in consideration. A more relevant comparison is the stance prediction task [Mohammad et al. 2017], discussed above, which is based on the topic expressed in the tweet.

5.2.2 Comparison with user level aggregated labels

User level label aggregation has been proposed in [Preoiuc-Pietro et al. 2017] for political ideology identification [Preotiuc-Pietro et al. 2016] as Machiavellianism, Narcissism, Psychopathy. However, these classification schema are global and don't account for the context provided when the user expresses their opinion about a given topic. Furthermore, these classification schemas aim at classifying the user using all their tweets, along with profile information. Our approach is based on classifying tweets and thereafter aggregating the level of enthusiasm and support expressed in tweets relevent to the topic to get a user level measure of enthusiasm and support towards a topic.

5.3 Data

Mishra et al. [Mishra et al. 2014; 2019][1] introduced a dataset of tweets collected on the following topics: **cyberbullying (CB)**, **Chronic Traumatic Encephalopathy (concussions) in National Football League (CTE)**, and **Lesbian, Gay, Bisexual, and Transgender rights (LGBT)**. Each tweet in this dataset was annotated by two coders, ensuring that annotators provide the labels based solely on the text of the tweet and in the absence of their own opinions.

Table 5.1: Annotator (Anno.) Label Stats for each dataset. E: Enthusiastic, P: Passive, S: Supportive, NS: Non-supportive.

Lable	**CTE**		**CB**		**LGBT**	
Anno.	**1**	**2**	**1**	**2**	**1**	**2**
NS	39	35	23	30	43	46
S	156	166	190	206	227	233
E	215	181	227	226	195	181
P	232	201	82	84	207	221

The distribution of the number of tweets per label in the dataset is shown in table 5.1. Although the dataset for a specific issue can be small (sometimes yielding less than 100 samples for certain classes), the resulting dataset has a high inter-annotator agreement identified using percentage agreement (% = in table 5.2) as well as Cohen's κ [Cohen 1960] (see table 5.2). However, a limitation of Cohen's κ is that it uses a baseline of chance agreement [Pontius and Millones 2011], which may hide disagreement. In this chapter, we conduct several additional experiments to assess the similarity of annotator labels for training machine learning models. These experiments allow us to confirm if tweets with similar features have similar annotations.

Table 5.2: Inter annotator agreement between two annotators. % = is percentage agreement, and κ is Cohen's kappa.

	Enthusiasm			**Support**		
	%=	κ	**N**	**%=**	κ	**N**
CTE	0.96	0.91	379	0.98	0.92	165
CB	0.94	0.86	309	1.00	1.00	209
LGBT	0.93	0.87	395	0.97	0.89	257

[1] Data: https://doi.org/10.13012/B2IDB-2603648_V1

5.4 Evaluating data robustness for training classification models

We assess the robustness of the annotated data towards its suitability for training generalizable prediction models. The data were prepared for analysis as follows: The texts were tokenized using a Twitter tokenizer in NLTK[2]. Each term was lemmatized using the NLTK lemmatizer. A document is represented in terms of the TF-IDF score of its unigrams along with bigrams and trigrams (identified via pointwise mutual information in each dataset).

Our first analysis focuses on the top salient terms identified for each dataset, label, and annotator combination. The salient terms are identified using mean TF-IDF scores. Table 5.3 shows that for the majority of datasets and labels, the salient terms identified across the annotated tweets are highly similar across annotators. The similarity in top salient terms correlates with relatively high inter-annotator agreement scores presented in table 5.2. This finding also provides support for the claim that the data for each label are similar in their word distribution across the annotators. Table 5.3 reveals that salient terms for enthusiastic are more conversational compared to other labels, and express proclamations, e.g., *screw* (CTE), *emoticon* (CB), and *you's* (LGBT). The analysis of salient terms leads the way for our next experiments on assessing feature importance for training generalizable tweet classifiers.

The second analysis examines the quality of logistic regression models trained on data using labels from only one annotator. This allows us to assess the consistency of an annotator (similar to the concept of intra-coder reliability). We conduct three-fold cross-validation. All evaluation scores represent micro-F1 scores (unless specified otherwise). Table 5.4 shows that for each dataset, cross-validated models trained using labels from only one annotator result in $\sim 70\%$ F1 score for the dimension of enthusiasm and $\sim 83\%$ for support. The scores are higher for identifying supportive versus non-supportive tweets compared to enthusiastic versus passive tweets. The high F1 scores are correlated with high inter-annotator agreement. Since this experiment relied on training and evaluation using a single annotator's labels, the F1 scores also provide information about the consistency in annotations by that annotator. Hence, it appears that it was easier for the annotators to label tweets for the support dimension than for the enthusiasm dimension. This can be due to codebook instructions or the nature of the texts.

In the third analysis, we identified if a model trained on two datasets (e.g., CTE+LGBT) and a single annotator's labels will generalize to the remaining dataset (e.g., CB) annotated by a different annotator. The results of this experiment are shown in table 5.5. We observe that a model trained on a given annotator's labels consistently gets comparable evaluation scores when tested on the remaining dataset and labels from either the same or the other annotator. It is important to note that the classifier performance for the support

[2] https://www.nltk.org/

Table 5.3: Salient n-grams identified in annotations of each annotator (A), for each label (L), across datasets (D).

D	L	A	Word (mean TF-IDF)
CTE	E	1	screw (0.96), chilling (0.87), rtnew (0.77), prevalent (0.74), fund (0.70)
		2	screw (0.96), chilling (0.87), prevalent (0.74), fund (0.70), wow (0.69)
	P	1	explained (0.82), reminds (0.80), coverage (0.76), possible (0.75), jermaine (0.71)
		2	explained (0.82), coverage (0.76), difference (0.76), possible (0.75), jermaine (0.71)
	S	1	chilling (0.87), coverage (0.76), tragedy (0.69), terrible (0.67), reveal (0.67)
		2	coverage (0.76), tragedy (0.69), johnathan (0.68), terrible (0.67), mild (0.64)
	NS	1	isn't (0.53), CTE (0.51), pool (0.47), brandon (0.47), weeden (0.47)
		2	isn't (0.53), CTE (0.51), faced (0.50), sea (0.50), pool (0.47)
CB	E	1	): (0.98), convicted (0.94), truce (0.91), boot (0.90), bro (0.84)
		2	): (0.98), convicted (0.94), truce (0.91), boot (0.90), bro (0.84)
	P	1	ali (0.96), pledge (0.94), watching (0.86), actually (0.83), white_people (0.77)
		2	ali (0.96), pledge (0.94), actually (0.83), favorite (0.78), watching (0.78)
	S	1	ali (0.96), pledge (0.94), convicted (0.94), youre (0.90), bro (0.84)
		2	): (0.98), ali (0.96), pledge (0.94), convicted (0.94), bro (0.84)
	NS	1	gay (0.71), best (0.64), go (0.62), caleb (0.60), raver (0.60)
		2	actually (0.83), gay (0.71), best (0.64), live (0.62), caleb (0.60)
LGBT	E	1	opinion (0.95), intended (0.70), sexless (0.68), you's (0.68), uncomfortable (0.67)
		2	opinion (0.95), maybe (0.92), intended (0.70), sexless (0.68), you's (0.68)
	P	1	legalized (0.97), heart (0.84), outside (0.75), drink (0.73), 10_thing (0.71)
		2	dont (0.89), outside (0.75), drink (0.73), 10_thing (0.71), new_campaign (0.71)
	S	1	legalized (0.97), outside (0.75), drink (0.73), 10_thing (0.71), biblical (0.70)
		2	legalized (0.97), outside (0.75), drink (0.73), 10_thing (0.71), biblical (0.70)
	NS	1	larry (0.69), glb (0.69), passion (0.65), kill (0.62), ship (0.62)
		2	actually (0.70), larry (0.69), glb (0.69), kill (0.62), ship (0.62)

Table 5.4: Cross-validation micro-F1 scores for training the model with three-fold cross-validation using an individual dataset with labels from a single annotator.

	Annotator	max		min		mean		std.	
Data	Model	1	2	1	2	1	2	1	2
CTE	Enthusiasm	0.872	0.858	0.517	0.523	0.713	0.697	0.138	0.131
	Support	0.877	0.879	0.800	0.821	0.823	0.835	0.033	0.019
CB	Enthusiasm	0.806	0.796	0.625	0.538	0.740	0.719	0.053	0.070
	Support	0.929	0.910	0.875	0.861	0.899	0.881	0.018	0.017
LGBT	Enthusiasm	0.866	0.815	0.515	0.548	0.667	0.654	0.114	0.085
	Support	0.854	0.839	0.822	0.809	0.839	0.831	0.010	0.009

dimension is considerably worse ($\sim 60\%$ compared to $\sim 83\%$) when the model trained on CTE+CB data is tested on LGBT data. However, the comparison between the drop in F1 scores is not fully comparable as the data in table 5.4 are the mean evaluation scores across three splits on one-third of the full data for a given topic (e.g., LGBT), whereas the scores in table 5.5 are based on the full dataset for that topic.

Table 5.5: Evaluation using micro-F1 scores for testing on one dataset from a single annotator and training on other data from the other annotator.

Test Data	**Annotator Model**	**Test → Train ↓**	**1**	**2**
CTE	**Enthusiasm**	**1**	0.729	0.715
		2	0.772	0.743
	Support	**1**	0.800	0.826
		2	0.800	0.826
CB	**Enthusiasm**	**1**	0.761	0.758
		2	0.738	0.742
	Support	**1**	0.873	0.843
		2	0.883	0.864
LGBT	**Enthusiasm**	**1**	0.729	0.694
		2	0.634	0.604
	Support	**1**	0.604	0.602
		2	0.596	0.599

In the last analysis, we evaluated the transferability of a model trained on two of the datasets with combined annotator labels on the remaining dataset with combined annotator labels. Instead of taking the majority vote of the labels and discarding instances with conflicting labels, we created a dataset such that $(X, y) = \bigcup_i (X_i, y_i)$, here X is the feature matrix, and y is the label vector, y_i are the labels by annotator i. Table 5.6 shows that the model yields evaluation scores comparable to those presented in table 5.5. The evaluation score for the support dimension of the LGBT dataset is again lower compared to the last experiment (table 5.5). The lower F1 score as well as Cohen's κ signifies least inter-annotator agreement for the LGBT data, as shown in table 5.2. Our analysis analysis also supports the hypothesis that the models trained using all annotator labels are often more accurate that those trained on single annotator label.

Table 5.6: Evaluation using micro-F1 scores for testing on one dataset and training on the others, combining annotations from all annotators.

Test Data	**Enthusiasm**	**Support**
CTE	0.749	0.801
CB	0.763	0.855
LGBT	0.731	0.559

Finally, we created a combined model which was trained on all data from all annotators. This model was tuned using three-fold cross-validation. The model evaluation scores are summarized in table 5.7. The

combined model achieves high mean F1 scores, suggesting the appropriateness of the extracted features that were used for classification. We also investigated the top features for each label identified by these combined models as shown in table 5.8. These features include the presence of URLs and account mentions, while some of the top features are also related to the respective datasets, e.g., *support_gay_right*, *cyber_bully_white_people*, and *nhl_concussion_case*. Table 5.8 also shows that top features for enthusiasm contain emotive words like *agree*, *lol*, and *great*, as well as mentions of accounts. Top features for the passive class include the presence of URLs, and mention of news outlets like *Reuters*. Similarly, top features for supportive include URL mentions, and explicit mentions of n-grams containing the word *support*, while top features for non-supportive include words like *hate*, *angry*, and *kill*.

The presence of dataset-specific features, i.e., specific words from a dataset, among top features can be smoothed out by the use of word embeddings, which can allow the model to learn more general features. We did not experiment with word embedding based models, owing to the small size of our training data and our need to use cross-validation to identify the variability in model evaluation scores.

Table 5.7: Cross-validation micro-F1 scores on training the model with three-fold cross-validation using combined annotations from all annotators across all datasets.

Model	**max**	**min**	**mean**	**std**
Enthusiasm	0.943	0.544	0.798	0.174
Support	0.972	0.845	0.902	0.058

Table 5.8: Top features for each class in the combined model trained using all datasets and annotator labels.

Model	**Label**	**Feature scores**
Enthusiasm	**E**	@account (-44.63), ! (-11.62), rt (-10.02), great (-6.70), read (-6.49), lol (-6.13), thechronicleher (-6.05), war (-5.96), agree (-5.85), ? (-5.79)
	P	head (5.00), scan (5.00), testing_company (5.47), love (5.68), reuters (5.89), supreme_court (5.92), actually (6.53), legalization (7.88), nhl_concussion_case (8.65), URL.COM (10.44)
Support	**NS**	kill (-5.66), go (-4.90), as (-4.85), hate (-4.21), mlk (-4.10), cyber_bully_white_people (-4.00), surveillance (-3.91), angry (-3.72), ? (-3.62), annie (-3.58)
	S	suicide (2.38), URL.COM (2.43), life (2.47), support_gay_right (2.54), absolutely (2.56), 100 (2.65), did (2.72), asshole (2.75), cyber_bully (3.12), sad (3.46)

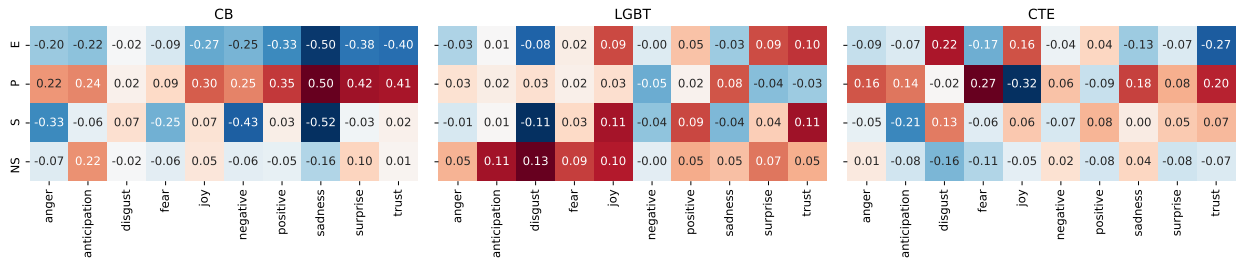


Figure 5.3: Correlation between enthusiasm and support labels across datasets and proportion of EmoLex categories present in a tweet

5.5 Comparison with EmoLex

The EmoLex lexicon was introduced in [Mohammad et al. 2017; 2018]. It classifies words into eight emotion categories namely, *anger, anticipation, disgust, fear, joy, sadness, surprise, and trust*, and two sentiment categories, namely *positive, and negative*. In this section, we compute the correlation between our four labels, namely *E, P, S, NS*, and the ten EmoLex categories. For each tweet in our dataset, we tag each token with its EmoLex categories, then we sum the EmoLex categories per tweet. We divide the total for each EmoLex category by the number of tokens in each tweet to avoid bias in the results because of tweet length. Finally, we compute the correlation between the EmoLex category scores and the tweet labels as shown in figure 5.3. We find that the correlation is not consistent across the datasets. In particular, the LGBT dataset has the most variation from the other datasets. We find that passive labels have higher correlation with sadness, surprise, and trust in the CB and CTE datasets. Exploring this correlation further, we identify the top 20 words in the combined corpora, along with their EmoLex labels. As shown in table 5.9. It is evident that the majority of words in the sadness category comes from words like case, and lawsuit which are prominently used in the CTE dataset. Similarly, words like marriage are likely to be present in the LGBT dataset and do not actually convey an emotion. This analysis also highlights the usefulness of full tweet based labeling as opposed to dictionary based scoring of tweets, as dictionary based approach might capture domain specific tokens, leading to higher false positives. However, a dictionary based approach can help us is comparing a new labeling schema to an existing labeling schema without training a new model.

5.6 Network based user and hashtag identification

In this section, we describe the construction of two types of networks for identifying the most enthusiastic and supportive accounts in a corpus of tweets related to a given social cause.

Table 5.9: Top 20 words in the combined corpora with their word counts and EmoLex labels

word	count	EmoLex labels
marriage	347	anticipation, joy, positive, trust
concussion	202	anger, negative, sadness
bully	96	anger, fear, negative
ban	73	negative
don	43	positive, trust
lawsuit	36	anger, disgust, fear, negative, sadness, surprise
law	35	trust
legal	34	positive, trust
love	32	joy, positive
court	29	anger, anticipation, fear
believed	26	trust
dazed	25	negative
good	23	anticipation, joy, positive, surprise, trust
league	23	positive
time	21	anticipation
case	19	fear, negative, sadness
watch	19	anticipation, fear
equality	19	joy, positive, trust
show	18	trust
join	17	positive

5.6.1 Network construction

In order to identify the top account for each type (e.g., enthusiastic or supportive), we create a network of user mentions. If $user_1$ mentions $user_2$ in tweet t then we create a directed edge between $user_1$ and $user_2$. Additionally, we use the probability of t being predicted as either enthusiastic (E), passive (P), supportive (S), or non-supportive (NS) as edge attribute, we all add a weight $w = 1$, as an attribute for each edge. Finally, if the same edge occurs multiple times in a corpus, we sum the scores for each of E, P, S, NS, and w. Furthermore, for each directed edge between n_1 and n_2, we sum the above mentioned for all the outgoing edges to get a score for n_1.

A similar network based on account hashtag mentions is also constructed. Here the edge is between *user* and *hashtag* instead of $user_1$ and $user_2$. This network allows for the identification of top hashtags about a social cause along the enthusiastic and the supportive dimension, using a tweet corpus.

5.6.2 Identification of top nodes in the network

In order to identify the top nodes of each type, we use the weighted personalized page rank algorithm. The personalization weights are identified as the $exp(\sum_j score_j^1 - \sum_j score_j^0)$, where $score_j^1$ and $score_j^0$ represent node score for enthusiastic (or supportive) and passive (or non-supportive) respectively. The exp is used

ensure the weights are positive (a requirement of the personalized page rank algorithm).

Using the above algorithm, we identify the top users in the user mention graph as well as the top user/hashtags in the user-hashtag graphs as described above. The top enthusiastic and supportive nodes are shown in table 5.10 and table 5.11. The tables highlight that the personalized page rank based methods allow us to identify a unique set of users and hashtags compared to the simple page rank approach (All). These usernames and hashtags have a higher importance in enthusiasm or support aligned social networks. For example, for CTE, one of the top nodes in E/P mention as well as hashtag network is *@Sports_Brain*, which is the Twitter handle of a company that provides concussion management programs. Similarly, the NFL account is among the top supportive accounts for CTE. For cyberbullying, the top hashtag for enthusiasm as well as support is *#cyberbullying*, while the top account in the mention network is *USR2* who is a Youtuber, hence likely to have faced cyberbullying. Finally, for LGBT, the top enthusiastic and supportive users are *@free_equal*, which is a United Nations initiative for LGBT equality, and *USR_FilmExpert*, who is an LGBT film expert.

Table 5.10: Top 3 nodes in the mention network based on different PageRank algorithms (**PR**=PageRank score). In the *All* row, ranking and scores are based on overall PageRank. Accounts of individuals were replaced with USR to protect privacy.

	CTE		**CB**		**LGBT**	
	Account	**PR**	**Account**	**PR**	**Account**	**PR**
E/P	USR1	0.191	USR2	0.050	free_equal	0.033
	Sports_Brain	0.191	USR4	0.050	UN_Women	0.030
	USR3	0.041	USR5	0.043	USR_FilmExpert	0.030
S/NS	USR6	0.186	USR2	0.062	free_equal	0.044
	USR12	0.068	USR4	0.062	HRC	0.033
	NFL	0.066	USR5	0.054	USR_FilmExpert	0.028
All	USR7	0.021	USR8	0.009	HRC	0.024
	NFL	0.015	USR9	0.008	Tedofficialpage	0.010
	frontlinepbs	0.009	USR10	0.008	USR11	0.010

5.7 Comparison with Net Promoter Score

Net Promoter Score (NPS) [Reichheld 2003] is a score used by companies to quantify customer satisfaction. It is based on the aggregate of customer responses to the question: *"Would you recommend this company to a friend?"*, on a scale of 1-10. This answer helps in identifying if a customer is a *promoter* ($score \in (9, 10]$) of the company, *passive* ($score \in (7, 8]$) or its *detractor* ($score < 7$). The difference between the percentage of promoters and the detractors gives us the net promoter score for that company. In the same light, we can consider enthusiastic support of a user towards a topic, as they recommend the topic to their friends

Table 5.11: Top 3 nodes in the hashtag network based on different PageRank algorithms (**PR**=PageRank score). In the *All* row, ranking and scores are based on overall PageRank. Accounts of individuals were replaced with USR to protect privacy.

	CTE		**CB**		**LGBT**	
	Account	**PR**	**Account**	**PR**	**Account**	**PR**
E/P	Sports_Brain	0.264	USR5	0.479	USR1	0.234
	USR2	0.264	#cyberbullying	0.116	#lgbt	0.105
	#cte	0.137	#parenting	0.102	USR3	0.032
S/NS	#nfl	0.062	WestYorksPolice	0.357	USR4	0.427
	#cte	0.058	#cyberbullying	0.122	#lgbt	0.101
	Sports_Brain	0.051	#bullying	0.094	#gay	0.087
All	#nfl	0.048	#cyberbullying	0.048	#lgbt	0.063
	#cte	0.023	#cdnpoli	0.018	#gay	0.008
	#concussion	0.015	#cyberbully	0.015	#questionnier	0.006

(social network). A passive support/non-support will be similar to the NPS passive category, while an enthusiastic non-support expressing can be used to classify the user as a detractor. Similar to the use case of NPS, a promoter of a social issue is likely to be a loyal supporter of the cause, and might be likely to help towards the growth of that cause, within their social network. This parallel conceptualization of our user-level classification can help in utilizing the vast literature of effective uses of NPS to grow social issues.

5.8 Conclusions

In this work, we have evaluated the EPSNS classification schema for labeling tweets, and subsequent applications for identifying accounts and hashtags expressing enthusiasm and support towards topics of public interest. More specifically, we evaluated the robustness of the annotation of three datasets on different topics through a series of experiments. Our findings demonstrate the robustness of the annotations and the generalization capability of the models trained on these data. Furthermore, we utilized the tweet level classification scores in the personalized PageRank algorithm for identifying top accounts and hashtags that express enthusiasm and support towards the three considered topics.

We compared our approach to the EmoLex lexicon, as well as drew parallels between our user based aggregation and the net promoter's score. This helped us shed light on the use case for identifying promoters and detractors for social issues.

Our approach is limited by the small dataset and the usage of simple linear models. Furthermore, a direct comparison with a stance classification, which could further establish the utility of the EPSNS classification task for account labeling, is not provided. However, since the goal of this work was to introduce the core idea of identifying accounts based on the measurement of enthusiasm and support expressed in tweets,

the comparison with other tasks can be pursued in future studies. Our work can help in merging text classification and network analysis for account labeling on social media platforms.

5.8.1 Connection with DSTD

The socially relevant sentiment labels proposed in this chapter along with the methodology to identify enthusiastic and supportive users based on their co-mention network is a direct application of IE labels used to make inference about unlabeled nodes (users) in DSTD. The labels extracted here can be combined with the SCTG visualization proposed in chapter 4.

Chapter 6

Meta data association with sentiment

Content in this chapter is based on our paper Mishra, S. and Diesner, J. (2018). Detecting the Correlation between Sentiment and User-level as well as Text-Level Meta-data from Benchmark Corpora. In *Proceedings of the 29th on Hypertext and Social Media - HT '18*, pages 2–10, New York, New York, USA. ACM Press.

6.1 Introduction

Sentiment prediction is a well-studied text classification problem [Liu 2012, Pang and Lee 2008] that has mostly been applied to reviews, e.g., of movies [Pang et al. 2002, Socher et al. 2013] and consumer products [Pang and Lee 2008]. Sentiment analysis is also frequently used to identify the valence of social media posts and other types of text data [Fan et al. 2014, Go et al. 2009, Pak and Paroubek 2010]. Additionally, sentiment detected from text data has been shown to be correlated with or predictive of individual as well as aggregated behavior, e.g., the political leaning of people [Tumasjan et al. 2010] or stock market trends [Bollen et al. 2011]. Many of these applications involve quantifying the distribution of sentiment classes, a task that is commonly referred to as sentiment quantification [Gao and Sebastiani 2015].

A major limitation of existing sentiment classification systems, when applied in the social media domain, is their reliance on mainly the text content of a post or tweet. However, platforms such as Twitter provide access to rich meta-data along with the text of the post. These meta-data include properties of social media posts and their authors, which may provide useful context for studying the sentiment conveyed in a tweet, and can complement the text features for the sentiment classification task. Earlier research has used tweet-based meta-data, such as the existence or number of URLs, hashtags, and mentions, as features for tweet sentiment classification [Mishra et al. 2014; 2015], as well as user-level meta-data for creating sentiment-based user networks [Mishra et al. 2014]. However, there is a limited body of literature on using or incorporating meta-data of tweets for improving sentiment classification, and most of this prior work is based on non-public and non-standard datasets [Tan et al. 2011, Vosoughi et al. 2015]. With this chapter, we aim to contribute to a more comprehensive understanding of the relationship between these meta-data

features and the sentiment of tweets across multiple datasets. This work is enabled by the availability of large-scale standardized sentiment-annotated Twitter corpora, such as the Semantic Evaluation's Twitter sentiment task corpus [Nakov et al. 2016b;a, Rosenthal et al. 2015], another recently available dataset of 1.6 million multilingual tweets [Mozetič et al. 2016], and a few other public datasets, which allow us to search for the existence of any meaningful relationships between the meta-data of tweets and tweet sentiment.

In this chapter, we identify how various meta-data are (on average) related to the sentiment of tweets in existing sentiment annotated benchmark corpora. Our analysis is limited in that we identify patterns at an aggregate level across all datasets considered. However, we further support our observations by including additional data from users in our dataset, and observing the correlation between meta-data and sentiment (as predicted by a baseline classifier). The goal of this research is to understand the distribution of meta-data characteristics across these datasets, and to identify if these meta-data can reveal biases in sentiment annotation. Finally, we also detect how using these meta-data can help as features in a classifier to improve sentiment classifiers as well as sentiment quantification.

Our contributions with this chapter are 1) an analysis of the relationship between sentiment (as per annotation) of tweets and tweet meta-data, 2) a validation of observed relationships between sentiment and meta-data by using additional tweets from users in benchmark data annotated with sentiment using a baseline classifier, 3) using the meta-data of tweets along with tweet text content for predicting sentiment, 4) a system called Meta-data Enhanced Sentiment Classification (MESC) for efficiently incorporating meta-data-based sentiment information of a tweet into existing text-based classifiers in a model-agnostic way, and 5) demonstrating the use of standard sentiment classification datasets for non-text-based sentiment analysis, thereby providing a baseline to compare other work against. The code reproducing this work as well as additional supplementary analysis are available at:
https://github.com/napsternxg/TwitterSentimentBenchmarks

6.2 Background

Achieving high accuracy rates for sentiment classification is challenging, especially for social media data. This is evident from the top accuracy rates of state of the art systems, which are often below 90% for movie reviews [Kim 2014, Socher et al. 2013], and even lower for Twitter data [Abbasi et al. 2014, Mohammad et al. 2013, Nakov et al. 2016b;a; 2013, Rosenthal et al. 2015]. One possible reason for this effect is the occasionally implicit assumption that the sentiment of a post is fully conveyed in the text; disregarding the text's context. Furthermore, sentiment classification models based on text do not necessarily perform

well when applied across domains [Mishra et al. 2015] due to factors such as diverse language use, concept evolution, and concept drift [Masud et al. 2010]. Recently, there has been an interest in quantifying the distribution of sentiment in a given collection of tweets [Gao and Sebastiani 2015, Nakov et al. 2016b]. This topic deals with the focus of earlier studies on using aggregates of sentiment distributions to model changes in peoples mood [Bollen et al. 2011], election results [Tumasjan et al. 2010], reviews [Asur and Huberman 2010], and the stock market [Bollen et al. 2011]. Our approach is methodologically closest to the research by Tan and colleagues [Tan et al. 2011], who used the full network of user follower, friend, and user mention along with the tweet text to infer the sentiment of tweets by using a computationally expensive graphical model. Our approach differs from that in several ways; for example, we only conduct analyses at the aggregate level of user and tweet meta-data, and our method can more easily be plugged into existing systems where text-based sentiment classification is already implemented.

6.3 Data

Most existing sentiment datasets categorize the data into three classes, namely negative, neutral, and positive. We use the same set of labels for our analysis, and only consider datasets annotated with those labels. Additionally, we also consider a different set of binary class labels to identify if tweets are opinionated (either positive or negative) or non-opinionated (neutral). Furthermore, we selected only datasets with tweet IDs for each tweet label. This is important for collecting user and tweet-level meta-data using the Twitter API. Finally, to infer any meaningful relationship between meta-data and sentiment labels, we want to avoid any dataset specific idiosyncrasies in annotation and tweet distribution. We address this bias mitigation need by using sentiment labeled datasets from various time periods, on different topics, and labeled by using different annotation guidelines and interfaces (but still the same classes). Using this approach, we hope to infer general relationships between tweet meta-data and sentiment labels after pooling the selected eligible datasets.

Based on our above-mentioned criteria, we identified six high quality, publicly available datasets as eligible for our analysis. The first dataset (referred to as SemEval) is from the recurring Twitter sentiment classification task of SemEval [Nakov et al. 2016b; 2013, Rosenthal et al. 2015], and includes all training, development, and test data from 2013 throughout 2016. We only consider the data for the tasks where the goal was to classify tweet sentiment as either negative, neutral, or positive. The second dataset is a large collection of multilingual tweets from European countries from a study by Mozetič and colleagues [Mozetič et al. 2016]. We only work with the English tweets from this dataset. This dataset is available from the

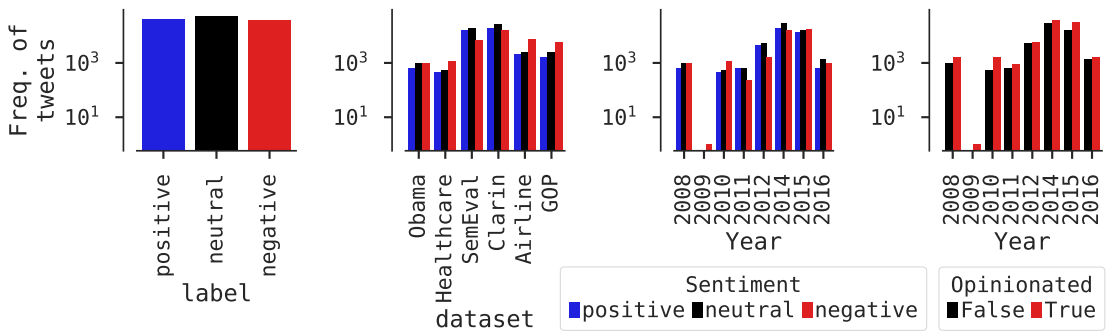


Figure 6.1: Frequency of sentiment labels across datasets and years. Opinionated tweet are either positive or negative.

CLARIN data repository and therefore referred to as Clarin. The next two datasets, namely, Airline and GOP, were generated on the Crowdflower platform and are hosted on Kaggle[1]. These two datasets include crowd sourced sentiment annotations for tweets about various Airlines as well as the first GOP debate of 2016. The final two datasets come from Saif and colleagues [23], and are about the Obama-McCain debate (referred as Obama) and healthcare (referred as Healthcare).

Our analysis considers user-level and tweet-level meta-data. Since the Twitter terms of service do not allow for tweet data to be (re-)distributed, we collected the tweet JSON data using the Twitter API, and then merged these data with the labels provided in each dataset. For evaluating the effect of meta-data features on tweet classification, we consider a training, development, and test split of each dataset. For the SemEval dataset, we use the provided training, development, and test splits, while for the other datasets, we create training, development, and test splits using a 72%, 8%, and 20% ratio of the datasets. The frequency of instances across the various datasets, labels, and data splits is presented in Table 6.1. Furthermore, the aggregate distribution of instances across the datasets and labels is presented in figure 6.1. This figure shows that our datasetś sizes are distributed across three orders of magnitude: large datasets with numbers of instances around 40K-60K, which include SemEval and Clarin, followed by smaller datasets, which are Airline and GOP, and finally, the smallest datasets of around 2K instances, namely Obama and Healthcare.

A major strength of the set of datasets that we consider their temporal diversity, with tweet instances ranging from 2008 to 2016 (Figure 6.1). Both SemEval and Clarin were collected over lengthy time periods (SemEval during 2011-16, Clarin during 2013-15) [Nakov et al. 2013, Mozetič et al. 2016]. However, the English tweets in the Clarin dataset are limited to 2014. The Healthcare dataset spans seven months between 2009-2010. The Airline dataset entails 2 days (2015), and the GOP (2015) and Obama (2008) dataset span 1 day each. All other datasets cover a shorter duration. A possible limitation with existing

[1]https://www.kaggle.com/crowdflower/datasets

Table 6.1: Distribution of the instances across datasets, labels, and data splits.

