Stony Brook University

Statistical Models for Linguistic Variation in Online Media

A Dissertation Presented

by

Vivek Kulkarni

to

The Graduate School

in Partial Fulfillment of the Requirements

for the Degree of

Doctor of Philosophy

in

Computer Science

Stony Brook University

May 2017

Stony Brook University

The Graduate School

Vivek Kulkarni

We, the dissertation committee for the above candidate for the Doctor of Philosophy degree, hereby recommend acceptance of this dissertation.

Steven Skiena – Dissertation Advisor
Distinguished Teaching Professor, Department of Computer Science

Niranjan Balasubramanian – Chairperson of Defense
Assistant Professor, Department of Computer Science

H. Andrew Schwartz
Assistant Professor, Department of Computer Science

David Bamman
Assistant Professor
University of Berkeley, California

This dissertation is accepted by the Graduate School.

Charles Taber
Dean of the Graduate School

Abstract of the Dissertation

Statistical Models for Linguistic Variation in Online Media

by

Vivek Kulkarni

Doctor of Philosophy

in

Computer Science

Stony Brook University

2017

Language on the Internet and social media varies due to time, geography, and social factors. For example, consider an online chat forum where people from different regions across the world interact. In such scenarios, it is important to track and detect regional variation in language. A person from the UK, who is in conversation with someone from the USA could say *"he is stuck in the lift"* to mean *"he is stuck in an elevator"*, since the word `lift` means an `elevator` in the UK. Note that in the US, `lift` does not refer to an `elevator`. Modeling such variation can allow for applications to prompt or suggest the intended meaning to the other participants of the conversation.

In this thesis, we conduct two related lines of inquiry focusing on (a) language itself and the variation it manifests and (b) the user and what we can infer about them based on their language use on social media.

First, we develop computational methods to track and detect

changes in word usage, including semantic and syntactic variation. We examine three modalities: time, geography and domains. Specifically, we outline methods to use distributional word representations (word embeddings) to detect semantic variation in word usage. Our methods are scalable to large datasets, making them particularly suited for social media. Second, we turn our attention towards users. In particular, we model latent traits of users based on their everyday language use on social media. We develop latent factor models, that explicitly seek to build representations of each user based on their inferred latent traits. These models capture latent traits that serve as useful co-variates for a wide variety of tasks like predicting what topics users like on social media and the number of friends in their social circle.

This work has broad applications in several fields like information retrieval, semantic web applications, socio-variational linguistics, and computational social science including digital health care and ad-targeting.

To my parents

Contents

List of Figures

List of Tables

Acknowledgements

My journey to a PhD has been an exciting and fulfilling one. During this undertaking, several people helped me along the way without whom this journey would not have been possible.

First, I would like to thank my advisor Steven Skiena. Using a unique recipe comprising of just the right proportions of healthy skepticism and intellectual freedom, he has influenced my research outlook profoundly. It was during my discussions with Steve, that I was forced to distill my research ideas, evaluate them critically and ask the right research questions – lessons that gently guided me onto the path of excellent research, for which I am forever grateful to Steve.

During the course of my PhD, I have been fortunate to have great collaborators. I would like to thank Prof. H. Andrew Schwartz for the very insightful research discussions and exceptional mentoring. His unique inter-disciplinary perspectives have undoubtedly shaped my research directions and this thesis. It is through my extensive interactions with Andy that I have grown to develop an active research interest at the intersection of natural language processing and computational social science. I would like to thank Rami Al-Rfou' and Bryan Perozzi for the numerous paper writing sprints and the fun brunches at "The Toast". I would also like to thank all my other collaborators at our lab (Yingtao, Hao Chen, and Junting Ye and many others). I also thank my many mentors and colleagues during my summer internships at Google! and Yahoo Labs! (including Tanya, Ismail and Yashar). I also thank my larger network of colleagues at Stony Brook namely the NLP Coffee hour gang and HLAB colleagues (including Niranjan Balasubramanian, Veronica Lynn, Young Seo, Mohammad Zamani, Fatemeh and many others).

I would like to take a moment to thank the funding agencies that enabled my research. Specifically I would like to thank Renaissance Technologies for their generous fellowship which enabled me to present my work at many more venues. I also thank NSF and Google for their grants awarded to Steve which enabled the research described in this thesis.

I would also like to thank all my friends in graduate school (including Amogh Akshintala, Shikha Singh and Hirak Sarkar). I especially thank Alok Katiyar,

Nitin Rastogi, Sagardeep Mahapatra for enriching my summer experience in the Silicon Valley and allowing me to crash at their apartments all summer. I also thank my friends Devashish Thakur, Kaushik Devarajiah and Parikshit Bhattacharjee for all the fun times (It was fun having Sunday brunches at IHOP with you!). I specially thank Abhradeep Guha Thakurta for encouraging me to pursue graduate school, and for all the insightful discussions.

Finally I owe all of this to my family, who believed in my abilities and without whose love and support I would never have undertaken this journey.

Chapter 1

Introduction

All models are wrong, but some are useful.

George E.P Box

The Internet is global. Half of the world's population has access to the Internet and more than one-third use on-line social media regularly.[1] This wide-spread use now enables researchers to analyze human language at an unprecedented scale and resolution. Moreover, language on social media is increasingly personal, and reflects the thoughts and emotions of users enabling researchers to analyze human behavior at a scale that was not previously possible using traditional methods like questionnaires and surveys.

Consequently, computational text analysis methods provide one approach to understanding textual content (natural language) on the Internet and enable applications like Google Now, Amazon Alexa and Microsoft Cortana. These include methods for tasks like text normalization [1–4], part of speech tagging [5, 6], named entity recognition[7–9], semantic role labeling, discourse analysis and knowledge base construction.

Most of these methods/techniques analyze text by treating it as a standalone entity. However text is associated with a rich and varied context. For example, consider the book title *"The gay science" by Nietsche*. A natural language processing (NLP) system might incorrectly conclude that the book is about homo-sexuality unless it incorporates the knowledge that the book was written in 1900's when the word `gay` meant `cheerful`. More generally, language use on the Internet displays rich *variation* across multiple dimensions: time, geography, domains, and social variables like age, sex and ethnicity.

[1] These statistics are as of January 2017 based on http://www.smartinsights.com/social-media-marketing/social-media-strategy/new-global-social-media-research/

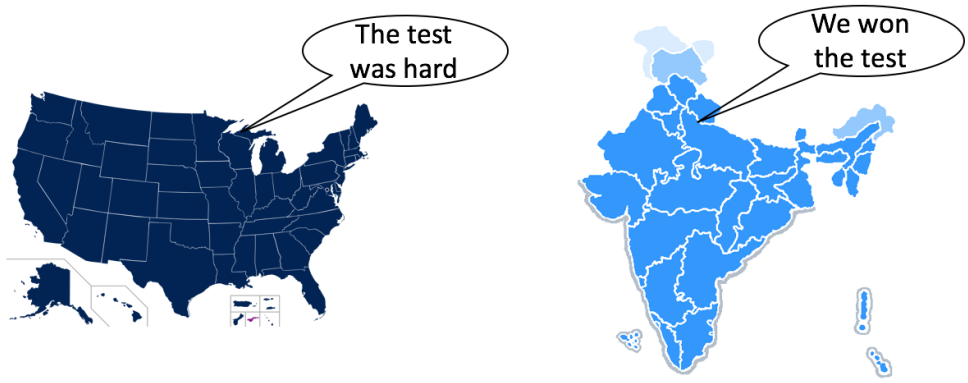


Figure 1.1: An illustration of the differing semantics associated with the word `test` between the US and India. While in the US, `test` dominantly refers to an exam, in India `test` can also refer to a game of cricket.(*Image source for maps: http://wikimedia.commons.org)*

In this thesis, we explore this variety of human language on the Internet through the lens of natural language processing, machine learning and computational social science and build models to incorporate such contextual information to improve natural language understanding. We propose statistical models that can track language variation across multiple modalities: time, geography and domains. Our models can uncover that words like `gay` or `tape` accquired new senses over time or that `test` can refer to a cricket match in India (see Figure 1.1). As a concrete application, we demonstrate that incorporating un-covered domain specific linguistic cues can boost performance on the task of Named Entity Recognition in a domain-adaptation setting. Finally, we turn our attention towards the language of individual users on social media. With the evolution of social media, it is now possible to perform large scale analyses of human behavior based on their everyday language use. Analyzing the language of social media users is a rich playground for computational social science research and can potentially provide insights into their emotions/sentiments, demographic variables and even their personality. In this vein, we attempt to characterize latent traits revealed by such *everyday language use* of people on social media. and demonstrate that our inferred latent traits are generalizable to a wide variety of predictive tasks (like predicting Income, IQ etc) and stable across time.

1.1 Thesis Overview

The thesis is broadly structured on two focal points (a) Language and (b) Users. The next three chapters are devoted to tracking and detecting linguistic variation across multiple modalities. The fifth chapter shifts the focus towards users and outlines models to infer latent user traits from their everyday language use on social media that characterize variation in human behavior. Finally we summarize conclusions and outline directions for future work in Chapter 6. These chapters are briefly described below:

Statistically Significant Detection of Linguistic Change over Time We propose a new computational approach for tracking and detecting statistically significant linguistic shifts in the meaning and usage of words. Such linguistic shifts are especially prevalent on the Internet, where the rapid exchange of ideas can quickly change a word's meaning. Our meta-analysis approach constructs property time series of word usage, and then uses statistically sound change point detection algorithms to identify significant linguistic shifts.

We consider and analyze three approaches of increasing complexity to generate such linguistic property time series, the culmination of which uses distributional characteristics inferred from word co-occurrences. Using recently proposed deep neural language models, we first train vector representations of words for each time period. Second, we warp the vector spaces into one unified coordinate system. Finally, we construct a distance-based distributional time series for each word to track its linguistic displacement over time.

We demonstrate that our approach is scalable by tracking linguistic change across years of micro-blogging using Twitter, a decade of product reviews using a corpus of movie reviews from Amazon, and a century of written books using the Google Book-ngrams. Our analysis reveals interesting patterns of language usage change commensurate with each medium.

Quantifying Geographical Variation in Internet Language We present a new computational technique to detect and analyze statistically significant geographic variation in language. While previous approaches have primarily focused on lexical variation between regions, our method identifies words that demonstrate semantic and syntactic variation as well.

We extend recently developed techniques for neural language models to learn word representations which capture differing semantics across geographical regions. In order to quantify this variation and ensure robust detection of true regional differences, we formulate a null model to determine whether observed

changes are statistically significant. Our method is the first such approach to explicitly account for random variation due to chance while detecting regional variation in word meaning.

To validate our model, we study and analyze two different massive online data sets: millions of tweets from Twitter spanning not only four different countries but also fifty states, as well as millions of phrases contained in the Google Book Ngrams. Our analysis reveals interesting facets of language change at multiple scales of geographic resolution – from neighboring states to distant continents.

Linguistic Variation across Domains with Applications to Named Entity Recognition Content on the Internet is heterogeneous and arises from various domains like News, Entertainment, Finance and Technology. Understanding such content requires identifying named entities (persons, places and organizations) as one of the key steps. Traditionally Named Entity Recognition (NER) systems have been built using available annotated datasets (like CoNLL, MUC) and demonstrate excellent performance. However, these models fail to generalize onto other domains like Sports and Finance where conventions and language use can differ significantly. Furthermore, several domains do not have large amounts of annotated labeled data for training robust Named Entity Recognition models. A key step towards this challenge is to adapt models learned on domains where large amounts of annotated training data are available to domains with scarce annotated data.

We propose methods to effectively adapt models learned on one domain onto other domains using distributed word representations. First we analyze the linguistic variation present across domains to identify key linguistic insights that can boost performance across domains. We propose methods to capture domain specific semantics of word usage in addition to global semantics. We then demonstrate how to effectively use such domain specific knowledge to learn NER models that outperform previous baselines in the domain adaptation setting.

Learning Latent User Traits from Language on Social Media We propose a new construct of user traits derived from unprompted language use over a large social media dataset. By deriving user traits from social media like Twitter and Facebook we discover latent human differences based on *everyday behavior* (i.e. language use), an approach that has only recently become viable with the availability of data at a large scale spanning millions of users. Leveraging these developments, we perform a computational analysis of social media text to infer latent traits that distinguish people. We subject our

construct to a comprehensive set of evaluations, establishing their stability (i.e. across time and populations), and generalizability (i.e. ability to predict other psychological attributes and behavior).

1.2 Thesis Statement

This thesis advocates for models that incorporate contextual information associated with language to improve natural language understanding especially on the Internet where language demonstrates rich variation. Language is not a stand-alone entity but is associated with a rich context – the time at which it was generated, the geographical region it is associated with, the domain of usage and ultimately even the user who communicates. We argue that statistical models which incorporate this contextual information can improve natural language understanding and ultimately enable richer, more personalized and user-centric applications.

We look into this perspective by proposing data-driven statistical models that track and detect linguistic variation across time, geography and domains. At their heart, these models capture semantic differences in word usage across these modalities by learning "focused" representations of words that capture word semantics. The methods and models outlined have broad applications to several fields like information retrieval, socio-variational linguistics, semantic web applications, ad-targeting, and personalization etc. In summary, this thesis demonstrates how incorporating contextual information (like time) when analyzing language can reveal linguistic insights and improve textual content understanding on the Internet and social media.

Chapter 2

Statistically Significant Detection of Linguistic Change over Time

Time changes all things; There is no reason why language should escape this universal law.

Ferdinand de Saussure

Natural languages are inherently dynamic, evolving over time to accommodate the needs of their speakers. This effect is especially prevalent on the Internet, where the rapid exchange of ideas can change a word's meaning overnight.

In this chapter, we study the problem of detecting such linguistic shifts on a variety of media including micro-blog posts, product reviews, and books. Specifically, we seek to detect the broadening and narrowing of semantic senses of words, as they continually change throughout the lifetime of a medium.

We propose the first computational approach for tracking and detecting statistically significant linguistic shifts of words. To model the temporal evolution of natural language, we construct a time series per word. We investigate three methods to build our word time series. First, we extract *Frequency* based statistics to capture sudden changes in word usage. Second, we construct *Syntactic* time series by analyzing each word's part of speech (POS)

Work described in this chapter was done in collaboration with Rami Al-Rfou', Bryan Perozzi and Steven Skiena and published at WWW, 2015.

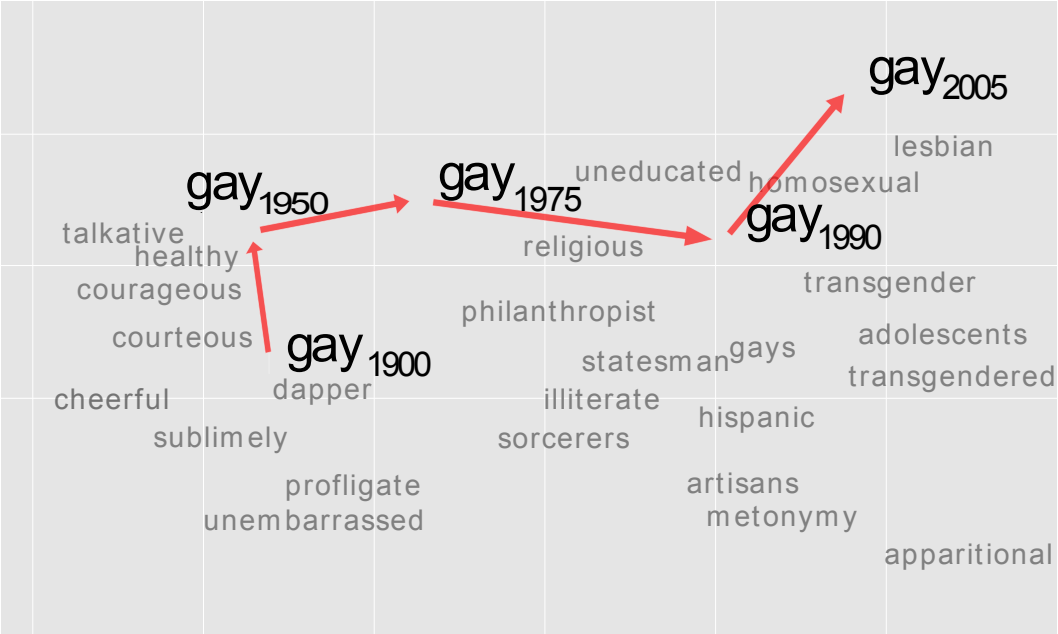


Figure 2.1: A 2-dimensional projection of the latent semantic space captured by our algorithm. Notice the semantic trajectory of the word `gay` transitioning meaning in the space.

tag distribution. Finally, we infer contextual cues from word co-occurrence statistics to construct *Distributional* time series. In order to detect and establish statistical significance of word changes over time, we present a change point detection algorithm, which is compatible with all methods.