Dataset	Train			Development			Test			Total
Labels	Negative	Neutral	Positive	Negative	Neutral	Positive	Negative	Neutral	Positive	
Airline	5,515	1,843	1,467	613	205	163	1,532	512	408	12,258
Clarin	11,485	19,418	13,496	1,276	2,158	1,500	3,191	5,394	3,749	61,667
GOP	4,230	1,818	1,173	471	202	130	1,175	505	326	10,030
Healthcare	834	378	321	93	42	36	232	106	89	2,131
Obama	715	707	455	80	79	50	199	197	126	2,608
SemEval	4,313	13,031	11,405	479	1,448	1,268	1,198	3,620	3,169	39,931

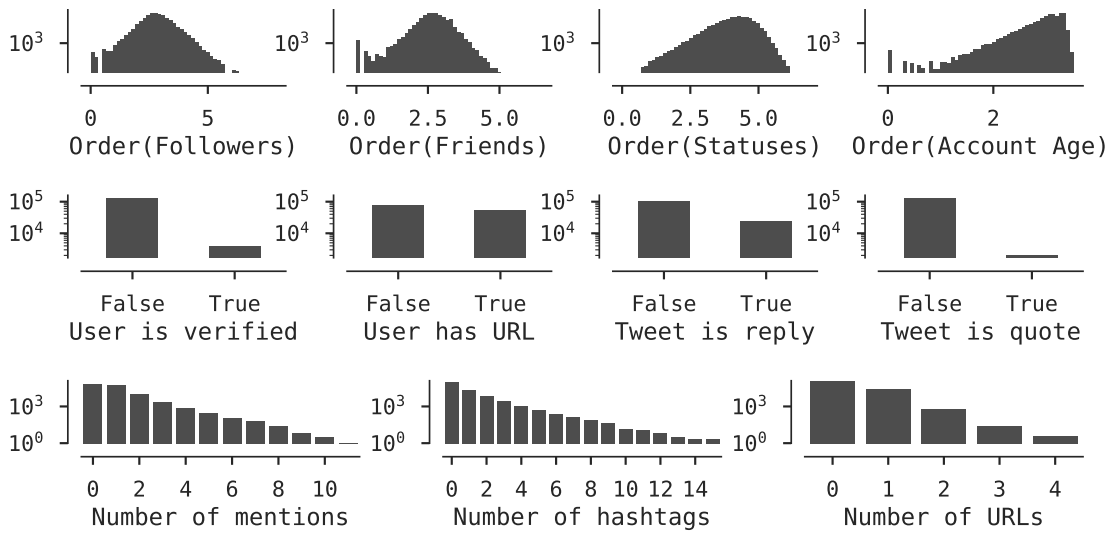


Figure 6.2: Frequency of user-level and tweet-level meta-data. $Order(x) = \log_{10}(x)$

research on Twitter sentiment classification is the analysis of tweets from a specific period, which may result in a failure to capture trends across years as well as in overfitting on trends from a specific period. Using multiple datasets in this study aims at mitigating this issue.

For each tweet instance in our dataset, we extract a) user-level, and b) tweet-level meta-data from each tweet's JSON file. User meta-data includes number of statuses, followers, and friends, user account age (in days) based on account creation date and tweet creation date, if the user account is verified, and if the user profile has a URL. Tweet meta-data include number of mentions, URLs, and hashtags, if the tweet is a retweet, and if the tweet quotes another tweet.

Since the distribution of the user-level meta-data is highly skewed and the tail of this distribution extends to large values, we transform the values by using a log transform with base 10, capturing their order. This allows our analysis to be robust to changes in meta-data values for user accounts over time as the log value changes are gradual compared to raw count changes. A distribution of the user and tweet meta-data is shown in Figure 6.2. Finally, a major advantage of jointly considering multiple datasets is that they are freer from selection and annotation biases than single sets with respect to the properties we are studying. It is common practice to perform the annotation task using only the text of the tweet [Nakov et al. 2016a], hence any bias in annotation because of user-level meta-data features being studied is less likely. We do acknowledge that the actual original tweet collections might still feature multiple types of sampling biases.

6.4 Methods

In the following sections, we describe our methods for analyzing the relationship of sentiment with user and tweet level meta-data.

6.4.1 Relationship between sentiment and user meta-data

We define the following properties of a user and the respective measurement of these properties from the user meta-data:

1. **Activity level** is measured in terms of the number of statuses posted by the user.
2. **Social status** of a user is defined as the amount of incoming connections to the user on the platform, and measured as the number of followers of the user.
3. **Social interest** of a user is defined as the amount of outgoing connections user make on the platform. The Twitter API defines this measure as the number of friends of a user. We measure it as the number of users that a user follows.
4. **Account age** is measured as the number of days that the account has existed until the user posted a given tweet.
5. **Profile authenticity** is measured using Twitter specific information, such as presence of a URL in the user profile, as well as the Twitter-provided verified user tag. This is a binary measure.

As mentioned earlier, each numeric measure was analyzed using its order instead of the raw count. The order is defined as $f(x) = \log_{10}(1+x)$, where x denotes the quantity being measured. We consider the order instead of the absolute value of the measure to prevent an effect of outliers on our analysis. We study the relationship between the sentiment of a tweet and its user-level meta-data using the log odds ratio (logOR) of the tweet belonging to a given class. Specifically, the log odds ratio of the correct class $C = 1$, for a meta-data value, $X = x$, relative to the meta-data value, $X = x_0$, is given as

$$logOR(x) = \ln\left(\frac{P(C=1|X=x)}{P(C=0|X=x)}\right) - \ln\left(\frac{P(C=1|X=x_0)}{P(C=0|X=x_0)}\right). \tag{6.1}$$

For the empirical analysis, the numeric attributes are partitioned into equal sized bins, and x_0 refers to the central bin. To investigate the interactive effect of correlated user meta-data features, we examine the relationship between the ratio of the numeric user meta-data features and the log odds ratio for a given class.

6.4.2 Relationship between sentiment and tweet meta-data

The tweet-level meta-data capture certain content properties of tweets. The placement of URLs, mentions, and hashtags can be aimed at providing evidence, shout-outs, and topical information, respectively. Furthermore, whether a tweet is a reply or quotes an existing status can provide an additional signal for the sentiment prediction. We study the relationship between the tweet-level meta-data features and sentiment class in the same way for the user-meta-data features.

6.4.3 Meta-data model

We use the user-level and tweet-level meta-data-based features to model the log odds of a tweet belonging to a specific class. We consider three settings: 1) only user-level meta-data features, 2) only tweet-level meta-data features, and 3) a linear combination of user and tweet-level meta-data features. We model the log odds of the tweet belonging to a given sentiment class by conditioning on all user/tweet/user+tweet level meta-data features. Numeric features are log transformed as described above. This is done by parameterizing a logistic regression model per class label; using a linear combination of the meta-data features. Based on the empirical relationship between the log odds and the meta-data features, certain features (e.g., social status, social influence, and activity level) are parametrized using an additional quadratic term. Models are fit on the aggregate of all datasets. We refer to the model with user and tweet meta-data features as the meta-data model.

6.4.4 MESC - Meta-data Enhanced Sentiment Classification

In this section, we describe our MESC system. The goal of this system is to seamlessly allow existing text-based classification systems to utilize meta-data-based attributes for enhancing the classification performance of existing text-based classifiers. We hypothesize that the sentiment class probabilities from the meta-data-based models can be used to enhance the prediction accuracy of text-based classifiers for social media texts. The MESC system runs through the following steps:

1. Get the score (can be log probabilities or SVM score) for each sentiment class from the text-based model (**text model**).
2. Get the score (can be log probabilities or SVM score) for each sentiment class from the meta-data model (**meta model**).
3. Train a multinomial logistic regression model (**joint model**) using the class-based scores from the text model and the metadata model as the only features.

4. The final sentiment of the tweet is the one predicted by the joint model.

The framework described above considers the text model and meta-data model as black-box models, and is independent of the features used to train these models.

6.5 Results

In this section, we describe the results obtained using our analysis methods.

6.5.1 Relationship between sentiment and user meta-data

The relationship between the log odds ratio of a tweet belonging to a given class based on various meta-data features is shown in Figure 6.3.

First, we discuss the correlation between a tweet user's activity level (order of statuses) with the sentiment label. We observe positive linear trend in the log odds ratio of a tweet being neutral with the activity level of its users. This might be partially explained by the fact that many of the accounts with high numbers of statuses are corporate or organizational accounts, e.g., @AmazonHelp, which has posted 1.25M statuses. These accounts might be less likely to engage in opinionated conversations. However, the relationship for low activity levels is highly variable, suggesting higher sentiment diversity among low activity accounts. Additionally, we observed that the overall relationship between activity and sentiment also holds for each of the individual datasets. Furthermore, tweets from users with mean activity level are more likely to be opinionated. Amongst the activity levels of opinionated users, we observe a quadratic relationship between the tweet being labeled as positive and the user having more than 10 tweets (order 1). This suggests that these median activity level users are more likely to tweet with positive sentiment, compared to others. However, no such trend is seen for the negative class, where the downward trend plateaus after the median activity level.

Second, we analyze the effect of the user's social status (the order of the number of followers of the user) on predicting the sentiment of the tweet. Figure 6.3a shows a strong quadratic trend across all classes for this feature. Tweets from high follower accounts are more likely to be more neutral than opinionated.

Third, we examine the relationship between tweet sentiment and the users social interest (as quantified by the order of the number of followers of the user). Figure 6.3a shows a strong quadratic relationship between both variables for the positive class. Furthermore, as the order of number of friends increases, the tweets from those users are less likely to be neutral, and more likely to be negative after crossing the median value. This might reflect that users with extreme social interest (as defined in this chapter, i.e., either very

low or very high order of number of users they follow) are less likely to post positive tweets, while the average social interest users might be more likely to express positive sentiments.

Fourth, the account age significantly correlates with the sentiment classes: older accounts tend to post less positive or neutral tweets, and are more likely to post negative tweets. This might reflect veteran users who criticize issues or actively take part in social media conversations rather than just sharing neutral tweets.

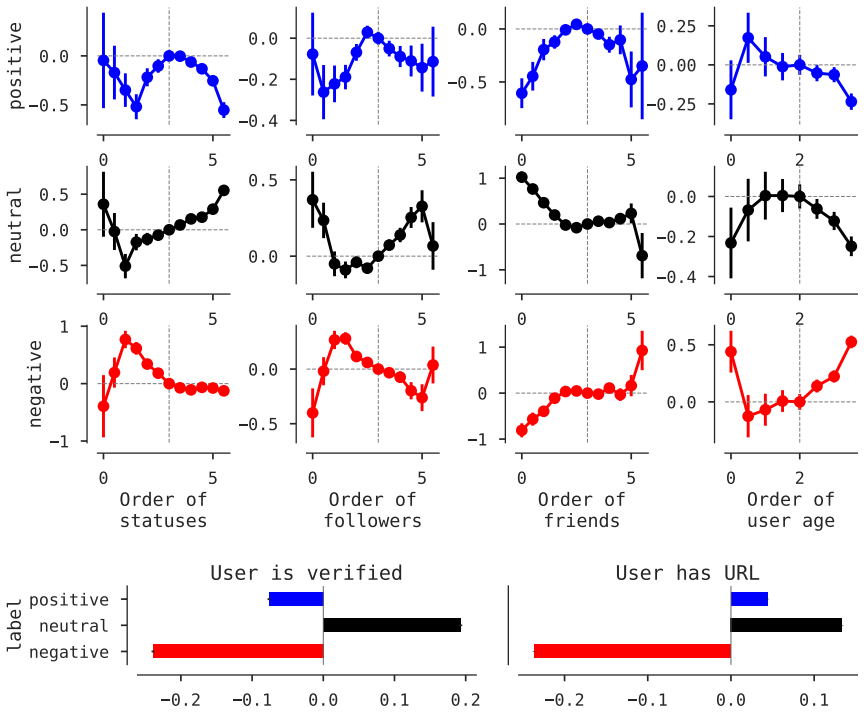
Fifth, we study the user meta-data features that reflect profile authenticity (results shown in Figure 6.3b). We found that the presence of a URL in the user's profile is correlated with user postings being more neutral or positive, while the lack of a URL reflects a higher likelihood of negative tweets. Similarly, verified users are more likely to post neutral tweets compared to non-verified users. Both findings suggest that user authenticity is related to opinionated tweeting behavior. This trend might suggest that non-authentic users are more likely to share negatively perceived posts, while authentic profiles share more positive and neutral posts.

Finally, to test for the correlation of features, we further examined the Pearson correlation between the numeric features. We observe a positive correlation between measures of social status and social interest. We also observe a low positive correlation between social activity and social status. Based on this insight, we further examine the relationship between the sentiment class with the ratio of the numeric user-level meta-data features. These quantities are provided in Figure 6.4.

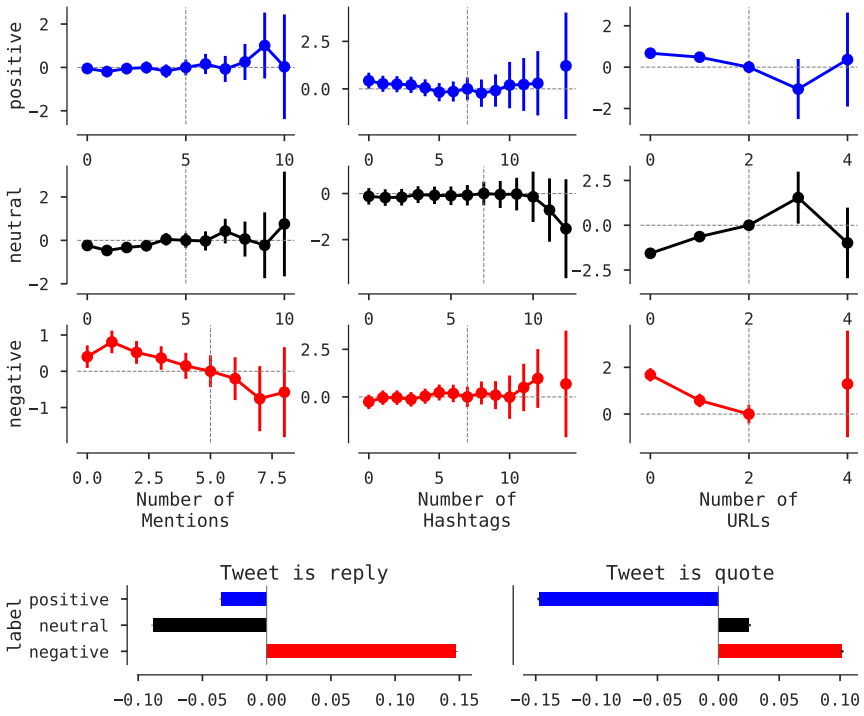
We found a strong relationship between the order of ratio of statuses and friends across all sentiment classes. Specifically, the log odds of neutral sentiment increases as the order of the ratio increases, while it decreases for negative sentiment. This reflects that low order ratio neutral tweets are less likely compared to high order ratio. This may suggest that users with a high number of statuses compared to their number of friends are mostly sharing neutral (non-opinionated) content (like the @AmazonHelp account mentioned before).

6.5.2 Relationship between sentiment and tweet meta-data

We now turn to the relationship between tweet sentiment and tweet-level meta-data (Figure 6.3b). A distinct pattern can be seen between the number of URLs and the sentiment class: as the number of URLs increases, the probability of the tweet being neutral also increases. This might be partially accounted for by the fact that news agencies or blogging services share the URL of their content via Twitter. Furthermore, the presence of a URL in non-neutral tweets is more likely to reflect a positive tweet. We also observe a decline in the probability of a negative tweet with an increase in the number of user mentions in a tweet. However, Figure 6b shows that tweets that are replies or direct quotes are more likely to be negative than neutral or



(a) User-level meta-data



(b) Tweet-level meta-data

Figure 6.3: Meta-data features vs. sentiment classes. Y-axis in top plots and X-axis in bottom plots, is log-odds ratio, with respect to point at dashed lines.

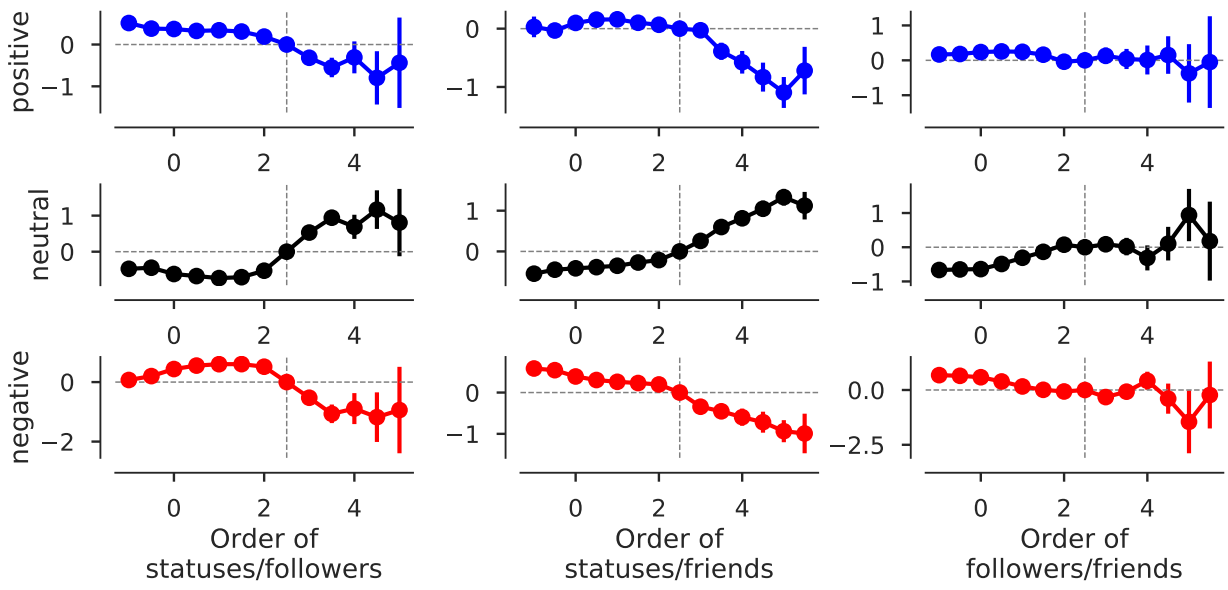


Figure 6.4: Ratio of user meta-data features vs. sentiment

positive.

6.5.3 Analysis with additional user tweets

The analyses up here have focused on sentiment-annotated data where the original annotators used the text of a given tweet to provide a sentiment label. One valid criticism of studying correlations between user meta-data and sentiment is that a tweet may exhibit multiple sentiments. However, in this study, we are only interested in the most common patterns of relationships between sentiments of tweets and their meta-data. Furthermore, we are not interested in causal analyses, but in the correlation between sentiment and meta-data features. More specifically, our current analysis is only reflective of the expected and most likely correlation of a user or tweet and the meta-data.

We conduct an additional set of experiments, this time based on data from all 110,388 users in our dataset. We collected their most recent 200 tweets (for 98% of the users we were able to collect more than 190 tweets). The choice of the number of recent tweets was made to reduce the computational complexity of processing the data. We collected around 20 million tweets from the accounts in our dataset. Since this data was not annotated with sentiment, we decided to annotate it with a highly accurate lexicon and rule-based sentiment analysis system tailored for Twitter data (Vader Sentiment) [Hutto and Gilbert 2014]. Once the sentiment labels were assigned, we conducted the same analyses as before for the various meta-data feature categories. Our results are presented in Figure 6.5.

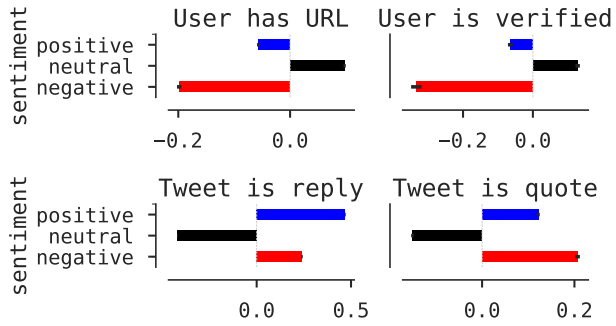
Among categorical attributes, the observed trends are consistent with our findings (Figure 6.3) for user-

level attributes except for the correlation between positive sentiment and the user profile having a URL (see Figure 6.5a). For the latter case, the results show a reversal in the correlation, but this can be attributed to the low correlation in our original analysis. For tweet quotes, we see quite a different trend for the positive and negative label, which is likely to be caused by the classifier's inaccuracy. Similar patterns exist for the numerical attributes: we observe similar but more noisy (compared to the human annotated data) patterns for all numerical user meta-data (Figure 6.5b). Note that these plots differ in the log odds ratio values from previous plots because of the selection of different baseline values. Another important point is the general trend for each of the curves, which are similar to those observed in the analysis based on the annotated data. Finally, we found that the patterns of ratio of user-level meta-data from our original data analysis are persistent in this version of the data. Figure 6.5c shows that the trends are similar to those observed in the original data, with the exception of neutral sentiment for the statuses/followers plot.

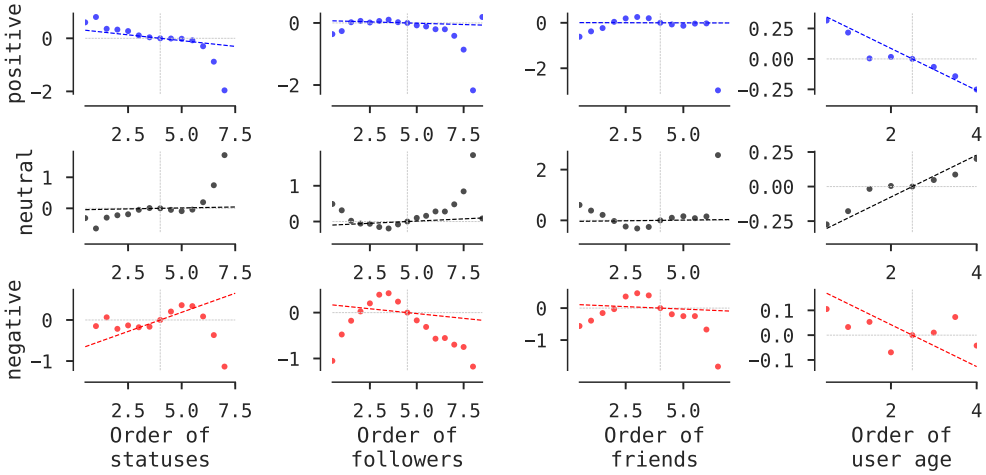
6.5.4 Meta-data model

First, we consider the aggregated effect of using all user-level meta-data features in modeling the probability of a given sentiment of a tweet. Table 6.2 shows the model parameters for each sentiment class. The model parameters confirm the observation of high user activity levels being correlated with higher odds of neutral sentiment and low odds of negative or neutral sentiment (Figure 6.3a). Similarly, average activity levels are associated with a higher probability of positive as well as negative sentiments. Similarly, the relationships for social interest are also consistent with the earlier observation that greater social interest is related to more negative tweet sentiment. Additionally, we observe that the coefficients of social status are very small and not particularly significant for all sentiment classes. Furthermore, the strong relationship between profile authenticity and sentiment class holds true across all three sentiment classes. This confirms the earlier observation that profile authenticity might be correlated with tweet sentiment.

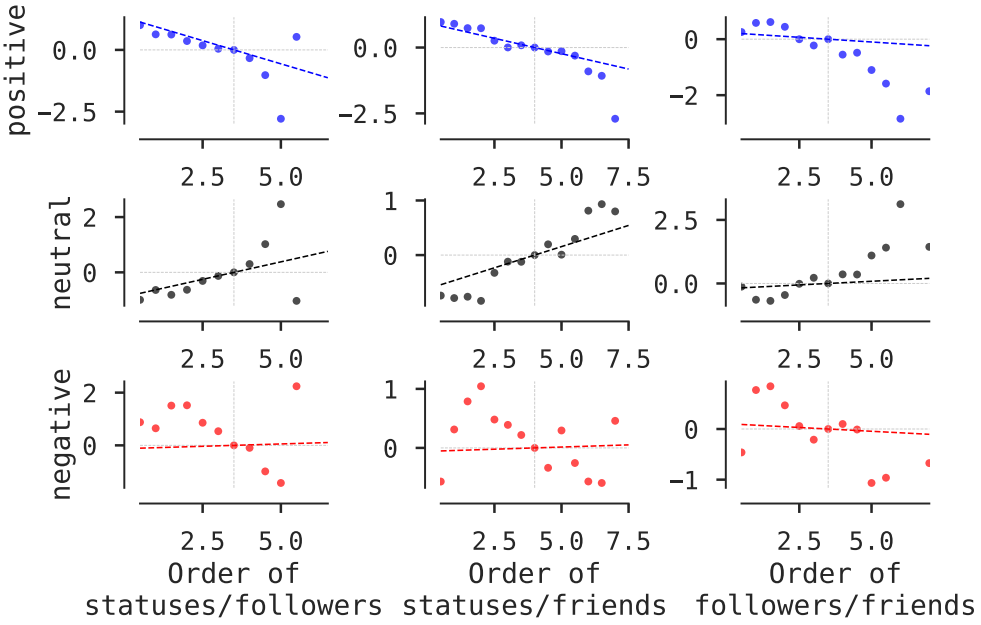
Second, we model all tweet-level meta-data measures (like the process used for the user meta-data) to study their cumulative effect on the odds of each sentiment class. Table 6.2 shows the model coefficients for each sentiment class. This model confirms our empirical observations: high numbers of URLs increase the probability of a neutral sentiment, while decreasing the probability of negative and positive sentiment. This effect is larger for negative sentiment. However, the trend is reversed for the number of user mentions. The tweet-level meta-data model associates large number of mentions with slightly higher odds of negative sentiment compared to positive and neutral sentiment. Furthermore, we observe a new pattern in the number of hashtags and the sentiment classes, indicating that higher numbers of hashtags are related to more negative sentiment. Next, we consider the joint effect of the user and tweet-level meta-data on modelling the



(a) Categorical meta-data



(b) User-level numerical meta-data



(c) Ratio of user-level numerical meta-data

Figure 6.5: Meta-data features vs. sentiment classes using recent 200 tweets for each user in the data. Sentiment predicted using VADER Sentiment [Hutto and Gilbert 2014]. X-axis in 6.5a, and Y-axis in 6.5b and 6.5c are log-odds ratio, with respect to point at the dashed lines.

probability of the sentiment classes. Table 6.2 shows the coefficients of the joint model per class. We observe that the effects of the profile authenticity remain quite close to its value in the user meta-data models. We make the same observation for the activity levels, social interest effects, and the tweet meta-data measures. Overall, we observe that after controlling for all other factors, social status is least correlated with any of the sentiment classes.

6.5.5 Evaluation of the MESC system

In this section, we evaluate our MESC (Meta-data Enhanced Sentiment Classification) system using a simple text-based as well as our meta-data-based sentiment classifier. For the text model, we consider a unigram bag-of-words (BOW) model, where each word was lower-cased. We removed all user mentions, hashtags, and URLs from the tweet text. Finally, we use the TF-IDF (term-frequency * inverse-document frequency) weight for each unigram as the feature of each tweet. The text model is trained using a multinomial logistic regression, which is suitable for modelling the predicted probabilities for each sentiment class. For the meta model, we trained a multinomial logistic regression classifier using the user+tweet meta model features described above. Finally, the joint model uses a linear combination of the class scores (log probabilities) from the text and the meta model, as well as the pairwise products between the scores from the text and the meta model. Evaluation of sentiment classification was done using the overall accuracy, macro-averaged value for precision, recall, and F1 score. Table 6.3 shows that the joint model results in significant gains over the text-based model on all the datasets. The gain is especially evident for the Healthcare, GOP, and SemEval datasets, where the F1 score of the joint model on the test data increases by 8.5%, 4.2% and 1.9%, respectively. The lack of significant improvement on the Clarin dataset is probably because the simple text-based model is already performing at the level of inter annotator agreement between the tweets as reported in [Mozetič et al. 2016]. Finally, we studied the effect of using the joint model for quantifying the distribution of tweets. For this analysis, we only considered the test dataset, and compared the true class distribution to the predicted distribution of classes from the various models using Kullback-Leibler (KL) divergence [Kullback and Leibler 1951] (a standard measure for measuring the distance between probability distributions) as used in prior research [Gao and Sebastiani 2015]. Table 6.3 shows that the distributions produced by the joint model is closer to the true distribution compared to the text-based model. The overall evaluation of the models on the test data is presented in Table 6.3. We observe that the recall and F1 scores of the joint model are consistently higher than for the text model (by 0.5-4%), however, there is a slight dip in precision and accuracy. The lower precision and accuracy, whereas higher recall and F1, for the text joint models compared to text-based models reflects the ability of the joint models to correctly predict a larger

proportion of labels at the cost of increasing the mistakes on these predictions.

6.6 Discussion and Conclusion

We have presented an analysis of the relationship between various meta-data features and the sentiment of tweets. Our findings suggest that certain user characteristics, such as their activity levels, profile authenticity, and the amount of profiles the users follow, can be highly correlated with the sentiment labels of tweets. Our proposed approach for integrating sentiment information correlated with meta-data into existing text-based classifiers results in a consistent increase in evaluation performance for sentiment classification and quantification tasks. We believe that this approach of using the meta-data-based sentiment correlation information of the tweets can serve as a prior for machine learning, which might help to improve the classification performance of text-based systems. This may be especially useful in cases where the tweet text has a high out of vocabulary (OOV) token rate. One major limitation of our approach is the usage of linear and pairwise combinations of prediction scores from the base model as well as the meta-data-based model. Although this approach results in a simple combination of models, more sophisticated approaches using deep neural networks can also be used for improving the prediction accuracy for the joint models. Furthermore, in our current experiments we used a standard unigram-based sentiment prediction model as a text model. It can be improved by using more sophisticated text classification algorithms based on current state of the art practices, thereby allowing us to further investigate the benefits of using meta-data models.

Another limitation of our analysis is the availability of labeled corpora that are annotated based on the text of the tweet. A more rigorous evaluation of our method could be done by annotating tweets based on both their meta-data and text content. This can help to better understand if the human annotators change their mind about the best fitting sentiment label when they also consider the meta-data of tweets. The methods we have described for studying correlation can also be applied to other social media corpora, such as Reddit or Wikipedia comments. We believe that our results can support the exploration of additional meta-data-based features for complementing text-based sentiment analysis research of social media data, and the creation of standard datasets that capture these effects in detail. Finally, our results matter for the advancement of social media analytics: knowing expected tweet sentiments based on user-level meta-data enables a) the detection of outlier tweets, which may signal special relevance of individual data points, and b) the calibration of individual users within samples of multiple users. The second point can help to address a major issue with sampling biases for social media data, i.e., the normalization of individual users who have unexpectedly high or low sentiments in comparison to their user-level features. In classic survey research,

identifying individual tendencies for responding in an overly positive or negative way is of high relevance, and such work can inform social media research. This chapter offers a remedy for starting to fix this need. Finally, we provide code that can be used for reproducing the results along with supplementary analysis.

6.6.1 Connection with DSTD

This chapter provides a useful insight into using user and tweet level metadata which is available in DSTD (chapter 1). The DSTD representation of social media data is further supported by the consistency of metadata correlation with sentiment labels as shown in our analysis using 200 recent tweets of users.