Figure 2.1 illustrates a 2-dimensional projection of the latent semantic space captured by our *Distributional* method. We clearly observe the sequence of semantic shifts that the word `gay` has undergone over the last century (1900-2005). Initially, `gay` was an adjective that meant `cheerful` or `dapper`. Observe for the first 50 years, that it stayed in the same general region of the semantic space. However by 1975, it had begun a transition over to its current meaning —a shift which accelerated over the years to come.

The choice of the time series construction method determines the type of information we capture regarding word usage. The difference between frequency-based approaches and distributional methods is illustrated in Figure 2.2. Figure 2.2a shows the frequencies of two words, `Sandy` (red), and `Hurricane` (blue) as a percentage of search queries according to Google Trends[1]. Observe the sharp spikes in both words' usage in October 2012, which corresponds to a

[1] `http://www.google.com/trends/`

storm called `Hurricane Sandy` striking the Atlantic Coast of the United States. However, only one of those words (`Sandy`) actually acquired a new meaning. Indeed, using our distributional method (Figure 2.2b), we observe that only the word `Sandy` shifted in meaning where as `Hurricane` did not.

Our computational approach is scalable, and we demonstrate this by running our method on three large datasets. Specifically, we investigate linguistic change detection across years of micro-blogging using Twitter, a decade of product reviews using a corpus of movie reviews from Amazon, and a century of written books using the Google Books Ngram Corpus.

Despite the fast pace of change of the web content, our method is able to detect the introduction of new products, movies and books. This could help semantically aware web applications to better understand user intentions and requests. Detecting the semantic shift of a word would trigger such applications to apply focused sense disambiguation analysis.

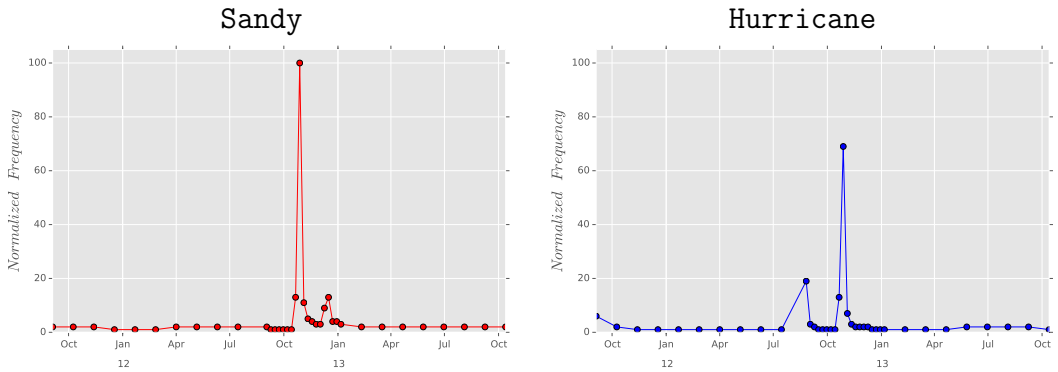
In summary, our contributions are as follows:

- **Word Evolution Modeling**: We study three different methods for the statistical modeling of word evolution over time. We use measures of frequency, part-of-speech tag distribution, and word co-occurrence to construct time series for each word under investigation.(Section 2.2)

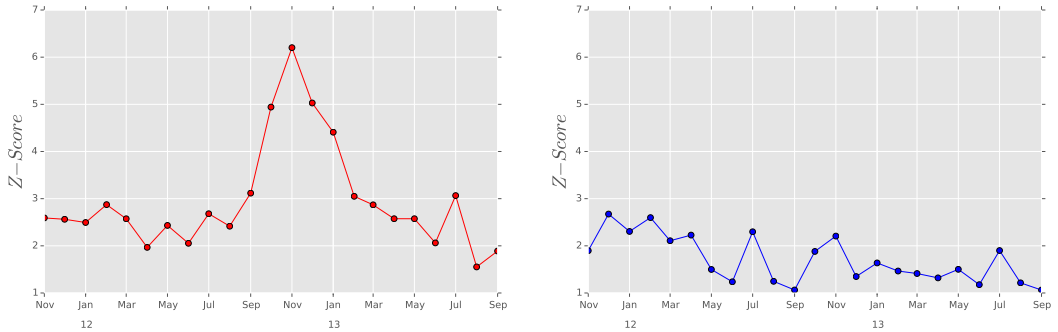
- **Statistical Soundness**: We propose (to our knowledge) the first statistically sound method for linguistic shift detection. Our approach uses change point detection in time series to assign significance of change scores to each word. (Section 2.3)

- **Cross-Domain Analysis**: We apply our method on three different domains; books, tweets and online reviews. Our corpora consists of billions of words and spans several time scales. We show several interesting instances of semantic change identified by our method. (Section 2.5)

The rest of the chapter is structured as follows. In Section 2.1 we define the problem of language shift detection over time. Then, we outline our proposals to construct time series modeling word evolution in Section 2.2. Next, in Section 2.3, we describe the method we developed for detecting significant changes in natural language. We describe the datasets we used in Section 2.4, and then evaluate our system both qualitatively and quantitatively in Section 2.5. We follow this with a treatment of related work in Section 2.6, and finally conclude with a discussion of the limitations and possible future work in Section 2.7.



(a) Frequency method (Google Trends)



(b) Distributional method

Figure 2.2: Comparison between Google Trends and our method. Observe how Google Trends shows spikes in frequency for both `Hurricane` (blue) and `Sandy` (red). Our method, in contrast, models change in usage and detects that only `Sandy` changed its meaning and not `Hurricane`.

2.1 Problem Definition

Our problem is to quantify the linguistic shift in word meaning and usage across time. Given a temporal corpora $\mathcal{C}$ that is created over a time span $\mathcal{S}$, we divide the corpora into n snapshots $\mathcal{C}_t$ each of period length P. We build a common vocabulary $\mathcal{V}$ by intersecting the word dictionaries that appear in all the snapshots (i.e, we track the same word set across time). This eliminates trivial examples of word usage shift from words which appear or vanish throughout the corpus.

To model word evolution, we construct a time series $\mathcal{T}(w)$ for each word $w \in \mathcal{V}$. Each point $\mathcal{T}_t(w)$ corresponds to statistical information extracted from corpus snapshot $\mathcal{C}_t$ that reflects the usage of w. In Section 2.2, we propose several methods to calculate $\mathcal{T}_t(w)$, each varying in the statistical information

used to capture w's usage.

Once these time series are constructed, we can quantify the significance of the shift that occurred to the word in its meaning and usage. Sudden increases or decreases in the time series are indicative of shifts in the word usage. Specifically we pose the following questions:

1. How statistically significant is the shift in usage of a word w across time (in $\mathcal{T}(w)$)?

2. Given that a word has shifted, at what point in time did the change happen?

2.2 Time Series Construction

Constructing the time series is the first step in quantifying the significance of word change. Different approaches capture different aspects of word's semantic, syntactic and usage patterns. In this section, we describe three approaches (Frequency, Syntactic, and Distributional) to building a time series that capture different aspects of word evolution across time. The choice of time series significantly influences the types of changes we can detect —a phenomenon which we discuss further in Section 2.5.

2.2.1 Frequency Method

The most immediate way to detect sequences of discrete events is through their change in frequency. Frequency based methods are therefore quite popular, and include tools like Google Trends and Google Books Ngram Corpus, both of which are used in research to predict economical and public health changes [10, 11]. Such analysis depends on keyword search over indexed corpora.

Frequency based methods can capture linguistic shift, as changes in frequency can correspond to words acquiring or losing senses. Although crude, this method is simple to implement. We track the change in probability of a word appearing over time. We calculate for each time snapshot corpus $\mathcal{C}_t$, a unigram language model. Specifically, we construct the time series for a word w as follows:

$$\mathcal{T}_t(w) = \log \frac{\#(w \in \mathcal{C}_t)}{|\mathcal{C}_t|}, \tag{2.1}$$

where $\#(w \in \mathcal{C}_t)$ is the number of occurrences of the word w in corpus snapshot $\mathcal{C}_t$. An example of the information we capture by tracking word frequencies over time is shown in Figure 2.3. Observe the sudden jump in late 1980s of the word `gay` in frequency.

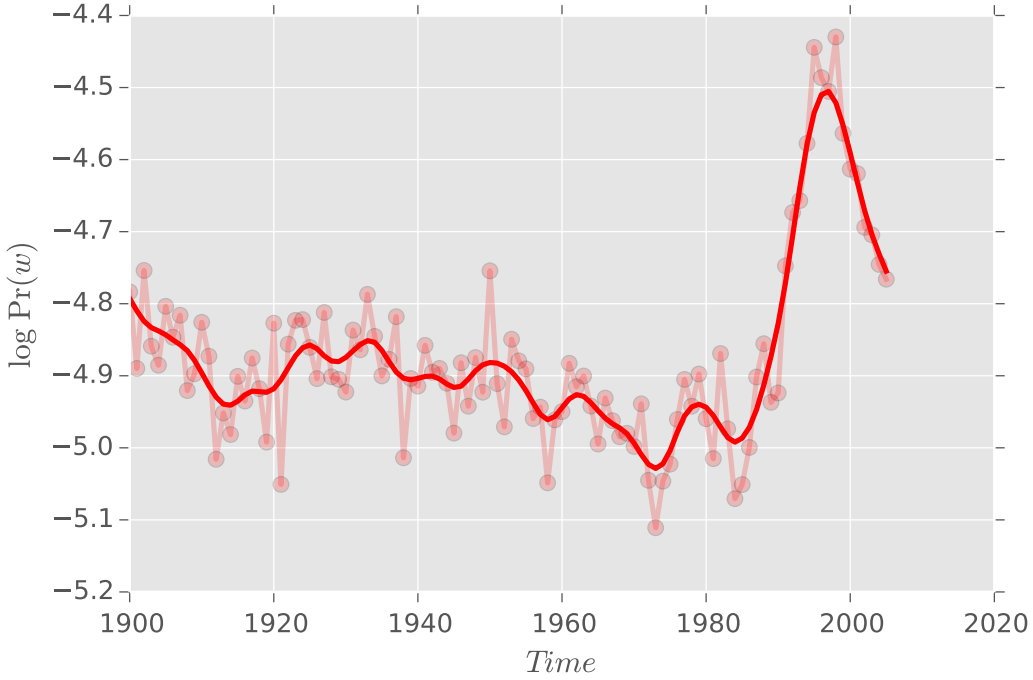


Figure 2.3: Frequency usage of the word `gay` over time, observe the sudden change in frequency in the late 1980s.

2.2.2 Syntactic Method

While word frequency based metrics are easy to calculate, they are prone to sampling error introduced by bias in domain and genre distribution in the corpus. Temporal events and popularity of specific entities could spike the word usage frequency without significant shift in its meaning, recall `Hurricane` in Figure 2.2a.

Another approach to detect and quantify significant change in the word usage involves tracking the syntactic functionality it serves. A word could evolve a new syntactic functionality by acquiring a new part of speech category. For example, `apple` used to be only a "Noun" describing a fruit, but over time it acquired the new part of speech "Proper Noun" to indicate the new sense describing a technical company (Figure 2.4). To leverage this syntactic knowledge, we annotate our corpus with part of speech (POS) tags. Then we calculate the probability distribution of part of speech tags Q_t given the word w and time snapshot t as follows: $Q_t = \Pr_{X \sim \text{POS Tags}}(X|w, \mathcal{C}_t)$. We consider the POS tag distribution at $t = 0$ to be the initial distribution Q_0. To quantify the temporal change between two time snapshots corpora, for a specific word w,

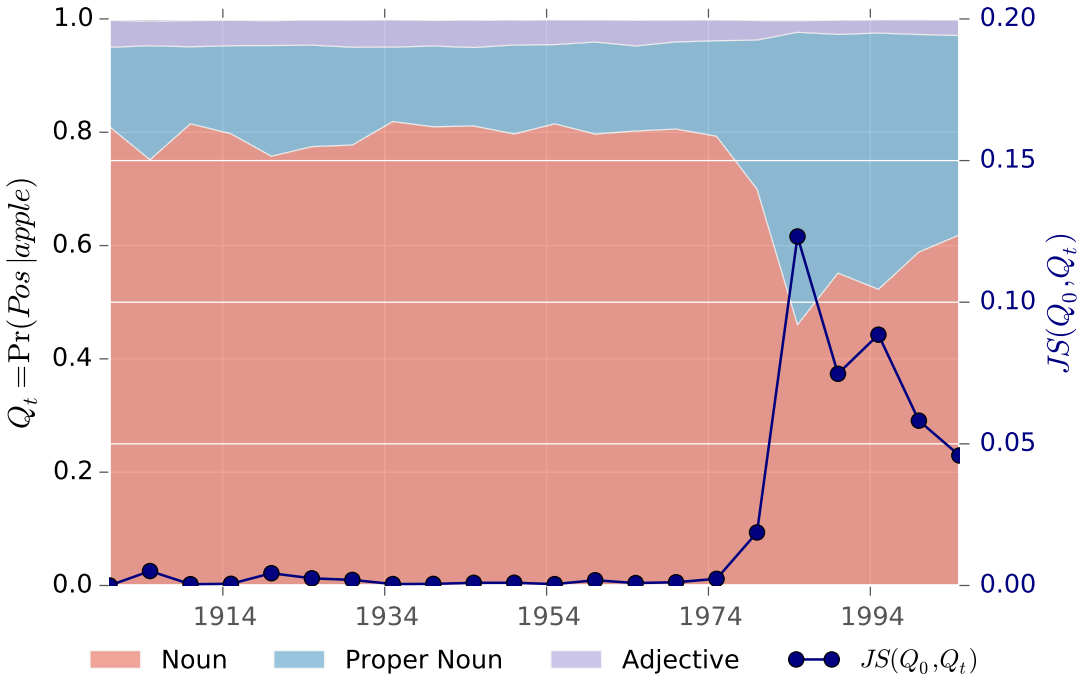


Figure 2.4: Part of speech tag probability distribution of the word `apple` (stacked area chart). Observe that the "Proper Noun" tag has dramatically increased in 1980s. The same trend is clear from the time series constructed using Jenssen-Shannon Divergence (dark blue line).

we calculate the divergence between the POS distributions in both snapshots.

Specifically, we construct the time series as follows:

$$\mathcal{T}_t(w) = \quad JSD(Q_0, Q_t) \tag{2.2}$$

where JSD is the Jenssen-Shannon divergence [12].

Figure 2.4 shows that the JS divergence (dark blue line) reflects the change in the distribution of the part of speech tags given the word `apple`. In 1980s, the "Proper Noun" tag (blue area) increased dramatically due to the rise of `Apple Computer Inc.`, the popular consumer electronics company.

2.2.3 Distributional Method

Semantic shifts are not restricted to changes to part of speech. For example, consider the word `mouse`. In the 1970s it acquired a new sense of "computer

input device", but did not change its part of speech categorization (since both senses are nouns). To detect such subtle semantic changes, we need to infer deeper cues from the contexts a word is used in.

The distributional hypothesis states that words appearing in similar contexts are semantically similar [13]. Distributional methods learn a semantic space that maps words to continuous vector space $\mathbb{R}^d$, where d is the dimension of the vector space. Thus, vector representations of words appearing in similar contexts will be close to each other. Recent developments in representation learning (*deep learning*) [14] have enabled the scalable learning of such models. We use a variation of these models [15] to learn word vector representation (*word embeddings*) that we track across time.

Specifically, we seek to learn a temporal word embedding $\phi_t : \mathcal{V}, \mathcal{C}_t \mapsto \mathbb{R}^d$. Once we learn a representation of a specific word for each time snapshot corpus, we track the changes of the representation across the embedding space to quantify the meaning shift of the word (as shown in Figure 2.1).

In this section we present our distributional approach in detail. Specially we discuss the learning of word embeddings, the aligning of embedding spaces across different time snapshots to a joint embedding space, and the utilization of a word's displacement through this semantic space to construct a distributional time series.