Table 6.2: Feature weights for models of tweet sentiment based on user and tweet metadata. (*) marked coefficients are statistically NOT significant ($p > 0.005$)

Model types	User			Tweet			User + Tweet		
Labels	Negative	Neutral	Positive	Negative	Neutral	Positive	Negative	Neutral	Positive
Intercept	-0.79	0.02 *	-1.78	-0.85	-0.56	-0.69	-0.55	-0.36	-1.54
Activity level	-0.75	0.31	0.47	-	-	-	-0.72	0.28	0.47
Activity level ˆ2	0.08	-0.01 *	-0.08	-	-	-	0.08	-0.01 *	-0.08
Social status	-0.11 *	-0.09 *	0.17	-	-	-	-0.13	-0.04 *	0.13
Social statusˆ2	0.00 *	0.01 *	-0.01 *	-	-	-	0.01 *	0.00 *	-0.00 *
Social interest	0.51	-0.6	0.34	-	-	-	0.27	-0.36	0.26
Social interestˆ2	-0.05	0.08	-0.07	-	-	-	-0.02 *	0.04	-0.05
Account age	0.34	-0.17	-0.13	-	-	-	0.37	-0.2	-0.13
User has URL	-0.32	0.22	0.07	-	-	-	-0.22	0.1	0.1
User verified	-0.11 *	0.26	-0.21	-	-	-	-0.15	0.29	-0.21
# Mentions	-	-	-	0.3	-0.07 *	-0.22	0.13	0.08 *	-0.23
# Hashtags	-	-	-	0.73	-0.22	-0.47	0.78	-0.24	-0.5
# URLs	-	-	-	-4.09	3.35	-0.73	-3.94	3.19	-0.67
Is reply	-	-	-	0.05 *	0.05	-0.1	0.03 *	0.06	-0.09
Is quote	-	-	-	1.17	-0.75	-0.05 *	1.1	-0.68	-0.05 *

Table 6.3: Evaluation scores of various models on the test split across all datasets. (Acc.=accuracy, P=precision, R=recall, F1=F1 score, KLD=KL divergence). Acc., P, R, F1 are measured as percentages and higher score means better. For KLD lower means better.

Dataset	Model	Acc.	P	R	F1	KLD
Airline	meta	63.9	61.1	36.8	32.8	0.663
	text	80.0	78.3	69.0	72.4	0.026
	joint	80.3	76.6	72.0	**74.0**	0.005
Clarin	meta	45.7	42.1	40.9	37.8	0.238
	text	64.1	64.5	62.2	62.9	0.012
	joint	64.1	64.0	63.0	**63.4**	0.000
GOP	meta	59.9	54.3	37.5	33.6	0.776
	text	66.4	63.7	51.4	53.6	0.111
	joint	65.6	59.9	56.5	**57.8**	0.006
Healthcare	meta	56.7	36.8	39.4	35.1	0.717
	text	64.2	71.3	49.5	51.0	0.233
	joint	65.6	61.6	58.3	**59.5**	0.007
Obama	meta	39.3	37.0	35.1	32.0	0.282
	text	61.5	64.8	59.7	60.9	0.030
	joint	62.3	63.2	61.6	**62.2**	0.002
SemEval	meta	47.0	31.0	36.2	33.0	0.845
	text	65.5	64.1	58.0	59.5	0.032
	joint	65.6	62.7	60.5	**61.4**	0.001

Part II

Improving text information extraction for DSTD construction

Chapter 7

Construction of hierarchical subject headings for computer science

7.1 Introduction

Domain-specific controlled vocabularies help to identify, classify, and disambiguate concepts in scholarly articles. An example for such a vocabulary is the Medical Subject Headings (MeSH) for the field of biomedicine. MeSH is particularly useful because of its hierarchical and non-cyclical nature, which allows efficient search for related terms. Furthermore, MeSH, along with an annotated biomedical corpus like MEDLINE, can be utilized to track the evolution of biomedical concepts over time and create concept profiles of authors [Mishra and Torvik 2016]. While the fields of mathematics and physics also have their own dedicated classification schemas, namely, Mathematics Subject Classification (MSC) and Physics Subject Headings (PhySH), respectively, there is no such comprehensive schema for CS. The most popular schema for CS is the ACM computing classification system (ACM CCS). However, ACM CCS is limited to identifying sub-fields of computing in a hierarchical fashion instead of identifying key terms in each sub-field. To address this gap, we construct a large-scale and hierarchical controlled vocabulary for CS. We utilize the Wikipedia Category Tree, which organizes categories in a hierarchical pattern, along with classifying pages with these categories. Furthermore, we utilize Wikipedia redirect links, which allows us to capture multiple name variants of the same concept, and is useful for disambiguating key phrases to concepts present in the controlled vocabulary.

7.2 Methods

We propose a heuristic algorithm to extract a sub-hierarchy from the complete Wikipedia category hierarchy. The heuristics aim to eliminate redundant terms related to non-computer science concepts, which is a major challenge with our approach. The following top-level categories are chosen as starting points because of their relevance for CS:

- Areas of computer science
- Mathematical concepts

- Linguistic research
- Mathematics
- Information science
- Computer engineering
- Computational statistics
- Statistical data
- Statistical methods
- Statistical software
- Statistical theory

Each category sub-tree is traversed to collect all its children that do not violate our rules for discarding spurious and non-relevant categories, e.g., sports teams and names of individual people. Traversing the category tree, 61,231 categories (with 94,062 edges and 27 levels starting from the top level) were extracted related to the parent categories identified above. Many of these categories are still spurious and non-relevant, but these false positives can be refined by using additional filtering processes based on terms identified in large-scale corpora of CS papers, e.g., DBLP. Finally, 1,629,964 pages were identified as being related to the final collection of extracted categories. Additionally, 1,039,718 redirects were also identified for each of the pages, resulting in a larger collection of name variants for the same concept. A comparison our our extracted vocabulary with existing vocabularies is shown in 7.1

Table 7.1: Comparison of our constructed vocabulary with existing vocabularies.

Name	Type	Size	Curation	Domain
MeSH	Fine grained	25K	National Library of Medicine	Biomedicine
PhySH	Fine grained	-	Americal Physical Society	Physics
Wikipedia	Fine grained	1M+	Wikipedia contributors	Open domain
MCS	Subject level	6.1K	Mathematical Reviews and Zentralblatt MATH	Mathematics
ACM CCS	Subject level	2.1K	Association of Computer Machinery	Computer Science
PACS	Subject level	9.1K	American Institute of Physics	Physics
LCSH	Subject level	342K	Library of Congress	Open domain
Our work	Fine grained	3.2M	Semi-automatic	Computer Science

7.3 Evaluation on CS papers from Korean authors

The resulting concept hierarchy is utilized to extract and map key phrases in a dataset of 60K CS articles of Korean authors published between 1950-2016. This dataset was curated by the Korean Institute of Science and Technology Information (KISTI), and includes manually disambiguated author names and complete

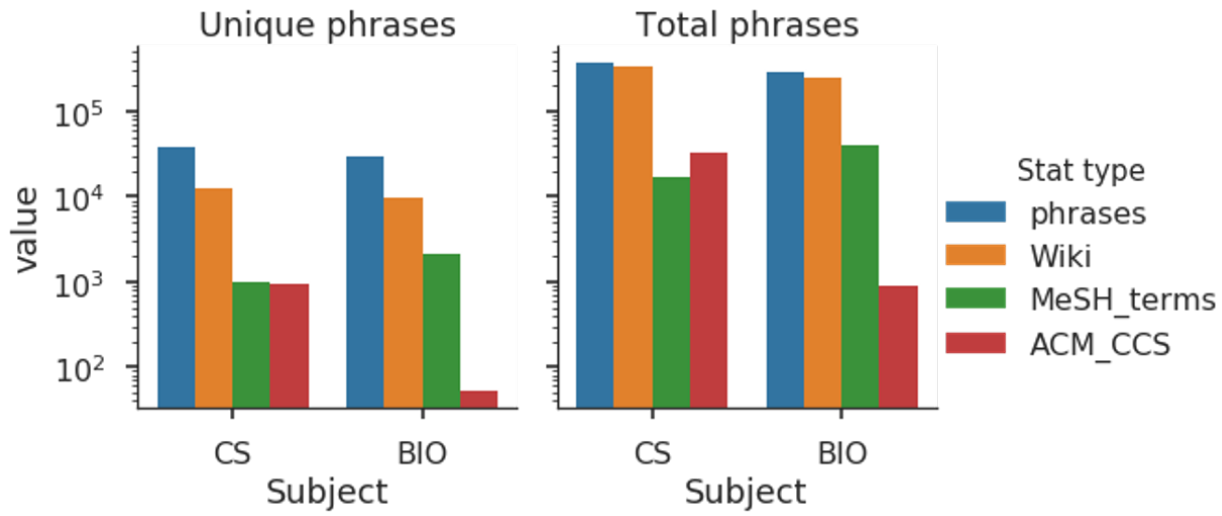


Figure 7.1: Distribution of mapped phrases in the KISTI CS and BIO corpora. Total phrases refers to all the identified phrases in the corpora. Unique phrases is only the unique phrases in the whole corpus.

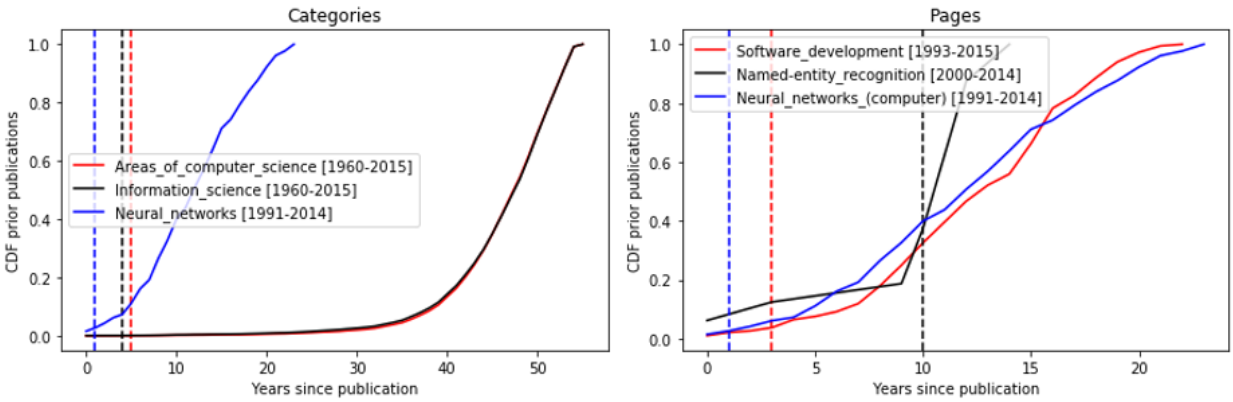


Figure 7.2: Growth of various concepts mapped to Wikipedia categories and pages in our hierarchy in the KISTI corpus.

author information. The AutoPhrase [Liu et al. 2015] algorithm is used to extract key-phrases from the titles and abstracts of all the papers in the KISTI data set. In total, 372K phrase instances were extracted resulting in 38K unique phrases. These extracted key phrases were used to identify the temporal profile of concepts in each dataset (examples are shown in figure 7.2). We utilize a manual disambiguation approach to map the key phrases to the concept hierarchy.

7.3.1 Comparison with other other controlled vocabularies

To evaluate the quality and coverage of our concept hierarchy, we compare the mapped key-phrases against the domain specific controlled vocabularies, i.e., ACM CCS and MeSH. We were able to map 32% of unique

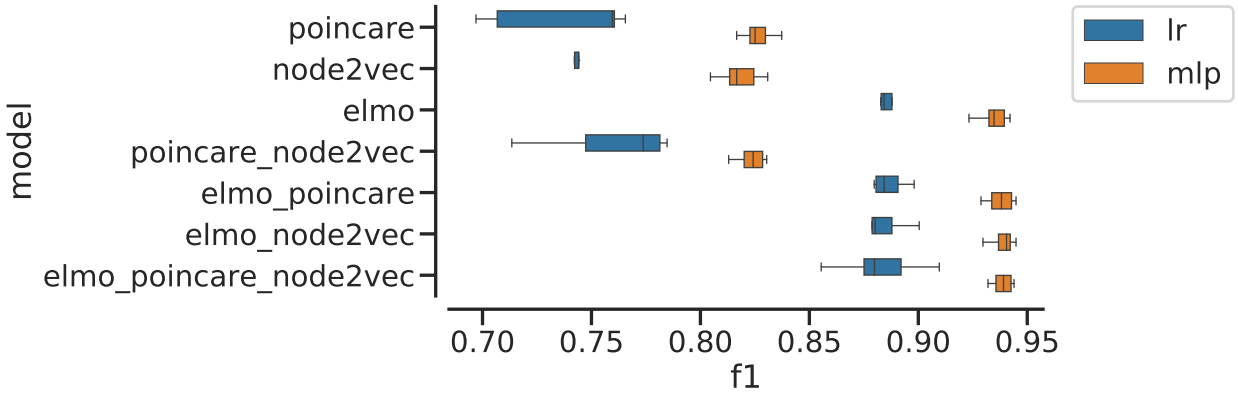


Figure 7.3: Cross validated scores for various models for identifying relevant categories for CS.

phrases and 90% of all identified phrases to Wikipedia. A more detailed breakdown of the distribution of mapped phrases is shown in figure 7.1. Most of our mappings are in our extracted concept hierarchy, followed by a higher proportion in ACM CCS. We could not identify concepts for 22% of the CS papers. This is likely due to the AutoPhrase algorithm not being able to find phrases in these papers, or the identified phrases being very noisy or not present in our concept hierarchy.

7.4 Embeddings for noise reduction in identified categories

One strategy to reduce noise in our extracted categories is to use a machine learning model trained on relevant versus noisy categories to filter out noisy categories. Embedding based approaches have been shown to perform well for node classification in graphs. In our case the Wikipedia category tree is a graph, and our aim is to predict if a category is relevant for computer science or not. We utilize three ways of embedding each category. The first is based on the category text. We utilize Elmo [Peters et al. 2018] embeddings of the category text to get a text only embedding. Next we utilize the graph structure of the category tree to generate node2vec embeddings [Grover and Leskovec 2016] for each category. The final embedding utilizes the tree like structure of our data. We utilize poincare embeddings [Nickel and Kiela 2017] for that. We construct a training dataset using accepted categories from the top 10 levels and rejected categories from the top 20 levels. We then train a logisitic regression model as well as a multi-layer perceptron model. Then we train different models using a single type of embedding and also combination of embeddings. The cross validated model performance is shown in 7.3. We find that Elmo based model perform best, and the addition of graph based features improves the performance slightly.

7.5 Conclusion

To conclude, we present a hierarchical controlled vocabulary for Computer Science that was constructed based on Wikipedia data, along with an algorithm for mapping paper concepts to these categories. We show that our constructed vocabulary of CS concepts has higher number of terms as well as higher coverage of identified phrases on a computer science publication data. We show how identified concepts in papers can be used to study the evolution of concept in computer science. We presented a method for reducing the noise in out taxonomy using embeddings of categories. Finally, these concept mappings can be utilized to identify the conceptual novelty of articles [Mishra and Torvik 2016], along with concept level expertise of authors [Mishra et al. 2018a].

7.5.1 Connection with DSTD

As described in chapter 1, DSTD construction requires accurate data about its components. In scholarly domains, computation of novelty (chapter 2) and expertise (chapter 3) require accurate identification of concepts in a paper. Using the hierarchical subject headings for CS described in this chapter, we can accurately identify the concepts related to each CS papers and use it to answer questions related to novelty and expertise for CS domain. Furthermore, the hierarchical CS subject headings can be combined with key-phrase extraction to address the major IE task of concept identification and linking in scholarly domains.

Chapter 8

Incremental training of text classifiers with human in the loop learning

Content in this chapter is based on our paper Mishra, S., Diesner, J., Byrne, J., and Surbeck, E. (2015). Sentiment Analysis with Incremental Human-in-the-Loop Learning and Lexical Resource Customization. In *Proceedings of the 26th ACM Conference on Hypertext & Social Media - HT '15*, pages 323–325, New York, New York, USA. ACM Press.

8.1 Introduction

In opinion extraction tasks, the emphasis is usually on classifying new data for opinion labels by learning a model using limited training data. However, these models when applied to social media data are likely to suffer from the issue of over-fitting to features in a given domain. This problem and some of its solutions have been previously discussed in the machine learning literature under the name of domain adaptation [Sarawagi 2008, Daumé III 2007]. The problem gets compounded for social media data as the vocabulary and language usage continuously evolve over time. Furthermore, the ways of expressing the same opinion also change with time. For example, the opinion label of the phrase "you are just like *subject*", will depend on the general opinion about *"subject"* when the phrase was expressed. Similarly, many new words are coined on social media [Eisenstein 2013, Gupta et al. 2010]. This poses a challenge for maintaining opinion classification systems. In [Mishra et al. 2015], we propose an approach to alleviate this issue by creating a system based on active human-in-the-loop learning which incrementally updates an existing classifier by requiring an annotator to provide few new examples from the new data. We achieve this by a) allowing for manually updating an existing sentiment lexicon, and b) asking the annotator to provide labels for a few instances in the new data based on our model uncertainty. We retrain our model using the newly acquired data.

8.2 Model

Our model is built with the following features in mind:

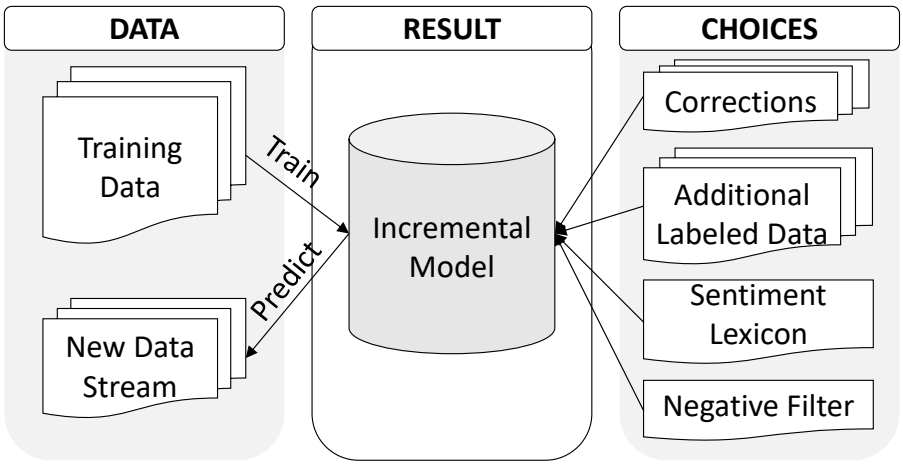


Figure 8.1: Model for training sentiment using human-in-the-loop incremental learning

- Low cost of continuous training data acquisition
- Incorporation of domain knowledge using lexicon
- Efficient update of model using only the newly acquired training data.

A description of our model is shown in 8.4

8.2.1 Data Pre-processing

Each tweet is tokenized and pre-processed by normalizing all mentions of hashtags, URLs, and mentions. We also use a large sentiment lexicon[1]. Furthermore, we suggest including a domain specific negative filter, i.e., words which should not be used to identify classification signals. For sentiment classification this can be entities in the copora.

8.3 Comparison of query selection strategies

A core requirement of active learning algorithms [Settles 2009] is to identify most informative instances from unlabeled data that can be used to construct a high quality ground truth dataset. In active learning literature, the act of identifying informative instances is called **query selection**. In order to select an instance for labeling we first need to rank the instances from the unlabeled data based on a score. We

[1]https://github.com/juliasilge/tidytext/blob/master/data-raw/sentiments.csv

consider two types of score:

- $entropy = \sum_i p_i * log(p_i)$ - higher is better
- $min - margin = \max_{i \neq \star}\{p_i - p_\star \mid p_\star = \max_j p_j\}$ - lower is better

The entropy based scoring favors model predictions with highest randomness. The min-margin based scoring is useful in ensuring that the difference between the top prediction score and the second top prediction score is less.

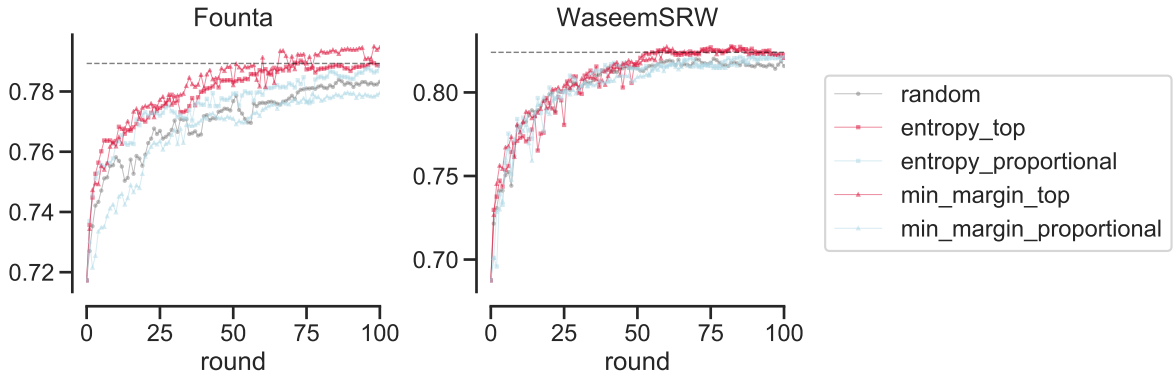
In order to simulate the human annotation process for the active learning algorithm, we bootstrap the model with 100 random samples from the training data. The rest of the training data is used as unlabeled data. In each round we select k (k=100) instances from unlabeled data based on the scoring criterion described above. The selection is done using three strategies:

- Instances are selected randomly without considering their scores. This acts as a baseline.
- Top k instances are selected based on the scoring criterion.
- k instances are sampled proportional to their scores. This adds a degree of randomness to the top k strategy.

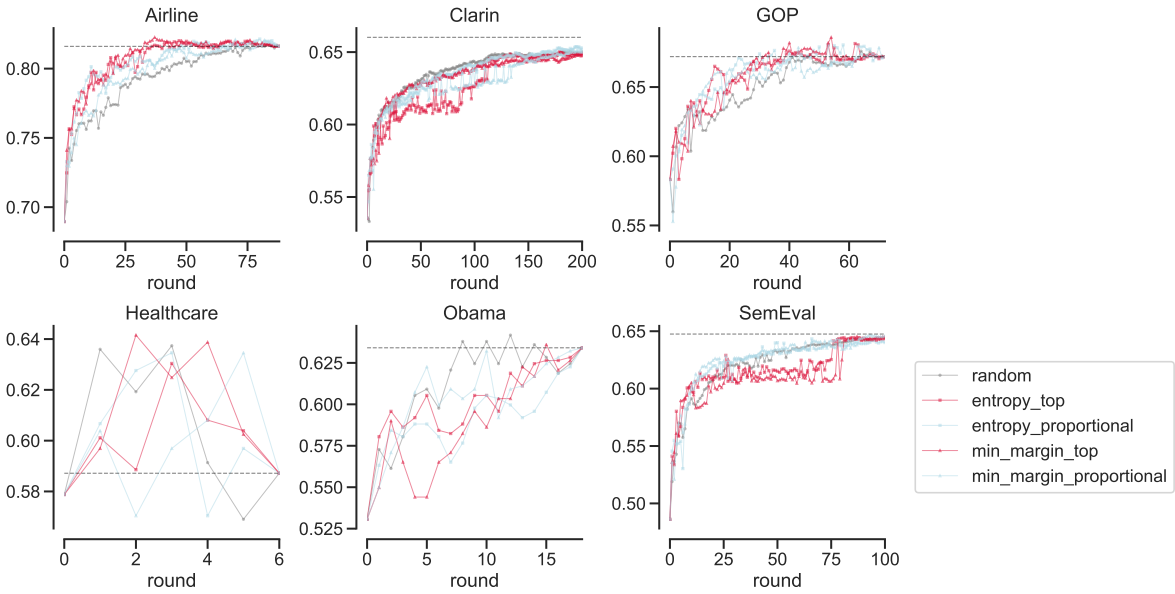
These new instances are then added to the selected instances, and the model is retrained. We use a logistic regression model with L_2 regularization. The regularization parameter is selected for each model using cross validation. We track the model scores on the held out test as well as validation data. For experiments we use the datasets for tweet classification described in detail in chapter 11. We conduct 100 rounds of active learning (200 for Clarin as it is a very large dataset) and evaluate the models using the micro-f1 score. We also compare the model evaluation scores against a model trained on the full data. The goal of this experiment is to understand how quickly active learning can close the evaluation score gap with the model trained on the full data.

The experimental results on the test split of each data are shown in figure 8.2. We observe that the top k strategy is usually the best followed by the proportional strategy across all data. For larger datasets we see that the model closes the gap very soon, likely due to the 10% training data used for bootstrapping the model.

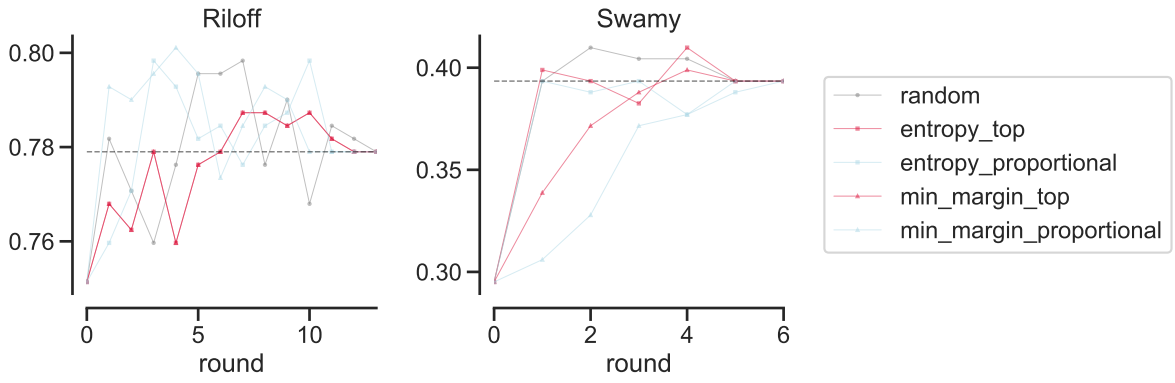
We also show experimental results on the unselected part of the training data in figure 8.3. We observe that the top k strategy is consistently the best, followed by the proportional strategy across all data. The increase in performance on the unselected is also indicative of the fact that active learning ensures that



(a) Abusive content detection



(b) Sentiment classification



(c) Uncertainty indicators

Figure 8.2: Progression of active learning classifier performance (micro f1-score) on the respective test set across 100 rounds of active learning (200 for Clarin). The annotation budget for each round is 100 instances, and the model is warm started with 100 random samples of the training data. Black dotted line is the classifier performance when trained on all of the training data. Data ordered alphabetically and X and Y axes are not shared.

the remaining data is actually easy to annotate without human correction. This evaluation presents a more practical usage pattern of ML models. This usage pattern requires annotating pre-selected, large, and unlabeled dataset. In reality, once the dataset is selected, one is interested in reducing the amount of training data created to efficiently annotate the data. This is where, according to us, the active learning approach may shine the most, as evident from our experiments. If in the end, the user can achieve high labeling accuracy by annotating few samples, then the user's job is done. An important thing to note is that the unselected data reduces proportionally in each active learning round.

8.4 Incremental learning of models with human in the loop

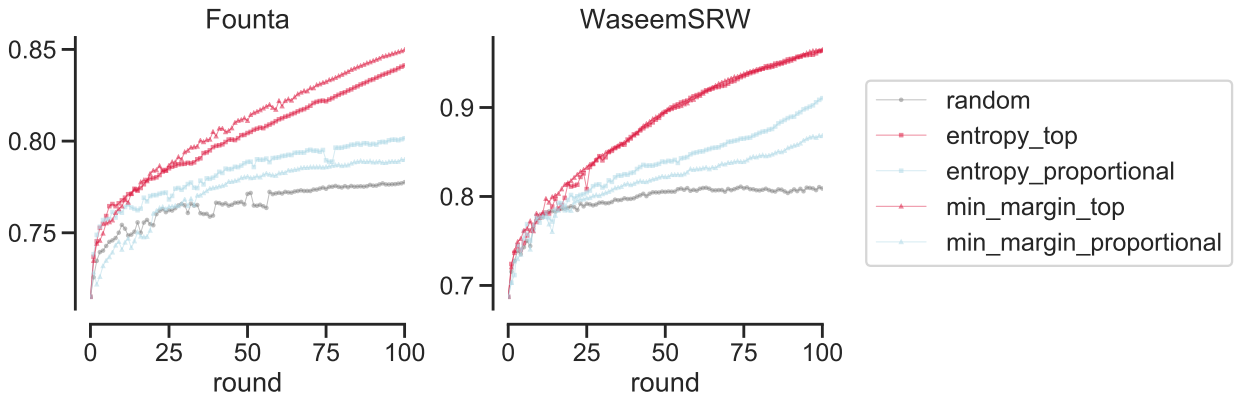
In this section we describe a tool called SAIL (Sentiment Analysis and Incremental Learning), that is focused on sentiment classification of SemEval data using the approach described above. The tool is aimed to ease efficient construction of training data using active learning, while supporting incremental learning of models using only the most recent data.

In order to perform incremental learning, we update our model with new batches of labeled data using stochastic gradient descent (SGD) [Bottou 1991; 2010].

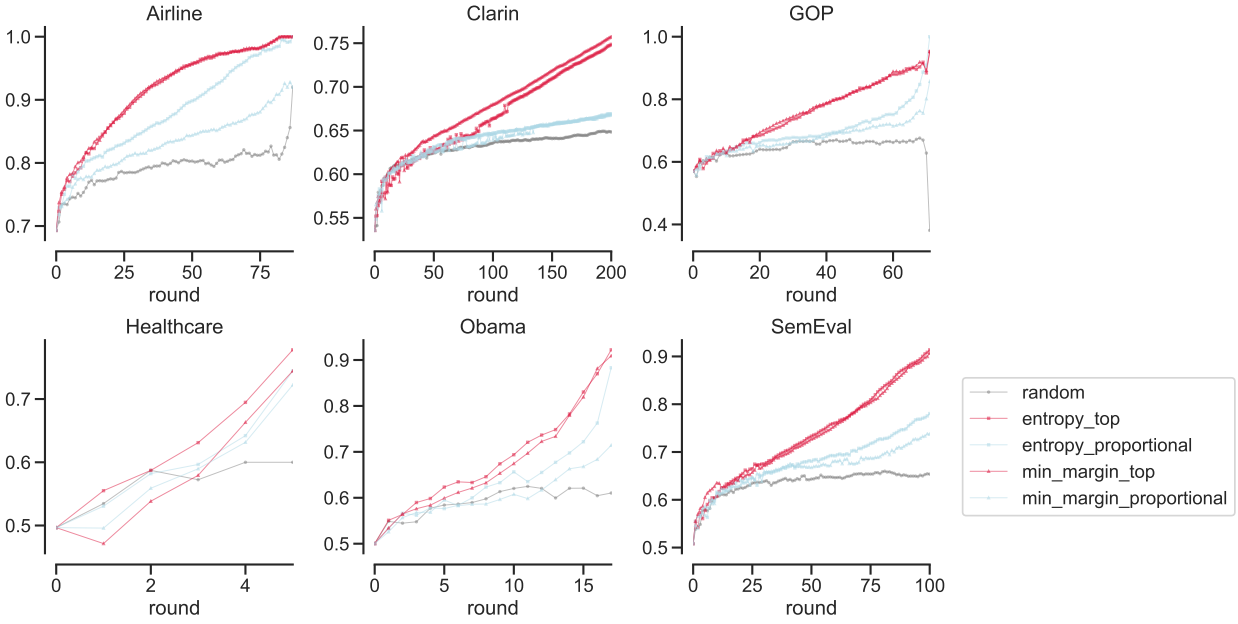
SAIL also considers an expanded set of features for each tweet, which are as follows:

- Count of hashtags, URLs, mentions, emoticons, and double quotes, and query terms,
- Count of POS tags extracted using the TweetNLP tool [Owoputi et al. 2013; 2012],
- Binary indicators for top 10 unigram and bigram words,
- Number of positive and negative words as identified in a sentiment lexicon (we use [Wilson et al. 2005]).

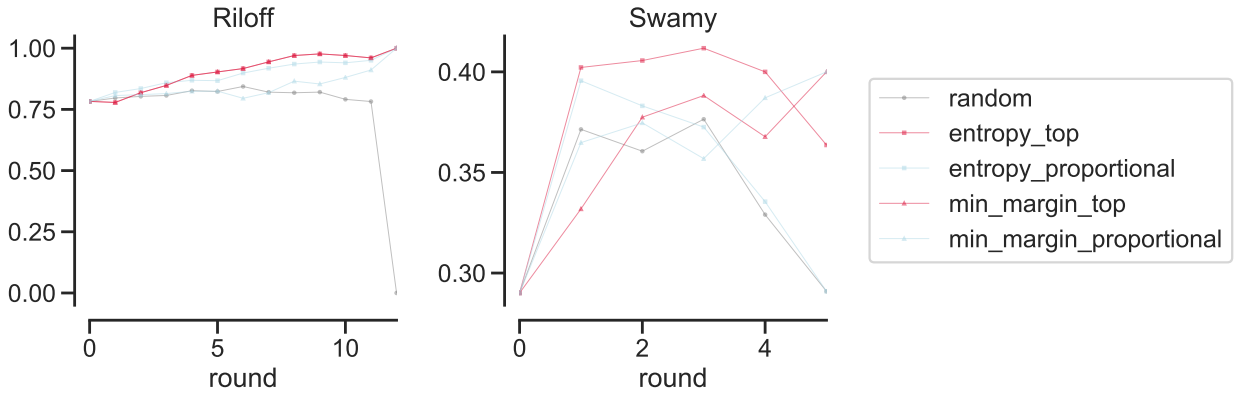
SAIL uses the pre-trained model to suggest top k instances to the annotator (see figure 8.4 (a)). The annotator can sort the instances using the scoring criterion. In order to reduce the cognitive work of labeling an instance from scratch, the annotator is shown the model predictions (as well as the label probability). The annotator is only required to edit the labels if they disagree. Model predictions for all the unlabeled instances from the top suggestions are now used as gold labels and fed to the model during the update process (this is similar to self-supervision with the possibility of human intervention). The annotator is also shown the prominent features for that instance. The annotator is only required to edit the labels, reducing the cognitive load per annotation as the model's label acts as a useful prior information. Once the model update has happened, the annotator is provided feedback on the change in model evaluation on a held out data (see figure 8.4 (b)).



(a) Abusive content detection

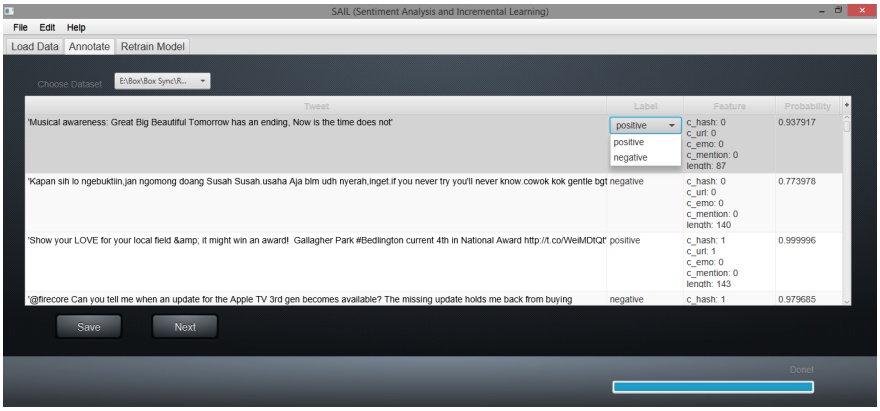


(b) Sentiment classification

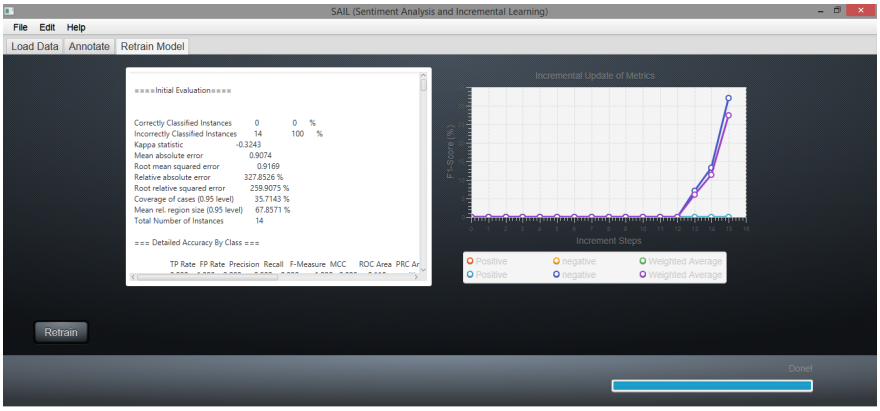


(c) Uncertainty indicators

Figure 8.3: Progression of active learning classifier performance (micro f1-score) on the respective unselected data set across 100 rounds of active learning (200 for Clarin). The annotation budget for each round is 100 instances, and the model is warm started with 100 random samples of the training data. Data ordered alphabetically and X and Y axes are not shared.



(a) Human editing of predictions



(b) Model loading

Figure 8.4: Human in the loop application interface

As an additional input the annotator can also update the sentiment lexicon as well as the negative filter lexicon before updating the model.

We use the SemEval dataset [Nakov et al. 2016b] partitioned into years. The comparison shows that SVM (as implemented in Weka [9]) is only outperformed by SGD (by about 0.9%) when using a large amount of tokens for the word feature (Table 8.1).

Table 8.1: Prediction accuracy depending on training algorithm and feature sets

Features considered			**Accuracy (F1)**	
Meta	**POS**	**Word**	**SVM**	**SGD**
X	X		70.50%	70.40%
X	X	X (N=2K)	85.70%	85.60%
X	X	X (N=20K)	86.60%	87.50%

8.5 Conclusion

In this chapter we described our experiments for evaluating active learning approaches for text classification tasks on tweet data. We further described a user interface for incremental learning of models by only requiring the annotator to update the labels for the model prediction if required.

SAIL is publicly available as an open source tool at https://github.com/uiuc-ischool-scanr/SAIL. SAIL is distributed with a model pre-trained on SemEval data [Nakov et al. 2016a]. It also allows users to train their model from scratch using new training data. A newer version of SAIL with different query selection strategies described above is now part of SocialMediaIE toolkit and available at https://github.com/socialmediaie/SocialMediaIE.

8.5.1 Connection with DSTD

As described in chapter 1, a DSTD is a temporally evolving dataset. Hence, taking a random sample from the full dataset to train a representative and generalizable model is difficult. In this chapter we have presented an approach based on human-in-the-loop learning to facilitate the rapid and accurate training of models in an online fashion using limited annotations. Our experiments on social media text classification tasks, suggest that this approach for mixing training and data annotation is computationally effective. Finally, other IE tasks on DTSD based on text features can be efficiently facilitated using this technique.

Chapter 9

Semi-supervised entity recognition

Content in this chapter is based on our paper Mishra, S. and Diesner, J. (2016). Semi-supervised Named Entity Recognition in noisy-text. In *Proceedings of the 2nd Workshop on Noisy User-generated Text (WNUT)*, pages 203–212, Osaka, Japan. The COLING 2016 Organizing Committee.

9.1 Introduction

A common task in IE is the identification of named entities from free text, also referred to as Named Entity Recognition (NER) [Sarawagi 2008]. In the machine learning and data mining literature, NER is typically formulated as a sequence prediction problem, where for a given sequence of tokens, an algorithm or model need to predict the correct sequence of labels. Additionally, most of the NER systems are designed or trained based on monolingual newswire corpora, which are written with proper linguistic syntax. However, noisy and user generated text data, which are common on social media, pose several challenges for generic NER systems, such as shorter and multilingual texts, ever evolving word forms and vocabulary, improper grammar, and shortened or incorrectly spelled words. Let us consider a fictional tweet: "r u guyz goin to c da #coldplay show madisonsqrgrdn?". This tweet contains two named entities, namely: "Coldplay", a music band, and "Madison Square Garden, NYC, USA", a geolocation, which references the place at which the band is playing. Many of the terms present in the exemplary tweet would be considered as out of vocabulary (OOV) terms by traditional NER systems. Furthermore, using a large set of such OOV tokens for training a classifier is likely to result in a sparse and high dimensional feature space, thereby increasing computing time. The phenomenon of concept-drift, i.e., the meaning of terms shifting over time, has also been found to affect the accuracy of NER systems over time, resulting in poor performance of a classifier trained on older data [Derczynski et al. 2015, Fromreide et al. 2014, Cherry and Guo 2015, Masud et al. 2010, Hulten et al. 2001].

The Workshop on Noisy User-generated Text (WNUT) continued its 2015 shared task on NER on tweets [Baldwin et al. 2015] in 2016. In 2016, the task was divided into two parts: (1) identification of named

entities in tweets, and (2) NER on 10 types of entities, namely person, geo-location, other, company, sports-team, facility, product, music-artist, movie, and tv-show. In this chapter we introduce two solutions to perform NER on tweets. The first system, which we will refer to as the submitted solution [ST], which was submitted as an entry to the WNUT 2016 NER shared task. It uses random feature [RF] dropout for up-sampling the dataset. This system was improved into a semi-supervised solution (our 2nd solution [SI]), which uses additional, unsupervised features. These features were found to be useful in prior information extraction and NER tasks. The semi-supervised approach circumvents the need to include word n-gram features from any tweets, and builds upon the successful usage of word representations [Collbert et al. 2011], and word clusters [Lin and Wu 2009, Ratinov and Roth 2009, Miller et al. 2004, Turian et al. 2010] for NER by utilizing large amounts of unlabeled data or models pre-trained on a large vocabulary. The SI system was designed to mitigate the various issues mentioned above, and utilizes the unlabeled tokens from the all the available datasets (including unlabeled test data) to improve the prediction quality on the evaluation datasets, a form of transductive learning [Joachims 2003]. The SI system outperforms ST by 7% (F1 score) when using the development set for evaluation, and by 11% when using the test set (1% higher than the 2nd best team in the task). The SI model does not utilize any word n-gram lexical features. We believe that the approach taken for SI is useful for situations that require refinement or adaptation of an existing classifier to perform well on a new test set. We have released our experimental setup and code at https://github.com/napsternxg/TwitterNER.

9.2 Data

The training, development, and test dataset were provided by the task organizers. The training set consists of 2,394 tweets with a total of 1,499 named entities. The organizers provided two separate development datasets, which we merged to create a dataset of 1,420 tweets with 937 named entities. This merged dataset was used as the development dataset for all of our experiments. The test dataset comprises 3,856 tweets with 3,473 named entities. Most of the tweets in the provided data lack any entities mentions (42% in training, 59% in development, and 47% in test data), resulting in sparse training samples. Furthermore, certain types of entities, such as movies and tvshows have only a few instances. The frequency distribution of the different types of named entities in the training, development, and test data are shown in Figure 9.1. Additionally, we found that the training, development, and test data have an average of 19.4 (7.6), 16.2 (6.8), and 16.1 (6.6) tokens per sequence, respectively, and mostly contain less than 3 entities per tweet. This implies that the presence of certain entity types might be reflective of the category of the tweet, e.g. movie entities

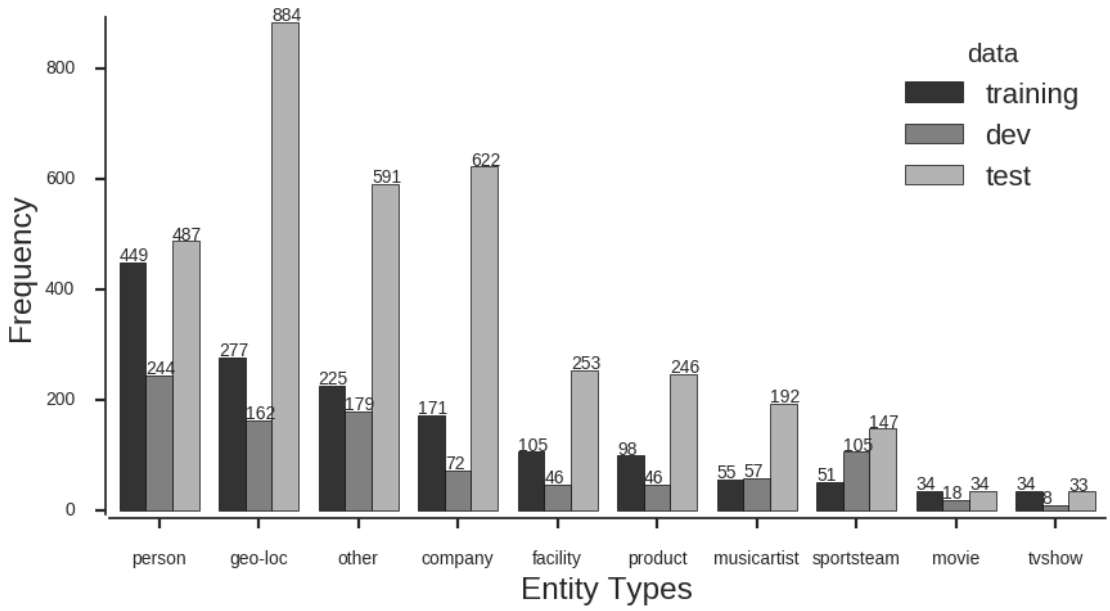


Figure 9.1: Frequency of named entity types in training, development, and test datasets

will occur in tweets about movies, and sports-team entities will occur in tweets about sports. Additionally, some types of entities are more likely to co-occur with each other than others. Using the provided data, we found that both person and geo-location entities were most likely to co-occur with entities of other 8 types, compared to the co-occurrence of the rest of the entities. Although the original dataset was tagged using the Begin-Inside-Outside (BIO) encoding, we converted that into the Begin-Inside-End-Outside-Unigram (BIEOU) encoding, which has been found to be more efficient for sequence classification tasks [Ratinov and Roth 2009]. However, the predicted tags were converted back to the BIO encoding to make our submission compatible with the evaluation system. The dataset is further described in [Han et al. 2016].

9.2.1 Background

Semi-supervised learning [Zhu 2008] can be useful for many tasks where we have large amounts of unlabeled data as well as some labeled data. The motivation behind using semi-supervised learning is that both these datasets can be utilized efficiently to build a more generalizable classifier than simply using the labeled data. The key idea is to use the unlabeled data as some kind of guiding prior for model [Zhu 2008]. Semi-supervised learning can be classified into two categories: transductive and inductive [Zhu 2008]. In transductive setting, we have the test data as an unlabeled source and can use it to train the model. In inductive setting, we wish to infer the correct mapping between the data and the labels given some unlabeled data.

9.3 Feature Engineering

We trained our system using multiple combinations of features. Features were chosen with the intent to increase the generalizability and scalability of our classifier. Some of the considered features can be updated with the availability of new unlabelled data, while other features capture the general token patterns in tweets. All features are described in detail in the following subsections.

9.3.1 Regex Features [RF]

Regular expressions are rules describing regularities in data, and are typically empirically derived. For example, in regular English corpora, named entities usually being with capital letters. Although regex based approaches can be effective, they are likely to result in retrieving large amounts of false positives. Most NER systems use token level regex features [Baldwin et al. 2015, Ratinov and Roth 2009]. We extended these regex features by including features that detect syntax patterns of tokens commonly present in tweets. Our patterns return "true" if the regex pattern matches the token. A detailed list of our regex features is described below:

isHashtag Identifies if token is a hashtag

isMention Identifies if token is a user mention

isMoney Identifies if token represents monetary values

isNumber Identifies if token is a number

isDigits Identifies if token only consists of digits

isAllCapitalWord Identifies if token only consists of capital alphabets

isAllSmallCase Identifies if token only consists of small alphabets

isWord Identifies if token only consists of letters

isAlphaNumeric Identifies if token only consists of digits and letters

isSingleCapLetter Identifies if token only consists of single capital letter

isSpecialCharacter Identifies if token only consists of special characters such as: #;:-/<>'"()&

endsWithDot Identifies if token only consists of alphanumeric and ends with a '.', e.g. Dr

containsDashes Identifies if token only contains dashes

containsDigits Identifies if token only contains digits

singlePunctuation Identifies if token is only single punctuation

repeatedPunctuation Identifies if token only consists of repeated punctuations

singleDot Identifies if token only consists of a single dot

singleComma Identifies if token only consists of a single comma

fourDigits Identifies if token only consists of four digits

singleQuote Identifies if token only consists of a single quotation mark

These features were extracted per token, and every pair of the neighbouring tokens' regex features were multiplied to create pairwise features.

9.3.2 Gazetteer features [GZ]

The task organizers provided a set of gazetteer lists. Although being helpful, these lists include some irregularities, such as words composed of or containing non-ascii characters, garbled strings, and missing names of important named entities in many categories. Furthermore, the provided gazetteers did not include names of movies or music artists. We increased the given set of gazetteers by including an additional 41K person names, 63K music artist names, 8K TV show titles, 2K sports team names, and 110K movie titles from WikiData (https://www.wikidata.org), additional 8.3M locations from GeoNames (http://www.geonames.org/), and 4.5M music artist names and their 1.4M name variants from the Discogs' public data dump (http://data.discogs.com/). Improved gazetteer features were also used as features in 2015 version of the shared task [Derczynski et al. 2015]. The gazetteer features were implemented on a per token level, where we look up a gazetteer phrase in a range of window sizes W (min=1 and max=6) both left and right of the current token. Additionally, we encode the window size and identified gazetteer name. Finally, we include interaction terms computed as the product of all pairs of gazetteer features for each token.

9.3.3 Word representation [WR]

Distributed word representations have been shown to improve the accuracy of NER systems [Collbert et al. 2011, Turian et al. 2010]. We used 200 dimensional GloVe word representations [WRG] [Pennington et al. 2014], which were pre-trained on 6 billion tweets. Furthermore, we built a set of word clusters by performing an agglomerative clustering of word representations [WRFTC], and fine tuning them on the training plus development dataset by running the word2vec model [Mikolov et al. 2013b;a].

9.3.4 Word clusters [WC]

Word clusters are word groupings that get generated in an unsupervised fashion, and they have been successfully used as features for NER tasks [Ratinov and Roth 2009] (Lin & Wu, 2009; Miller et al., 2004; Turian et al., 2010). One algorithm for creating such sets is Brown clustering (Brown et al., 1992), which produces

a hierarchical cluster of words in the corpus while optimizing the likelihood of a language model based on a Hidden Markov Model (HMM). We used pre-trained 1000 brown clusters [WC_{BPT}] that were prepared by using a large corpus of tweets [Owoputi et al. 2013, Gimpel et al. 2011]. Additionally, we built another set of brown clusters [WC_{BD}] with a cluster size of 100 based on all of the available data by using the code provided by Liang (2005) [1]. Furthermore, we also used an implementation[2] of the algorithm proposed by [Clark 2003] to create 32 (default option) additional word clusters from our training plus development data based on the regex and sequential features of the words. We choose to call these Clark clusters [WC_{CC}]. Additionally, for each token, we also included all word cluster features for their immediate neighbours along with interactive terms; with the latter capturing the product of the token cluster with the neighbouring cluster.

9.3.5 Additional Features

Even though the strength of our system lies in its semi-supervised nature and its non-reliance on data specific features such as lexical tokens [LT], we still included lexical tokens for comparison. Additionally, we used certain global features [GF] for helping with the prediction. Global features capture the overall composition of the sequence. We constructed the GF using the average values of the word representations and the binary presence of cluster and dictionary features. Additionally, another feature was constructed, which approximates the probability of the sequence being of a certain type. This feature adds an additional context to the token level prediction task, e.g., a tweet about sports is more likely to mention a sports team, and similarly, a tweet about a company is more likely to mention a product and vice-versa. To use this global feature, we first trained a Logistic regression classifier to predict if a tweet is about any of each of the 10 types of entities. The predicted probability per type is used as a feature for each of the tokens in the sequence.

9.3.6 Random up-sampling with feature dropout [RS_{FD}]

Since the training dataset is comparatively small and its features are sparse, we created synthetic examples by dropping interaction and lexical features with probability p. These features were chosen for random dropout because our earlier experiments had shown that the classifier identifies large weights for these features. We further scaled the training data size by a factor of k. This technique is inspired by the success of the dropout technique [Srivastava et al. 2014], which serves as a regularization function for deep neural networks. However, our technique is slightly different in that we use dropout to create a larger number of

[1] https://github.com/percyliang/brown-cluster
[2] https://github.com/ninjin/clark_pos_induction

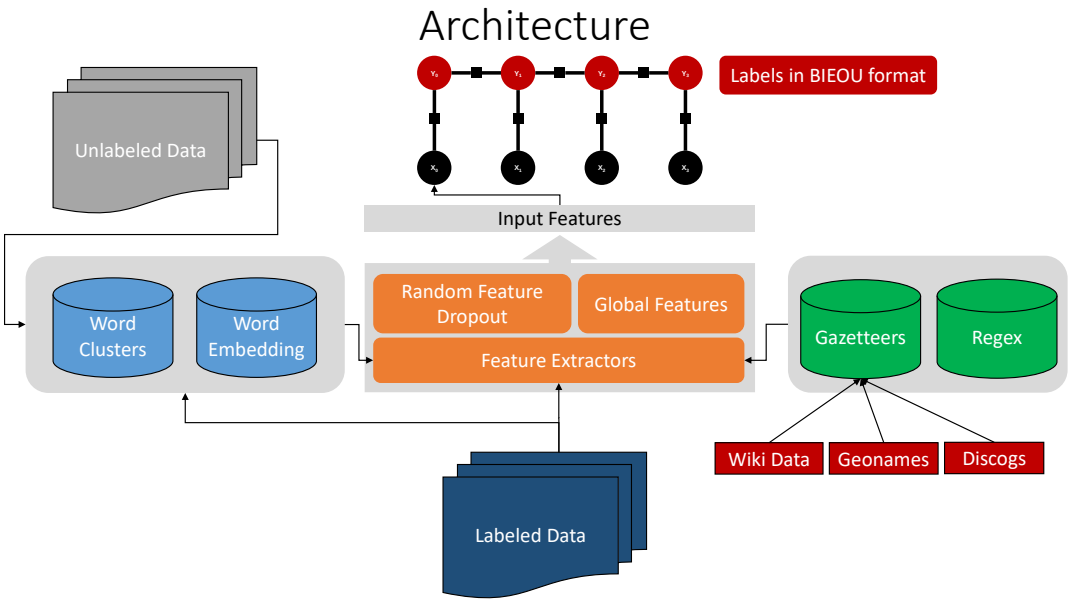


Figure 9.2: Model architecture

noisy samples from our data. Also, in contrast to the basic dropout technique, we did not re-weight the feature weights using the dropout probability [Srivastava et al. 2014] during evaluation.

9.4 NER classification algorithm

We used a linear chain CRF [Lafferty et al. 2001](McCallum & Li, 2003) as implemented in the CRFSuite [Okazaki 2007] package for training all our models. The models were trained using stochastic gradient decent (SGD) with an L2 norm ($C = 10^{-3}$). We also tested some of the recently popular deep learning based approaches, such as word embedding based and character based recurrent neural networks, for our prediction task. However, these techniques did not yield competitive results and were too slow to converge on CPU. Furthermore, training the CRF model was faster (average training time of the CRF algorithms was $\sim$ 3 mins on CPU, compared to $>$ 15 minutes for the character/word based 3-layer deep recurrent neural network solution), and gave interpretable results while beating the baseline model provided by the task organizers. In the following sections, we will first describe the model we used in our submission to the shared task, and then our improvement over the initial model and results. A schematic diagram of our model is shown in 9.2.

9.4.1 Shared task submission solution [ST] based on random feature dropout up-sampling

Our original submission to the shared task [ST] was based on a system that uses the lexical, regex, and dictionary based features with random feature dropout based up-sampling. All the interaction terms were randomly dropped out with p=0.5, and the scaling factor k was chosen to be 5. The dictionary based features were created using a context window of size 2 to the left and right of the token. Additional interaction features were included by calculating the product of the dictionary features of the token and the neighbouring tokens. Finally, ST was based on a classifier trained only on the training dataset, and was corpus specific in that it used the vocabulary created from the training data.

9.4.2 Semi-supervised word clusters and representation based solution [SI]

The described lexicon based solution [ST] had one major drawback: The most highly weighted features were mainly tokens descriptive of entity types that occurred in the training data. For example, the highest weighted feature for the label U-person was word_normed:pope. Similarly, for many of the other entity types, the highest weighted features were the names or labels of popular entities. Although these features help to achieve a decent evaluation score on the development dataset, they can lead to overfitting of the classifier to the vocabulary of the training corpus. In order to circumvent this issue, a semi-supervised [Blum 1998, Blum and Mitchell 1998] solution builds on the general recent success of using word representations and word clusters in NER tasks, while disregarding lexical vocabulary based features. The intuition behind our approach to the 2nd solution [SI] was to ensure that the classifier learns higher level representations of the observed tokens. All the features used for our second solution augment the tokens present in the given tweets. This allows us to scale-up the underlying resources, such as gazetteers, and improve word representations and clusters using the new unlabeled test data, while still being able to update the classifier from the initially provided, limited training data. We replicate this behavior in our classifiers by training our clusters on all of the unlabeled data generated by merging tweet texts from the training, development, and test data (only unlabeled) [TDT_E] (Blum & Mitchell, 1998), and comparing the resulting performance to that obtained with unsupervised training that does not consider the test data [TD]. Although it might appear that our classifier has access to the unlabeled test data sequences while learning, it rather is the case that we resemble an online setting where we continuously update our unsupervised features using the new batch of unlabeled test data, and then retrain our model on the original training data [Blum 1998, Blum and Mitchell 1998, Chapelle et al. 2006, Zhu and Goldberg 2009, Turian et al. 2010, Liang 2005, Carlson et al. 2010]. In this case, the unlabeled data prevent the classifier from over-fitting to the training data by acting

as a regularization factor. An alternative approach would be to train these clusters on a large number of unlabeled tweets that match the time range and search domain of the test tweets.

9.5 Results

In the following sections, we describe the evaluation of the accuracy of both the ST and SI system in comparison to BL and against each other. All evaluations were done by using the evaluation script provided by the organizers. We use the classifier provided by the organizers as the baseline (BL) system. The baseline system uses lexical, gazetteer, and regex features.

9.5.1 Performance in WNUT NER shared task

Using BL as a point of comparison, ST scored 1.1% (F1 score) higher for the 10-types task (based on the development set), and 1.2% (F1) lower for the no-types task. Our ST is based on random feature dropout based sampling. Among the 10 participating teams, our solution placed 7th for the 10-types category with an overall F1 score of 36.95%, and 6th in the no-type category with an overall F1 score of 51.38%. The top team on both tasks (same team in both cases) achieved F1 scores of 52.41% and 65.89%, respectively. Overall, we found that ST performed best on the geo-location type (F1 score of 64.72%), and behind the top two teams (score of 72.61% and 68.36%, respectively) for this category. We placed 3rd in terms of F1 (37%) in the facility category shown Table 9.1.

9.5.2 Improved model performance [SI]

In this section, we describe the evaluation of our improved system SI, which was developed after the release of the shared task results. Since we received the gold standard labels for the test-set late in the process, we evaluated most of the improved models based on the development set. We present the additive effect of a series of features to the model in Table 3. Additionally, that table also shows the performance of ST and BL. We do not include any lexical features in SI, however, lexical features were part of the ST and BL models. We found that the addition of the gazetteer [GZ] features improved the classification accuracy considerably. The next two big jumps accuracy improvements in SI came from using brown clusters [$\mathrm{WC_{BTP}}$] and fine-tuned word representations based clusters [$\mathrm{WC_{FTC}}$]. From all of the improved models that we trained, we selected the 10-types category model with the highest overall F1 score, namely $RF + GZ + WR_G + WC_{BPT} + WC_{CC} + WR_{FTC}$ model, also referred to as SI herein. Only the SI model was also evaluated on the test data with [$\mathrm{TDT_E}$] as well as without [TD], using the test data for enriching the

Table 9.1: Results of the WNUT NER 2016 shared task. Rank denotes the rank of the winning team, which we use as an ID to identify the evaluation performance of each of the participating teams in the shared task. Our solution was ranked 7th (in bold) and (6th not shown) in the 10-types and no-types categories, respectively. Columns with TD and TDT_E show the performance of the improved model on the test data, and their ranks denote the best rank in the competition which they beat.

Rank	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**TD**	**TDT_E**
10-types over-all	52.4	46.2	44.8	40.1	39.0	37.2	**37.0**	36.2	29.8	19.3	**46.4**	**47.3**
No-types	65.9	63.2	60.2	59.1	55.2	51.4	**47.8**	46.7	44.3	40.7	57.3	59.0
company	57.2	46.9	43.8	31.3	38.9	34.5	**25.8**	42.6	24.3	10.2	42.1	46.2
facility	42.4	31.6	36.1	36.5	20.3	30.4	**37.0**	40.5	26.3	26.1	37.5	34.8
geo-loc	72.6	68.4	63.3	61.1	61.1	57.0	**64.7**	60.9	47.4	37.0	70.1	71.0
movie	10.9	5.1	4.6	15.8	2.9	0.0	**4.0**	5.0	0.0	5.4	0.0	0.0
musicartist	9.5	8.5	7.0	17.4	5.7	37.2	**1.8**	0.0	2.8	0.0	7.6	5.8
other	31.7	27.1	29.2	26.3	21.1	22.5	**16.2**	13.0	22.6	8.4	31.7	32.4
person	59.0	51.8	52.8	48.8	52.0	42.6	**40.5**	52.3	34.1	20.6	51.3	52.2
product	20.1	11.5	18.3	3.8	10.0	7.3	**5.7**	15.4	6.3	0.8	10.0	9.3
sportsteam	52.4	34.2	38.5	18.5	34.6	15.9	**9.1**	19.7	11.0	0.0	31.3	32.0
tvshow	5.9	0.0	4.7	5.4	7.3	9.8	**4.8**	0.0	5.1	0.0	5.7	5.7
Rank	1	2	3	4	5	6	**7**	8	9	10	**∼2**	**∼2**

unsupervised features. Although the model with the global features [See +GF in Table9.3] is not the top one in terms of the F1 score, it achieved considerably high scores for the movie, and tvshow class, which have very few training instances. Similarly, the random dropout upsampling based solution showed improvements by 15% and 6% F1 score in terms of predicting named entities of the types movie and music-artist, respectively. Finally, these models were trained in almost half the time as the ST models.

9.5.3 Features learned by the model

We extracted the learned features from the top performing model on the 10-types category (the $RF+GZ+WR_G+WC_{BPT}+WC_{CC}+WR_{FTC}$ model). The features with the highest positive and negative weights for each of the category labels are presented in Table 9.2. The table also shows that for person, product, movie, and tvshow the top features were specific dimensions of the pre-trained word embedding. Furthermore, the brown cluster ids of the token word are more informative for the named entities of geo-location, other, and company types, while the brown cluster id of neighbouring tokens is likely to indicate if a named entity is of types musicartist, sportsteam, or facility. Additionally, if the token belongs to a phrase in a gazetteer of music artist names, then it is less likely to be a geo-loc, company, or product.

We also investigated the transition features of the linear chain CRF model. The transition matrix (based on transition weights) is presented in Figure 9.3, and colored as red for negative weights and black for positive weights. Some trends become obvious from the transition matrix: For most entity types, the model is able

Table 9.2: Feature weights (w) in the SI model for each of the 10 entity types. WV is word vector; BC is brown cluster. Superscript $^{\pm num}$ denotes features for left or right neighbour, respectively. → BIEOU denotes which boundary type for the entity type the feature belongs to.

Type	**Most positive weight**		**Most negative weight**	
Entity	**feature**	**w**	**feature**	**w**
person	$WV_{90} \rightarrow U$	1.27	$WV_{46} \rightarrow U$	-1.02
other	$BC_2 : 1001000 \rightarrow$ U	1.28	isAllSmallCase—isAlphaNumeric$^{+1} \rightarrow$ U	-0.88
geo-loc	$BC_0 : 11100110101 \rightarrow$ U	2.33	DICT=musicartist_names → U	-0.88
facility	$BC_2^{-1} : 1001111110 \rightarrow$ B	1.63	$WV_{185} \rightarrow B$	-0.68
company	$BC_0 : 111001100001 \rightarrow$ U	1.30	DICT=musicartist_namevars → U	-0.77
product	$WV_{199} \rightarrow U$	1.07	DICT=musicartist_namevars → U	-0.97
musicartist	$BC_2^{-1} : 11110010 \rightarrow$ U	1.21	DICT=geonames → U	-0.80
movie	$WV_{75} \rightarrow B$	0.76	isAlphaNumeric$^{+1} \rightarrow$ E	-0.50
sportsteam	$BC_0^{+1} : 1111011010 \rightarrow$ B	1.29	$WV_{30} \rightarrow U$	-0.86
tvshow	$WV_{154} \rightarrow U$	0.76	isInitCapitalWord—singlePunctuation$^{+1} \rightarrow$ E	-0.40

to find high transition weights for going from B to I to E, while penalizing transitions between the other states. The choice of using BIEOU tagging is supported by the results shown in the transition matrix since for most entity types, there is a high negative weight for going from the B or I tag to the O tag. However, a transition from the U tag to O tag is usually supported. Our earliest experiments (not reported here) revealed that there was a considerable improvement from using the BIEOU tagging scheme, this is in line with the findings of Ratinov and Roth [2009], and findings of others that argue for the usage of this tagging scheme for NER tasks.