Learning embeddings

Given a time snapshot $\mathcal{C}_t$ of the corpus, our goal is to learn ϕ_t over $\mathcal{V}$ using neural language models. At the beginning of the training process, the words vector representations are randomly initialized. The training objective is to maximize the probability of the words appearing in the context of word w_i. Specifically, given the vector representation $\mathbf{w}_i$ of a word w_i ($\mathbf{w}_i = \phi_t(w_i)$), we seek to maximize the probability of w_j through the following equation:

$$\Pr(w_j \mid \mathbf{w}_i) = \frac{\exp\left(\mathbf{w}_j^T \mathbf{w}_i\right)}{\sum_{w_k \in \mathcal{V}} \exp\left(\mathbf{w}_k^T \mathbf{w}_i\right)} \tag{2.3}$$

In a single epoch, we iterate over each word occurrence in the time snapshot $\mathcal{C}_t$ to minimize the negative log-likelihood of the context words. Context words are the words appearing to the left or right of w_i within a window of size m

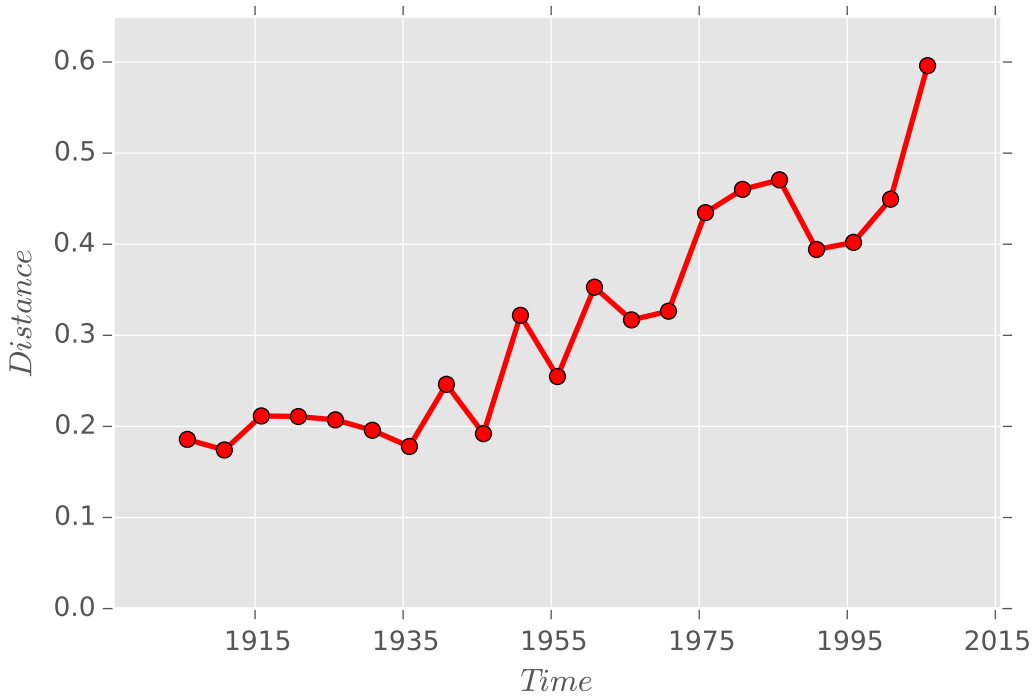


Figure 2.5: Distributional time series for the word `tape` over time using word embeddings. Observe the change of behavior starting in the 1950s, which is quite apparent by the 1970s.

(Equation 2.4).

$$J = \sum_{w_i \in \mathcal{C}_t} \sum_{\substack{j=i-m \\ j \neq i}}^{i+m} -\log \Pr(w_j \mid \mathbf{w}_i) \tag{2.4}$$

Notice that the normalization factor that appears in Equation 2.3 is not feasible to calculate if $|\mathcal{V}|$ is too large. To approximate this probability, we map the problem from a classification of 1-out-of-$\mathcal{V}$ words to a hierarchical classification problem [16, 17]. This reduces the cost of calculating the normalization factor from $\mathcal{O}(|\mathcal{V}|)$ to $\mathcal{O}(\log|\mathcal{V}|)$.

We optimize the model parameters using stochastic gradient descent [18], as follows:

$$\phi_t(w_i) = \phi_t(w_i) - \alpha \times \frac{\partial J}{\partial \phi_t(w_i)}, \tag{2.5}$$

where α is the learning rate. We calculate the derivatives of the model using the back-propagation algorithm[19]. We use the following measure of training

convergence:

$$\rho = \frac{1}{|\mathcal{V}|} \sum_{w \in \mathcal{V}} \frac{\phi^{k^T}(w)\phi^{k+1}(w)}{\|\phi^k(w)\|_2 \|\phi^{k+1}(w)\|_2}, \tag{2.6}$$

where ϕ^k is the model parameters after epoch k. We calculate ρ after each epoch and stop the training if $\rho \leq 1.0^{-4}$. After training stops, we normalize word embeddings by their L_2 norm, which forces all words to be represented by unit vectors.

In our experiments, we use *gensim* implementation of Skipgram models[2]. We set the context window size m to 10 unless otherwise stated. We choose the size of the word embedding space dimension d to be 200. To speed up the training, we subsample the frequent words by the ratio 10^{-5} [20].

Aligning Embeddings

Having trained temporal word embeddings for each time snapshot $\mathcal{C}_t$, we must now align the embeddings so that all the embeddings are in one unified coordinate system. This enables us to characterize the change between them. This process is complicated by the stochastic nature of our training, which implies that models trained on exactly the same data could produce vector spaces where words have the same nearest neighbors but not with the same coordinates. The alignment problem is exacerbated by actual changes in the distributional nature of words in each snapshot.

To aid the alignment process, we make two simplifying assumptions: First, we assume that the spaces are equivalent under a linear transformation. Second, we assume that the meaning of most words did not shift over time, and therefore, their local structure is preserved. Based on these assumptions, observe that when the alignment model fails to align a word properly, it is possibly indicative of a linguistic shift.

Specifically, we define the set of k nearest words in the embedding space ϕ_t to a word w to be k-NN$(\phi_t(w))$. We seek to learn a linear transformation $\mathbf{W}_{t' \mapsto t}(w) \in \mathbb{R}^{d \times d}$ that maps a word from $\phi_{t'}$ to ϕ_t by solving the following optimization:

$$\underset{t' \mapsto t}{W}(w) = \underset{\mathbf{W}}{\operatorname{argmin}} \sum_{\substack{w_i \in \\ k\text{-NN}(\phi_{t'}(w))}} \|\phi_{t'}(w_i)\mathbf{W} - \phi_t(w_i)\|_2^2, \tag{2.7}$$

which is equivalent to a piecewise linear regression model.

[2] https://github.com/piskvorky/gensim

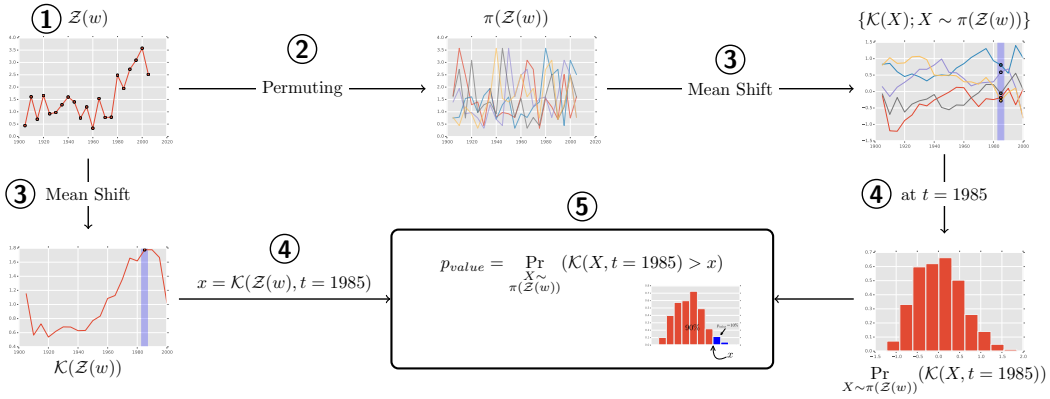


Figure 2.6: Our change point detection algorithm. In Step ①, we normalize the given time series $\mathcal{T}(w)$ to produce $\mathcal{Z}(w)$. Next, we shuffle the time series points producing the set $\pi(\mathcal{Z}(w))$ (Step ②). Then, we apply the mean shift transformation ($\mathcal{K}$) on both the original normalized time series $\mathcal{Z}(w)$ and the permuted set (Step ③). In Step ④, we calculate the probability distribution of the mean shifts possible given a specific time ($t = 1985$) over the bootstrapped samples. Finally, we compare the observed value in $\mathcal{K}(\mathcal{Z}(w))$ to the probability distribution of possible values to calculate the p-value which determines the statistical significance of the observed time series shift (Step ⑤).

Time Series Construction

To track the shift of word position across time, we align all embeddings spaces to the embedding space of the final time snapshot ϕ_n using the linear mapping (Eq. 2.7). This unification of coordinate systems allows us to compare relative displacements that occurred to words across different time periods.

To capture linguistic shift, we construct our distributional time series by calculating the distance in the embedding space between $\phi_t(w)\mathbf{W}_{t \mapsto n}(w)$ and $\phi_0(w)\mathbf{W}_{0 \mapsto n}(w)$ as

$$\mathcal{T}_t(w) = 1 - \frac{(\phi_t(w)\mathbf{W}_{t \mapsto n}(w))^T (\phi_0(w)\mathbf{W}_{0 \mapsto n}(w))}{\|\phi_t(w)\mathbf{W}_{t \mapsto n}(w)\|_2 \|\phi_0(w)\mathbf{W}_{0 \mapsto n}(w)\|_2} \tag{2.8}$$

Figure 2.5 shows the time series obtained using word embeddings for `tape`, which underwent a semantic change in the 1950s with the introduction of magnetic tape recorders. As such recorders grew in popularity, the change becomes more pronounced, until it is quite apparent by the 1970s.

Algorithm 1 CHANGE POINT DETECTION ($\mathcal{T}(w)$, B, γ)

Input: $\mathcal{T}(w)$: Time series for the word w, B: Number of bootstrap samples, γ: Z-Score threshold
Output: ECP: Estimated change point, p-value: Significance score.

// Preprocessing
$Z(w) \leftarrow$ Normalize $\mathcal{T}(w)$.
Compute mean shift series $\mathcal{K}(Z(w))$
// Bootstrapping
$BS \leftarrow \emptyset$ {Bootstrapped samples}
repeat
  Draw P from $\pi(\mathcal{Z}(w))$
  $BS \leftarrow BS \cup P$
until $|BS| = B$
for $i \leftarrow 1, n$ **do**
  $p\text{-value}(w, i) \leftarrow \frac{1}{B} \sum_{P \in BS} [\mathcal{K}_i(P) > \mathcal{K}_i(Z(w))]$
end for
// Change Point Detection
$C \leftarrow \{j | j \in [1, n] \; and \; Z_j(w) >= \gamma\}$
$p\text{-value} \leftarrow \min_{j \in C} p\text{-value}(w, j)$
$ECP \leftarrow \text{argmin}_{j \in C} p\text{-value}(w, j)$
return p-value, ECP

2.3 Change Point Detection

Given a time series of a word $\mathcal{T}(w)$, constructed using one of the methods discussed in Section 2.2, we seek to determine whether the word changed significantly, and if so estimate the change point. We believe a formulation in terms of changepoint detection is appropriate because even if a word might change its meaning (usage) gradually over time, we expect a time period where the new usage suddenly dominates (tips over) the previous usage (akin to a phase transition) with the word `gay` serving as an excellent example.

There exists an extensive body of work on change point detection in time series [21–23]. Our approach models the time series based on the *Mean Shift* model described in [21]. First, our method recognizes that language exhibits a general stochastic drift. We account for this by first normalizing the time series for each word. Our method then attempts to detect a shift in the mean of the time series using a variant of mean shift algorithms for change point analysis. We outline our method in Algorithm 1 and describe it below. We also illustrate key aspects of the method in Figure 2.6.

Given a time series of a word $\mathcal{T}(w)$, we first normalize the time series. We

calculate the mean $\mu_i = \frac{1}{|\mathcal{V}|}\sum_{w\in\mathcal{V}} \mathcal{T}_i(w)$ and variance $Var_i = \frac{1}{|\mathcal{V}|}\sum_{w\in\mathcal{V}}(\mathcal{T}_i(w) - \mu_i)^2$ across all words. Then, we transform $\mathcal{T}(w)$ into a *Z-Score* series using:

$$\mathcal{Z}_i(w) = \frac{\mathcal{T}_i(w) - \mu_i}{\sqrt{Var_i}}, \tag{2.9}$$

where $\mathcal{Z}_i(w)$ is the Z-Score of the time series for the word w at time snapshot i.

We model the time series $\mathcal{Z}(w)$ by a *Mean shift model* [21]. Let $S = \mathcal{Z}_1(w), \mathcal{Z}_2(w), \ldots, \mathcal{Z}_n(w)$ represent the time series. We model $\mathcal{S}$ to be an output of a stochastic process where each $\mathcal{S}_i$ can be described as $\mathcal{S}_i = \mu_i + \epsilon_i$ where μ_i is the mean and ϵ_i is the random error at time i. We also assume that the errors ϵ_i are independent with mean 0. Generally $\mu_i = \mu_{i-1}$ except for a few points which are *change points*.

Based on the above model, we define the mean shift of a general time series S as follows:

$$\mathcal{K}(S) = \frac{1}{l-j}\sum_{k=j+1}^{l} S_k - \frac{1}{j}\sum_{k=1}^{j} S_k \tag{2.10}$$

This corresponds to calculating the shift in mean between two parts of the time series pivoted at time point j. Change points can be thus identified by detecting significant shifts in the mean.[3]

Given a normalized time series $\mathcal{Z}(w)$, we then compute the mean shift series $\mathcal{K}(\mathcal{Z}(w))$ (Line 2). To estimate the statistical significance of observing a mean shift at time point j, we use bootstrapping [24] (see Figure 2.6 and Lines 1-10) under the null hypothesis that there is no change in the mean. In particular, we establish statistical significance by first obtaining B (typically $B = 1000$) bootstrap samples obtained by permuting $\mathcal{Z}(w)$ (Lines 1-10). Second, for each bootstrap sample $\mathcal{P}$, we calculate $\mathcal{K}(P)$ to yield its corresponding bootstrap statistic and we estimate the statistical significance (p-value) of observing the mean shift at time i compared to the null distribution (Lines 8-10). Finally, we estimate the change point by considering the time point j with the minimum p-value score (described in [21]). While this method does detect significant changes in the mean of the time series, observe that it does not account for the magnitude of the change in terms of Z-Scores. We extend this approach to obtain words that changed significantly compared to other words, by considering only those time points where the Z-Score exceeds a user-defined threshold γ (we typically set γ to 1.75). We then estimate the change point as the time point with the minimum p-value exactly as outlined before (Lines 11-14).

[3] This is similar to the CUSUM based approach used for detecting change points which is also based on mean shift model.

2.4 Datasets

Here we report the details of the three datasets that we consider - years of micro-blogging from Twitter, a decade of movie reviews from Amazon, and a century of written books using the Google Books Ngram Corpus. Table 2.1 shows a summary of three different datasets spanning different modes of expression on the Internet: books, online forum and a micro-blogs.

The Google Books Ngram Corpus The Google Books Ngram Corpus project enables the analysis of cultural, social and linguistic trends. It contains the frequency of short phrases of text (*ngrams*) that were extracted from books written in eight languages over five centuries [25]. These ngrams vary in size (1-5) grams. We use the 5-gram phrases which restrict our context window size m to 5. Here, we show a sample of 5-grams we used:

- `thousand pounds less then nothing`
- `to communicate to each other`

We focus on the time span from $1900 - 2005$, and set the time snapshot period to 5 years (21 points). We obtain the POS Distribution of each word in the above time range by using the Google Syntactic Ngrams dataset [26–28].

Amazon Movie Reviews Amazon Movie Reviews dataset consists of movie reviews from Amazon. This data spans August 1997 to October 2012 (13 time points), including all 8 million reviews. However, we consider the time period starting from 2000 as the number of reviews from earlier years is considerably small. Each review includes product and user information, ratings, and a plain-text review. A sample review text is shown below:

```
This movie has it all.Drama, action, amazing battle scenes -
the best I've ever seen.It's definitely a must see.
```

Twitter Data This dataset consists of a sample of that spans 24 months starting from September 2011 to October 2013. Each Tweet includes the Tweet ID, Tweet and the geo-location if available. A sample Tweet text is shown below:

```
I hope sandy doesn't rip the roof off the pool while we're
swimming ...
```

	Google Ngrams	Amazon	Twitter
Span (years)	105	12	2
Period	5 years	1 year	1 month
# words	$\sim 10^9$	$\sim 9.9 \times 10^8$	$\sim 10^9$
$\lvert\mathcal{V}\rvert$	$\sim$50K	$\sim$50K	$\sim$100K
# documents	$\sim 7.5 \times 10^8$	$8. \times 10^6$	$\sim 10^8$
Domain	Books	Movie Reviews	Micro Blogging

Table 2.1: Summary of our datasets

2.5 Experiments

In this section, we apply our method for each dataset presented in Section 2.4 and identify words that have changed usage over time. We describe the results of our experiments below.

2.5.1 Time Series Analysis

As we shall see in Section 2.5.4, our proposed time series construction methods differ in performance. Here, we use the detected words to study the behavior of our construction methods.

Table 2.2 shows the time series constructed for a sample of words with their corresponding p-value time series, displayed in the last column. A dip in the p-value is indicative of a shift in the word usage. The first three words, `transmitted`, `bitch`, and `sex`, are detected by both the *Frequency* and *Distributional* methods. Table 2.3 shows the previous and current senses of these words demonstrating the changes in usage they have gone through.