9.5.4 Discussion and conclusion

Prior work has shown that semi-supervised algorithms can perform decently for NER tasks with sparse labelled data [Blum 1998, Blum and Mitchell 1998, Chapelle et al. 2006, Zhu and Goldberg 2009, Turian et al. 2010, Liang 2005, Carlson et al. 2010]. We leverage this fact in our SI model via the use of unsupervised word clusters, word representations, and refined gazetteers; all of which contributed to a cumulative increase in accuracy over our initial submission [ST] by 11% when using the test data for evaluation. Furthermore, the transition features learned by our model are reflective of correct learning of NER sequences and demonstrate the strength of using the BIEUO encoding scheme. Additionally, the supervised training of our classifier on features extracted from the unlabeled data, as opposed to lexical token features, reduces the dimensionality of the training data for the classifier and results in increased performance in terms of both accuracy and training time. Furthermore, our model can be adjusted on the arrival of new unlabelled data by updating the underlying learned word clusters and representations, and retraining the model on the existing labelled data. As identified by [Turian et al. 2010], the importance of word representations and word clusters increases as the

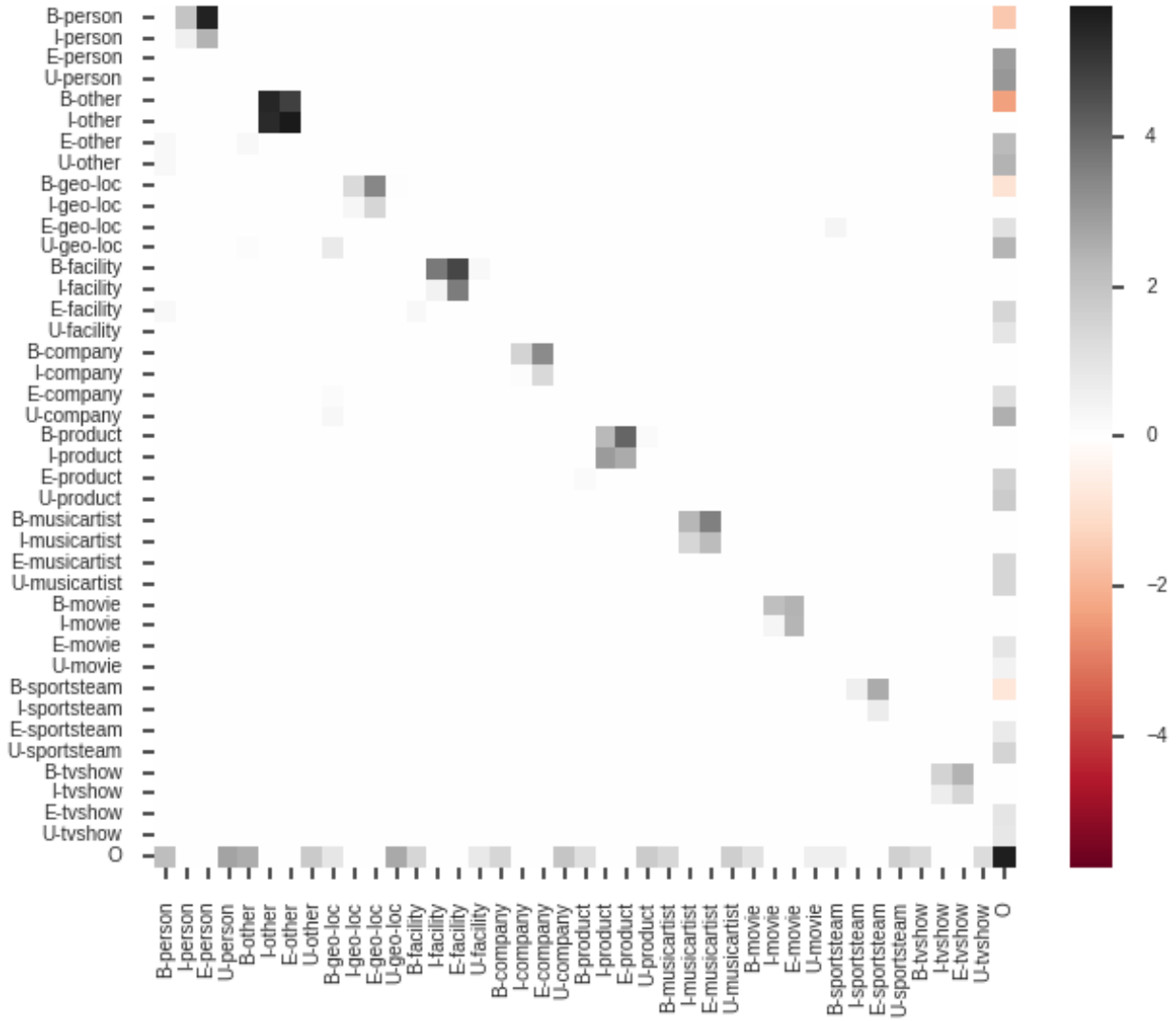


Figure 9.3: Transition weights learned by the SI model

availability of unlabelled data increases. We can add additional entity names to the gazetteers. Retraining the model on the same training data would then allow for accommodating to the new feature representations. Finally, the random feature dropout based up-sampling can help to increase the amount of training data available, and can also be improved by random swapping of entity types in the training data with their nearest neighbours in the word representations and clusters, or by choosing entities from the most correlated gazetteers. We believe that our described models can help in improving NER on noisy-text, and our open source implementation can be further extended.

9.5.5 Connection with DSTD

This chapter described an approach to improve named entity recognition using semi-supervised learning. This technique is very suitable for DSTD which are temporally evolving and allows efficient annotation of text components of DSTD similar to the human-in-the-loop learning techniques of chapter 8. The NER technique described here is also beneficial for identifying key concepts in scholarly corpora and can be combined with the approach described in chapter 7 to facilitate novelty (chapter 2) and expertise (chapter 3), computation.

Table 9.3: Change in F1 score for the NER classifier on the development dataset on incremental addition of different types of features (from left to right). ST refers to submitted solution, BL refers to baseline solution provided by the organizers. Bold values are the best scores across classifiers.

Features	RF	$^{+}$GZ	$^{+}\text{WR}_{\text{G}}$	$^{+}\text{WC}_{\text{BPT}}$	$^{+}\text{WC}_{\text{CC}}$	$^{+}\text{WR}_{\text{FTC}}$	$^{+}$GF	$^{+}\text{RS}_{\text{FD}}$	ST	BL	TD	TDT_{E}
10-types	5.3	34.8	36.7	41.6	41.0	43.3	40.9	40.0	36.2	35.1	46.4	47.3
company	0.0	30.0	34.5	33.3	35.2	33.3	32.0	33.3	27.7	26.2	42.1	46.2
facility	0.0	12.4	9.6	20.8	18.6	17.9	14.5	16.7	30.4	19.2	37.5	34.8
geo-loc	5.2	47.2	48.1	53.8	54.4	55.9	56.7	56.1	49.7	48.4	70.1	71.0
movie	8.0	7.4	6.5	8.3	7.7	9.5	23.5	28.6	8.3	0.0	0.0	0.0
musicartist	0.0	6.6	8.5	9.1	9.5	12.7	6.5	14.7	0.0	0.0	7.6	5.8
other	5.8	18.6	18.7	22.5	20.9	26.6	22.1	17.7	24.2	27.7	31.7	32.4
person	11.4	55.1	58.5	63.4	63.8	64.8	65.0	60.2	53.4	50.2	51.3	52.2
product	2.9	12.7	20.0	16.7	18.2	15.4	10.8	11.9	9.0	11.9	10.0	9.3
Sportsteam	0.0	12.9	27.9	30.5	29.0	28.1	27.7	25.4	12.8	13.1	31.3	32.0
tvshow	0.0	0.0	0.0	16.7	16.7	16.7	18.2	13.3	0.0	14.3	5.7	5.7
No-types	13.1	48.3	52.5	56.7	56.4	57.4	53.7	52.9	50.5	51.7	57.3	59.0

Chapter 10

Deep multi-dataset multi-task learning for sequence tagging

Content in this chapter is based on my paper Mishra, S. (2019a). Multi-dataset-multi-task Neural Sequence Tagging for Information Extraction from Tweets. In *Proceedings of the 30th ACM Conference on Hypertext and Social Media - HT '19*, pages 283–284, New York, New York, USA. ACM Press

10.1 Introduction

Many social media research publications rely on extracting structured information from social media text, which is usually achieved by applying existing tools or models. However, social media text is different from the newswire text. Social media text usually features (a) rapidly increasing vocabulary [Eisenstein 2013], (b) semantic and syntactic drift in language usage [Eisenstein 2013, Derczynski et al. 2013b], and (c) short text context [Eisenstein 2013, Derczynski et al. 2013a;b, Ritter et al. 2011]. Recent publications have demonstrated a major drop in accuracy of existing systems trained on newswire corpora when they are applied to social media text [Derczynski et al. 2013a, Ritter et al. 2011, Derczynski et al. 2013b; 2015]. The problem of building efficient models for IE from social media is worsened by the lack of large scale and consistently annotated corpora based on social media data. Existing datasets for social media data are often small scale, noisily annotated, and can differ across adopted annotation and preprocessing practices. This limits the ability of using traditional machine learning algorithms for training models for each task. Recent success of using deep neural networks to build end to end differentiable models [Collbert et al. 2011] has enabled the development of efficient multi-task learning algorithms that can utilize datasets from across multiple tasks and domains. Many of these models have resulted in the development of end-to-end architectures, which can be utilized for multiple tasks [Bingel and Søgaard 2017, Alonso and Plank 2017, Søgaard and Goldberg 2016], while achieving state-of-the art performance. A major reason cited for the success of these models is that they tackle the data-sparsity issue [Alonso and Plank 2017], and different tasks provide intrinsic regularization [Caruana 1993] for learning a robust shared representation of the data. However, application of these techniques for tackling social media sequence tagging tasks has not yet been

explored.

In this chapter, we study the effectiveness of deep **multi-dataset multi-task (MDMT)** learning models. We define multi-dataset learning as utilizing multiple datasets annotated for the same task using same or similar labeling schemes. Multi-task learning is the expansion of multi-dataset and single task learning to include multiple tasks. The proposed models are trained on four different tasks commonly used in social media research, namely, part-of-speech (PoS) tagging, phrase chunking, named entity recognition (NER), and supersense (CCG) tagging. For the purpose of our research, we pre-process multiple existing datasets into a common format, resulting in a meta-corpus for multi-task sequence tagging of tweets. Our work is focused on demonstrating the utility of multi-task learning compared to single task learning. Hence, we design our experiments to study these aspects of our models.

We achieve state of the art results on many of the included test datasets using the proposed MDMT models. Additionally, to ease the adoption of these models in future social media research, we make an open source implementation of our methods available for public use. We hope our work will help social media researchers who use Twitter data in utilizing state of the art models for processing their research data, leading to more robust findings.

10.2 Background

10.2.1 Multi task learning

Multi-task learning [Caruana 1993, Caruana and Niculescu-Mizil 2006] is an approach used to train models on multiple tasks with the assumptions that different task signals will guide the model to learn more generalizable internal representations of the data. The rise of automatic differentiation tools like Tensorflow, Pytorch, etc. has led to renewed interest in multi-task learning through gradient based methods. For a comprehensive review of multi-task learning using deep neural networks we refer the reader to [Ruder 2017]. [Collobert and Weston 2008] introduced one of the first large scale study of using neural network models with multi-task learning for NLP tasks. [Bingel and Søgaard 2017] report evaluation of multi-task models for a range of NLP tasks, and assess the utility of pairing tasks for multi-task learning. [Alonso and Plank 2017] also discuss multi-task learning for many sequence tagging tasks on general English corpora. Based on our knowledge, this is the first study to evaluate the effectiveness of multi-task learning by utilizing only Twitter corpora. A more recent study of [Changpinyo et al. 2018] provides an extensive evaluation of different architectures of multi-task learning focused on sequence tagging tasks, but it also focuses on general English corpora instead of Twitter corpora. Finally, [Søgaard and Goldberg 2016] also evaluate

Table 10.1: Description of POS datasets. Datasets that are clustered together are enclosed between horizontal lines.

data	split	labels	sequences	vocab	total
Owoputi	train	25	1547	6572	22326
	dev	23	327	2036	4823
	test	23	500	2754	7152
TwitIE	dev	43	269	1229	2998
	test	44	250	1182	2841
Ritter	train	45	632	3539	12196
	dev	38	71	695	1362
	test	42	84	735	1627
Tweetbankv2	dev	17	710	3271	11759
	train	17	1639	5632	24753
	test	17	1201	4699	19095
DiMSUM2016	train	17	4799	85	73826
	test	17	1000	100	16500
Foster	test	12	250	1068	2841
lowlands	test	12	1318	4805	19794

relative ordering of internal layers of a model for multi-task sequence prediction tasks on English data. Our multi-task learning architectures are inspired by the MTL-DEC architecture of [Changpinyo et al. 2018], and the model of [Søgaard and Goldberg 2016]. Our selection of tasks is inspired from those selected in [Søgaard and Goldberg 2016]. Multi task learning is a rapidly growing research field with numerous recent advances on studying task relatedness, reducing catastrophic forgetting [Kirkpatrick et al. 2017], and learning meta-models to generate task specific models [Finn et al. 2017]. This chapter is limited to demonstrating the effectiveness of multi-dataset and multi-task learning using no feature engineering, as opposed to developing better multi-task models or identifying better features for each task.

10.3 Tasks and Data

PoS tagging and NER are the most commonly studied sequence tagging task in the context of IE from tweets. For our experiment, we identified previously published datasets for PoS tagging, and NER. We also identified datasets for phrase chunking and Combinatory Categorical Grammar (CCG) supersense tagging from prior studies. This enabled us to assess our algorithm for four sequence tagging tasks. For PoS and NER, some of the datasets came in non-standard format, followed different tokenization, or were tagged using different annotation schemes. In order to reduce the possible number of annotation labels, we clustered datasets based on shared or overlapping label sets. For each dataset, its *train*, *dev*, and *test* splits were used for its respective purposes during our experiments. In order to train our model, we converted all datasets into

Table 10.2: Description of NER datasets. Datasets that are clustered together are enclosed between horizontal lines.

data	split	labels	sequences	vocab	tokens
YODIE	train	13	396	2554	7905
	test	13	397	2578	8032
Ritter	train	10	1900	7695	36936
	dev	10	240	1731	4612
	test	10	254	1776	4921
WNUT2016	train	10	2394	9068	46469
	test	10	3850	16012	61908
	dev	10	1000	5563	16261
WNUT2017	train	6	3394	12840	62730
	dev	6	1009	3538	15733
	test	6	1287	5759	23394
NEEL2016	train	7	2588	9731	51669
	dev	7	88	762	1647
	test	7	2663	9894	47488
Finin	train	3	10000	19663	172188
	test	3	5369	13027	97525
Hege	test	3	1545	4552	20664
BROAD	train	3	5605	19523	90060
	dev	3	933	5312	15169
	test	3	2802	11772	45159
MultiModal	train	4	4000	20221	64439
	dev	4	1000	6832	16178
	test	4	3257	17381	52822
MSM2013	train	4	2815	8514	51521
	test	4	1450	5701	29089

Table 10.3: Description of Chunking datasets

data	split	labels	sequences	vocab	tokens
Ritter	train	9	551	3158	10584
	dev	8	118	994	2317
	test	8	119	988	2310

Table 10.4: Description of CCG Supersense tagging datasets

data	split	labels	sequences	vocab	tokens
Ritter	train	40	551	3174	10652
	dev	37	118	1014	2242
	test	40	118	1011	2291
Johannsen2014	test	37	200	1249	3064

CoNLL style format, with only two columns, namely, token and label. The following sections describe how the datasets for each task were combined.

Throughout this chapter, a multi task setting will include using dataset from all the four tasks described below. Also, each label cluster will be referred to as a dataset unless noted otherwise. Hence, a multi-dataset setting will use all the datasets (i.e., clusters) from the same task. The baseline setting will be a single-task-single-dataset.

10.3.1 Part of Speech Tagging (PoS)

Table 10.1 describes the statistics of the PoS tagging datasets considered for our experiments. For our analysis, we clustered the datasets based on their label types. We found three common labeling schema, namely *ark* [1] (described in [Owoputi et al. 2012; 2013]), *ud* [2] (based on universal dependencies PoS tags), and *ptb* [3] (based on the common Penn Treebank PoS tags). The Owoputi dataset from [Owoputi et al. 2012; 2013] and follows the *ark* schema. The TwitIE [Derczynski et al. 2013b],and Ritter [Ritter et al. 2011] datasets follow the *ptb* schema. Finally, the Tweetbankv2 [Liu et al. 2018], DiMSUM2016 [Schneider and Smith 2015], Foster [Hovy et al. 2014a], and lowlands [Hovy et al. 2014b;a] follow the *ud* schema. The TwitIE dataset is the same as the Foster dataset, but with a different tokenization scheme and different tagging schema. The DiMSUM2016 and Tweetbankv2 dataset were in the CoNLL-U format and were converted to the specified format. For all datasets we also corrected spelling error in PoS tags, e.g., the VPP tag in the Ritter test data was converted to VBP, and CONJ tags in all ud datasets were converted to CCONJ after manual inspection of token values.

10.3.2 Named Entity Recognition

Table 10.2 describes the used for various NER datasets considered for our experiments. We first considered the more commonly used NER datasets for Tweets. This included Ritter [Ritter et al. 2011]; WNUT 2016 [Strauss et al. 2016] [4]; WNUT 2017 [Derczynski et al. 2017] [5], Finin [Finin et al. 2010], Hege [Fromreide et al. 2014] and Broad [Derczynski et al. 2016] [6]. The Broad corpora is split into six datasets. We create a training-development-test split of the Broad corpora by splitting each of the six datasets into training (60%), development (10%), and test (30%). We also used MultiModal dataset [Zhang et al. 2018], in which the

[1] !, #, $, &, ,, @, A, D, E, G, L, M, N, O, P, R, S, T, U, V, X, Y, Z, ^, ~
[2] ADJ, ADP, ADV, AUX, CCONJ, DET, INTJ, NOUN, NUM, PART, PRON, PROPN, PUNCT, SCONJ, SYM, VERB, X
[3] ", (,), ,, :, CC, CD, DT, EX, FW, HT, IN, JJ, JJR, JJS, LS, MD, NN, NNP, NNPS, NNS, O, POS, PRP, PRP$, PUNCT, RB, RBR, RBS, RP, RT, SYM, TO, UH, URL, USR, VB, VBD, VBG, VBN, VBP, VBZ, WDT, WP, WRB
[4] sportsteam, geo-loc, movie, person, tvshow, musicartist, other, product, facility, company
[5] creative-work, group, person, corporation, product, location
[6] per, loc, org

texts were annotated for entities based on images in the tweets. Datasets that shared the same labels sets were clustered together. All datasets used the *Begin-Inside-Outside (BIO)* tagging schema for identifying entity spans.

We also utilized Twitter datasets which have been earlier used for entity linking and named entity disambiguation tasks. Many of these datasets contain identified entities as well as their types. We extracted these entity spans and types, and converted each of these datasets into our format. The datasets contained charecter indexes denoting the start and end of an entity. The tweets were identified using their Twitter ID. We first collected the tweets using the Twitter API. Then for each tweet, we split the text between each span index. Finally, each span was tokenized using the Twitter specific tokenizer in NLTK [7]. This helped us to generate additional training data for our NER tasks. The entity linking datasets included YODIE [Gorrell et al. 2015] [8], MSM2013 [Cano et al. 2013], and NEEL2016 [Rizzo et al. 2016] [9]. MultiModal and MSM2013 share the same label schema[10].

Further analysis of NEEL2016 revealed that many of the entity spans identified in NEEL2016 do not represent a proper entity. For example, when the extracted entity should have been "*StarWars*", the entity span only covered "*arWars xo*". There were quite a few instances of this type, and hence could not be fixed manually. The pattern was present in all data splits of NEEL2016. Another issue in the NEEL2016 dataset, is the non-tagging of many entities, e.g., only few mentions of #Trump are tagged as *PERSON*, while many others are tagged as *O*. However, there were many correctly tagged entities. In favor of utilizing more training data, we did not exclude NEEL2016 from our analysis. In our results, we describe the impact of excluding NEEL2016 from our best models.

10.3.3 Chunk tagging

We found only one chunking dataset for Tweets. It was introduced in [Ritter et al. 2011]. The details of this dataset are shown in table 10.3. This dataset used the *Begin-Inside-Outside (BIO)* tagging schema for identifying chunk spans. The chunks were labeled with the *ADJ, CONJP, PRT, VP, ADVP, ADJP, INTJ, SBAR, NP, and PP* labelset. There does not exist a training-development-test split of the data. Hence, we prepared one by splitting the data into training (70%), development (15%), and test (15%).

[7] https://www.nltk.org/_modules/nltk/tokenize/casual.html
[8] unk, sportsteam, geo-loc, movie, product, person, tvshow, musicartist, other, organization, location, company, facility
[9] character, person, thing, event, product, location, organization
[10] loc, per, misc, org

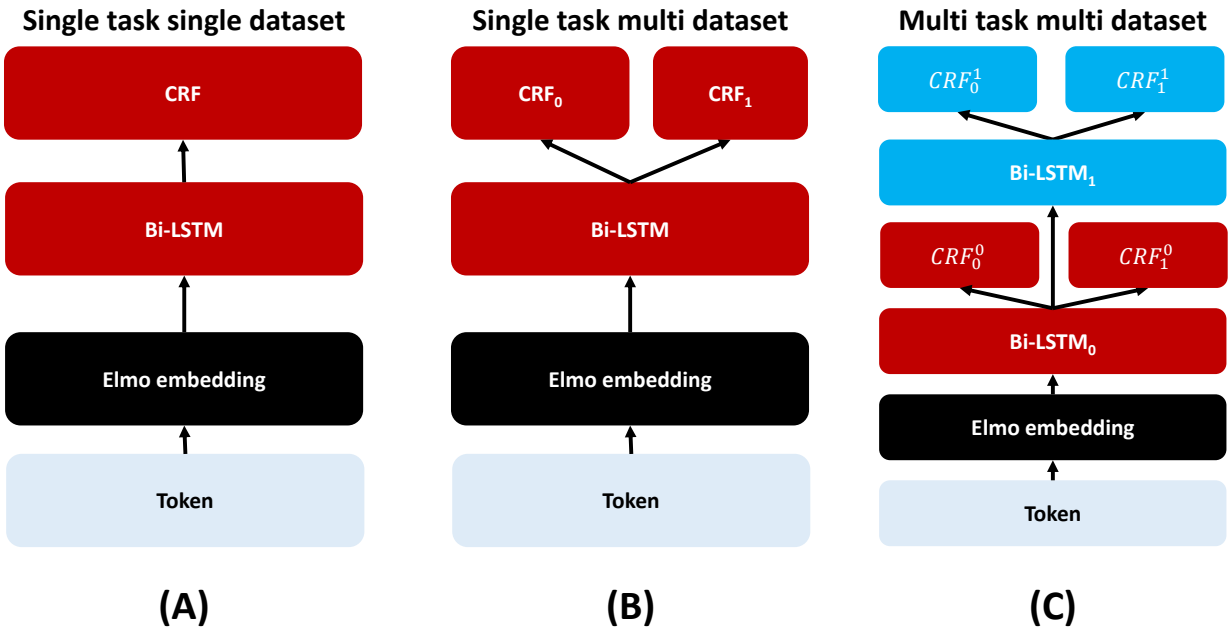


Figure 10.1: Possible model configurations. **(A)** Single task single dataset model. **(B)** Single task multi dataset model or multi task multi dataset model with shared internal layer. **(C)** Multi task multi dataset model with stacked internal layers (task specific layers coded with same color). Elmo weights are fixed during training.

10.3.4 Supersense tagging

We identified two datasets that used the same set of labels for CCG supersense tagging. The Ritter data was based on [Ritter et al. 2011] and was introduced in [Johannsen et al. 2014]. The [Johannsen et al. 2014] data was introduced in [Johannsen et al. 2014]. All datasets used the *Begin-Inside-Outside (BIO)* for schema for identifying supersense spans. The supersense span were labeled under two broad categories, namely, *NOUN*, and *VERB*. There were a total of forty sub-categories [11].

10.4 Methods

In this section we describe our MDMT model as well as the setup for comparing our model against baseline models. A schematic representation of the model is shown in 10.1.

[11] NOUN.BODY, NOUN.STATE, NOUN.ARTIFACT, NOUN.ATTRIBUTE, NOUN.FOOD, NOUN.TOPS, NOUN.COGNITION, NOUN.EVENT, NOUN.OBJECT, NOUN.MOTIVE, NOUN.SHAPE, NOUN.GROUP, VERB.COMMUNICATION, NOUN.PHENOMENON, VERB.POSSESSION, NOUN.FEELING, NOUN.POSSESSION, VERB.COMPETITION, VERB.SOCIAL, NOUN.ANIMAL, VERB.CREATION, VERB.CONSUMPTION, VERB.PERCEPTION, VERB.CONTACT, VERB.WEATHER, VERB.BODY, NOUN.LOCATION, NOUN.QUANTITY, NOUN.SUBSTANCE, NOUN.RELATION, NOUN.TIME, NOUN.PERSON, VERB.COGNITION, VERB.EMOTION, VERB.STATIVE, VERB.MOTION, NOUN.COMMUNICATION, NOUN.PROCESS, NOUN.ACT, VERB.CHANGE

10.4.1 Encoding tokens and labels

Traditional systems for sequence tagging represent each token in a sequence with a unique identified. However, this has several limitations, e.g., none (or common) representation for all out of vocabulary words, no similarity between the representations of related words. To overcome these issues, we decided to use pre-trained representations of the tokens in a sequence. While many options exist for pre-trained token representations, we decided to utilize the pretrained large-ELMO model [12] trained on 5.5 billion English language tokens. ELMO [Peters et al. 2017] uses a language modeling objective to train token representations for each token in a sentence. This ensures that the token representation is sensitive to other tokens as well as its position in the sentence. Furthermore, ELMO representation are based on the character representation of the token. These two aspects of ELMO representations are suitable for representing tokens in tweets, as tweet tokens are more likely to be out of vocabulary, as well as the meaning of these tokens vary based on the sentence structure. Furthermore, the focus of this chapter is on assessing the utility of MDMT models as opposed to evaluation of word-representations. We also do not utilize any word shape, or gazetteer based features in our experiments to limit our model search space, like these used in traditional sequence tagging worked.

10.4.2 Intermediate module

The word representations are then fed to an intermediate module. This intermediate module can be either a single neural network layer, or a collection of neural network layers. The intermediate layer also introduces dropout with probability 0.5. For most of our experiments, we utilize the bidirectional Long Short Term Memory (bilstm) layer. BiLSTM has been shown to perform accurately on many NLP tasks, and has been especially effective for sequence tagging tasks. We also conduct a few experiments using the recently introduced stacked self-attention layer as the intermediate module. The stacked self-attention (ssa) was introduced as part of the transformer architecture, and is faster to train compared to biLSTM, as all sequence elements are processed in parallel. However, the ssa layer consists of a stack of layers, which use self-attention to compute feature values for each sequence items. In our experiments, we a used ssa layer with 3 sub layers.

10.4.3 Output layer

Since each task is a sequence tagging task, we use a linear chain conditional random field (linear CRF) [Lafferty et al. 2001] loss function, to train our model. In particular, the linear CRF loss allows the model to

[12] https://allennlp.org/elmo

incorporate the correlation between neighboring labels apart from the correlation between the label vector and the token feature. We have a unique output layer for each dataset within each task. During inference, the correct sequence labels are identified using Viterbi decoding.

10.4.4 Single task single data model

The single dataset single task model serves as our baseline. This is equivalent to training the model without using data from other datasets or other tasks. This model is shown in figure 10.1 (A). We feed the ELMO embeddings to a bi-LSTM layer, and then feed the bi-LSTM output to the CRF loss.

10.4.5 Single-task or Multi-task (shared) multi-dataset model

In this setting, a single model is trained on all datasets for the same task. Each dataset has its own CRF loss function, but all datasets have their inputs passed through an intermediate layer (see figure 10.1 (B)). If models for multiple tasks are trained, they also share the same intermediate layer, but the CRF losses are different for each task-dataset pair. When the model is trained on a single dataset for a single task, we refer to it as single model (prefix **S**). When the model is trained on multiple datasets for the same task, we refer to it as multi-dataset model (prefix **MD**). Finally, when the model is trained on all the datasets for all the tasks, using the shared layer, we refer to it doctor as multi-task shared model (prefix **MTS**).

10.4.6 Multi-task (stacked) multi-dataset learning

This setting is an extention of the Multi-task (shared) multi-dataset model. In this case, all datasets within the same task share the same task specific intermediate layer. However, given an ordering of the tasks, the output of the previous intermediate layer is used as an input to the next intermediate layer (see figure 10.1 (C)). This model requires an ordering of the task. For our experiments, we use the task ordering as *[pos, chunk, supersense, ner]*. When the model is trained on all the datasets for all the tasks using the stacked layer, we refer to it as multi-task stacked model (prefix **MTL**).

10.4.7 Training routine and data sampling

The models were optimized using the Adam optimizer using a mini-batch of 16 instances. During each epoch, the dataset was shuffled before creating mini-batches. Each model was trained for a maximum of 10 epochs, terminating the training early if the validation loss stops improving after 3 epochs.

For the multi-dataset and multi-task setting, the model was trained using homogeneous mini-batches from the same dataset and same task. In order to ensure that all tasks and datasets are uniformly represented,

we sample from each task sequentially, and then generate a mini-batch for that task.

10.4.8 Implementation and hyper-parameters

The multi-data-multi-task models were implemented in SocialMediaIE [13] which is a deep learning library, focused on social media information extraction. SocialMediaIE is based on PyTorch deep learning library and uses the common language related deep learning modules from the AllenNLP library [Gardner et al. 2018]. For each intermediate layer, we used either a bidirectional long short term memory (biLSTM) module or a stacked self attention (ssa) module. The intermediate layer also uses a dropout rate of 0.5, has a fixed layer size equal to 100. We tried learning rates of Adam $\in \{0.01, 0.001\}$. L_2 regularization strength $\lambda \in \{0, 0.001\}$. The batch size was selected as 32. The models were trained on Google Colab platform with the GPU runtime. This machine had Intel(R) Xeon(R) CPU @ 2.30GHz with 13 GB of RAM, a Tesla T4 GPU with 16 GB memory. All the trained models have been uploaded for further analysis and reproducing our result, to the Illinois Data Bank[14].

10.5 Results

In this section, we use the following naming conventions for identifying the various models:

1. Models trained on single datasets have prefix *S*.
2. Models trained on all datasets of same task have prefix *MD*.
3. Models trained on all datasets have prefix. *MTS* for multitask models with shared module, and *MTL* for stacked modules.
4. Models with $lr = 0.001$ and no L_2 regularization have suffix *"*"*.
5. Models trained without NEEL2016 have suffix. *"#"*

Each model was evaluated across a variety of test data for each task. The relative rank (r), based on the evaluation metric (v) was identified for each model.