Observe that words like `her` and `desk` did not change, however, the *Frequency* method detects a change. The sharp increase of the word `her` in frequency around the 1960's could be attributed to the concurrent rise and popularity of the feminist movement. Sudden temporary popularity of specific social and political events could lead the *Frequency* method to produce many false positives. These results confirm our intuition we illustrated in Figure 2.2. While frequency analysis (like Google Trends) is an extremely useful tool to visualize trends, it is not very well suited for the task of detecting linguistic shift.

The last two rows in Table 2.2 display two words (`apple` and `diet`) that *Syntactic* method detected. The word `apple` was detected uniquely by the *Syntactic* method as its most frequent part of speech tag changed significantly from "Noun" to "Proper Noun". While both *Syntactic* and *Distributional*

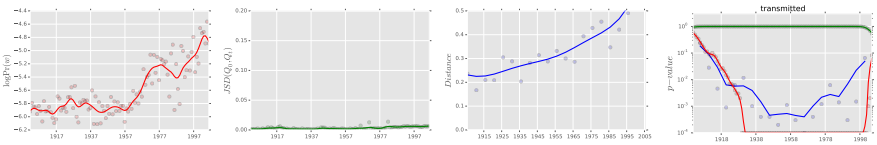
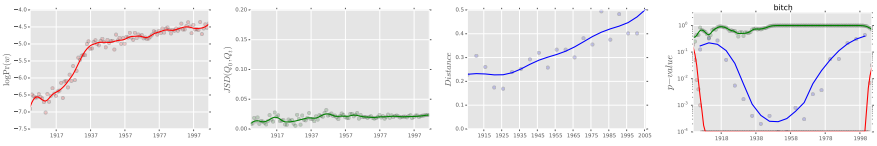
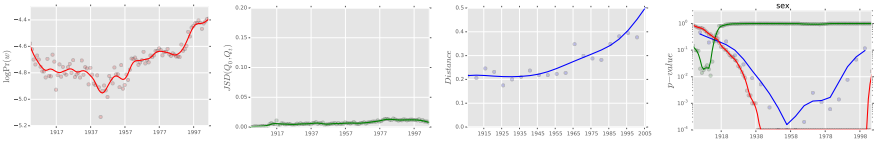
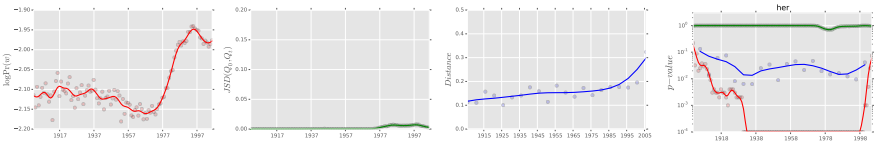
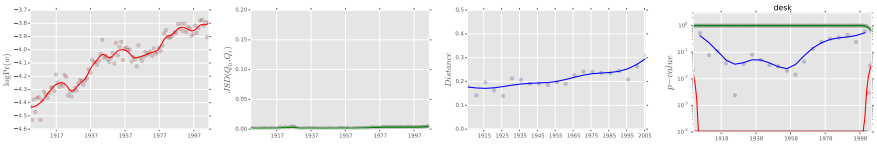
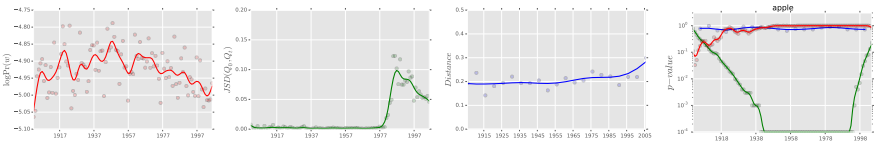
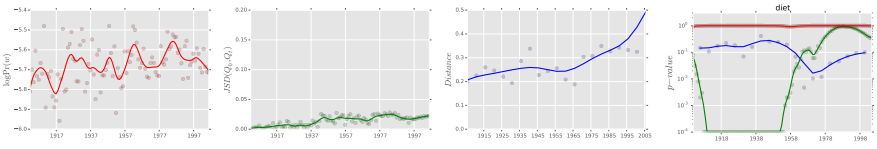
Word	Time Series			p-value
	Frequency	***Syntactic***	***Distributional***	
`transmitted`				
`bitch`				
`sex`				
`her`				
`desk`				
`apple`				
`diet`				

Table 2.2: Comparison of our different methods (—*Frequency*, —*Syntactic* and —*Distributional*) of constructing linguistic shift time series on the Google Books Ngram Corpus. The first three columns represent time series for a sample of words. The last column shows the p-value as generated by our point detection algorithm for each method.

methods indicate the change in meaning of the word `diet`, it is only the

	Word	ECP	p-value	Past usage	Present usage
Distributional better	`recording`	1990	0.0263	*to be ashamed of recording that*	*recording, photo-copying*
	`gay`	1985	0.0001	*happy and gay*	*gay and lesbians*
	`tape`	1970	<0.0001	*red tape, tape from her mouth*	*a copy of the tape*
	`checking`	1970	0.0002	*then checking himself*	*checking him out*
	`diet`	1970	0.0104	*diet of bread and butter*	*go on a diet*
	`sex`	1965	0.0002	*and of the fair sex*	*have sex with*
	`bitch`	1955	0.0001	*nicest black bitch* (Female dog)	*bitch* (Slang)
	`plastic`	1950	0.0005	*of plastic possibilities*	*put in a plastic*
	`transmitted`	1950	0.0002	*had been transmitted to him, transmitted from age to age*	*transmitted in electronic form*
	`peck`	1935	0.0004	*brewed a peck*	*a peck on the cheek*
	`honey`	1930	0.01	*land of milk and honey*	*Oh honey!*
				Past POS	**Present POS**
Syntactic better	`hug`	2002	<0.001	Verb (*hug a child*)	Noun (*a free hug*)
	`windows`	1992	<0.001	Noun (*doors and windows of a house*)	Proper Noun (*Microsoft Windows*)
	`bush`	1989	<0.001	Noun (*bush and a shrub*)	Proper Noun (*George Bush*)
	`apple`	1984	<0.001	Noun (*apple, orange, grapes*)	Proper Noun (*Apple computer*)
	`sink`	1972	<0.001	Verb (*sink a ship*)	Noun (*a kitchen sink*)
	`click`	1952	<0.001	Noun (*click of a latch*)	Verb (*click a picture*)
	`handle`	1951	<0.001	Noun (*handle of a door*)	Verb (*he can handle it*)

Table 2.3: Estimated change point (ECP) as detected by our approach for a sample of words on Google Books Ngram Corpus. *Distributional* method is better on some words (which *Syntactic* did not detect as statistically significant eg. sex, transmitted, bitch, tape, peck) while *Syntactic* method is better on others (which *Distributional* failed to detect as statistically significant eg. apple, windows, bush)

Distributional method that detects the right point of change (as shown in Table 2.3). The *Syntactic* method is indicative of having low false positive rate, but suffers from a high false negative rate, given that only two words in the table were detected. Furthermore, observe that *Syntactic* method relies on good linguistic taggers. However, linguistic taggers require annotated data sets and

	Word	p-value	ECP	Past Usage	Present Usage
Amazon Reviews	`instant`	0.016	2010	*instant hit, instant dislike*	*instant download*
	`twilight`	0.022	2009	*twilight as in dusk*	*Twilight* (The movie)
	`rays`	0.001	2008	*x-rays*	*blu-rays*
	`streaming`	0.002	2008	*sunlight streaming*	*streaming video*
	`ray`	0.002	2006	*ray of sunshine*	*Blu-ray*
	`delivery`	0.002	2006	*delivery of dialogue*	*timely delivery of products*
	`combo`	0.002	2006	*combo of plots*	*combo DVD pack*
Tweets	`candy`	<0.001	Apr 2013	*candy sweets*	*Candy Crush* (The game)
	`rally`	<0.001	Mar 2013	*political rally*	*rally of soldiers* (Immortalis game)
	`snap`	<0.001	Dec 2012	*snap a picture*	*snap chat*
	`mystery`	<0.001	Dec 2012	*mystery books*	*Mystery Manor* (The game)
	`stats`	<0.001	Nov 2012	*sport statistics*	*follower statistics*
	`sandy`	0.03	Sep 2012	*sandy beaches*	*Hurricane Sandy*
	`shades`	<0.001	Jun 2012	*color shade, shaded glasses*	50 *shades of grey* (The Book)

Table 2.4: Sample of words detected by our *Distributional* method on Amazon Reviews and Tweets.

also do not work well across domains.

We find that the *Distributional* method offers a good balance between false positives and false negatives, while requiring no linguistic resources of any sort. Having analyzed the words detected by different time series we turn our attention to the analysis of estimated changepoints.

2.5.2 Historical Analysis

We have demonstrated that our methods are able to detect words that shifted in meaning. We seek to identify the inflection points in time where the new senses are introduced. Moreover, we are interested in understanding how the new acquired senses differ from the previous ones.

Table 2.3 shows sample words that are detected by *Syntactic* and *Distributional* methods. The first set represents words which the *Distributional* method detected (*Distributional* better) while the second set shows sample words which *Syntactic* method detected (*Syntactic* better).

Our *Distributional* method estimates that the word `tape` changed in the

early 1970s to mean a "cassette tape" and not only an "adhesive tape". The change in the meaning of `tape` commences with the introduction of magnetic tapes in 1950s (Figure 2.5). The meaning continues to shift with the mass production of cassettes in Europe and North America for pre-recorded music industry in mid 1960s until it is deemed statistically significant.

The word `plastic` is yet another example, where the introduction of new products inflected a shift the word meaning. The introduction of Polystyrene in 1950 popularized the term "plastic" as a synthetic polymer, which was once used only to denote the physical property of "flexibility". The popularity of books on dieting started with the best selling book *Dr. Atkins' Diet Revolution* by Robert C. Atkins in 1972 [29]. This changed the use of the word `diet` to mean a life-style of food consumption behavior and not only the food consumed by an individual or group.

The *Syntactic* section of Table 2.3 shows that words like `hug` and `sink` were previously used mainly as verbs. Over time organizations and movements started using `hug` as a noun which dominated over its previous sense. On the other hand, the words `click` and `handle`, originally nouns, started being used as verbs.

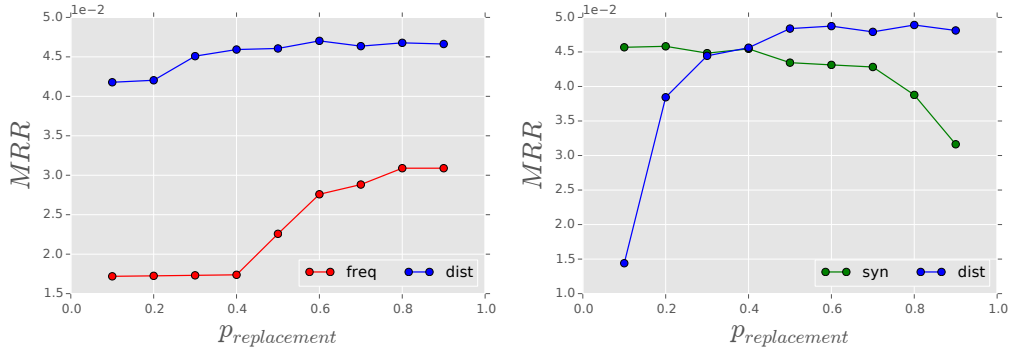
Another clear trend is the use of common words as proper nouns. For example, with the rise of the computer industry, the word `apple` acquired the sense of the tech company Apple in mid 1980s and the word `windows` shifted its meaning to the operating system developed by Microsoft in early 1990s. Additionally, we detect the word `bush` became widely used as proper noun in 1989, which coincides with George H. W. Bush's presidency in USA.

2.5.3 Cross Domain Analysis

Semantic shift can occur much faster on the web, where words can acquire new meanings within weeks, or even days. In this section we turn our attention to analyzing linguistic shift on Amazon Reviews and Twitter (content that spans a much shorter time scale as compared to Google Books Ngram Corpus).

Table 2.4 shows our *Distributional* method results on Amazon Reviews and Twitter datasets. New technologies and products introduced new meanings to words like `streaming, ray, rays, combo`. The word `twilight` acquired new sense in 2009 concurrent with the release of the Twilight movie in November 2008.

Similar trends can be observed in Twitter. The introduction of new games and cellphone applications changed the meaning of the words `candy`, `mystery` and `rally`. The word `sandy` acquired a new sense in September 2012 weeks before Hurricane Sandy hitting the East Coast of USA. Similarly we see that



(a) Frequency Perturbation **(b)** Syntactic Perturbation

Figure 2.7: Performance of our proposed methods under different scenarios of perturbation.

the word `shades` shifted its meaning with the release of the bestselling book *"Fifty Shades of Grey"* in June 2012.

These examples illustrate the capability of our method to detect the introduction of new products, movies and books. This could help semantically aware web applications to understand user intentions and requests better. Detecting the semantic shift of a word would trigger such applications to apply a focused disambiguation analysis on the sense intended by the user.

2.5.4 Quantitative Evaluation

The lack of any reference (gold standard) data, poses a challenge to quantitatively evaluate our methods. Therefore, we assess the performance of our methods using multiple approaches. We begin with a synthetic evaluation, where we have knowledge of ground-truth changes. Next we create a reference data set based on prior work and evaluate all three methods using it. We follow this with a human evaluation, and conclude with an examination of the agreement between the methods.

Synthetic Evaluation

To evaluate the quantitative merits of our approach, we use a synthetic setup which enables us to model linguistic shift in a controlled fashion by artificially introducing changes to a corpus.

Our synthetic corpus is created as follows: First, we duplicate a copy of a Wikipedia corpus[4] 20 times to model time snapshots. We tagged the

[4] http://mattmahoney.net/dc/text8.zip

Wikipedia corpora with part of speech tags using the *TextBlob* tagger[5]. Next, we introduce changes to a word's usage to model linguistics shift. To do this, we perturb the last 10 snapshots. Finally, we use our approach to rank all words according to their p-values, and then we calculate the Mean Reciprocal Rank ($MRR = 1/|Q|\sum_{i=1}^{|Q|} 1/rank(w_i)$) for the words we perturbed. We rank the words that have lower p-value higher, therefore, we expect the MRR to be higher in the methods that are able to discover more words that have changed.

To introduce a single perturbation, we sample a pair of words out of the vocabulary excluding functional words and stop words[6]. We designate one of them to be a donor and the other to be a receptor. The donor word occurrences will be replaced with the receptor word with a success probability $p_{\text{replacement}}$. For example, given the word pair (`location`, `equation`), some of the occurrences of the word `location` (Donor) were replaced with the word `equation` (Receptor) in the second five snapshots of Wikipedia.

Figure 2.7 illustrates the results on two types of perturbations we synthesized. First, we picked our (Donor, Receptor) pairs such that both of them have the same most frequent part of speech tag. For example, we might use the pair (boat, car) but not (boat, running). We expect the frequency of the receptor to change and its context distribution but no significant syntactic changes. Figure 2.7a shows the MRR of the receptor words on *Distributional* and *Frequency* methods. We observe that both methods improve their rankings as the degree of induced change increases (measured, here, by $p_{\text{replacement}}$). Second, we observe that the *Distributional* approach outperforms *Frequency* method consistently for different values of $p_{\text{replacement}}$.

Second, to compare *Distributional* and *Syntactic* methods we sample word pairs without the constraint of being from the same part of speech categories. Figure 2.7b shows that the *Syntactic* method while outperforming *Distributional* method when the perturbation statistically is minimal, its ranking continue to decline in quality as the perturbation increases. This could be explained by the fact that the quality of the tagger annotations decreases as the corpus at inference time diverges from the training corpus.

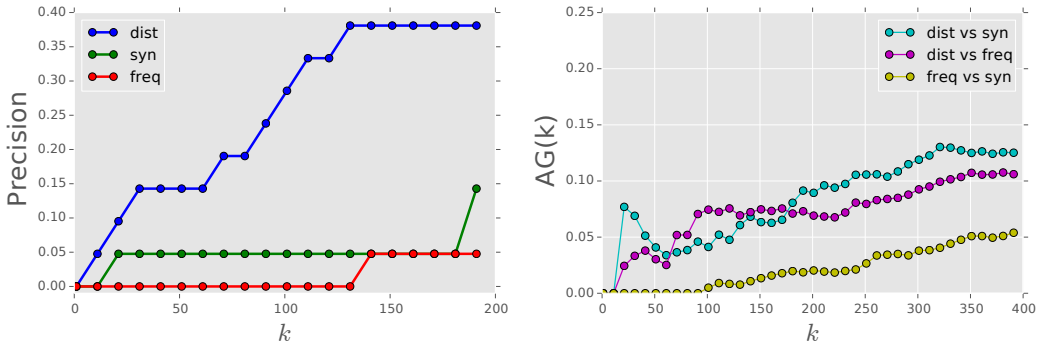
It is quite clear from both experiments, that the *Distributional* method outperforms other methods when $p_{replacement} > 0.4$ without requiring any language specific resources or annotators.

Evaluation on a Reference Dataset

In this section, we attempt to gauge the performance of the various methods on a reference data set. We created a reference data set D of 20 words that

[5] http://textblob.readthedocs.org/en/dev/

[6] NLTK Stopword List: http://www.nltk.org/



(a) Reference Dataset **(b)** Methods Agreement

Figure 2.8: Method performance and agreement on changed words in the Google Books Ngram Corpus.

have been suggested by prior work [30–33] as having undergone a linguistic change[7]. For each method, we create a list L of its changed words ordered by the significance scores of the change, and evaluate the Precision@k with respect to the reference data set constructed. Specifically, the Precision@k between L and D can be defined as:

$$\text{Precision@}k(L, D) = \frac{|L[1:k] \cap D|}{|D|} \tag{2.11}$$

Figure 2.8a depicts the performance of the different methods on this reference data set. Observe that the *Distributional* method outperforms other methods with the *Frequency* method performing the poorest (due to its high false positive rate). The *Syntactic* method which does not capture semantic changes well also performs worse than the *Distributional* method.

Human Evaluation

We chose the top 20 words claimed to have changed by each method and asked 3 human evaluators to independently decide whether each word experienced a linguistic shift. For each method, we calculated the percentage of words each rater believes have changed and report the mean percentage. We observed that on an average the raters believe that only 13.33% of the words reported by *Frequency* method and only 21.66% of the words reported by *Syntactic* method changed. However, in the case of *Distributional* method we observed that on

[7]The reference data set and the human evaluations are available at http://vivekkulkarni.net

an average the raters believe that 53.33% of the words changed. We conclude thus from this evaluation that the *Distributional* method outperforms other methods.

Method Agreement

In order to investigate the agreement between the various methods, we again consider the top k words that each method is most confident have changed. For each pair of methods, we then compute the fraction of words both methods agree on in their top k lists. Specifically given methods M_1 and M_2 let $M_1(k)$ and $M_2(k)$ represent the top k lists for M_1 and M_2 respectively. We define the agreement between these 2 lists as follows:

$$AG(M_1(k), M_2(k)) = \frac{|M_1(k) \cap M_2(k)|}{|M_1(k) \cup M_2(k)|} \tag{2.12}$$

which is the Jaccard Similarity between $M_1(k)$ and $M_2(k)$.

Figure 2.8b shows the agreement scores between each pair of methods for different values of k. We first note that the agreement between all methods is low, suggesting that the methods differ in aspects of word change captured. Observe that the agreement between *Distributional* and *Syntactic* is higher compared to that of *Syntactic* and *Frequency*. This can be explained by noting that *Distributional* method captures semantic changes along with elements of syntactic changes, and therefore agrees more with *Syntactic* method. We leave it to future work to investigate whether a single improved method can capture all of these aspects of word usage effectively.

2.6 Related Work

Because our work lies at the intersection of different fields, we will discuss the most relevant four areas of work: linguistic shift, word embeddings, change point detection, and Internet linguistics.

2.6.1 Linguistic Shift

There has been a surge in work about language evolution over time [25, 30, 31, 34–36]. Michel et al. [25] detected important political events by analyzing frequent patterns. Juola [36] compared language from different time periods and quantified the change. Different from both studies, we quantify linguistic change by tracking individual shifts in words meaning. This fine grain detection

and tracking still allows us to quantify the change in natural language as a whole, while still being able to interpret these changes.

Previous work on topic modeling and distributional semantics [30, 31, 34, 35, 37] either restrict their period to two language snapshots , or do not suggest a change point detection algorithm. Some of the above work is also restricted to detecting changes in entities (e.g. Iraq). Mitra et al. [38] use a graph based approach relying on dependency parsing of sentences. Our proposed time series construction methods require minimal linguistic knowledge and resources enabling the application of our approach to all languages and domains equally. Compared to the sequential training procedure proposed by Kim et al. [32] work, our technique warps the embeddings spaces of the different time snapshots after the training, allowing for efficient training that could be parallelized for large corpora.

Moreover, our work is unique in the fact that our datasets span different time scales, cover larger user interactions and represent a better sample of the web.

2.6.2 Word Embeddings

Hinton [39] proposed distributed representations (word embeddings), to learn a mapping of symbolic data to continuous space. Bengio et al. [40] used these word embeddings to develop a neural language model that outperforms traditional ngram models. Several efforts have been proposed to scale and speed up the computation of such big networks [16, 17, 41, 42]. Word embeddings are shown to capture fine grain structures and regularities in the data [20, 43]. Moreover, they proved to be useful for a wide range of natural language processing tasks [44, 45]. The same technique of learning word embeddings has been applied recently to learning graph representations [46].

2.6.3 Change point detection

Change Point Detection and Analysis is an important problem in the area of Time Series Analysis and Modeling. Taylor [21] describes control charts and CUSUM based methods in detail. Adams and MacKay [23] describes a Bayesian approach to Online Change Point Detection.The method of bootstrapping and establishing statistical significance is outlined in [24]. Basseville and Nikiforov [22] provides an excellent survey on several elementary change point detection techniques and time series models.

2.6.4 Relation to Internet Linguistics

Internet Linguistics is concerned with the study of language in media influenced by the Internet (online forums, blogs, online social media) and also other related forms of electronic media like Text Messaging. Schiano et al. [47] and Tagliamonte and Denis [48] study how teenagers use messaging media focusing on their usage patterns and the resulting implications on design of e-mail and Instant messaging (IM). Merchant [49] study the language use by teenagers in online chat forums. An excellent survey on Internet Linguistics is provided by [50] and includes linguistic analyses of social media like Twitter, Facebook or Google+.

2.7 Conclusions And Future Work

In this chapter, we proposed three approaches to model word evolution across time through different time series construction methods. We designed a computational approach to detect statistically significant linguistic shifts. Finally, we demonstrated our method on three different data sets each representing a different medium. By analyzing the Google Books Ngram Corpus, we were able to detect historical semantic shifts that happened to words like `gay` and `bitch`. Moreover, in faster evolving medium like Tweets and Amazon Reviews, we were able to detect recent events like storms and game and book releases. This capability of detecting meaning shift, should help decipher the ambiguity of dynamical systems like natural languages. We believe our work has implications to the fields of Semantic Search and the recently burgeoning field of Internet Linguistics.

Chapter 3

Quantifying Geographic Variation in Internet Language

Those who know nothing of foreign languages know nothing of their own.

Johann Wolfgang von Goethe

Detecting and analyzing regional variation in language is central to the field of socio-variational linguistics and dialectology (eg. [51–54]). Since online content is an agglomeration of material originating from all over the world, language on the Internet demonstrates geographic variation. The abundance of geotagged online text enables a study of geographic linguistic variation at scales that are unattainable using classical methods like surveys and questionnaires.

Characterizing and detecting such variation is challenging since it takes different forms: lexical, syntactic, and semantic. Most existing work has focused on detecting lexical variation prevalent in geographic regions [55–58]. However, regional linguistic variation is not limited to lexical variation.

In this chapter we address this gap. Our method, GeoDist, is the first computational approach for tracking and detecting statistically significant linguistic shifts of words across geographical regions. GeoDist detects syntactic and semantic variation in word usage across regions, in addition to purely lexical differences. GeoDist builds on recently introduced neural language models that learn word representations (*word embeddings*), extending them to

Work described in this chapter was done in collaboration with Bryan Perozzi and Steven Skiena and published at ICWSM, 2016.

capture region-specific semantics. Since observed regional variation could be due to chance, GeoDist explicitly introduces a null model to ensure detection of only statistically significant differences between regions.

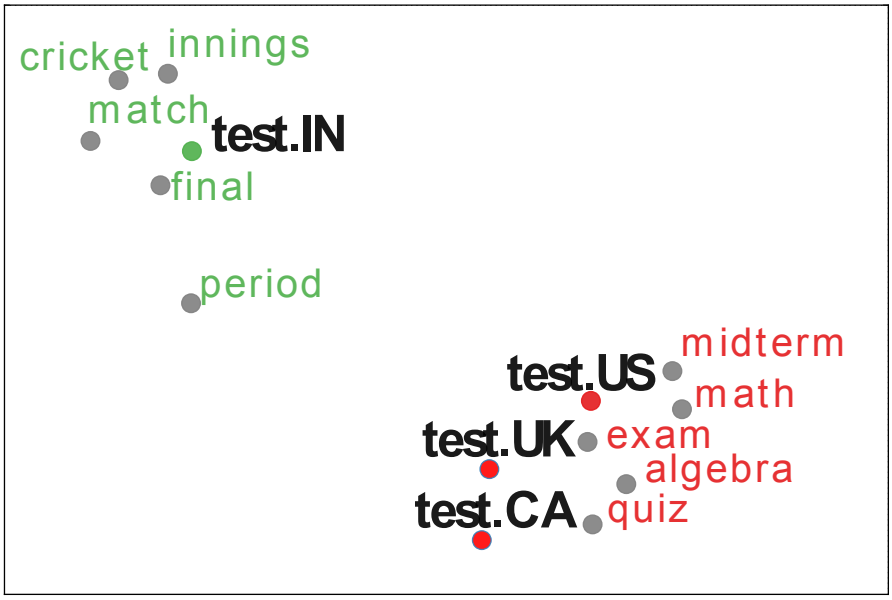
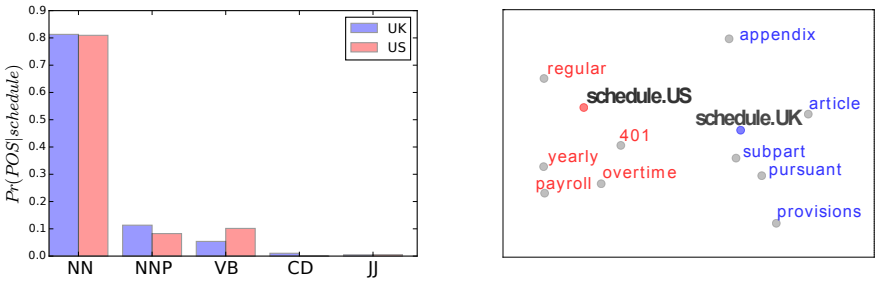


Figure 3.1: The latent semantic space captured by our method (GeoDist) reveals geographic variation between language speakers. In the majority of the English speaking world (e.g. US, UK, and Canada) a `test` is primarily used to refer to an `exam`, while in India a `test` additionally indicates a lengthy cricket match which is played over five consecutive days.

Figure 3.1 presents a visualization of the semantic variation captured by GeoDist for the word `test` between the United States, the United Kingdom, Canada, and India. In the majority of English speaking countries, `test` almost always means an `exam`, but in India (where cricket is a popular sport) `test` almost always refers to a lengthy form of cricket match. One might argue that simple baseline methods like (analyzing part of speech) might be sufficient to identify regional variation. However because these methods capture different modalities, they detect different types of changes as we illustrate in Figure 3.2.

We evaluate our methods on several large datasets at multiple geographic resolutions. We investigate linguistic variation across Twitter at multiple scales: (a) between four English speaking countries and (b) between fifty states in USA. We also investigate regional variation in the Google Books Ngram Corpus data. Our methods detect a variety of changes including regional dialectical variations, region specific usages, words incorporated due to code mixing and differing semantics.



(a) Part of Speech distribution for `schedule` (*Syntactic*)

(b) Latent semantic space captured by GeoDist method.

Figure 3.2: The word `schedule` differs in its semantic usage between US and UK English which GeoDist (see Figure 3.2b) detects. While `schedule` in the USA refers to a *"scheduling time"*, in the UK `schedule` also has the meaning of an *"addendum to a text"*. However the *Syntactic* method (see Figure 3.2a) does not detect this semantic change since `schedule` is dominantly used as a noun (NN) in both UK and the USA.

Specifically, our contributions are as follows:

- **Models and Methods**: We present our new method GeoDist which extends recently proposed neural language models to capture semantic differences between regions (Section 3.2.2). GeoDist is a new statistical method that explicitly incorporates a null model to ascertain statistical significance of observed semantic changes.
- **Multi-Resolution Analysis**: We apply our method on multiple domains (Books and Tweets) across geographic scales (States and Countries). Our analysis of these large corpora (containing billions of words) reveals interesting facets of language change at multiple scales of geographic resolution – from neighboring states to distant continents (Section 3.4).

3.1 Problem Definition

We seek to quantify shift in word meaning (usage) across different geographic regions. Specifically, we are given a corpus $\mathcal{C}$ that spans R regions where $\mathcal{C}_r$ corresponds to the corpus specific to region r. We denote the vocabulary of the corpus by $\mathcal{V}$. We want to detect words in $\mathcal{V}$ that have region specific semantics (not including trivial instances of words exclusively used in one region). For each region r, we capture statistical properties of a word w's usage in that

region. Given a pair of regions (r_i, r_j), we then reduce the problem of detecting words that are used differently across these regions to an outlier detection problem using the statistical properties captured.

In summary, we answer the following questions:

1. In which regions does the word usage drastically differ from other regions?
2. How statistically significant is the difference observed across regions?

3.2 Methods

In this section we discuss methods to model regional word usage.

3.2.1 Baseline Methods

Frequency Method. One standard method to detect which words vary across geographical regions is to track their frequency of usage. Formally, we track the change in probability of a word across regions as described in [59]. To characterize the difference in frequency usage of w between a region pair (r_i, r_j), we compute the ratio $\textsc{Score}(w) = \frac{P_{r_i}(w)}{P_{r_j}(w)}$ where $P_{r_i}(w)$ is the probability of w occurring in region r_i. An example of the information we capture by tracking word frequencies over regions is shown in Figure 3.3. Observe that `touchdown` (an American football term) is used much more frequently in the US than in UK. While this naive method is easy to implement and identifies words which differ in their usage patterns, one limitation is an overemphasis on rare words. Furthermore frequency based methods overlook the fact that word usage or meaning changes are not exclusively associated with a change in frequency.
Syntactic Method. A method to capture syntactic variation in word usage through time was proposed by [59]. Along similar lines, we can capture regional syntactic variation of words. The word `lift` is a striking example of such variation: In the US, `lift` is dominantly used as a verb (in the sense: "to lift an object"), whereas in the UK `lift` also refers to an elevator, thus predominantly used as a common noun. Given a word w and a pair of regions (r_i, r_j) we adapt the method outlined in [59] and compute the Jennsen-Shannon Divergence between the part of speech distributions for word w corresponding to the regions.

Figure 3.4 shows the part of speech distribution for a few words that differ in syntactic usage between the US and UK. In the US, `remit` is used primarily as a verb (as in "to remit a payment"). However in the UK, `remit` can refer "to an area of activity over which a particular person or group has authority,

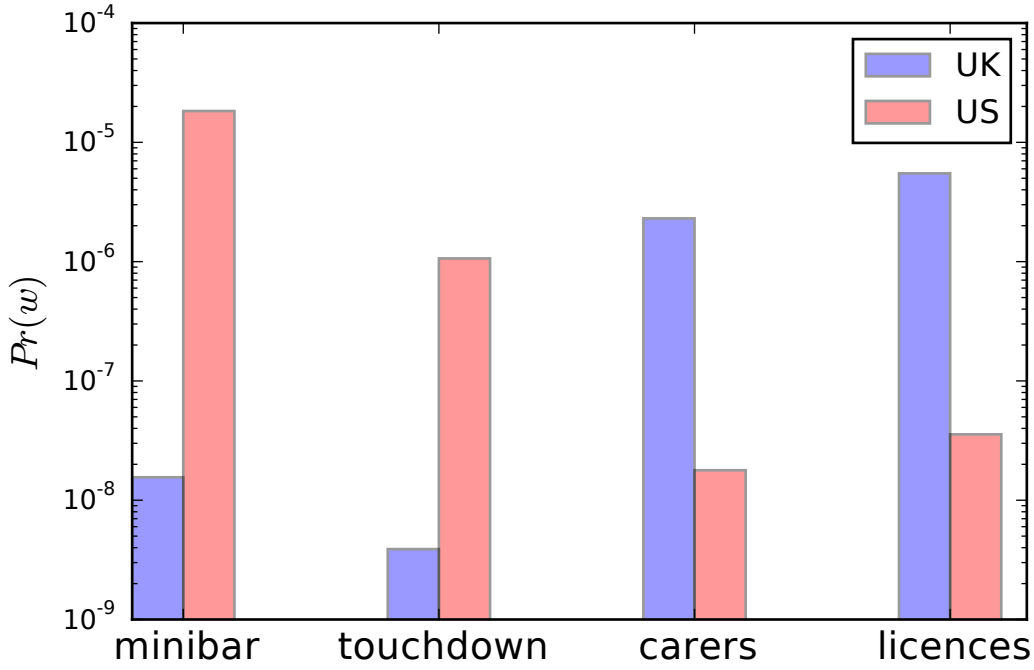


Figure 3.3: Frequency usage of different words in English UK and English US. Note that `touchdown`, an American football term is much more frequent in the US than in UK. Words like `carers` and `licences` are used more in the UK than in the US. `carers` are known as `caregivers` in the US and `licences` is spelled as `licenses` in the US.

control or influence" (used as "A remit to report on medical services")[1]. The word `curb` is used mostly as a noun (as "I should put a curb on my drinking habits.") in the UK but it is used dominantly as a verb in the US (as in "We must curb the rebellion.").