10.5.1 Part of speech tagging

The evaluation metric for PoS tagging is the accuracy of the labels. The results on PoS tagging are shown in table 10.5. We found that for PoS tagging, the MDMT models perform the best without any regularization. In presence of regularization, the single task models perform better, but the regularized single task models

[13] https://github.com/napsternxg/SocialMediaIE
[14] https://databank.illinois.edu/datasets/IDB-0934773

Table 10.5: Accuracy for part of speech tagging datasets (**r** = rank, **v** = accuracy).

file	DiMSUM2016		Foster		Owoputi		Ritter		Tweetbankv2		TwitIE		lowlands	
model	r	v	r	v	r	v	r	v	r	v	r	v	r	v
S bilstm	11	85.30	5	69.10	11	89.46	9	90.23	12	91.57	10	89.86	10	66.52
MD bilstm	12	85.22	13	68.11	12	89.37	12	88.57	10	92.05	12	88.70	11	66.35
MTS bilstm	13	85.08	9	68.67	13	86.81	13	79.10	14	89.46	13	79.66	14	65.27
MTL bilstm	14	84.71	14	67.97	14	86.56	14	77.81	13	89.50	14	76.66	13	65.40
S bilstm *	9	85.73	1	69.34	7	90.88	10	89.86	5	92.27	9	89.90	9	67.01
MD bilstm *	4	86.39	7	68.74	4	91.39	4	91.46	9	92.10	7	90.36	5	67.36
MTS bilstm *	5	86.35	4	69.17	5	91.36	5	91.40	11	91.57	1	91.62	4	67.65
MTL bilstm *	2	86.45	2	69.31	2	91.61	2	91.70	1	92.44	2	91.27	2	68.02
MTS bilstm * #	1	86.77	11	68.53	3	91.40	3	91.52	2	92.34	8	90.32	3	67.89
MTL bilstm * #	3	86.39	5	69.10	1	91.76	1	92.01	4	92.31	6	90.50	1	68.10
S ssa *	10	85.46	3	69.20	10	90.63	11	89.67	6	92.24	11	89.55	7	67.10
MD ssa *	8	85.83	12	68.36	6	90.93	7	90.96	2	92.34	5	90.67	6	67.13
MTS ssa *	6	86.02	8	68.71	9	90.72	6	91.15	7	92.23	3	90.85	8	67.07
MTS ssa * #	7	85.98	10	68.64	7	90.88	8	90.35	8	92.16	4	90.74	12	66.30

Table 10.6: Micro-F1 for named entity recognition datasets (r = rank, v = micro-f1).

file	BROAD		Finin		Hege		MSM2013		MultiModal		Ritter		WNUT2016		WNUT2017		YODIE	
model	r	v	r	v	r	v	r	v	r	v	r	v	r	v	r	v	r	v
S bilstm	13	73.15	14	52.53	13	87.16	15	77.55	12	71.98	13	65.84	9	49.58	14	40.38	11	58.56
MD bilstm	14	71.74	12	53.39	9	87.98	13	77.96	14	68.94	15	51.75	15	43.63	13	40.91	13	54.03
MTS bilstm	15	69.07	16	51.13	14	87.08	14	77.87	15	67.47	14	53.70	14	45.94	15	39.20	16	21.55
MTL bilstm	16	64.66	15	51.61	15	86.06	16	74.79	16	64.92	16	40.00	16	32.51	16	38.94	15	44.71
S bilstm *	3	76.76	10	54.20	5	88.59	1	80.46	4	72.83	12	71.94	12	47.41	12	42.53	14	52.60
MD bilstm *	6	76.35	4	55.28	2	89.08	3	79.99	1	73.39	6	82.07	1	53.16	6	45.88	7	62.83
MTS bilstm *	2	76.85	6	55.22	4	88.69	10	78.53	3	72.98	4	83.40	8	50.22	1	49.86	10	60.31
MTL bilstm *	4	76.59	1	56.42	6	88.44	5	79.74	13	71.96	2	85.06	11	47.46	7	45.45	6	62.88
MD bilstm * #	1	77.40	3	55.40	1	89.45	2	80.32	2	73.38	3	84.85	2	50.73	2	48.35	1	65.39
MTS bilstm * #	5	76.40	5	55.24	3	89.07	12	78.05	5	72.81	5	82.89	6	50.43	5	46.56	9	61.74
MTL bilstm * #	7	75.76	2	56.18	12	87.42	7	79.46	10	72.08	1	86.04	10	48.37	11	42.87	5	63.01
S ssa *	9	75.11	13	52.58	8	88.22	6	79.55	9	72.09	7	80.15	13	46.40	10	43.36	12	55.43
MD ssa *	8	75.50	11	53.62	7	88.42	4	79.80	8	72.40	8	78.88	3	50.68	3	48.12	3	63.35
MTS ssa *	11	74.93	7	55.08	10	87.87	8	79.18	6	72.66	10	76.98	4	50.57	4	46.83	8	62.36
MD ssa * #	12	74.84	8	54.72	11	87.59	9	78.59	7	72.40	9	77.74	5	50.47	8	45.00	2	63.87
MTS ssa * #	10	74.96	9	54.53	16	83.64	11	78.43	11	72.07	11	76.23	7	50.37	9	43.60	4	63.08

Table 10.7: Micro-f1 for chunking datasets (r = rank, v = micro-f1).

file	Ritter	
model	r	v
S bilstm	5	85.32
MTS bilstm	10	78.51
MTL bilstm	11	72.73
S bilstm *	7	83.54
MTS bilstm *	4	87.17
MTL bilstm *	2	87.85
MTS bilstm * #	3	87.68
MTL bilstm * #	1	88.92
S ssa *	9	80.70
MTS ssa *	6	85.09
MTS ssa * #	8	82.55

Table 10.8: Micro-f1 for supersense tagging datasets (r = rank, v = micro-f1).

file	Johannsen2014		Ritter	
model	r	v	r	v
S bilstm	6	34.33	4	56.98
MTS bilstm	10	20.19	10	33.44
MTL bilstm	11	19.96	11	26.34
S bilstm *	8	32.23	7	52.13
MTS bilstm *	4	36.38	5	54.20
MTL bilstm *	7	32.39	9	48.20
MTS bilstm * #	3	37.38	6	53.77
MTL bilstm * #	9	31.57	8	48.80
S ssa *	5	36.28	3	58.17
MTS ssa *	2	40.56	2	58.36
MTS ssa * #	1	42.38	1	59.16

Table 10.9: Model performance on NEEL2016 dataset.

model	precision	recall	f1
S bilstm	3.09	20.32	5.36
MD bilstm	4.30	17.12	6.88
MTS bilstm	3.02	3.77	3.35
MTL bilstm	3.00	6.39	4.08
S bilstm *	4.32	24.43	7.35
MD bilstm *	4.06	21.58	6.84
MTS bilstm *	3.26	22.26	5.69
MTL bilstm *	2.50	18.15	4.39
MTL bilstm *	3.47	19.86	5.91
S ssa *	4.80	27.51	8.17
MD ssa *	3.84	20.21	6.45
MTS ssa *	3.94	21.69	6.67

are not performing better than an un-regularized multi-task model. We also found that the biLSTM based models consistently out performed the stacked self-attention models. The stacked multitask models are the best across most test data. For 4 out of the 7 test data, the models without the NEEL2016 dataset performed better, compared the ones with NEEL2016. The accuracy on the Foster and lowlands test sets is lower ($\tilde{6}$8-69%) compared to the other datasets. Next, we compare our best model against model's in published literature. For the Ritter dataset, we achieve 92.01% accuracy, which is better than 90.0% as reported in [Owoputi et al. 2013]. For Owoputi, we achieve 91.76% compared to 88.89% as reported in [Owoputi et al. 2013]. For Foster and lowlands dataset, our model is off by a large margin (69.3% for Foster, and 68.1%$for lowlands$) compared to the best reported results in [Wulff and Søgaard 2015], which state 83.2% for lowlands, and 90.4%. It is important to note that for Foster and lowlands, we do not have an equivalent in-domain training dataset. Although, already published models also suffer from this limitation, our training proccess with multi-task learning is more susceptible to lack of in-domain training data, as learned models are more fine-tuned towards existing data. The model in [Wulff and Søgaard 2015] uses Brown clusters and is trained on the a larger newswire corpora based part of the OntoNotes corpus, which may explain the reason for the discrepancy of our models, which are only trained on Twitter data. For TwitIE dataset, we compare our model against the model in [Derczynski et al. 2013b], which achieves 89.37%, while our model achieves 91.62%. Since the TwitIE is the same as the Foster dataset, but with a different tokenization and mapping of PoS tags to a different scheme, we assume that the lower performance of our models on Foster because of this discrepancy as well. For DiMSum2016, the best tagging accuracy reported in [Schneider and Smith 2015] is 82.49%, comparatively our model achieves 86.77%. The best performance on Tweetbankv2 as reported in [Liu et al. 2018] is 93.3%, which is based on the model trained on the combination of Universal Dependencies English Web Text corpora (which includes 254,830 tokens) and the TweetBankv2 corpora. Our model achieves 92.44%, which is slightly lower, as our model only uses Twitter specific data.

10.5.2 Named entity recognition

The results for NER are shown in table 10.6. NER models are evaluated based on the correct prediction of each entity span and its type. Hence, the evaluation metric for NER is span based micro-F1 score across all label types. Similar to the PoS tagging results, we found that for NER the MDMT models perform the best without any regularization. In presence of regularization, the single task and multi-dataset models are better than multi-task models, but the regularized single task models are not better then an un-regularized multi-task/multi-dataset models. We also found that the biLSTM based models consistently out-performed

the stacked self-attention models. The multi-dataset models are the best across most test data, followed by multi-task stacked models. For 4 out of the 9 test datasets, the usage of NEEL2016 helps, while with the others, the usage of NEEL2016 is detrimental to model performance.

Next, we compare our best model to the best performing model for each test dataset. For WNUT2016, the best model as reported in [Strauss et al. 2016] achieved 52.41%. Our best model is 0.75% better. For WNUT2017, the best performing model as reported in [Derczynski et al. 2017] achieved 41.86%. Another model using multi task learning for NER achieved 45.55% [Aguilar et al. 2018], while another model reports 49.49% [Akbik et al. 2019][15]. Our best model achieved 49.86%, which is higher. For Ritter, MSM2013, and Finin the best model performance metrics are from [Augenstein et al. 2017] (Table 3). The best performance for Ritter is 52.14%, Finin (referred as UMBC) is 32.43%, and MSM2013 is 58.72%. Our best models are consistently better than each of the best performing models reported in the [Augenstein et al. 2017]. For Hege and Ritter, [Cherry and Guo 2015] report F1 scores of 86.9% and 82.6%, respectively. Our best model is again better in this case. For MultiModal dataset, the original paper reports a performance of 70.69%, our best model leads to 73.38%. For the remaining datasets we are the first to report the results.

10.5.3 Phrase chunk tagging

Similar to NER, the evaluation of phrase chunk tagging is done using span based micro-F1 scores, across label types. The results on phrase chunk tagging are shown in table 10.7. For the phrase chunking task as well, we have the multi-task stacked model, trained without the NEEL2016 data, which performs the best. There does not exist any evaluation of the chunking task for tweets. Furthermore, as we used our own training-validation-test split, we cannot compare the method against any existing implementation.

10.5.4 Super sense tagging

Similar to NER and chunk tagging, the supersense tagging model is also evaluated based on span based micro-f1 score, across label types. The results on super sense tagging are shown in table 10.8. For supersense tagging also, the multi-task model is better than the single/multi-dataset models. For this task, the stacked self-attention module consistently out performs the biLSTM based module compared to other models. It is important to note that the Ritter dataset is the only dataset shared across all tasks. Hence, the benefit of multi-task learning is more prominent on the Johannsen2014 dataset, which is only present in the supersense tagging data.

Next, we compare our best model with the best performing supervised domain adaptation models on

[15]https://github.com/zalandoresearch/flair

these datasets as reported in [Johannsen et al. 2014]. The best score for the Ritter dataset was 57.14%, while the one for Johannsen2014 was 42.42%. For Johannsen2014, our MTL model is comparable, albeit slightly less accurate by 0.04%.

10.5.5 Evaluation on NEEL2016 dataset and its impact

For chunking and supersense tagging, the models with the best evaluation scores were trained after excluding the NEEL2016 dataset.

When our models were evaluated on the test data of NEEL2016, the span based precision, recall, and micro-F1 scores were poor. The complete results are shown in table 10.9. Furthermore, the results on test data of the NEEl2016 dataset are quite poor compared to the results reported in the original task paper [Cano et al. 2013], where the best micro-F1 score on identifying the labels was reported to be 67%, while the least was 39.9%. The recall of our trained models is higher than the precision. This shows that the models identify high proportion of entities, but the labeling of these entities is not correct, or many of the identified entities are incorrectly labeled in the test data. Some reasons for this discrepancy are discussed in detail in the NER data section above. One potential solution might be to fix the issues in the test data, and then redo the training. We plan to conduct these experiments in the future.

10.5.6 Investigating label representations

In the last layer of the model, the score for each label is computed by taking a dot product of the input token representation with the vector for each of the labels. This results in a vector representation being learned for each label. We can investigate the space of these label representations to know the similarity of two labels. Since the label vectors are of 100 dimensions, we can perform a dimensionality reduction on the space of all labels and see a two dimensional representation of the labels. We utilize a dimensionality reduction algorithm called Uniform Manifold Approximation and Projection (UMAP) [McInnes et al. 2018b;a]. UMAP projects high dimensional data into a lower dimension while ensuring that relative distances between items in higher dimensions are also preserved in a lower dimensional space. We use the **MTL bilstm * #** model for our analysis as this model is one of the most consistent models in our previous results.

UMAP projection of PoS labels is shown in 10.2. We observe that labels which represent similar things across different datasets like nouns and verbs, are clustered around each other in their representation.

UMAP projection of NER labels is shown in 10.3. Again, we see a common clustering of labels referring to the same entity types across datasets, like PERSON, as well as similar entity types, like LOC, GEO-LOC, and FACILITY. Furthermore, there is a clear partitioning of B, I, and O tags, which are clustered together.

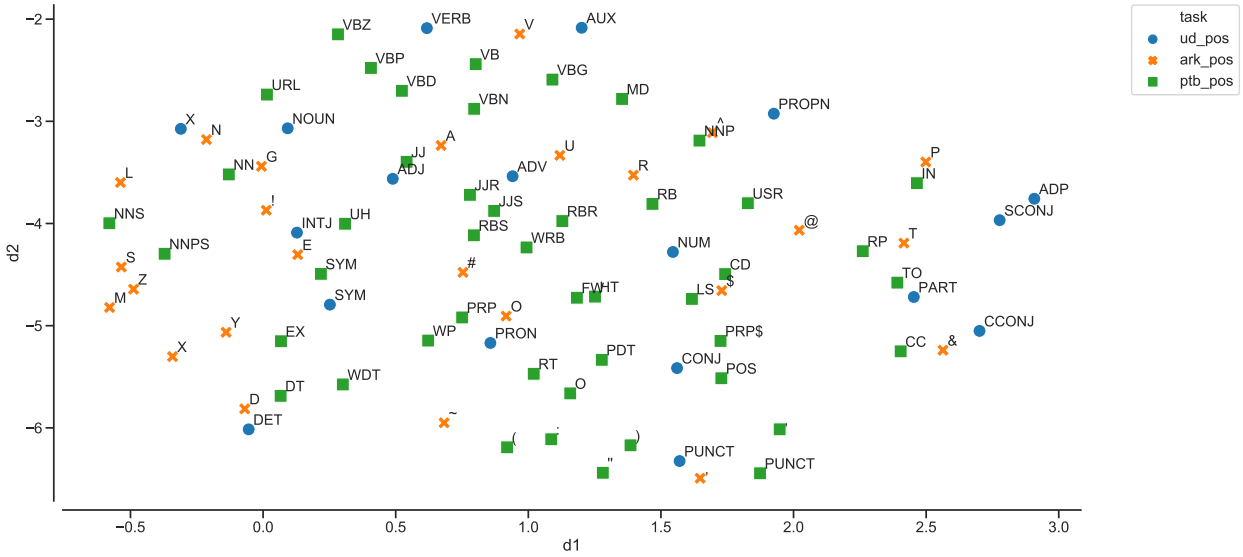


Figure 10.2: UMAP representation for POS embeddings of MTL model **MTL bilstm * #**

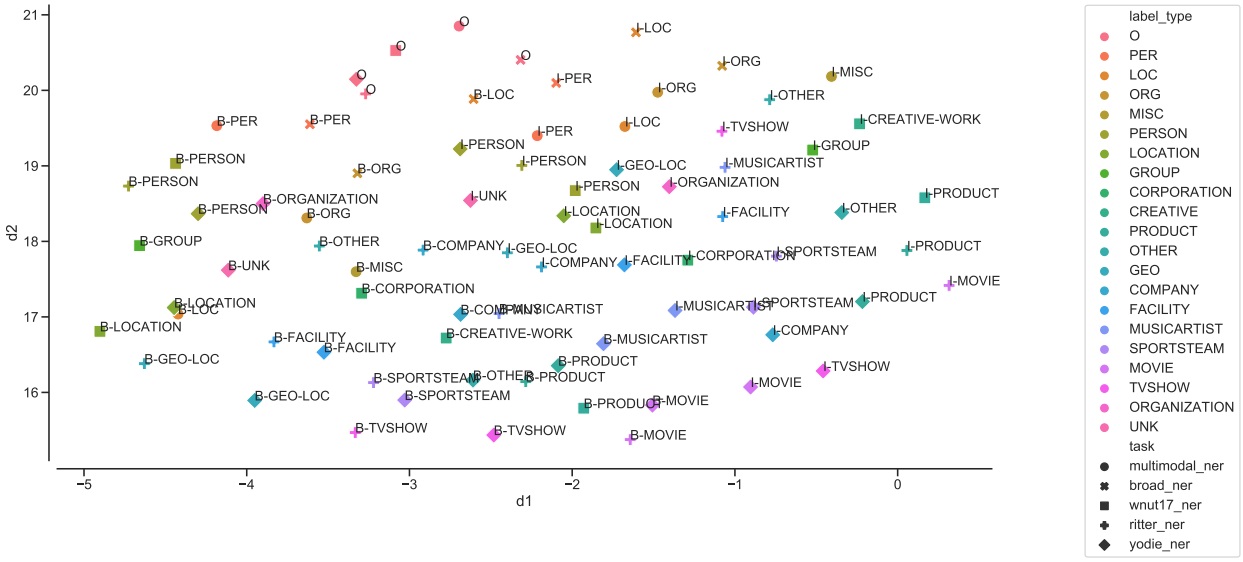


Figure 10.3: UMAP representation for NER embeddings of MTL model **MTL bilstm * #**

UMAP projection of CHUNK labels is shown in 10.4. The results are similar to NER, and since there is only one dataset for chunking, we see a consistent pattern in the direction from B tags to I tags. Again, the clustering based on label types is present here as well.

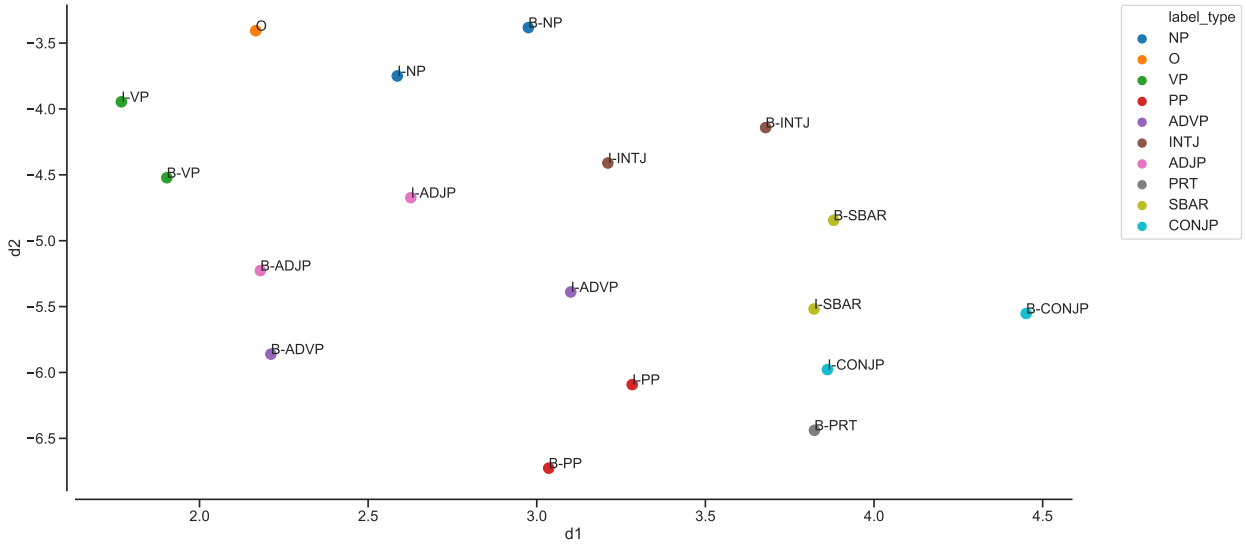


Figure 10.4: UMAP representation for CHUNK embeddings of MTL model **MTL bilstm * #**

UMAP projection of super-sense labels is shown in 10.5. This task has the maximum number of labels, but the primary type of the labels is either NOUN, VERB, or O. We decided to color the labels using the primary type. We observe a clear cluster of labels which are NOUNs or are VERBs. The direction of B to I labels is also quite similar across each cluster of primary labels types.

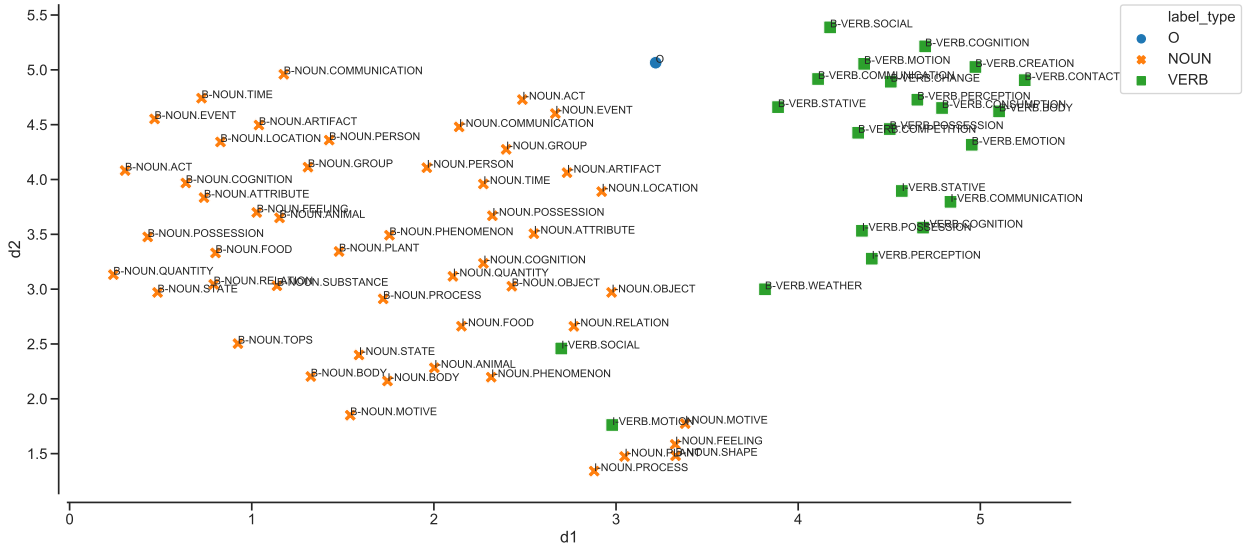


Figure 10.5: UMAP representation for super-sense embeddings of MTL model **MTL bilstm * #**

Human inspection of label representations shows that our model is indirectly learning the mapping between similar labels across different datasets.

10.5.7 Unified web interface for serving multi task models

We also created a model serving interface focused on showing the results of multi task models for sequence tagging. A screen-shot of the model outputs for a sample tweet is shown in figure 10.6. The example used an ill formed text with spelling mistakes for the noun phrase *Donald Trump*, being spelled as *donal drumph*. We find that the multimodal_ner output of our model correctly identifies this as referring to a person, while other outputs do not identify it as a person. We also see that SpaCy NER[16] and Stanford CoreNLP [17] do not identify this as named entity. Standford CoreNLP however, does identify this as an entity; using the OpenIE output as part of a longer span of text. Our interface further facilitates comparitive assesement of multi class outputs for sequence tagging.

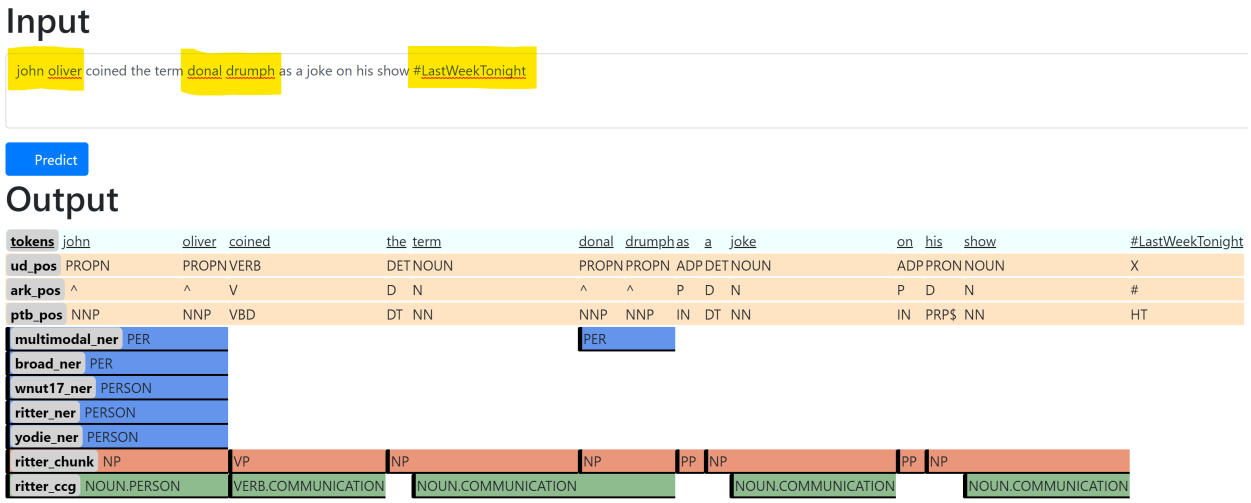
10.6 Conclusion

In this chapter, we have introduced a combined meta-corpus of Twitter data annotated for four sequence tagging tasks, namely, part of speech tagging, phrase chunking, supersense tagging, and named entity recognition. We utilized these datasets to evaluate the effectiveness of multitask learning models using neural network architecture. Our analysis reveals the effectiveness of multitask learning for the majority of the datasets and tasks compared to single dataset and task models. Furthermore, our trained models outperform most published models across different test datasets. We open source a library called SocialMediaIE and our trained models, which allows researchers to use our models for information extraction from tweets. We believe our work can aide further investigation into the application of multi-task learning for social media data.

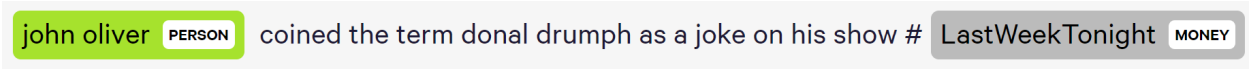
10.6.1 Connection with DSTD

Sequence tagging tasks are often used to perform IE on text data. In order to construct high quality DSTD, we need accurate and efficient models which can identify named entities, tag each word with PoS labels, and identify chunks and supersense tags. The MDMT models result in state-of-the art performance on most social media tagging benchmark corpus.

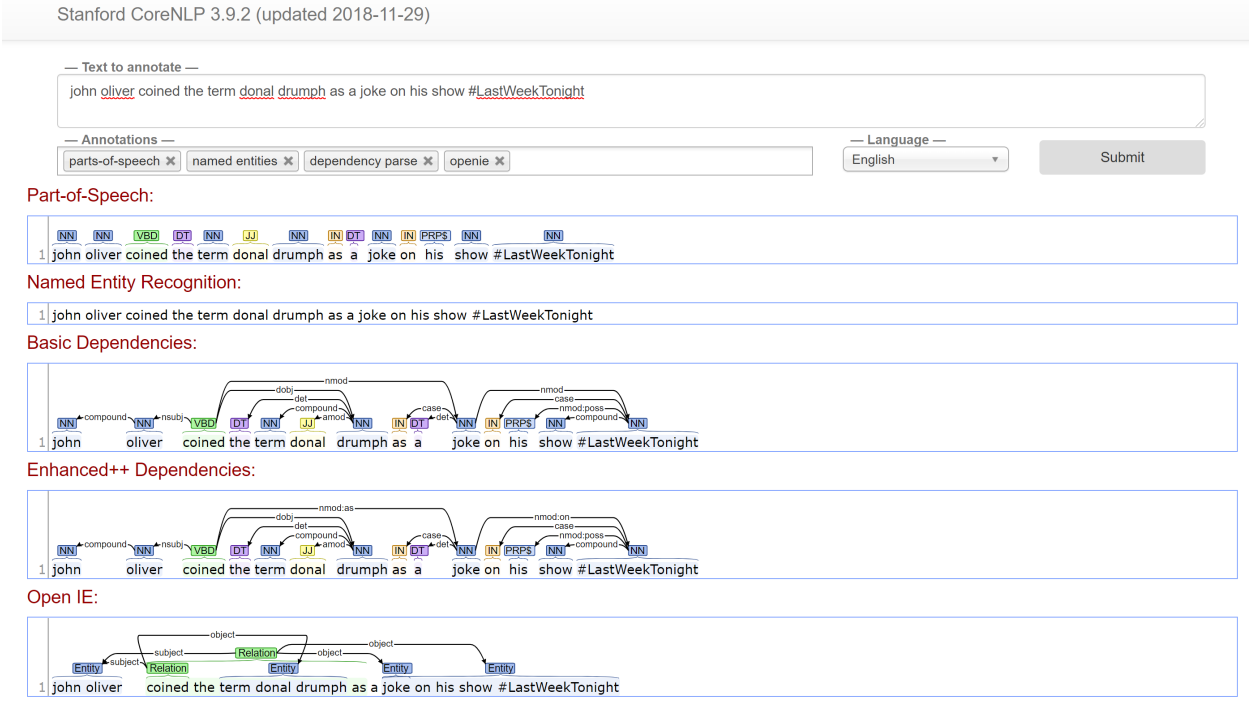
[16] https://explosion.ai/demos/displacy-ent
[17] https://corenlp.run



(a) SocialMediaIE - MTL tagger



(b) SpaCy NER



(c) Stanford CoreNLP

Figure 10.6: Comparison of outputs of SocialMediaIE tagger to other popular NER models.

Chapter 11

Deep multi-dataset multi-task learning for text classification

This chapter extends the approach in chapter 10 for text classification tasks in social media. Most of the methodology and experimental framework are the same. Changes are highlighted in their respective sections.

11.1 Tasks and Data

Sentiment classification is one of the most popular tasks for tweet text classification. However, in order to assess the performance improvements on multi task learning, we also considered two additional tasks. The first one is abusive content identification, and the second one is predicting uncertainty indicators. For each task, we consider data from various datasets as was the case for sequence tagging. Similar to the sequence tagging chapter, in this chapter, the multi data setting refers to the case when a model is trained on all datasets with diverse labelsets pertaining to the same task, whereas multi task setting will refer to models trained on all datasets for all tasks.

11.1.1 A note on reproducibility of annotated tweet datasets

It is important to note that most of the datasets are distributed as tweet id and label pairs based on Twitter's terms of service[1]. The user is expected to collect the tweet text using Twitter's API [2]. Since tweets can be deleted overtime and many user accounts can be deleted or deactivated, this approach for re-constructing tweet datasets results in a loss of tweets, some-times biased towards a particular label. Hence, the results achieved from re-doing the experiments presented here might not be comparable to the ones reported in the original papers that introduced the dataset. This issue was not encountered for the datasets used in chapter 10, as all of the data used there was distributed using only the text and the corresponding token labels.

Additionally, many datasets did not come with a pre-specified train, dev, and test split. Hence, I did the split for these datasets, using the same rule as the one used in chapter 10, i.e., 20% of the original data is

[1] https://developer.twitter.com/en/developer-terms/more-on-restricted-use-cases.html
[2] https://developer.twitter.com/en/docs/tweets/search/overview

used as test data. Of the remaining 80%, 10% (8% of the original) is used as dev, and the remaining (72%) used as train.

An exception to the above issues is the sentiment classification data, which I have compared against the baseline implemented in our earlier paper [Mishra and Diesner 2018]. That paper uses the same data as well as same train, dev, and test splits. Hence, these results can be compared directly.

11.1.2 Sentiment classification

For sentiment classification we use the same data as in [Mishra and Diesner 2018]. A description of these data is shown in table 11.1. Clarin Mozetič et al. [2016] and SemEval are the two largest corpora. However, SemEval has a larger test set. All the sentiment datasets use the traditional labels of positive, neutral, and negative for labeling the tweets.

Table 11.1: Description of sentiment classification datasets. Datasets clustered together are enclosed between horizontal lines. Labels are *negative*, *neutral*, *positive*.

data	**split**	tokens	tweets	vocab
Airline	**dev**	20079	981	3273
	test	50777	2452	5630
	train	182040	8825	11697
Clarin	**dev**	80672	4934	15387
	test	205126	12334	31373
	train	732743	44399	84279
GOP	**dev**	16339	803	3610
	test	41226	2006	6541
	train	148358	7221	14342
Healthcare	**dev**	15797	724	3304
	test	16022	717	3471
	train	14923	690	3511
Obama	**dev**	3472	209	1118
	test	8816	522	2043
	train	31074	1877	4349
SemEval	**dev**	105108	4583	14468
	test	528234	23103	43812
	train	281468	12245	29673

11.1.3 Abusive content classification

The second task we consider is abusive content classification. This task has recently gained prominence, owing to the the growth of abusive content on social media platforms. We utilize two datasets of abusive content. The first data is Founta from Founta et al. [2018], which tags tweets as *abusive, hateful, normal, spam*. The second dataset is WaseemSRW from Waseem and Hovy [2016]. It tags the data as *none, racism,*

sexism. The rationale for including both these data under the same task it the core idea of identifing abusive content either direct or using racist or sexist variation. A description of these data is shown in table 11.2.