Whereas the *Syntactic* method captures a deeper variation than the frequency methods, it is important to observe that semantic changes in word usage are not limited to syntactic variation as we illustrated before in Figure 3.2.

3.2.2 Distributional Method: GeoDist

As we noted in the previous section, linguistic variation is not restricted only to syntactic variation. In order to detect subtle semantic changes, we need to infer cues based on the contextual usage of a word. To do so, we use distributional

[1] http://www.oxfordlearnersdictionaries.com/us/definition/english/remit_1

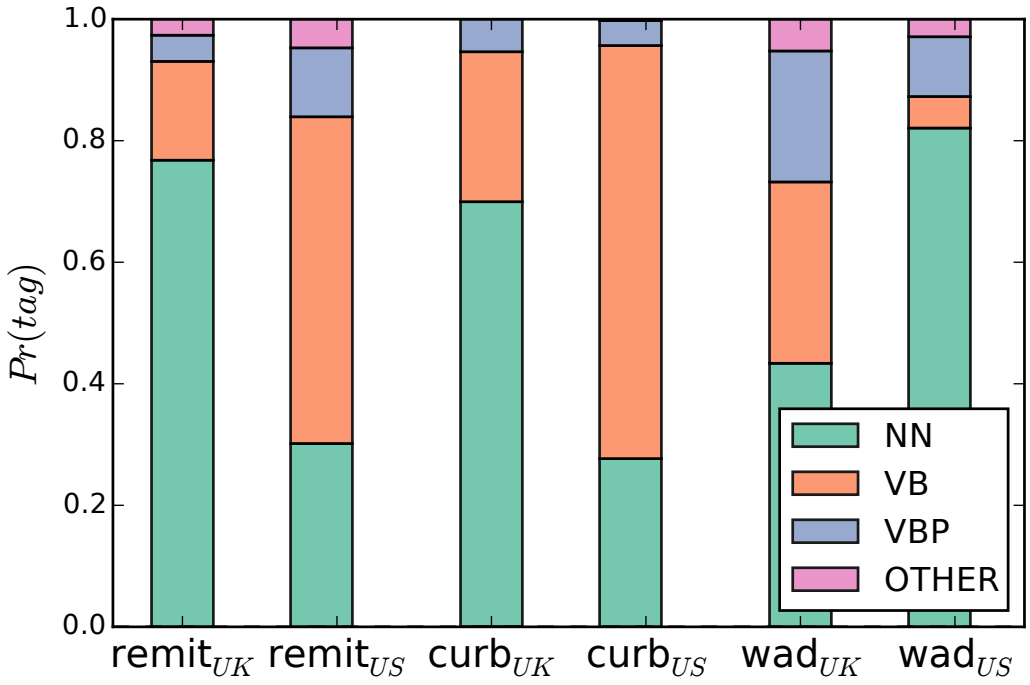


Figure 3.4: Part of speech tag probability distribution of the words which differ in syntactic usage between UK and US. Observe that `remit` is predominantly used as a verb (VB) in the US but as a common noun (NN) in the UK.

methods which learn a latent semantic space that maps each word $w \in \mathcal{V}$ to a continuous vector space $\mathbb{R}^d$.

We differentiate ourselves from the closest work related to our method [60], by *explicitly* accounting for random variation between regions, and proposing a method to detect statistically significant changes.

Learning region specific word embeddings

Given a corpus $\mathcal{C}$ with R regions, we seek to learn a region specific word embedding $\phi_r : \mathcal{V}, \mathcal{C}_r \mapsto \mathbb{R}^d$ using a neural language model. For each word $w \in \mathcal{V}$ the neural language model learns:

1. A global embedding $\delta_{\text{MAIN}}(w)$ for the word ignoring all region specific cues.

2. A differential embedding $\delta_r(w)$ that encodes differences from the global embedding specific to region r.

The region specific embedding $\phi_r(w)$ is computed as: $\phi_r(w) = \delta_{\text{MAIN}}(w) + \delta_r(w)$. Before training, the global word embeddings are randomly initialized while the

differential word embeddings are initialized to $\mathbf{0}$. During each training step, the model is presented with a set of words w and the region r they are drawn from. Given a word w_i, the context words are the words appearing to the left or right of w_i within a window of size m. We define the set of active regions $\mathcal{A} = \{r, \text{MAIN}\}$ where MAIN is a placeholder location corresponding to the global embedding and is always included in the set of active regions. The training objective then is to maximize the probability of words appearing in the context of word w_i conditioned on the active set of regions $\mathcal{A}$. Specifically, we model the probability of a context word w_j given w_i as:

$$\Pr(w_j \mid w_i) = \frac{\exp\left(\mathbf{w}_j^T \mathbf{w}_i\right)}{\sum_{w_k \in \mathcal{V}} \exp\left(\mathbf{w}_k^T \mathbf{w}_i\right)} \tag{3.1}$$

where $\mathbf{w}_i$ is defined as $\mathbf{w}_i = \sum_{a \in \mathcal{A}} \delta_a(w_i)$.

During training, we iterate over each word occurrence in $\mathcal{C}$ to minimize the negative log-likelihood of the context words. Our objective function J is thus given by:

$$J = \sum_{w_i \in \mathcal{C}} \sum_{\substack{j=i-m \\ j \neq i}}^{i+m} -\log \Pr(w_j \mid \mathbf{w}_i) \tag{3.2}$$

When $|\mathcal{V}|$ is large, it is computationally expensive to compute the normalization factor in Equation 3.1 exactly. Therefore, we approximate this probability by using hierarchical soft-max [16, 17] which reduces the cost of computing the normalization factor from $\mathcal{O}(|\mathcal{V}|)$ to $\mathcal{O}(\log|\mathcal{V}|)$. We optimize the model parameters using stochastic gradient descent [18], as $\phi_t(w_i) = \phi_t(w_i) - \alpha \times \frac{\partial J}{\partial \phi_t(w_i)}$ where α is the learning rate. We calculate the derivatives using the back-propagation algorithm [19]. We set $\alpha = 0.025$, context window size m to 10 and size of the word embedding d to be 200 unless stated otherwise.

Distance Computation between regional embeddings

After learning word embeddings for each word $w \in \mathcal{V}$, we then compute the distance of a word between any two regions (r_i, r_j) as $\textsc{Score}(w) = \textsc{CosineDistance}(\phi_{r_i}(w), \phi_{r_j}(w))$ where $\textsc{CosineDistance}(u, v)$ is defined by $1 - \frac{u^T v}{\|u\|_2 \|v\|_2}$.

Figure 3.5 illustrates the information captured by our GeoDist method as a two dimensional projection of the latent semantic space learned, for the word `theatre`. In the US, the British spelling `theatre` is typically used only

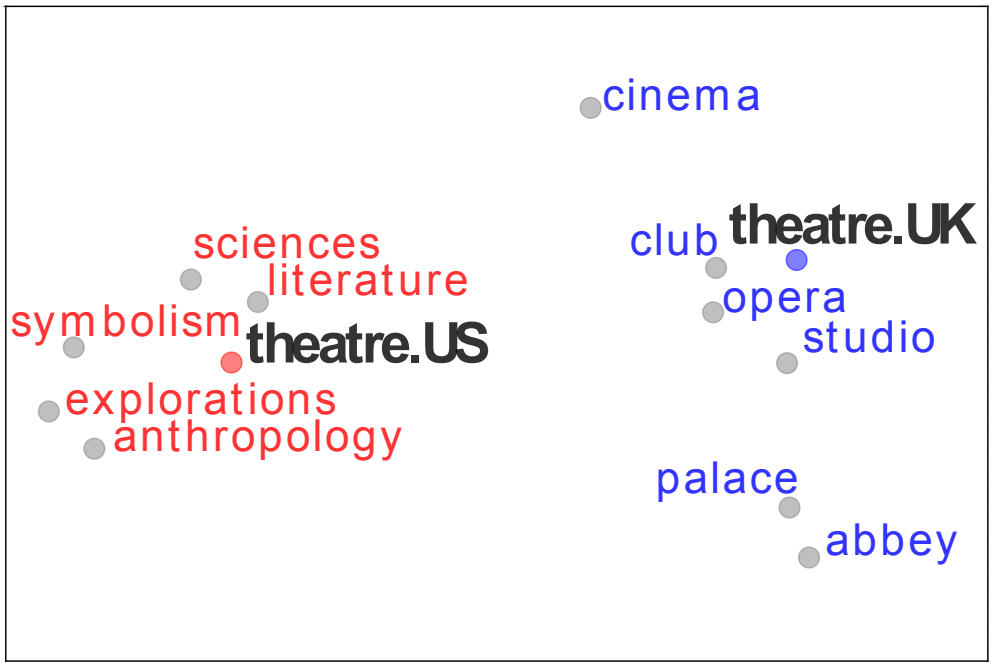


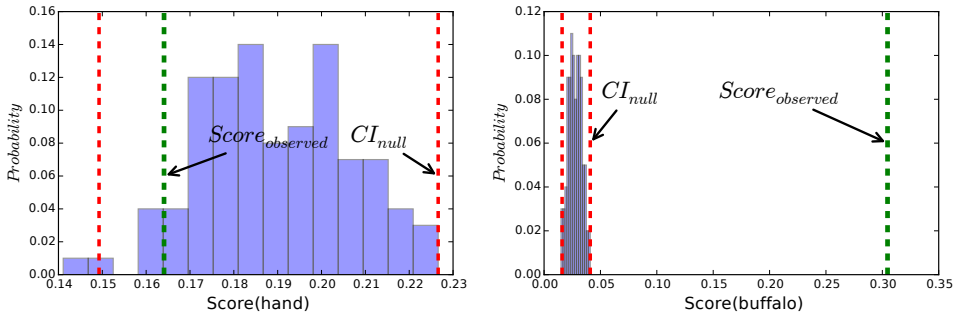
Figure 3.5: Semantic field of `theatre` as captured by GeoDist method between the UK and US. `theatre` is a field of study in the US while in the UK it primarily associated with opera or a club.

to refer to the performing arts. Observe how the word `theatre` in the US is close to other subjects of study: `sciences, literature, anthropology`, but `theatre` as used in UK is close to places showcasing performances (like `opera`, `studio`, etc). We emphasize that these regional differences detected by GeoDist are inherently *semantic*, the result of a level of language understanding unattainable by methods which focus solely on lexical variation [61].

3.2.3 Statistical Significance of Changes

In this section, we outline our method to quantify whether an observed change given by Score(w) is significant. i

In our method, Score(w) could vary due to random stochastic processes (even possibly pure chance), whether an observed score is significant or not depends on two factors: (a) the magnitude of the observed score (*effect size*) and (b) probability of obtaining a score more extreme than the observed score, even in the absence of a true effect.



(a) Observed score for `hand` **(b)** Observed score for `buffalo`

Figure 3.6: The observed scores computed by GeoDist (in - - -) for `buffalo` and `hand` when analyzing regional differences between New York and USA overall. The histogram shows the distribution of scores under the null model. The 98% confidence intervals of the score under null model are shown in - - -. The observed score for `hand` lies well within the confidence interval and hence is not a statistically significant change. In contrast, the score for `buffalo` is far outside the confidence interval for the null distribution indicating a statistically significant change.

Specifically, given a word w with a score $E(w) = \textsc{Score}(w)$ between regions (r_i, r_j) we ask the question: *"What is the chance of observing $E(w)$ or a more extreme value assuming the absence of an effect?"*

First our method explicitly models the scenario when there is no effect, which we term as the *null model*. Next we characterize the distribution of scores under the null model. Our method then compares the observed score with this distribution of scores to ascertain the significance of the observed score. The details of our method are described in Algorithm 2 and below.

We simulate the null model by observing that under the null model, the labels of the text are *exchangeable*. Therefore, we generate a corpus C' by a random assignment of the labels (regions) of the given corpus C. We then learn a model using C' and estimate $\textsc{Score}(w)$ under this model. By repeating this procedure B times we estimate the distribution of scores for each word under the null model (Lines 1 to 10).

After we estimate the distribution of scores we then compute the $100\alpha\%$ confidence interval on $\textsc{Score}(w)$ under the null model. Thus for each word w, we specify two measures: (a) observed effect size and (b) $100\alpha\%$ confidence interval corresponding to the null distribution (Lines 16-17). When the observed effect is not contained in the confidence interval obtained for the null distribution, the effect is statistically significant at the $1 - \alpha$ significance level.

Even though p-values have been traditionally used to report significance, recently researchers have argued against their use as p-values themselves do

not indicate what the observed effect size was and hence even very small effects can be deemed statistically significant [63, 64]. In contrast, reporting effect sizes and confidence intervals enables us to factor in the magnitude of effect size while interpreting significance. In a nutshell therefore, we deem a change observed for w as statistically significant when:

1. The effect size exceeds a threshold β which ensures the effect size is large enough.

2. It is rare to observe this effect as a result of pure chance. This is captured by our comparison to the null model and the confidence intervals computed.

Figure 3.6 illustrates this for two words: `hand` and `buffalo`. Observe that for `hand`, the observed score is smaller than the higher confidence interval, indicating that `hand` has not changed significantly. In contrast `buffalo` which is used differently in New York (since `buffalo` refers to a place in New York) has a score well above the higher confidence interval under the null model.

As we will also see in Section 3.4, the incorporation of the null model and obtaining confidence estimates enables our method to efficaciously tease out effects arising due to random chance from statistically significant effects.

3.3 Datasets

Here we outline the details of two online datasets that we consider - Tweets from various geographic locations on Twitter and Google Books Ngram Corpus.

The Google Books Ngram Corpus The Google Books Ngram Corpus corpus [25] contains frequencies of short phrases of text (*ngrams*) which were taken from books spanning eight languages over five centuries. While these ngrams vary in size from $1-5$, we use the 5-grams in our experiments. Specifically we use the Google Books Ngram Corpus corpora for *American English* and *British English* and use a random sample of 30 million ngrams for our experiments. Here, we show a sample of 5-grams along with their region:

- `drive a coach and horses (UK)`
- `years as a football coach (US)`

We obtained the POS Distribution of each word in the above corpora using Google Syntactic Ngrams[26, 27].

	Word	US/UK Δ	Explanation
Books	`zucchini`	2.3	*"zucchinis" are known as "courgettes" in UK*
	`touchdown`	2.4	*"touchdown" is a term in American football*
	`bartender`	2.5	*"bartender" is a very recent addition to the pub language in UK.*

	Word	US/UK Δ	Explanation
Tweets	`freshman`	2.7	*"freshman" are referred to as "freshers" in the UK*
	`hmu`	2.5	*hit me up a slang which is popular in USA*
		US/AU Δ	
	`maccas`	−3.3	*McDonald's in Australia is called maccas*
	`wickets`	−2.9	*wickets is a term in cricket, a popular game in Australia*
	`heaps`	−2.7	*Australian colloquial for "alot"*

Table 3.1: Examples of words detected by the *Frequency* method on Google Book NGrams and Twitter. (Δ is difference in log probabilities between countries). A positive value indicates the word is more probable in the US than the other region. A negative value indicates the word is more probable in the other region than the US.

	Word	JS	US Usage	UK Usage
Books	`remit`	0.173	*remit the loan*	*The jury investigated issues within its remit (an assigned area).*
	`oracle`	0.149	*Oracle the company*	*a person who is omniscient*
	`wad`	0.143	*a wad of cotton*	*Wad the paper towel and throw it! (used as "to compress")*
Tweets	`sort`	0.224	*He's not a bad sort*	*sort it out*
	`lift`	0.220	*lift the bag*	*I am stuck in the lift (elevator)*
	`ring`	0.200	*ring on my finger*	*give him a ring (call)*
	`cracking`	0.181	*The ice is cracking*	*The girl is cracking (beautiful)*
	`cuddle`	0.148	*Let her cuddle the baby (verb)*	*Come here and give me a cuddle (noun)*
	`dear`	0.137	*dear relatives*	*Something is dear (expensive)*
			US Usage	**AU Usage**
	`kisses`	0.320	*hugs and kisses (as a noun)*	*He kisses them (verb)*
	`claim`	0.109	*He made an insurance claim (noun)*	*I claim ... (almost always used as a verb)*

Table 3.2: Examples of words detected by the *Syntactic* method on Google Book NGrams and Twitter. (JS is Jennsen Shannon Divergence)

Word	Effect Size	CI(Null)	US Usage	UK Usage
theatre	0.6067	(0.004,0.007)	*great love for the theatre*	*in a large theatre*
schedule	0.5153	(0.032,0.050)	*back to your regular schedule*	*a schedule to the agreement*
forms	0.595	(0.015, 0.026)	*out the application forms*	*range of literary forms (styles)*
extract	0.400	(0.023, 0.045)	*vanilla and almond extract*	*extract from a sermon*
leisure	0.535	(0.012, 0.024)	*culture and leisure (a topic)*	*as a leisure activity*
extensive	0.487	(0.015, 0.027)	*view our extensive catalog list*	*possessed an extensive knowledge (as in impressive)*
store	0.423	(0.02, 0.04)	*trips to the grocery store*	*store of gold (used as a container)*
facility	0.378	(0.035, 0.055)	*mental health,term care facility*	*set up a manufacturing facility (a unit)*

(a) Google Book NGrams: Differences between English usage in the United States and United Kingdoms

Word	Effect Size	CI(Null)	US Usage	IN Usage
high	0.820	(0.02,0.03)	*I am in high school*	*by pass the high way (as a road)*
hum	0.740	(0.03, 0.04)	*more than hum and talk*	*hum busy hain (Indian English)*
main	0.691	(0.048, 0.074)	*your main attraction*	*main cool hoon (I am cool)*
ring	0.718	(0.054, 0.093)	*My belly piercing ring*	*on the ring road (a circular road)*
test	0.572	(0.03, 0.061)	*I failed the test*	*We won the test*
stand	0.589	(0.046, 0.07)	*I can't stand stupid people*	*Wait at the bus stand*

(b) Twitter: Differences between English usage in the United States and India

Table 3.3: Examples of statistically significant geographic variation of language detected by our method, GeoDist, between English usage in the USA and (a) UK (b)India (CI - the 98% Confidence Intervals under the null model)

Algorithm 2 ScoreSignificance (C, B, α)

Input: C: Corpus of text with R regions, B: Number of bootstrap samples, α: Confidence Interval threshold

Output: E: Computed effect sizes for each word w, CI: Computed confidence intervals for each word w

// Estimate the NULL distribution.

BS $\leftarrow \emptyset$ {Corpora from the NULL Distribution}. NULLSCORES(w) {Store the scores for w under null model.}

repeat

  Permute the labels assigned to text of C uniformly at random to obtain corpus C'

  BS $\leftarrow$ BS $\cup\, C'$

  Learn a model N using C' as the text.

  for $w \in \mathcal{V}$ **do**

    Compute Score(w) using N.

    Append Score(w) to NULLSCORES(w)

  end for

until $|\text{BS}| = B$

// Estimate the actual observed effect and compute confidence intervals.

Learn a model M using C as the text.

for $w \in \mathcal{V}$ **do**

  Compute Score(w) using M.

  $E(w) \leftarrow$ Score(w)

  Sort the scores in NULLSCORES(w).

  HCI(w) $\leftarrow 100\alpha$ percentile in NULLSCORES(w)

  LCI(w) $\leftarrow 100(1-\alpha)$ percentile in NULLSCORES(w)

  CI(w) $\leftarrow$ (LCI(w), HCI(w))

end for

return E, CI

Twitter Data This dataset consists of a sample of Tweets spanning 24 months starting from September 2011 to October 2013. Each Tweet includes the Tweet ID, Tweet and the geolocation if available. We partition these tweets by their location in two ways:

1. *States in the USA*: We consider Tweets originating in the United States and group the Tweets by the state in the United States they originated from. The joint corpus consists of 7 million Tweets.

2. *Countries*: We consider 11 million Tweets originating from USA, UK, India (IN) and Australia (AU) and partition the Tweets among these

four countries.

In order to obtain part of speech tags, for the tweets we use the TweetNLP POS Tagger[65].

3.4 Results and Analysis

In this section, we apply our methods to various data sets described above to identify words that are used differently across various geographic regions. We describe the results of our experiments below.

3.4.1 Geographical Variation Analysis

Table 3.1 shows words which are detected by the *Frequency* method. Note that `zucchini` is used rarely in the UK because a `zucchini` is referred to as a `courgette` in the UK. Yet another example is the word `freshman` which refers to a student in their first year at college in the US. However in the UK a `freshman` is known as a `fresher`. The *Frequency* method also detects terms that are specific to regional cultures like `touchdown`, an American football term and hence used very frequently in the US.

As we noted in Section 3.2.1, the *Syntactic* method detects words which differ in their syntactic roles. Table 3.2 shows words like `lift`, `cuddle` which are used as verbs in the US but predominantly as nouns in the UK. In particular `lift` in the UK also refers to an *elevator*. While in the USA, the word `cracking` is typically used as a verb (as in "the ice is cracking"), in the UK `cracking` is also used as an adjective and means "stunningly beautiful". The *Frequency* method in contrast would not be able to detect such syntactic variation since it focuses only on usage counts and not on syntax.

In Tables 3.3a and 3.3b we show several words identified by our GeoDist method. While `theatre` refers primarily to a building (where events are held) in the UK, in the US `theatre` also refers primarily to the study of the performing arts. The word `extract` is yet another example: `extract` in the US refers to food extracts but is used primarily as a verb in the UK. While in the US, the word `test` almost always refers to an `exam`, in India `test` has an additional meaning of a cricket match that is typically played over five days. An example usage of this meaning is "We are going to see the test match between India and Australia" or the "The test was drawn.". We reiterate here that the *Distributional* method picks up on finer distributional cues that the *Syntactic* or the *Frequency* method cannot detect. To illustrate this, observe that `theatre` is still used predominantly as a noun in both UK and the USA, but they differ in semantics which the *Syntactic* method fails to detect.

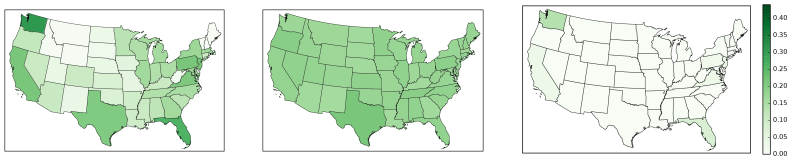
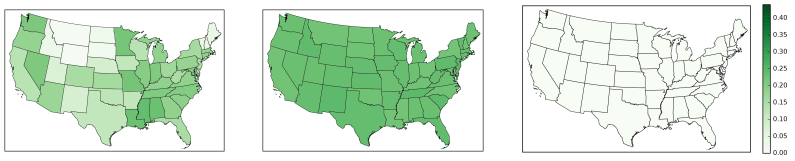
Word	Distances		
	Naive Distances	NULLMODEL	GeoDist(**Our Method**)
`buffalo`			
`twins`			
`space`			
`golden`			
`hand`			

Table 3.4: Sample set of words which differ in meaning (semantics) in different states of the USA. Note how incorporating the null model highlights only statistically significant changes. Observe how our method GeoDist correctly detects no change in `hand`.

Another clear pattern that emerges are "code-mixed words", which are regional language words that are incorporated into the variant of English (yet still retaining the meaning in the regional language). Examples of such words include `main` and `hum` which in India also mean "I" and "We" respectively in addition to their standard meanings. In Indian English, one can use `main` as "the main job is done" as well as "main free at noon. what about you?". In the second sentence `main` refers to "I" and means "I am free at noon. what about

you?".

Furthermore, we demonstrate that our method is capable of detecting changes in word meaning (usage) at finer scales (within states in a country). Table 3.4 shows a sample of the words in states of the USA which differ in semantic usage markedly from their overall semantics globally across the country.

Note that the usage of `buffalo` significantly differs in New York as compared to the rest of the USA. `buffalo` typically would refer to an animal in the rest of USA, but it refers to a place named *Buffalo* in New York. The word `queens` is yet another example where people in New York almost always refer to it as a place.

Other clear trends evident are words that are typically associated with states. Examples of such words include `golden`, `space` and `twins`. The word `golden` in California almost always refers to *The golden gate bridge* and `space` in Washington refers to *The space needle*. While `twins` in the rest of the country is dominantly associated with twin babies (or twin brothers), in the state of Minnesota, `twins` also refers to the state's baseball team *Minnesota Twins*.

Table 3.4 also illustrates the significance of incorporating the null model to detect which changes are significant. Observe how incorporating the null model renders several observed changes as being not significant thus highlighting statistically significant changes. Without incorporating the null model, one would erroneously conclude that `hand` has different semantic usage in several states. However on incorporating the null model, we notice that these are very likely due to random chance thus enabling us to reject this as signifying a true change.

These examples demonstrate the capability of our method to detect wide variety of variation across different scales of geography spanning regional differences to code-mixed words.

3.5 Related Work

Most of the related work can be organized into two areas: (a) Socio-variational linguistics (b) Word embeddings

Socio-variational linguistics: A large body of work studies how language varies according to geography and time [32, 55, 57, 59–61, 66, 67].

While previous work like [30, 32, 66, 68, 69] focus on temporal analysis of language variation, our work centers on methods to detect and analyze linguistic variation according to geography. A majority of these works also

either restrict themselves to two time periods or do not outline methods to detect when changes are significant. Recently [59] proposed methods to detect statistically significant linguistic change over time that hinge on timeseries analysis. Since their methods explicitly model word evolution as a time series, their methods cannot be trivially applied to detect geographical variation.

Several works on geographic variation [55–57, 70] focus on lexical variation. Bamman et al. [55] study lexical variation in social media like Twitter based on gender identity. Eisenstein et al. [57] describe a latent variable model to capture geographic lexical variation. Eisenstein et al. [58] outline a model to capture diffusion of lexical variation in social media. Different from these studies, our work seeks to identify semantic changes in word meaning (usage) not limited to lexical variation. The work that is most closely related to ours is that of Bamman et al. [60]. They propose a method to obtain geographically situated word embeddings and evaluate them on a semantic similarity task that seeks to identify words accounting for geographical location. Their evaluation typically focuses on named entities that are specific to geographic regions. Our work differs in several aspects: Unlike their work which does not explicitly seek to identify which words vary in semantics across regions, we propose methods to detect and identify which words vary across regions. While our work builds on their work to learn region specific word embeddings, we differentiate our work by proposing an appropriate null model, quantifying the change and assessing its significance. Furthermore our work is unique in the fact that we evaluate our method comprehensively on multiple web-scale datasets at different scales (both at a country level and state level).

Word Embeddings: The concept of using distributed representations to learn a mapping from symbolic data to continuous space dates back to Hinton [39]. In a landmark paper, Bengio et al. [40] proposed a neural language model to learn word embeddings and demonstrated that they outperform traditional n-gram based models. Mikolov et al. [71] proposed Skipgram models for learning word embeddings and demonstrated that they capture fine grained structures and linguistic regularities [20, 43]. Also Perozzi et al. [72] induce language networks over word embeddings to reveal rich but varied community structure. Finally these embeddings have been demonstrated to be useful features for several NLP tasks [44, 45, 73, 74].

3.6 Conclusions

In this chapter, we proposed a new method to detect linguistic change across geographic regions. Our method explicitly accounts for random variation,

quantifying not only the change but also its significance. This allows for more precise detection than previous methods.

We comprehensively evaluate our method on large datasets at different levels of granularity – from states in a country to countries spread across continents. Our methods are capable of detecting a rich set of changes attributed to word semantics, syntax, and code-mixing.

Chapter 4

Linguistic Variation across Domains with Applications to Named Entity Recognition

Now, on the St. Louis team we have Who's on first, What's on second, I Don't Know is on third.

Bud Abbott

Named Entity Recognition (NER) is a critical task for understanding textual content. While most NER systems demonstrate very good performance, this performance is typically measured on test data drawn from the same domain as the training data.

For example, most competitive Named Entity Recognition systems are trained on large amounts of labeled data from a given domain (like CoNLL or MUC) and evaluated on a held out test set drawn from the same domain [44, 75–78]. While such systems demonstrate high performance in-domain, content on the Internet can originate from multiple domains like Finance and Sports over which these systems perform quite poorly. Moreover one typically does not have access to large amounts of labeled examples on these domains to train robust domain specific models. This challenge is typically addressed through domain adaptation techniques [79–83]. Most existing work on domain adaptation like Feature Subsetting [81], Structural Correspondence

Work described in this chapter was done in collaboration with Yashar Mehdad and Troy Chevalier at Yahoo! Research.

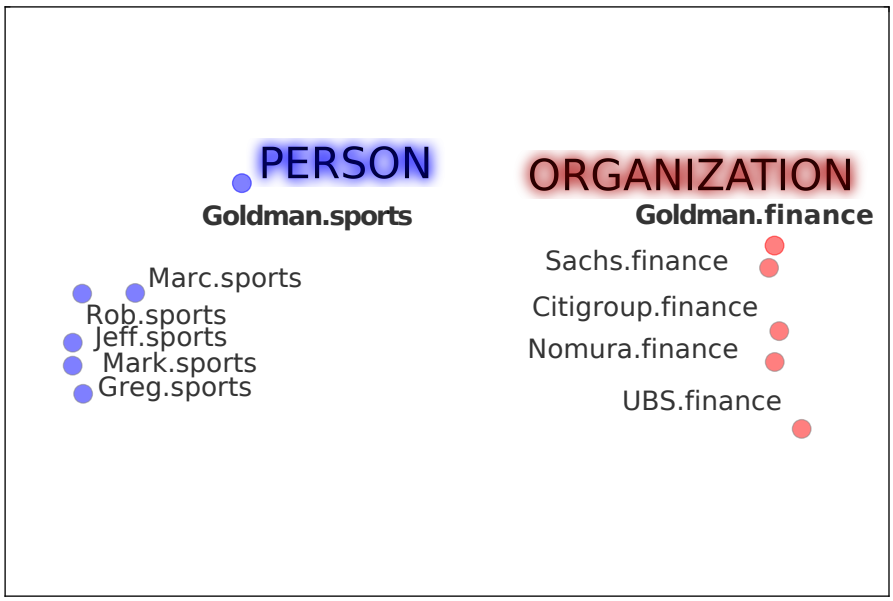


Figure 4.1: A 2-D projection of the semantic space learned using DomainDist capturing domain specific differences in the usage of the word `Goldman` between Sports and Finance. Note how `Goldman` is close to other banks in Finance domain, but close to other person names in Sports. Capturing such domain specific differences explicitly can allow a model to more effectively infer that `Goldman` is an *Organization* in Finance but a *Person* in Sports.

Learning [79, 84], learn a subset of features or learn dense representations of features that are more suited for domain adaptation. Different from these works, we explore word embeddings that *explicitly* capture domain specific differences while still capturing shared semantics across domains, and show that our proposed methods outperform several competitive baselines on domain adaptation for NER.

With recent advances in representation learning, word embeddings have been shown to be very useful features for several NLP tasks like POS Tagging, NER, and Sentiment Analysis [44, 73, 74]. One drawback of using generic word embeddings is that these word vectors do not capture domain specific differences in word semantics and usage. To illustrate this, consider articles from two distinct domains: (a) Sports and (b) Finance. The word `tackle` in the Sports domain is generally associated with moves in *football* and used as *"A defensive tackle"*. However in the domain of Finance, `tackle` is used to indicate problem solving as in *"The company needs to tackle the rising costs immediately"*.

Explicitly modeling such domain specific differences allows us to capture

linguistic variation between domains that serve as distinctive features to boost performance of a machine learning model on NLP tasks. In this work we propose methods to effectively model such domain specific differences of language. We then apply our methods to analyze domain specific differences in word semantics. Finally, we demonstrate the effectiveness of using domain specific word embeddings for the task of Named Entity Recognition in the domain adaptation setting. Figure 4.1 shows the domain specific differences captured by our method across two domains (a) Sports and (b) Finance. Observe how the domain specific embeddings that our method learns can easily capture the distinct usages of a word (in this case as a Person or an Organization). As we will show in Section 4.3 such distinctive representations can improve performance of Named Entity Recognition in different domains outperforming competitive baselines.

In a nut shell, our contributions are as follows:

- **Linguistic Variation across Domains**: Given a word w how does its usage differ across different domains? We analyze variation in word usage (semantics) across different domains like Finance and Sports using distributed word representations (Section 4.1.1).
- **NER systems for Sports and Finance**: We propose methods to effectively use such domain specific knowledge captured by word embeddings towards the task of Named Entity Recognition. In particular we show how to build state of the art NER systems for domains with scarce amount of annotated training data by adapting NER models learned primarily on domains with large amounts of annotated training data (Section 4.1.2).

4.1 Methods

In this section we propose (a) Two methods to model domain specific word semantics in order to explicitly capture linguistic differences between domains and (b) Two methods that use domain specific word embeddings to learn robust Named Entity Recognition models for different domains using domain adaptation.