Table 11.2: Description of abusive content classification datasets. Datasets which are clustered together are enclosed between horizontal lines. Labels for Founta are *abusive*, *hateful*, *normal*, and *spam*. Labels for WaseemSRW are *none*, *racism*, and *sexism*.

data	split	tokens	tweets	vocab
Founta	**dev**	102534	4663	22529
	test	256569	11657	44540
	train	922028	41961	118349
WaseemSRW	**dev**	25588	1464	5907
	test	64893	3659	10646
	train	234550	13172	23042

11.1.4 Uncertainty indicators

Finally, we consider a collection of datasets for the task of identifying uncertainty indicators. Uncertainty indicators are defined as indicators in text which capture a level of uncertainty about the text, e.g., veridictality or sarcasm (uncertainty in intended meaning). We consider two datasets for this task as well. The first dataset is Riloff from Riloff et al. [2013]. This dataset consists of tweets annotated for sarcasm and non-sarcasm. The second dataset is Swamy from Swamy et al. [2017]. This dataset tries to identify the level of veridictality or degree of belief expressed in the tweet. The label set for this data is *definitely no, probably no, uncertain, probably yes, definitely yes*. A description of these data is shown in 11.3.

Table 11.3: Description of uncertainty indicators dataset. Datasets which are clustered together are enclosed between horizontal lines. Labels for Riloff are *sarcasm* and *not sarcasm*. Labels are for Swamy are *definitely no*, *definitely yes*, *probably no*, *probably yes*, and *uncertain*.

data	split	tokens	tweets	vocab
Riloff	**dev**	2126	145	1002
	test	5576	362	1986
	train	19652	1301	5090
Swamy	**dev**	1597	73	738
	test	3909	183	1259
	train	14026	655	2921

11.2 Model

We use the same modeling approach as described in chapter 10. The only difference is that instead of using the sequential output, we use the mean vector of all the outputs for each token from the intermediate layers.

We also try with two different intermediate layers, biLSTM and convolution. For the convolution layer, we use filters of sizes 2, 3, 4, and 5. The last layer of the model is a softmax for each task. Similar to the last experiment we also try out a model with learning rate of $1e-3$ and $\lambda = 0$ for L_2 regularization. For stacked models, we use the task ordering as *sentiment, abusive, and uncertainty*. Our decision for this type of task ordering for classification is not based on task hierarchy, but on data size. We have kept the task with larger data in a lower layer to allow tasks with lesser data to have their own representation of the output.

11.3 Results

In this section, we use the following naming convention for identifying the various models:

1. Models trained on single datasets have prefix *S*.
2. Models trained on all datasets of same task have prefix *MD*.
3. Models trained on all datasets have prefix. *MTS* for multitask models with shared module, and *MTL* for stacked modules.
4. Models with $lr = 0.001$ and no L_2 regularization have suffix *"*"*.

Each model was evaluated using micro-f1 score across the respective test data for each data withing the task. The relative rank (r), based on the evaluation metric (v), was identified for each model.

11.3.1 Sentiment classification

The results of our experiments for sentiment classification are shown in table 11.4. Similar to the observation in chapter 10, we observe that the results for un-regularized model are better than the regularized ones. Furthermore, we see comparitively less significant gains in using multi task models for sentiment classification. The single CNN model performs best on GOP, Obama, and Clarin data, all of which are political topics. The MTL biLSTM model is the best on the SemEval dataset which is one of the most competitive benchmark for tweet sentiment classification. However, in all cases the multi task models are very close to the the best models, and hence should be preferred owing to their lower computational requirements like model sizes and runtimes. All of our models perform better in terms of the evaluation scores compared to the models in Mishra and Diesner [2018].

11.3.2 Abusive content identification

The results of our experiments for abusive content identification are shown in table 11.5. Here, the MD cnn * model is the most accurate performer on both datasets, and the top model for WaseemSRW, while not

Table 11.4: Micro F1 for sentiment classification datasets. (r = rank, v = micro-f1)

file	Airline		Clarin		GOP		Healthcare		Obama		SemEval	
model	r	v	r	v	r	v	r	v	r	v	r	v
S bilstm	8	80.46	8	65.71	5	67.05	6	63.88	9	59.00	9	65.57
MD bilstm	9	79.77	9	65.28	8	65.95	9	60.95	8	59.58	6	67.05
MTS bilstm	11	63.21	10	47.37	10	56.78	10	60.25	11	38.89	11	40.43
MTL bilstm	10	63.70	11	47.00	11	45.21	11	59.69	10	44.64	10	49.92
S bilstm *	6	81.69	3	67.71	3	67.55	3	65.97	1	62.64	7	66.47
MD bilstm *	5	81.85	7	66.23	7	66.50	4	64.85	3	61.69	3	68.98
MTS bilstm *	7	81.65	6	66.55	4	67.45	2	66.81	7	60.34	1	69.52
MTL bilstm *	2	82.22	4	67.60	2	68.10	1	67.09	6	61.30	2	69.10
S cnn *	3	82.10	1	68.18	1	68.89	8	62.34	1	62.64	8	66.19
MD cnn *	1	82.54	5	67.01	6	66.65	7	63.18	5	61.49	4	68.04
MTS cnn *	4	82.06	2	67.72	9	64.81	5	64.57	3	61.69	5	67.63

showing any significant drop for Founta compared to other models. The best reported model for WaseemSRW is 73.93% in Waseem and Hovy [2016]. In Founta et al. [2018], the authors do not report any text classification results.

Table 11.5: Micro F1 for uncertainty indicators datasets. (r = rank, v = micro-f1)

file	Founta		WaseemSRW	
model	r	v	r	v
S bilstm	8	79.33	8	81.72
MD bilstm	9	79.03	9	81.31
MTS bilstm	11	61.48	11	68.57
MTL bilstm	10	69.26	10	70.13
S bilstm *	1	80.60	3	82.95
MD bilstm *	2	80.35	2	83.22
MTS bilstm *	6	80.11	7	81.99
MTL bilstm *	4	80.23	5	82.78
S cnn *	3	80.25	4	82.89
MD cnn *	5	80.18	1	84.42
MTS cnn *	7	79.92	6	82.67

11.3.3 Uncertainty indicators

The results of our experiments for uncertainty indicators are shown in table 11.6. Here MD cnn * model is the top performer on both datasets, being the top model for WaseemSRW, while not showing significant drop for Founta compared to other models. Here again, the MTL biLSTM * model is the top performer on both datasets, and the second best model for the Riloff data. Even though the results cannot be compared directly but for context, the micro-f1 reported in Riloff et al. [2013] using 10-fold cross validation is 51%. Similarly, the micro-f1 reported in Swamy et al. [2017] is 68%, which uses identification of entity, target and

opponent as features. Our model for the Swamy data is not comparable as we are treating the problem as simple text classification problem.

Table 11.6: Micro F1 for sentiment abusive content datasets. (r = rank, v = micro-f1)

file	Riloff		Swamy	
model	r	v	r	v
S bilstm	6	81.22	5	38.80
MD bilstm	9	79.28	1	39.34
MTS bilstm	10	58.84	10	27.87
MTL bilstm	11	58.01	11	23.50
S bilstm *	3	83.43	1	39.34
MD bilstm *	7	80.94	1	39.34
MTS bilstm *	5	82.60	6	38.25
MTL bilstm *	2	83.98	1	39.34
S cnn *	1	85.64	7	35.52
MD cnn *	4	83.15	8	32.79
MTS cnn *	8	80.11	9	31.15

11.3.4 Investigating label representations

Similar to our tagging model in chapter 10, we again investigate the space of these label representations for label similarity. Our labels vectors for classification models are 100 dimensional. We again utilize UMAP [McInnes et al. 2018b;a] for dimensionality reduction. We use the **MTS bilstm *** model for our analysis as it is one of the most consistent multi-task models in our previous results.

UMAP projection of classification labels is shown in 11.1. We observe that labels which represent similar meaning across different datasets cluster together in this representation space. Some interesting observations are the closeness of the following labels: neutral, uncertain, none, normal, and not_sarcasm. Each of these classes represents the base class of their dataset, and we observe that they are quite similar in their representation space. Similarly, negative, abusive, hateful, sexism, and racism are clustered together compared to positive. This result again shows that out model is learning relationship between the labels, which can be utilized for model verification.

11.3.5 Unified web interface for serving multi task models

We also created a model serving interface focused on showing the results of multi task models for text classification. A screenshot of the model outputs for a sample tweet is shown in figure 11.2. Our interface further facilitates comparative assessment of multi task output for text classification tasks.

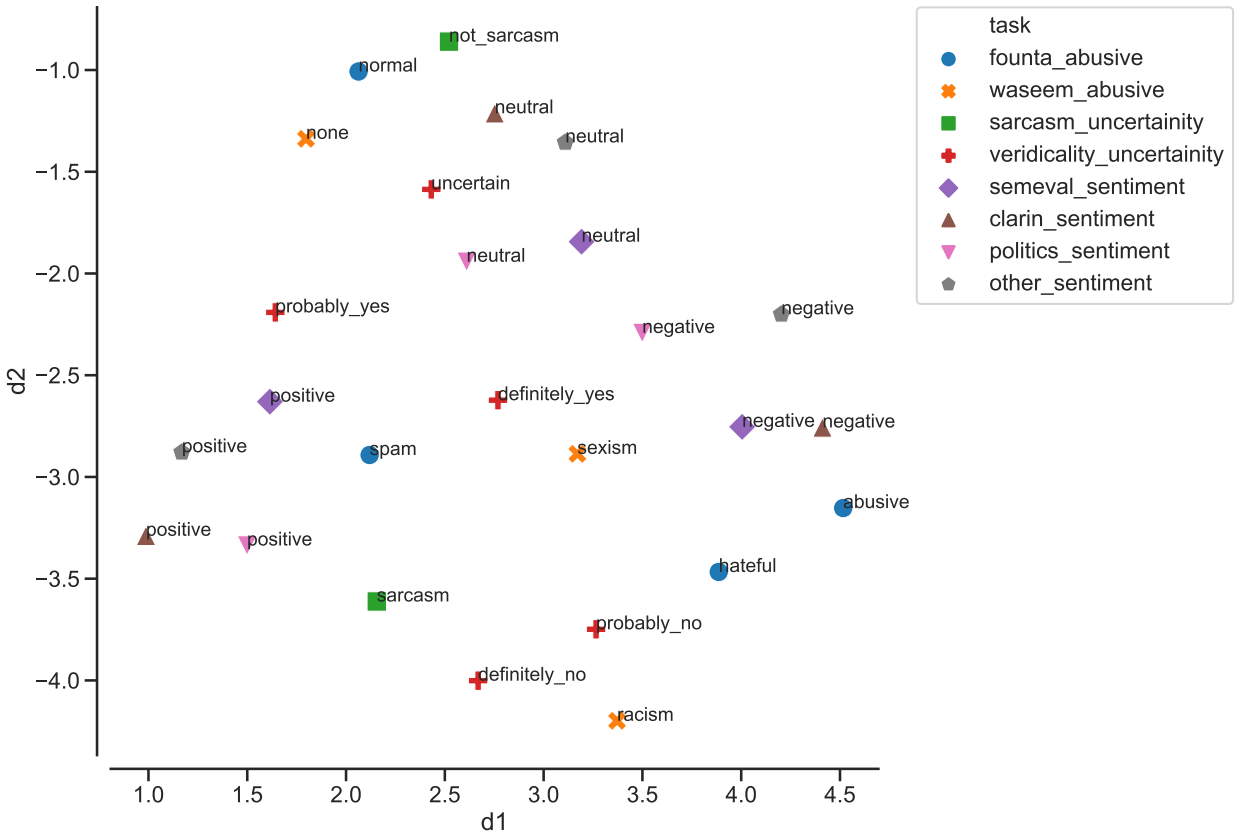


Figure 11.1: UMAP representation for classification embeddings of multi-task model **MTS bilstm ***

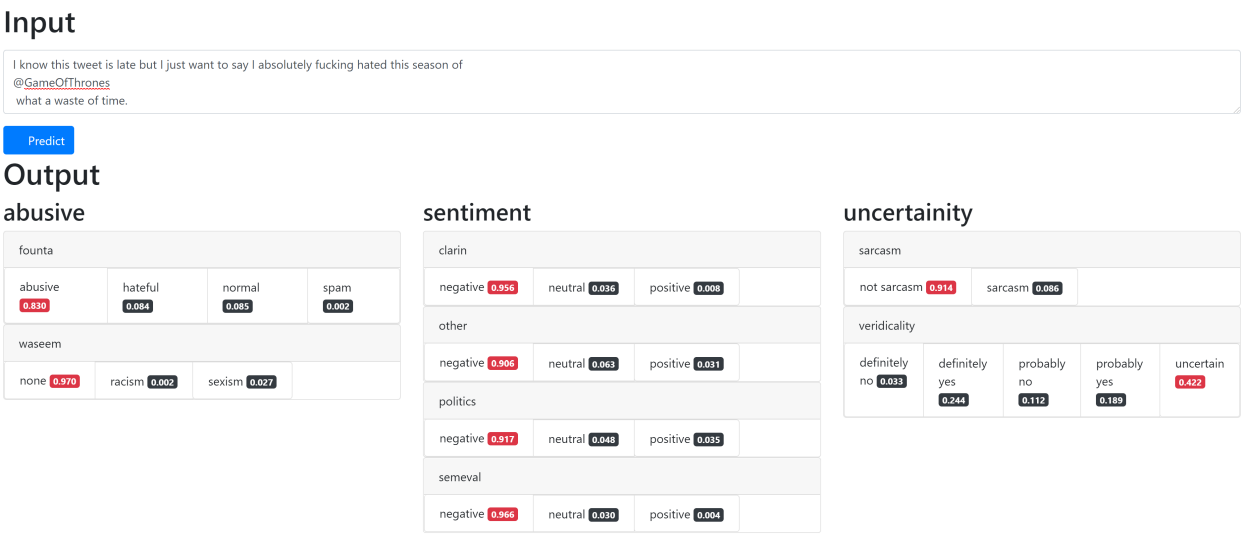


Figure 11.2: Outputs of SocialMediaIE multi task classifier, on tweets.

11.4 Conclusion

In this chapter, we demonstrated the effectiveness of multi task learning for three classification tasks. We observe that MTL may not always give the best model but is always competitive against single task models, and can be a good alternative for efficient model serving in resource constrained scenarios. We make our models and code available via a web based python tool.

11.4.1 Connection with DSTD

Similar to chapter 10, this chapter also describes compute efficient and accurate MDMT models which can be run on text components of DSTD, constructed from social media data. These text classification labels can later be used as metadata of DSTD nodes or can be aggregated over nodes, e.g., sentiment aggregated for user over time. A visual example of how this DSTD will appear, is shown in figure 4.2 from chapter 4.

Part III

Moving forward

Chapter 12

Thesis conclusions

This dissertation has tried to fill a gap in information extraction for social media and scholarly data. Using the unifying abstraction of Digital Social Trace Data, the dissertation has opened the doors to cross-pollinate ideas from the domains of social media and scholarly publications. In various chapters of the thesis, I have tried to answer the research questions identified in chapter 1. Below, I will revisit these questions and summarize how various contributions of this thesis helps to answer each of them.

RQ 1 *How to use all information available to improve the efficiency and accuracy of IE from DSTD?:* I suggest using active human-in-the-loop learning (chapter 8), semi-supervised learning (chapter 9), and multi-task learning for improving sequence tagging (chapter 10) and text classification in tweets (chapter 11). Also, Wikipedia information can be efficiently utilized for improving information extraction in scholarly data from various domains (chapter 7).

RQ 2 *What information to extract?:* I describe how extracting temporal profiles of concepts and authors in scholarly data can help quantify novelty (chapter 2), and expertise (chapter 3). I suggest an alternative orthogonal set of labels and annotated data which identify if a tweet supports or opposes the cause and if it conveys an author's enthusiasm or passiveness towards the cause (chapter 5). Finally, I propose the extraction of bias towards user and tweet meta-data in sentiment annotated corpora (chapter 6).

RQ 3 *How can the extracted information be presented and utilized?:* Present a visualization framework for DSTD which allows presenting temporal, network, and meta-data aspects of the corpus (chapter 4).

12.1 Other related work

I have also worked on other applications of using IE from DSTD. These are listed below:

1. In Mishra et al. [2018c], we investigated gender bias in self-citation patterns in scholarly data from biomedical data.
2. In Collier et al. [2017] and Collier et al. [2019], we worked on extracting information from Facebook comments about tuition free college to understand the discourse around the topic as well as alignment

between sentiment and political identities.

3. In an ongoing work, we extend the findings and data of Mishra and Diesner [2018] to study how sentiment evolves temporally for tweets of a given user.
4. In another ongoing work, we investigate the similarity between datasets used in chapters 10 and 11 to identify their suitability for multi-dataset multi-task learning.
5. In Addawood et al. [2017], we have developed an information source lexicon that can be utilized for classifying data from various URL sources into scientific, fakenews, social media, etc.
6. We are investigating how can social networks be adversarial perturbed, leading to incorrect conclusions. Since DSTD are an example of temporal networks, this work can be utilized to assess the robustness of inferences drawn from DSTD under adversarial attacks. Some preliminary findings on simulated social networks are described in Avram et al. [2019].
7. In Mishra and Mishra [2019], Mishra et al. [2020], we investigated how pretrained deep neural networks for text can be fine-tuned to achieve state of the art performance on hate speech and offensive content identification in Indo-European languages.

12.2 Limitations of our work and approach

Although the work presented here offers new insight into IE tasks, it is limited in the following ways.

- Utilizing DSTD structure for large scale data processing is cumbersome. The methodologies described in chapters 2 and 3, overcome this by utilizing parallel computing. This might not always be feasible when working on large data using limited computational resources. This problem is further excaberated for visualizing DSTD using the SCTG visualization framework of chapter 4.

- The models developed and described in chapters 8, 9, 10, and 11, will inherit the same socio-cultural biases as present in the existing data. In our current approach, we have not investigated either identification or rectification of these biases in the models learned. This should be an important consideration for scholars who are using our models and tools to draw inferences on data, which are likely to exhibit these biases.

12.3 Summary of contributions

Let us reiterate some of the major contributions of this thesis.

1. Proposed a unifying abstraction of DSTD to facilitate common research questions for social media and scholarly communications data.
2. Experimentally verified the validity and effectiveness of human-in-the-loop active learning, semi-supervised learning, and multi task learning for sequence tagging and text classification tasks.
3. Synthesized data from multiple publicly available sources to construct benchmark datasets for evaluating information extraction tasks applied to social media data.
4. Introduced a concept taxonomy for scholarly publications in computer science based on Wikipedia's category tree.
5. Demonstrated how concept taxonomies can be utilized to identify conceptual novelty and expertise from scholarly data.
6. Developed open source tools and user interfaces to facilitate information extraction from social media and other DSTD.

12.4 List of tools

This thesis has led to the development of multiple open source tools which can be utilized to reproduce and extend the research work presented in various chapters. These tools are:

1. **SocialMediaIE** - Tool for multiple social media information extraction techniques. https://github.com/socialmediaie
2. **TwitterNER** - Tool for named entity recognition for Tweets. https://github.com/napsternxg/TwitterNER
3. **SAIL** - Tool for human-in-the-loop incremental learning for sentiment classification. https://github.com/uiuc-ischool-scanr/SAIL
4. **Social Communication Temporal Graph** - Tool for building SCTG visualizations. https://shubhanshu.com/social-comm-temporal-graph/
5. **GIMLI** - Tool for visualizing and exploring novelty scores for authors in PubMed. http://abel.ischool.illinois.edu/gimli/. Source code: https://github.com/napsternxg/Novelty
6. **LEGOLAS** - Tool for visualizing and exploring article level expertise of authors in PubMed. http://abel.ischool.illinois.edu/legolas/
7. **liteAnnotate** - Web based interface for collecting annotated data in teams. https://github.com/napsternxg/liteAnnotate

12.5 List of datasets

This thesis has led to the development of multiple open access datasets that can be utilized to reproduce and extend the research work presented in various chapters. These datasets are:

1. Mishra et al. [2018b] - Mishra, S., Fegley, B. D., Diesner, J., and Torvik, V. I. (2018b). Self-citation analysis data based on PubMed Central subset (2002-2005)
2. Mishra and Torvik [2018] - Mishra, S. and Torvik, V. I. (2018). Conceptual novelty scores for PubMed articles
3. Mishra [2019c] - Mishra, S. (2019c). Trained models for multi-task multi-dataset learning for text classification as well as sequence tagging in tweets
4. Mishra [2019d] - Mishra, S. (2019d). Trained models for multi-task multi-dataset learning for text classification in tweets
5. Mishra [2019b] - Mishra, S. (2019b). Trained models for multi-task multi-dataset learning for sequence prediction in tweets
6. Mishra [2019e] - Mishra, S. (2019e). Wikipedia category embeddings - Node2Vec, Poincare, Elmo
7. Mishra et al. [2019] - Mishra, S., Agarwal, S., Guo, J., Phelps, K., Picco, J., and Diesner, J. (2019). Tweet IDs annotated for enthusiasm and support towards social causes: CTE, cyberbullying, and LGBT

12.6 Future directions

This dissertation has opened up many vistas for improving the methods and applications discussed. Here I list a few open areas of further research on this topic:

1. How to develop efficient learning algorithms that utilize the DSTD structure of the data?
2. How can visual interfaces for displaying large scale DSTD be improved?
3. What are some other domains where the DSTD representation of data will help?
4. How can humans be efficiently looped into the machine learning process?
5. How can temporal concept profiles be utilized for social media data?
6. What are the limits of multi-task learning? When should we trade-off efficiency of MTL to improve accuracy of dedicated models?

Finally, resources related to this thesis and updates on future work related to this thesis can be found at https://shubhanshu.com/phd_thesis/

References

Abbasi, A., Hassan, A., and Dhar, M. (2014). Benchmarking Twitter Sentiment Analysis Tools. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC-2014)*, pages 823–829, Reykjavik, Iceland. European Language Resources Association (ELRA).

Addawood, A., Rezapour, R., Mishra, S., Schneider, J., and Diesner, J. (2017). Developing an Information Source Lexicon. In *Prioritising Online Content workshop co-located at NIPS*.

Aguilar, G., López Monroy, A. P., González, F., and Solorio, T. (2018). Modeling Noisiness to Recognize Named Entities using Multitask Neural Networks on Social Media. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1401–1412, New Orleans, Louisiana. Association for Computational Linguistics.

Akbik, A., Bergmann, T., and Vollgraf, R. (2019). Pooled Contextualized Embeddings for Named Entity Recognition. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*. Association for Computational Linguistics.

Alonso, H. M. and Plank, B. (2017). When is multitask learning effective? Semantic sequence prediction under varying data conditions. In *15th Conference of the European Chapter of the Association for Computational Linguistics, EACL 2017 - Proceedings of Conference*, volume 1, pages 44–53, Valencia, Spain. Association for Computational Linguistics.

Asur, S. and Huberman, B. A. (2010). Predicting the Future with Social Media. In *2010 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology*, volume 1, pages 492–499. IEEE.

Aue, A. and Gamon, M. (2005). Customizing Sentiment Classifiers to New Domains: A Case Study. *Proceedings of Recent Advances in Natural Language Processing (RANLP)*, 3(3):16–18.

Augenstein, I., Derczynski, L., and Bontcheva, K. (2017). Generalisation in named entity recognition: A quantitative analysis. *Computer Speech and Language*, 44(C):61–83.

Avram, M. V., Mishra, S., Parulian, N. N., and Diesner, J. (2019). Adversarial perturbations to manipulate the perception of power and influence in networks. In *2019 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining*, pages 986–993, Vancouver, Canada. IEEE.

Bakshy, E., Hofman, J. M., Mason, W. A., and Watts, D. J. (2011). Everyone's an influencer: quantifying influence on twitter. In *Proceedings of the fourth ACM international conference on Web search and data mining - WSDM '11*, page 65, New York, New York, USA. ACM Press.

Baldwin, T., de Marneffe, M.-C., Han, B., Kim, Y.-B., Ritter, A., and Xu, W. (2015). Shared Tasks of the 2015 Workshop on Noisy User-generated Text: Twitter Lexical Normalization and Named Entity Recognition. In *Proceedings of the Workshop on Noisy User-generated Text*, pages 126–135, Stroudsburg, PA, USA. Association for Computational Linguistics.

Barre-Sinoussi, F., Chermann, J., Rey, F., Nugeyre, M., Chamaret, S., Gruest, J., Dauguet, C., Axler-Blin, C., Vezinet-Brun, F., Rouzioux, C., Rozenbaum, W., and Montagnier, L. (1983). Isolation of a T-lymphotropic retrovirus from a patient at risk for acquired immune deficiency syndrome (AIDS). *Science*, 220(4599):868–871.

Bates, T., Anić, A., Marušić, M., and Marušić, A. (2004). Authorship Criteria and Disclosure of Contributions. *JAMA*, 292(1):86–88.

Beck, F., Burch, M., Diehl, S., and Weiskopf, D. (2017). A Taxonomy and Survey of Dynamic Graph Visualization. *Computer Graphics Forum*, 36(1):133–159.

Bingel, J. and Søgaard, A. (2017). Identifying beneficial task relations for multi-task learning in deep neural networks. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, volume 2, pages 164–169, Valencia, Spain. Association for Computational Linguistics.

Blei, D., Carin, L., and Dunson, D. (2010). Probabilistic Topic Models. *IEEE Signal Processing Magazine*, 27(6):55–65.

Blum, A. (1998). On-Line Algorithms in Machine Learning. In *Proceedings of the Workshop on On-Line Algorithms*, pages 306–325.

Blum, A. and Mitchell, T. (1998). Combining labeled and unlabeled data with co-training. In *Proceedings of the eleventh annual conference on Computational learning theory - COLT' 98*, pages 92–100, New York, New York, USA. ACM Press.

Bollen, J., Mao, H., and Zeng, X. (2011). Twitter mood predicts the stock market. *Journal of Computational Science*, 2(1):1–8.

Bottou, L. (1991). Stochastic Gradient Learning in Neural Networks. In *Proceedings of Neuro-N\ˆimes 91*, Nimes, France. EC2.

Bottou, L. (2010). Large-Scale Machine Learning with Stochastic Gradient Descent. In *Proceedings of COMPSTAT'2010*, pages 177–186. Physica-Verlag HD, Heidelberg.

Brin, S. and Page, L. (1998). The anatomy of a large-scale hypertextual Web search engine. *Computer Networks and ISDN Systems*, 30(1-7):107–117.

Cano, A. E., Varga, A., Rowe, M., Stankovic, M., and Dadzie, A.-S. (2013). Making Sense of Microposts (#MSM2013) Concept Extraction Challenge. In *#MSM*.

Carlson, A., Betteridge, J., Wang, R. C., Hruschka, E. R., and Mitchell, T. M. (2010). Coupled semi-supervised learning for information extraction. In *Proceedings of the third ACM international conference on Web search and data mining - WSDM '10*, page 101, New York, New York, USA. ACM Press.

Caruana, R. and Niculescu-Mizil, A. (2006). An empirical comparison of supervised learning algorithms. In *Proceedings of the 23rd international conference on Machine learning - ICML '06*, volume C, pages 161–168, New York, New York, USA. ACM Press.

Caruana, R. A. (1993). Multitask Learning: A Knowledge-Based Source of Inductive Bias. In *Machine Learning Proceedings 1993*, pages 41–48. Elsevier.

Changpinyo, S., Hu, H., and Sha, F. (2018). Multi-Task Learning for Sequence Tagging: An Empirical Study. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 2965–2977, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

Chapelle, O., Scholkopf, B., and Zien, A. (2006). *Semi-supervised learning*. MIT Press.

Cherry, C. and Guo, H. (2015). The Unreasonable Effectiveness of Word Representations for Twitter Named Entity Recognition. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 735–745, Stroudsburg, PA, USA. Association for Computational Linguistics.

Clark, A. (2003). Combining distributional and morphological information for part of speech induction. In *Proceedings of the tenth conference on European chapter of the Association for Computational Linguistics - EACL '03*, volume 1, page 59, Morristown, NJ, USA. Association for Computational Linguistics.

Clement, T. P. (2014). Authorship Matrix: A Rational Approach to Quantify Individual Contributions and Responsibilities in Multi-Author Scientific Articles. *Science and Engineering Ethics*, 20(2):345–361.

Cohen, J. (1960). A Coefficient of Agreement for Nominal Scales. *Educational and Psychological Measurement*, 20(1):37–46.

Collbert, R., Weston, J., Bottou, L., Karlen, M., Kavukcuoglu, K., and Kuksa, P. (2011). Natural Language Processing (Almost) from Scratch. *Journal ofMachine Learning Research*, 12:2493–2537.

Collier, D., Mishra, S., Houston, D., Hensley, B., Mitchell, S., and Hartlep, N. (2019). Who is Most Likely to Oppose Federal Tuition-Free College Policies? Investigating Variable Interactions of Sentiments to Americas College Promise. *SSRN Electronic Journal*.

Collier, D. A., Mishra, S., Houston, D. A., Hensley, B. O., and Hartlep, N. D. (2017). Americans support the idea of tuition-free college: an exploration of sentiment and political identity signals otherwise. *Journal of Further and Higher Education*, pages 1–16.

Collobert, R. and Weston, J. (2008). A Unified Architecture for Natural Language Processing: Deep Neural Networks with Multitask Learning. In *Proceedings of the 25th International Conference on Machine Learning*, ICML '08, pages 160–167, New York, NY, USA. ACM.

Coulter, N. (1997). ACM'S computing classification system reflects changing times. *Communications of the ACM*, 40(12):111–112.

Dang, T. N., Pendar, N., and Forbes, A. G. (2016). TimeArcs: Visualizing Fluctuations in Dynamic Networks. *Computer Graphics Forum*, 35(3):61–69.

Daumé III, H. (2007). Frustratingly Easy Domain Adaptation. *Association for Computational Linguistic (ACL)s*, (June):256–263.

Derczynski, L., Bontcheva, K., and Roberts, I. (2016). Broad Twitter Corpus: A Diverse Named Entity Recognition Resource. *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 1169–1179.

Derczynski, L., Maynard, D., Aswani, N., and Bontcheva, K. (2013a). Microblog-genre Noise and Impact on Semantic Annotation Accuracy. In *Proceedings of the 24th ACM Conference on Hypertext and Social Media*, HT '13, pages 21–30, New York, NY, USA. ACM.

Derczynski, L., Maynard, D., Rizzo, G., van Erp, M., Gorrell, G., Troncy, R., Petrak, J., and Bontcheva, K. (2015). Analysis of named entity recognition and linking for tweets. *Information Processing & Management*, 51(2):32–49.

Derczynski, L., Nichols, E., van Erp, M., and Limsopatham, N. (2017). Results of the WNUT2017 Shared Task on Novel and Emerging Entity Recognition. In *Proceedings of the 3rd Workshop on Noisy User-generated Text*, pages 140–147, Copenhagen, Denmark. Association for Computational Linguistics.

Derczynski, L., Ritter, A., Clark, S., and Bontcheva, K. (2013b). Twitter Part-of-Speech Tagging for All: Overcoming Sparse and Noisy Data. *Proceedings of the International Conference Recent Advances in Natural Language Processing RANLP 2013*, pages 198–206.

Diesner, J. and Chin, C.-L. (2015). Usable ethics: practical considerations for responsibly conducting research with social trace data. *Proceedings of Beyond IRBs: Ethical Review Processes for Big Data Research.*

Diesner, J., Kim, J., and Pak, S. (2014). Computational Impact Assessment of Social Justice Documentaries. *The Journal of Electronic Publishing*, 17(3).

Drik, L. (1999). A Measure of Originality: The Elements of Science. *Social Studies of Science*, 29(5):765–776.

Eisenstein, J. (2013). What to do about bad language on the internet.

El-Assady, M., Sevastjanova, R., Gipp, B., Keim, D., and Collins, C. (2017). NEREx: Named-Entity Relationship Exploration in Multi-Party Conversations. *Computer Graphics Forum*, 36(3):213–225.

Evans, J. and Foster, J. (2015). Measuring Novelty by Simulating Discovery.

Evans, J. A. (2010). Industry Induces Academic Science to Know Less about More. *American Journal of Sociology*, 116(2):389–452.

Fan, R., Zhao, J., Chen, Y., and Xu, K. (2014). Anger Is More Influential than Joy: Sentiment Correlation in Weibo. *PLoS ONE*, 9(10):e110184.

Fegley, B. D. and Torvik, V. I. (2013). Has Large-Scale Named-Entity Network Analysis Been Resting on a Flawed Assumption? *PLoS ONE*, 8(7):e70299.

Felbo, B., Mislove, A., Søgaard, A., Rahwan, I., and Lehmann, S. (2017). Using millions of emoji occurrences to learn any-domain representations for detecting sentiment, emotion and sarcasm. *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1616–1626.

Finin, T., Murnane, W., Karandikar, A., Keller, N., Martineau, J., and Dredze, M. (2010). Annotating Named Entities in Twitter Data with Crowdsourcing. *Proceedings of the NAACL HLT 2010 Workshop on Creating Speech and Language Data with Amazon's Mechanical Turk*, 2010(January):8088.

Finn, C., Abbeel, P., and Levine, S. (2017). Model-Agnostic Meta-Learning for Fast Adaptation of Deep Networks. In Precup, D. and Teh, Y. W., editors, *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pages 1126–1135, International Convention Centre, Sydney, Australia. PMLR.

Founta, A.-M., Djouvas, C., Chatzakou, D., Leontiadis, I., Blackburn, J., Stringhini, G., Vakali, A., Sirivianos, M., and Kourtellis, N. (2018). Large Scale Crowdsourcing and Characterization of Twitter Abusive Behavior. In *International AAAI Conference on Web and Social Media.*

Fromreide, H., Hovy, D., and Søgaard, A. (2014). Crowdsourcing and annotating NER for Twitter #drift. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, pages 2544–2547. European language resources distribution agency.

Gallo, R., Sarin, P., Gelmann, E., Robert-Guroff, M., Richardson, E., Kalyanaraman, V., Mann, D., Sidhu, G., Stahl, R., Zolla-Pazner, S., Leibowitch, J., and Popovic, M. (1983). Isolation of human T-cell leukemia virus in acquired immune deficiency syndrome (AIDS). *Science*, 220(4599):865–867.

Gao, W. and Sebastiani, F. (2015). Tweet sentiment: From classification to quantification. In *Proceedings of the 2015 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining 2015 - ASONAM '15*, pages 97–104, New York, New York, USA. ACM Press.

Gardner, M., Grus, J., Neumann, M., Tafjord, O., Dasigi, P., Liu, N., Peters, M., Schmitz, M., and Zettlemoyer, L. (2018). AllenNLP: A Deep Semantic Natural Language Processing Platform. In *Proceedings of Workshop for NLP Open Source Software (NLP-OSS)*, pages 1–6, Melbourne, Australia.

Gimpel, K., Schneider, N., Mills, D., O'Connor, B., Das, D., Eisenstein, J., Heilman, M., Yogatama, D., Flanigan, J., Smith, N. A., Mills, D., Eisenstein, J., Heilman, M., Yogatama, D., Flanigan, J., and Smith, N. A. (2011). Part-of-speech tagging for twitter: Annotation, features, and experiments. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics*, pages 42–47.

Go, A., Bhayani, R., and Huang, L. (2009). Twitter Sentiment Classification using Distant Supervision.

Gorrell, G., Petrak, J., and Bontcheva, K. (2015). Using @Twitter Conventions to Improve #LOD-Based Named Entity Disambiguation. In *The Semantic Web. Latest Advances and New Domains*, pages 171–186. Springer International Publishing.

Grover, A. and Leskovec, J. (2016). node2vec: Scalable Feature Learning for Networks. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining - KDD '16*, pages 855–864, New York, New York, USA. ACM Press.

Gupta, M., Li, R., Yin, Z., and Han, J. (2010). Survey on social tagging techniques. *ACM SIGKDD Explorations Newsletter*, 12(1):58.

Han, B., Rahimi, A., Derczynski, L., and Baldwin, T. (2016). Twitter Geolocation Prediction Shared Task of the 2016 Workshop on Noisy User-generated Text. *Proceedings of the 2nd Workshop on Noisy User-generated Text (WNUT)*, pages 213–217.

Holme, P. and Saramäki, J. (2012). Temporal networks. *Physics Reports*, 519(3):97–125.

Hovy, D., Plank, B., and Søgaard, A. (2014a). Experiments with crowdsourced re-annotation of a POS tagging data set. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 377–382, Baltimore, Maryland. Association for Computational Linguistics.

Hovy, D., Plank, B., and Søgaard, A. (2014b). When POS data sets dont add up: Combatting sample bias. *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC-2014)*.

Howison, J., Wiggins, A., and Crowston, K. (2011). Validity Issues in the Use of Social Network Analysis with Digital Trace Data. *Journal of the association of information systems*, 12(12):767–797.

Hulten, G., Spencer, L., and Domingos, P. (2001). Mining time-changing data streams. In *Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining - KDD '01*, pages 97–106, New York, New York, USA. ACM Press.

Hutto, C. J. and Gilbert, E. (2014). Vader: A parsimonious rule-based model for sentiment analysis of social media text. In *International AAAI Conference on Web and Social Media*, Ann Arbor, Michigan, USA.

International Committee of Medical Journal Editors (2018). Defining the Role of Authors and Contributors.

Joachims, T. (2003). Transductive Learning via Spectral Graph Partitioning. In *Proceedings of the Twentieth International Conference on International Conference on Machine Learning*, ICML'03, pages 290–297. AAAI Press.

Johannsen, A., Hovy, D., Martínez Alonso, H., Plank, B., and Søgaard, A. (2014). More or less supervised supersense tagging of Twitter. In *Proceedings of the Third Joint Conference on Lexical and Computational Semantics (*SEM 2014)*, pages 1–11, Stroudsburg, PA, USA. Association for Computational Linguistics and Dublin City University.

Katz, J. and Martin, B. R. (1997). What is research collaboration? *Research Policy*, 26(1):1–18.

Kim, J. and Diesner, J. (2016). Distortive effects of initial-based name disambiguation on measurements of large-scale coauthorship networks. *Journal of the Association for Information Science and Technology*, 67(6):1446–1461.

Kim, J. and Diesner, J. (2017). Over-time measurement of triadic closure in coauthorship networks. *Social Network Analysis and Mining*, 7(1):9.

Kim, Y. (2014). Convolutional Neural Networks for Sentence Classification. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1746–1751, Stroudsburg, PA, USA. Association for Computational Linguistics.

Kiritchenko, S., Zhu, X., and Mohammad, S. M. (2014). Sentiment analysis of short informal texts. *Journal of Artificial Intelligence Research*, 50:723–762.

Kirkpatrick, J., Pascanu, R., Rabinowitz, N., Veness, J., Desjardins, G., Rusu, A. A., Milan, K., Quan, J., Ramalho, T., Grabska-Barwinska, A., Hassabis, D., Clopath, C., Kumaran, D., and Hadsell, R. (2017). Overcoming catastrophic forgetting in neural networks. *Proceedings of the National Academy of Sciences of the United States of America*, 114(13):3521–3526.

Kosinski, M., Matz, S. C., Gosling, S. D., Popov, V., and Stillwell, D. (2015). Facebook as a research tool for the social sciences: Opportunities, challenges, ethical considerations, and practical guidelines. *American Psychologist*, 70(6):543–556.

Kouloumpis, E., Wilson, T., and Moore, J. (2011). Twitter Sentiment Analysis: The Good the Bad and the OMG! *Fifth International AAAI Conference on Weblogs and Social Media*.

Kuhn, T. S. (1970). *The Structure of Scientific Revolutions*, volume II. University of Chicago Press.

Kullback, S. and Leibler, R. A. (1951). On Information and Sufficiency. *The Annals of Mathematical Statistics*, 22(1):79–86.

Kwak, H., Lee, C., Park, H., and Moon, S. (2010). What is Twitter, a social network or a news media? In *Proceedings of the 19th international conference on World wide web - WWW '10*, page 591, New York, New York, USA. ACM Press.

Lafferty, J., McCallum, A., and Pereira, F. C. N. (2001). Conditional random fields: Probabilistic models for segmenting and labeling sequence data.

Lazer, D., Pentland, A., Adamic, L., Aral, S., Barabasi, A.-L., Brewer, D., Christakis, N., Contractor, N., Fowler, J., Gutmann, M., Jebara, T., King, G., Macy, M., Roy, D., and Van Alstyne, M. (2009). Social science. Computational social science. *Science (New York, N.Y.)*, 323(5915).

Leimu, R. and Koricheva, J. (2005). Does Scientific Collaboration Increase the Impact of Ecological Articles? *BioScience*, 55(5):438–443.

Liang, P. (2005). *Semi-supervised learning for natural language*. PhD thesis.

Lin, D. and Wu, X. (2009). Phrase clustering for discriminative learning. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 2 - ACL-IJCNLP '09*, volume 2, page 1030, Morristown, NJ, USA. Association for Computational Linguistics.

Liu, B. (2012). *Sentiment Analysis and Opinion Mining*, volume 5.

Liu, J., Shang, J., Wang, C., Ren, X., and Han, J. (2015). Mining Quality Phrases from Massive Text Corpora. *Proceedings of the 2015 ACM SIGMOD International Conference on Management of Data*, pages 1729–1744.

Liu, Y., Zhu, Y., Che, W., Qin, B., Schneider, N., and Smith, N. A. (2018). Parsing Tweets into Universal Dependencies. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 965–975, New Orleans, Louisiana. Association for Computational Linguistics.

Lofi, C., Selke, J., and Balke, W.-T. (2012). Information Extraction Meets Crowdsourcing: A Promising Couple. *Datenbank-Spektrum*, 12(2):109–120.

Masud, M. M., Chen, Q., Khan, L., Aggarwal, C., Gao, J., Han, J., and Thuraisingham, B. (2010). Addressing concept-evolution in concept-drifting data streams. In *Proceedings - IEEE International Conference on Data Mining, ICDM*, pages 929–934. IEEE.

Maynard, D., Bontcheva, K., and Rout, D. (2012). Challenges in developing opinion mining tools for social media. *LREC 2012 Workshop @NLP can u tag #usergeneratedcontent*, page 8.

McInnes, L., Healy, J., and Melville, J. (2018a). UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction.

McInnes, L., Healy, J., Saul, N., and Großberger, L. (2018b). UMAP: Uniform Manifold Approximation and Projection. *Journal of Open Source Software*, 3(29):861.

Merton, R. K. (1968). The Matthew Effect in Science: The reward and communication systems of science are considered. *Science*, 159(3810):56–63.

Mikolov, T., Chen, K., Corrado, G., and Dean, J. (2013a). Efficient Estimation of Word Representations in Vector Space.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G., and Dean, J. (2013b). Distributed Representations of Words and Phrases and their Compositionality. *Proceedings of the 26th International Conference on Neural Information Processing Systems*, pages 3111–3119.

Miller, G. (2011). Sociology. Social scientists wade into the tweet stream. *Science (New York, N.Y.)*, 333(6051).

Miller, S., Guinness, J., and Zamanian, A. (2004). Name tagging with word clusters and discriminative training. In *In Proceedings of 2004 Human Language Technology conference / North American chapter of the Association for Computational Linguistics annual meeting (HLT-NAACL 2004)*, volume 4, pages 337–342.

Mintz, M., Bills, S., Snow, R., and Jurafsky, D. (2009). Distant supervision for relation extraction without labeled data. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 2 - Volume 2*, pages 1003–1011, Suntec, Singapore. Association for Computational Linguistics.

Mishra, S. (2017). SCTG: Social Communications Temporal Graph A novel approach to visualize temporal communication graphs from social data. In *UIUC Data Science Day*.

Mishra, S. (2019a). Multi-dataset-multi-task Neural Sequence Tagging for Information Extraction from Tweets. In *Proceedings of the 30th ACM Conference on Hypertext and Social Media - HT '19*, pages 283–284, New York, New York, USA. ACM Press.

Mishra, S. (2019b). Trained models for multi-task multi-dataset learning for sequence prediction in tweets.

Mishra, S. (2019c). Trained models for multi-task multi-dataset learning for text classification as well as sequence tagging in tweets.

Mishra, S. (2019d). Trained models for multi-task multi-dataset learning for text classification in tweets.

Mishra, S. (2019e). Wikipedia category embeddings - Node2Vec, Poincare, Elmo.

Mishra, S., Agarwal, S., Guo, J., Phelps, K., Picco, J., and Diesner, J. (2014). Enthusiasm and support. In *Proceedings of the 2014 ACM conference on Web science - WebSci '14*, pages 261–262, New York, New York, USA. ACM Press.

Mishra, S., Agarwal, S., Guo, J., Phelps, K., Picco, J., and Diesner, J. (2019). Tweet IDs annotated for enthusiasm and support towards social causes: CTE, cyberbullying, and LGBT.

Mishra, S. and Diesner, J. (2016). Semi-supervised Named Entity Recognition in noisy-text. In *Proceedings of the 2nd Workshop on Noisy User-generated Text (WNUT)*, pages 203–212, Osaka, Japan. The COLING 2016 Organizing Committee.

Mishra, S. and Diesner, J. (2018). Detecting the Correlation between Sentiment and User-level as well as Text-Level Meta-data from Benchmark Corpora. In *Proceedings of the 29th on Hypertext and Social Media - HT '18*, pages 2–10, New York, New York, USA. ACM Press.

Mishra, S. and Diesner, J. (2019). Capturing Signals of Enthusiasm and Support Towards Social Issues from Twitter. In *Proceedings of the 5th International Workshop on Social Media World Sensors - SIdEWayS'19*, pages 19–24, New York, New York, USA. ACM Press.

Mishra, S., Diesner, J., Byrne, J., and Surbeck, E. (2015). Sentiment Analysis with Incremental Human-in-the-Loop Learning and Lexical Resource Customization. In *Proceedings of the 26th ACM Conference on Hypertext & Social Media - HT '15*, pages 323–325, New York, New York, USA. ACM Press.

Mishra, S., Fegley, B. D., Diesner, J., and Torvik, V. I. (2018a). Expertise as an aspect of author contributions. In *WORKSHOP ON INFORMETRIC AND SCIENTOMETRIC RESEARCH (SIG/MET)*, Vancouver.

Mishra, S., Fegley, B. D., Diesner, J., and Torvik, V. I. (2018b). Self-citation analysis data based on PubMed Central subset (2002-2005).

Mishra, S., Fegley, B. D., Diesner, J., and Torvik, V. I. (2018c). Self-citation is the hallmark of productive authors, of any gender. *PLOS ONE*, 13(9):e0195773.

Mishra, S. and Mishra, S. (2019). 3Idiots at HASOC 2019: Fine-tuning Transformer Neural Networks for Hate Speech Identification in Indo-European Languages. In *Proceedings of the 11th annual meeting of the Forum for Information Retrieval Evaluation*.

Mishra, S., Prasad, S., and Mishra, S. (2020). Multilingual Joint Fine-tuning of Transformer models for identifying Trolling,Aggression and Cyberbullying at TRAC 2020. In *Proceedings of the Second Workshop on Trolling, Aggression and Cyberbullying (TRAC-2020)*.

Mishra, S. and Torvik, V. I. (2016). Quantifying Conceptual Novelty in the Biomedical Literature. *D-Lib Magazine*, 22(9/10).

Mishra, S. and Torvik, V. I. (2018). Conceptual novelty scores for PubMed articles.

Mohammad, S., Bravo-Marquez, F., Salameh, M., and Kiritchenko, S. (2018). SemEval-2018 Task 1: Affect in Tweets. In *Proceedings of The 12th International Workshop on Semantic Evaluation*, pages 1–17, Stroudsburg, PA, USA. Association for Computational Linguistics.

Mohammad, S. M., Kiritchenko, S., and Zhu, X. (2013). NRC-Canada: Building the State-of-the-Art in Sentiment Analysis of Tweets. *Proceedings of the seventh international workshop on Semantic Evaluation Exercises (SemEval-2013)*, 2(SemEval):321–327.

Mohammad, S. M., Sobhani, P., and Kiritchenko, S. (2017). Stance and Sentiment in Tweets. *ACM Transactions on Internet Technology*, 17(3):1–23.

Morgan, P. P. (1985). Originality, novelty and priority: Three words to reckon with in scientific publishing. *Canadian Medical Association Journal*, 132(1):8–9.

Mozetič, I., Grčar, M., Smailović, J., Alani, H., Mozetič, I., and Scala, A. (2016). Multilingual Twitter Sentiment Classification: The Role of Human Annotators. *PLOS ONE*, 11(5):e0155036.

Nakov, P., Ritter, A., Rosenthal, S., Sebastiani, F., and Stoyanov, V. (2016a). SemEval-2016 Task 4: Sentiment Analysis in Twitter. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*, pages 1–18, Stroudsburg, PA, USA. Association for Computational Linguistics.

Nakov, P., Rosenthal, S., Kiritchenko, S., Mohammad, S. M., Kozareva, Z., Ritter, A., Stoyanov, V., and Zhu, X. (2016b). Developing a successful SemEval task in sentiment analysis of Twitter and other social media texts. *Language Resources and Evaluation*, 50(1):35–65.

Nakov, P., Rosenthal, S., Ritter, A., and Wilson, T. (2013). SemEval-2013 Task 2: Sentiment Analysis in Twitter. *Proceedings of the International Workshop on Semantic Evaluation (SemEval-2013)*, 2(SemEval):312–320.

Newman, M. E. J. (2001). The structure of scientific collaboration networks. *Proceedings of the National Academy of Sciences*, 98(2):404–409.

Newman, M. E. J. (2004). Coauthorship networks and patterns of scientific collaboration. *Proceedings of the National Academy of Sciences*, 101(Supplement 1):5200–5205.

Nicholson, J. M. and Ioannidis, J. P. A. (2012). Conform and be funded. *Nature*, 492(7427):34–36.

Nickel, M. and Kiela, D. (2017). Poincaré Embeddings for Learning Hierarchical Representations. In Guyon, I., Luxburg, U. V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., and Garnett, R., editors, *Advances in Neural Information Processing Systems 30*, pages 6338–6347. Curran Associates, Inc.

Nivre, J., de Marneffe, M.-C., Ginter, F., Goldberg, Y., Hajic, J., Manning, C. D., McDonald, R., Petrov, S., Pyysalo, S., Silveira, N., Tsarfaty, R., and Zeman, D. (2016). Universal Dependencies v1: A Multilingual Treebank Collection. In Chair), N. C. C., Choukri, K., Declerck, T., Goggi, S., Grobelnik, M., Maegaard, B., Mariani, J., Mazo, H., Moreno, A., Odijk, J., and Piperidis, S., editors, *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, Paris, France. European Language Resources Association (ELRA).

Okazaki, N. (2007). CRFsuite: a fast implementation of Conditional Random Fields (CRFs).

Owoputi, O., O'Connor, B., Dyer, C., Gimpel, K., and Schneider, N. (2012). Part-of-Speech Tagging for Twitter: Word Clusters and Other Advances. *Cmu-Ml-12-107*.

Owoputi, O., O'Connor, B., Dyer, C., Gimpel, K., Schneider, N., and Smith, N. a. (2013). Improved Part-of-Speech Tagging for Online Conversational Text with Word Clusters. *Proceedings of NAACL-HLT 2013*, (June):380–390.

Packalen, M. and Bhattacharya, J. (2015a). Age and the trying out of new ideas. Technical report.

Packalen, M. and Bhattacharya, J. (2015b). New Ideas in Invention. *National Bureau of Economic Research Working Paper Series*, No. 20922.

Page, L., Brin, S., Motwani, R., and Winograd, T. (1998). The PageRank Citation Ranking: Bringing Order to the Web.

Pak, A. and Paroubek, P. (2010). Twitter as a Corpus for Sentiment Analysis and Opinion Mining. In *Proceedings of the Seventh conference on International Language Resources and Evaluation (LREC'10)*, Valletta, Malta. European Languages Resources Association (ELRA).

Pang, B. and Lee, L. (2008). Opinion Mining and Sentiment Analysis. *Foundations and Trends® in Information Retrieval*, 2(12):1–135.

Pang, B., Lee, L., and Vaithyanathan, S. (2002). Thumbs up? Sentiment Classification using Machine Learning Techniques. In *Proceedings of the ACL-02 conference on Empirical methods in natural language processing - EMNLP '02*, volume 10, pages 79–86, Morristown, NJ, USA. Association for Computational Linguistics.

Pennington, J., Socher, R., and Manning, C. D. (2014). GloVe: Global Vectors for Word Representation. *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*, pages 1532–1543.

Peters, M., Ammar, W., Bhagavatula, C., and Power, R. (2017). Semi-supervised sequence tagging with bidirectional language models. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*, pages 1756–1765, Stroudsburg, PA, USA. Association for Computational Linguistics.

Peters, M., Neumann, M., Iyyer, M., Gardner, M., Clark, C., Lee, K., and Zettlemoyer, L. (2018). Deep Contextualized Word Representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 2227–2237, Stroudsburg, PA, USA. Association for Computational Linguistics.

Plank, B., Hovy, D., McDonald, R., and Søgaard, A. (2014). Adapting taggers to Twitter with not-so-distant supervision. *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, pages 1783–1792.

Pontiki, M., Galanis, D., Papageorgiou, H., Manandhar, S., and Androutsopoulos, I. (2015). SemEval-2015 Task 12: Aspect Based Sentiment Analysis. In *Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015)*, pages 486–495, Stroudsburg, PA, USA. Association for Computational Linguistics.

Pontius, R. G. and Millones, M. (2011). Death to Kappa: birth of quantity disagreement and allocation disagreement for accuracy assessment. *International Journal of Remote Sensing*, 32(15):4407–4429.

Preotiuc-Pietro, D., Carpenter, J., Giorgi, S., and Ungar, L. (2016). Studying the Dark Triad of Personality through Twitter Behavior. In *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management - CIKM '16*, pages 761–770, New York, New York, USA. ACM Press.

Preoiuc-Pietro, D., Liu, Y., Hopkins, D., and Ungar, L. (2017). Beyond Binary Labels: Political Ideology Prediction of Twitter Users. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 729–740, Stroudsburg, PA, USA. Association for Computational Linguistics.

Ratinov, L. and Roth, D. (2009). Design challenges and misconceptions in named entity recognition.

Reichheld, F. F. (2003). The one number you need to grow. *Harvard business review*, 81(12):46–54.

Reitz, F. (2010). A Framework for an Ego-centered and Time-aware Visualization of Relations in Arbitrary Data Repositories.

Rennie, D. and Flanagin, A. (1994). Authorship! Authorship! Guests, Ghosts, Grafters, and the Two-Sided Coin. *JAMA*, 271(6):469.

Rennie, D., Yank, V., and Emmanuel, L. (1997). When Authorship Fails. *Jama*, 278(7):579.

Rezapour, R. and Diesner, J. (2017). Classification and Detection of Micro-Level Impact of Issue-Focused Documentary Films based on Reviews. In *Proceedings of the 2017 ACM Conference on Computer Supported Cooperative Work and Social Computing - CSCW '17*, pages 1419–1431, New York, New York, USA. ACM Press.

Riloff, E., Qadir, A., Surve, P., Silva, L. D., Gilbert, N., and Huang, R. (2013). Sarcasm as Contrast between a Positive Sentiment and Negative Situation. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 704–714, Seattle, Washington, USA. Association for Computational Linguistics.

Ritter, A., Clark, S., and Etzioni, O. (2011). Named entity recognition in tweets: an experimental study. In *Proceedings of Emperical Methods for Natural Langauge Processing*, pages 1524–1534.

Rizzo, G., Erp, M. v., Plu, J., and Troncy, R. (2016). Making Sense of Microposts (#Microposts2016) Named Entity rEcognition and Linking (NEEL) Challenge. In *Workshop on Making Sense of Microposts (#Microposts2016)*, Montréal.

Rosen-Zvi, M., Griffiths, T., Steyvers, M., and Smyth, P. (2004). The author-topic model for authors and documents. In *Proceedings of the 20th conference on Uncertainty in artificial intelligence*, UAI '04, page 487494, Arlington, VA, USA. AUAI Press.

Rosenthal, S., Nakov, P., Kiritchenko, S., Mohammad, S., Ritter, A., and Stoyanov, V. (2015). SemEval-2015 Task 10: Sentiment Analysis in Twitter. In *Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015)*, pages 451–463, Stroudsburg, PA, USA. Association for Computational Linguistics.

Rosso, P., Rangel, F., Potthast, M., Stamatatos, E., Tschuggnall, M., and Stein, B. (2016). Overview of PAN16: New challenges for authorship analysis: Cross-genre profiling, clustering, diarization, and obfuscation. In *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, volume 9822, pages 332–350. Springer, Cham.

Ruder, S. (2017). An Overview of Multi-Task Learning in Deep Neural Networks. *CoRR*, abs/1706.0.

Sarawagi, S. (2008). Information extraction. *Foundation and Trends in Databases*, 1(3):261–377.

Schneider, N. and Smith, N. A. (2015). A Corpus and Model Integrating Multiword Expressions and Supersenses. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1537–1547, Denver, Colorado. Association for Computational Linguistics.

Schoenmakers, W. and Duysters, G. (2010). The technological origins of radical inventions. *Research Policy*, 39(8):1051–1059.

Settles, B. (2009). Active Learning Literature Survey. Technical report, University of Wisconsin–Madison.

Settles, B. and Craven, M. (2008). An analysis of active learning strategies for sequence labeling tasks.

Settles, B., Craven, M., and Friedland, L. (2008). Active Learning with Real Annotation Costs. In *Proceedings of the NIPS Workshop on Cost-Sensitive Learning*.

Shen, H.-W. and Barabási, A.-L. (2014). Collective credit allocation in science. *Proceedings of the National Academy of Sciences of the United States of America*, 111(34):12325–30.

Shi, L., Wang, C., Wen, Z., Qu, H., Lin, C., and Liao, Q. (2015). 1.5D Egocentric Dynamic Network Visualization. *IEEE Transactions on Visualization and Computer Graphics*, 21(5):624–637.

Shneiderman, B. (1996). The eyes have it: a task by data type taxonomy for information visualizations. In *Proceedings 1996 IEEE Symposium on Visual Languages*, pages 336–343. IEEE Comput. Soc. Press.

Sinha, A., Shen, Z., Song, Y., Ma, H., Eide, D., Hsu, B.-J. P., and Wang, K. (2015). An Overview of Microsoft Academic Service (MAS) and Applications. In *Proceedings of the 24th International Conference on World Wide Web Companion (WWW 2015 Companion)*, pages 243–246, New York, New York, USA. ACM Press.

Sobhani, P., Mohammad, S., and Kiritchenko, S. (2016). Detecting Stance in Tweets And Analyzing its Interaction with Sentiment. In *Proceedings of the Fifth Joint Conference on Lexical and Computational Semantics*, pages 159–169, Stroudsburg, PA, USA. Association for Computational Linguistics.

Socher, R., Chen, D., Manning, C. D., and Ng, A. (2013). Reasoning With Neural Tensor Networks for Knowledge Base Completion. In *Proceedings of the 26th International Conference on Neural Information Processing Systems - Volume 1*, pages 926–934, Lake Tahoe, Nevada. Curran Associates Inc.

Søgaard, A. and Goldberg, Y. (2016). Deep multi-task learning with low level tasks supervised at lower layers. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 231–235. Association for Computational Linguistics.

Srivastava, N., Hinton, G. E., Krizhevsky, A., Sutskever, I., and Salakhutdinov, R. (2014). Dropout : A Simple Way to Prevent Neural Networks from Overfitting. *Journal of Machine Learning Research (JMLR)*, 15.

Stephan, P., Wang, J., and Veugelers, R. (2015). Bias against novelty in science: A cautionary tale for users of bibliometric indicators.

Steyvers, M., Smyth, P., Rosen-Zvi, M., and Griffiths, T. (2004). Probabilistic author-topic models for information discovery. In *Proceedings of the 2004 ACM SIGKDD international conference on Knowledge discovery and data mining - KDD '04*, number 1990 in KDD '04, page 306, New York, New York, USA. ACM Press.

Strauss, B., Toma, B., Ritter, A., Marneffe, M.-C. d., and Xu, W. (2016). Results of the WNUT16 Named Entity Recognition Shared Task. *Proceedings of the 2nd Workshop on Noisy User-generated Text (WNUT)*, pages 138–144.

Sun, Y. and Han, J. (2012). Mining Heterogeneous Information Networks: Principles and Methodologies. *Synthesis Lectures on Data Mining and Knowledge Discovery*, 3(2):1–159.

Swamy, S., Ritter, A., and de Marneffe, M.-C. (2017). "i have a feeling trump will win.....": Forecasting Winners and Losers from User Predictions on Twitter. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1583–1592, Stroudsburg, PA, USA. Association for Computational Linguistics.

Tan, C., Lee, L., Tang, J., Jiang, L., Zhou, M., and Li, P. (2011). User-level sentiment analysis incorporating social networks. In *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining - KDD '11*, page 1397, New York, New York, USA. ACM Press.

Torvik, V. I. and Smalheiser, N. R. (2009). Author Name Disambiguation in MEDLINE. *ACM transactions on knowledge discovery from data*, 3(3):1–29.

Torvik, V. I. and Smalheiser, N. R. (2018). Author-ity 2009 - PubMed author name disambiguated dataset.

Trapido, D. (2015). How novelty in knowledge earns recognition: The role of consistent identities. *Research Policy*, 44(8):1488–1500.

Tumasjan, A., Sprenger, T., Sandner, P., and Welpe, I. (2010). Predicting Elections with Twitter: What 140 Characters Reveal about Political Sentiment. In *International AAAI Conference on Web and Social Media*.

Turian, J., Ratinov, L., and Bengio, Y. (2010). Word representations : A simple and general method for semi-supervised learning. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics (ACL-10)*, number July, pages 384–394. Association for Computational Linguistics.

Uzzi, B., Mukherjee, S., Stringer, M., and Jones, B. (2013). Atypical Combinations and Scientific Impact. *Science*, 342(6157):468–472.

Vosoughi, S., Zhou, H., and Roy, D. (2015). Enhanced Twitter Sentiment Classification Using Contextual Information. In *Proceedings of the 6th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis*, pages 16–24, Stroudsburg, PA, USA. Association for Computational Linguistics.

Waseem, Z. and Hovy, D. (2016). Hateful Symbols or Hateful People? Predictive Features for Hate Speech Detection on Twitter. In *Proceedings of the NAACL Student Research Workshop*, pages 88–93, Stroudsburg, PA, USA. Association for Computational Linguistics.

Wilson, R. E., Gosling, S. D., and Graham, L. T. (2012). A Review of Facebook Research in the Social Sciences. *Perspectives on Psychological Science*, 7(3):203–220.

Wilson, T., Wiebe, J., and Hoffmann, P. (2005). Recognizing contextual polarity in phrase-level sentiment analysis. In *Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing - HLT '05*, volume 7, pages 347–354, Morristown, NJ, USA. Association for Computational Linguistics.

Wulff, J. and Søgaard, A. (2015). Learning finite state word representations for unsupervised Twitter adaptation of POS taggers. In *Proceedings of the Workshop on Noisy User-generated Text*, pages 162–166, Beijing, China. Association for Computational Linguistics.

Xing, W. and Ghorbani, A. (2004). Weighted PageRank algorithm. In *Proceedings. Second Annual Conference on Communication Networks and Services Research, 2004.*, pages 305–314. IEEE.

Yank, V. and Rennie, D. (1999). Disclosure of researcher contributions: A study of original research articles in the Lancet. *Annals of Internal Medicine*, 130(8):661–670.

Youn, H., Strumsky, D., Bettencourt, L. M. A., and Lobo, J. (2015). Invention as a combinatorial process: evidence from US patents. *Journal of the Royal Society, Interface / the Royal Society*, 12(106):20150272–.

Zhang, Q., Fu, J., Liu, X., and Huang, X. (2018). Adaptive Co-attention Network for Named Entity Recognition in Tweets.

Zhu, X. (2008). Semi-Supervised Learning Literature Survey. Technical report, Computer Sciences, University of Wisconsin-Madison.

Zhu, X. and Goldberg, A. B. (2009). Introduction to Semi-Supervised Learning. *Synthesis Lectures on Artificial Intelligence and Machine Learning*, 3(1):1–130.

Zubiaga, A., Liakata, M., Procter, R., Hoi, G. W. S., and Tolmie, P. (2016a). PHEME rumour scheme dataset: journalism use case.

Zubiaga, A., Liakata, M., Procter, R., Wong Sak Hoi, G., and Tolmie, P. (2016b). Analysing How People Orient to and Spread Rumours in Social Media by Looking at Conversational Threads. *PLOS ONE*, 11(3):e0150989.

Zuckerman, H. (1987). Citation analysis and the complex problem of intellectual influence. *Scientometrics*, 12(5-6):329–338.

Zuckerman, H. A. (1968). Patterns of Name Ordering Among Authors of Scientific Papers: A Study of Social Symbolism and Its Ambiguity. *American Journal of Sociology*, 74(3):276–291.