4.1.1 Domain Specific Linguistic Variation

DomainDist

Given a corpus $\mathcal{C}$ with K domains and vocabulary $\mathcal{V}$, we seek to learn a domain specific word embedding $\phi_k : \mathcal{V} \mapsto \mathbb{R}^d$ using a neural language model where

$k \in \{1 \cdots K\}$. We apply the method discussed in [60, 85] to learn domain specific word embeddings. We briefly describe this approach below as pertaining to learning domain specific embeddings. For each word $w \in \mathcal{V}$ the model learns (1) A global embedding $\delta_{\text{MAIN}}(w)$ for the word ignoring all domain specific cues and (2) A differential embedding $\delta_k(w)$ that encodes deviations from the global embedding for w specific to domain k. The domain specific embedding $\phi_k(w)$ is computed as: $\phi_k(w) = \delta_{\text{MAIN}}(w) + \delta_k(w)$. The global word embeddings are randomly initialized, while the differential word embeddings are initialized to $\mathbf{0}$. We use the Skipgram objective function with hierarchical soft-max to learn the global and the differential embeddings. We set the learning rate $\alpha = 0.025$, context window size m to 10 and word embedding size d to be 100. An example of the domain specific linguistic variation captured by DomainDist is illustrated in Figure 4.1.

DomainSense

Here we outline yet another method to capture semantic variation in word usage across domains. We model the problem as follows:

- **Sense Specific Embeddings** We assume each word w has potentially S senses where we seek to learn not only an embedding for each sense of w but also infer what these senses are from the corpus $\mathcal{C}$.
- **Sense Proportions in Domains** The usage of w in each domain k can be characterized by a probability distribution $\pi_k(w)$ over the inferred senses of w.

To learn sense specific embeddings, we use the Adaptive Skipgram model proposed by [86] to automatically infer (a) the different senses a word w exhibits (b) a probability distribution $\pi(w)$ over the the different senses a word exhibits in the corpus and (c) an embedding for each sense of the word. Specifically, we combine the sub-corpora of different domains to form a single corpus $\mathcal{C}$. We then learn sense specific embeddings for each word w in $\mathcal{C}$ using the Adaptive Skipgram model. We set the number of dimensions d of the embedding to 100, the maximum number of senses a word has $S = 5$ and restrict the vocabulary to only words that occur more than 100 times.

Finally, given a word w we quantify the difference in the sense usage of w between two domains d_i and d_j as follows:

1. Disambiguate each occurrence of w in d_i and d_j using the method described by [86]. We can then estimate the sense distribution of word w in domain d_i, $\pi_{d_i}(w)$ as $Pr(\pi_{d_i}(w) = s) = \frac{\#_{d_i}(Sense(w)=s)}{\#_{d_i}(w)}$ where $\#_{d_i}(X)$ represents the count of number of times X is true in domain d_i.

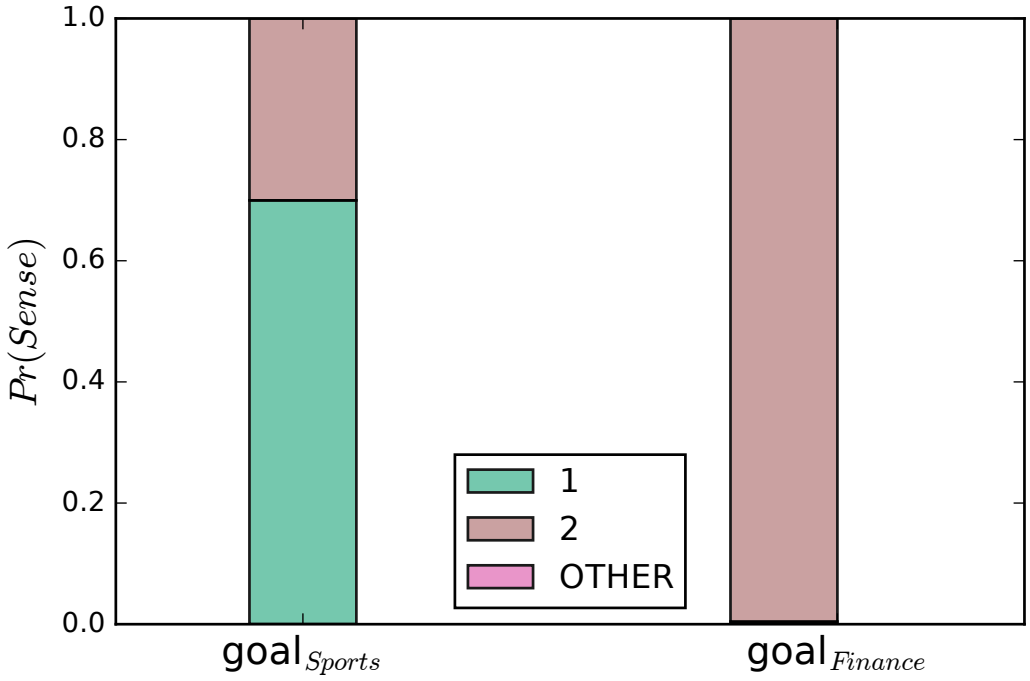


Figure 4.2: Different sense proportions of `goal` in Sports and Finance as computed by DomainSense. The word `goal` has two inferred senses as shown in Table 4.1: Sense1 corresponds to the sense of `goal` as a *score* in games or sports. Sense2 corresponds to the sense of `goal` as an *objective*. The usages of these senses is different in Sports and Finance. Note that in Sports, Sense1 is dominant while in Finance the usage is exclusively Sense2.

2. We then compute the Jennsen-Shannon Divergence (JSD) between the sense distributions of the word w between the two domains d_i and d_j to quantify the difference in sense usage of w between these domains.

Table 4.1 shows a small sample of words along with their inferred senses using this method. Figure 4.2 then depicts the domain specific difference in the sense usages of `goal` as computed by DomainSense.

While both DomainDist and DomainSense explicitly capture domain specific differences in word semantics, they differ in their underlying models. DomainDist captures domain specific word semantic/usage by directly learning domain specific word representations. DomainSense on the other hand infers different senses of a word and learns an embedding for each sense. Domain specific differences are then modeled by differences in sense usage of the word across domains. To illustrate this difference, consider the word `goal`. DomainDist will capture the fact that `goal` is associated with `match`, `winning`

Word	#(Senses)	Sense 1	Sense 2	Sense 3
tackle	2	handle, avoid, sidestep	linebacker, cornerback, defensive	–
track	3	field, swimming, lacrosse	song, album, remix	route, lane, dirt
board	2	committee, chairperson, chair	deck, boards, bench, boat, raft	–
heats	2	rounds, semifinals	cools, warmed, dried	–
goal	2	try, hat-trick, gamewinning	aim, mission, objective	–

Table 4.1: The senses inferred for a sample set of words by Adaptive Skipgram. Each word's sense is succinctly described by the nearest neighbors of that word's sense specific embedding. Note the different senses of words like `heats` and `tackle`. These senses are used in different proportions in various domains as shown for `goal` in Figure 4.2.

in Sports and capture this sense of `goal` in the *Sports* Specific Embedding. DomainSense in contrast will infer that `goal` has two senses overall (see Table 4.1) and then capture that in Sports both these senses are used. Moreover, the sense related to `score` is used 70% of the time while the sense associated with `objective` is estimated to be used 30% of the time in Sports. Finally, we empirically evaluate the effectiveness of both DomainDist and DomainSense for the task of NER (Section 4.3).

Error bounds on computation of JS Divergence for DomainSense

Lemma 1 (Lemma)**.** *Let $n_a(w)$ and $n_b(w)$ be the total number of occurrences of a word w in domains d_a and d_b respectively. The standard deviation in the JS divergence of the sense distribution of w across this domain pair is* $\mathcal{O}(\frac{1}{\sqrt{n_a(w)}} + \frac{1}{\sqrt{n_b(w)}})$

Proof. Assume that a word w has S senses where the probability distribution over the senses in domain d_a is given by $\mathbf{p} = (p_1, p_2, \ldots p_S)$ and that in domain d_b is given by $\mathbf{q} = (q_1, q_2, \ldots q_S)$.

The JS Divergence between the probability distributions $\mathbf{p}$ and $\mathbf{q}$ is given by:

$$JS(\mathbf{p}, \mathbf{q}) = \frac{\sum p_i \log p_i + \sum q_i \log q_i}{2} - \sum \frac{p_i + q_i}{2} \log \frac{(p_i + q_i)}{2} \tag{4.1}$$

Now note that each of p_i and q_i are sample MLE estimates of multinomial distribution. The standard deviations of each of these sample MLE estimates denoted by σ_{p_i} and σ_{q_i} are as follows:

$$\sigma_{p_i} = \sqrt{\frac{p_i^*(1-p_i^*)}{n_a(w)}} \tag{4.2}$$

$$\sigma_{q_i} = \sqrt{\frac{q_i^*(1-q_i^*)}{n_b(w)}} \tag{4.3}$$

□

where p_i^*, q_i^* are the true values of the particular probabilities. In order to quantify the standard deviation in the resulting computation of $JS(\mathbf{p}, \mathbf{q})$ which we denote by $\sigma_{JS(\mathbf{p},\mathbf{q})}$, we now apply the rule for *propagation of uncertainty*[1], which yields:

$$\sigma_{JS(\mathbf{p},\mathbf{q})} = \sqrt{\begin{array}{c}\sum (\frac{\partial JS(\mathbf{p},\mathbf{q})}{\partial p_i})^2 \sigma_{p_i^2} + \\ \sum (\frac{\partial JS(\mathbf{p},\mathbf{q})}{\partial q_i})^2 \sigma_{q_i^2}\end{array}} \tag{4.4}$$

The coefficients $\frac{\partial JS(\mathbf{p},\mathbf{q})}{\partial p_i}$ and $\frac{\partial JS(\mathbf{p},\mathbf{q})}{\partial q_i}$ are called the *sensitivity coefficients*. Substituting the computations for σ_{p_i} and σ_{q_i} in Equation 4.4 completes the proof. This fact that the uncertainty in the JS Divergence is inversely proportional to the square root of the sample size enables us to get reasonably accurate estimates by choosing an appropriate sample size. The bound above implies the following: (a) We can quantify the uncertainty in our estimates based on the frequency of the words in the corpus and interpret our results with greater confidence. Very rare words would have larger deviations. (b) Depending on an applications sensitivity to error, we can estimate the appropriate sample size needed. [2]

4.1.2 Domain Adaptation for Named Entity Recognition

In the previous section, we described methods to capture domain specific linguistic variation in word semantics/usage by learning word embeddings that

[1] We ignore covariance terms as each parameter is estimated independently.

[2] Our reported results (see Figure 4.3) computing JS Divergence all have counts >= 1000 in both domains and hence have low errors.

Feature	Description
Tokens	w_i for i in $\{-2, \cdots +2\}$, w_i and w_{i+1} for i in $\{-1, 0\}$
Embeddings	Embeddings for w_i for i in $\{-2, \cdots +2\}$
Morphological	Shape and capitalization features, token prefixes and suffixes (up-to length 4), numbers and punctuation.

Table 4.2: Summary of features we use for learning Named Entity Recognition (NER) models.

are domain specific. In this section, we outline how to learn NER models for the various domains using such word embeddings as features.

As in previous works, we treat NER as a sequence labeling problem. To train, we use CRFsuite [87] with L-BFGS algorithm. We use a BILOU label encoding scheme. The features we use are listed in Table 4.2. Our main features are tokens and word embeddings, within a small window of the target token. We investigate using different kinds of embeddings listed below:

- **Generic Word2vec embeddings**: We learn generic Skipgram embeddings using English Wikipedia.
- **Domain/Sense Specific Word Embeddings**: We experiment by using the embeddings learned using DomainDist and DomainSense.

DomainEmbNER

Here, we outline the supervised domain adaptation method that uses domain specific word embeddings to learn NER models that significantly outperform other baselines on NER task in the domain adaptation setting. In this setting, we are interested in a Named Entity Recognition system for domain $\mathcal{T}$. However training data available for domain $\mathcal{T}$ is scarce but we have access to a source domain $\mathcal{S}$ for which we have large number of training examples. We would like to perform domain adaptation by learning a model using the large amount of training data in source domain $\mathcal{S}$ and adapt it to work well on the target domain $\mathcal{T}$. There exist a number of methods for the task of supervised domain adaptation [82]. We use a simple method for this task outlined below:

1. Combine the training data from $\mathcal{S}$ and $\mathcal{T}$. Note again that $|\mathcal{S}| >> |\mathcal{T}|$ in our setting.

Algorithm 3 ActiveDomainEmbNER (S, T, B, k)

Input: S: Training data for NER in the source domain, T: Unlabeled data for the task of NER in the target domain which is separate and distinct from the final test set. B: Number of actively labeled examples, k: Batch size of actively labeled examples.

Output: M: NER model

```
C ← S
repeat
    Learn a model M using C
    Evaluate M on T.
    E ← Sort the evaluated phrases of T in ascending order of model confidence (probability) and remove top k least confident examples.
    Ask an expert to label each example in E and add them to C.
    C ← C ∪ E
until |C|≥ |S|+B
return M
```

2. Extract the features outlined for training the CRF model as out-lined in Table 4.2. Note that we experiment with different kinds of word embeddings and baselines.

3. Learn a CRF model using this training data.

4. Evaluate the learned CRF model on the domain specific test data set and report the performance.

   As we will show in Section 4.3, using domain specific word embeddings improves the performance of NER on these target domains significantly and outperforms previous baselines for this task.

ActiveDomainEmbNER

In this section, we describe how we can learn a Named Entity Recognition system, assuming we have no labeled training data in the target domain. We can however request for a small number of examples B to be labeled by annotators. In such a setting, one can actively choose the set of examples that need to be labeled which will be most useful to learn a good model. We propose a method to actively label examples for the purpose of domain adaptation which we describe succinctly in Algorithm 3. In Section 4.3 we show that by merely asking for an editorial to label 1500 sentences, we can achieve performance close to state of art in this setting.

	ConLL	Yahoo Finance	Yahoo Sports
# Sents (train)	14808	6439	4077
# Sents (test)	3648	4294	2719
Domain	News	Finance	Sports

Table 4.3: Summary of our editorially labeled data.

4.2 Datasets

In this section, we outline details of the datasets we consider for our experiments.

Our datasets can be classified into 2 categories (a) Unlabeled data for learning word embeddings and (b) Labeled data for the task of NER, each of which we describe below.

4.2.1 Unlabeled Data

We use the following unlabeled data sets for the purpose of learning word embeddings. We consider (a) all sentences of English Wikipedia (b) a random sample of 1 Million articles from Yahoo! Finance restricting our language to only English and (c) a random sample of 1 Million articles from Yahoo! Sports restricting our language to only English.

4.2.2 Labeled Data

We also use labeled data sets for the task of learning NER models which we summarize in Table 4.3.

4.3 Experiments

Here, we briefly describe the results of our experiments on (a) Domain Specific Linguistic Variation and (b) Domain Adaptation for Named Entity Recognition.

4.3.1 Domain Specific Linguistic Variation

Table 4.4 shows some of the semantic differences in word usage captured by DomainDist. Observe that the method is able to capture words like `quote, overtime, hurdles` that have alternative meanings (semantics) in a domain. For example, the word `hurdles` means `challenges` in Finance but a kind of athletic `race` in Sports. In addition to capturing words that differ in semantics, note that DomainDist also uncovers differing semantic usages of entities as well, as depicted in Figure 4.1. In the domain of Finance, `Anthem` refers

Word	Distance	Usage(Finance)	Usage(Sports)
quote	0.70	an official document (used as "details of the quote")	an aphorism (a saying)
selections	0.94	selections of menus, checkouts, products	selection to an honor (an award, recognition)
overtime	0.93	used as "overtime pay"	A checkpoint in a match (used as "double overtime")
Assists	0.89	Assist, Coordinate (as in help)	A term in American football
hurdles	0.89	setbacks, obstacles	a type of athletic race
Anthem	0.97	Health Insurance Company (similar to Aetna, Metlife)	Song (of a band, team etc) used as "Sing the anthem"
Hays	0.88	Hays Advertising	Last Name of person
Schneider	0.88	Schneider Electric (company)	Last name of a person
Hugo	0.88	Name of a company (like Hugo Boss)	First Name of a person

Table 4.4: Sample words that depict the differences (and the measured distance) in word semantics between Sports and Finance by DomainDist. Note that we capture semantic differences in words that are entities (`Anthem, Schneider`) and non-entities (`quote, overtime`).

to a health insurance company but `Anthem` in Sports dominantly refers to a `song` like a team anthem. In Figure 4.3 a sample set of words detected by DomainSense are shown. Note once again, that we are able to capture domain specific differences between words (both entities and non-entities). Furthermore, DomainSense is able to quantify the proportion of each word sense usage in various domains. For example, the word `tackle` is used exclusively in Finance as a verb that means `to solve`, whereas in Sports `tackle` is dominantly used to refer to an `American football move`. Note that in Sports, the sense of `tackle` that means `to solve` is only used 30% of the time.

This ability to capture differing entity roles (like Organizations and Persons) provides an insight into the effectiveness of domain specific embeddings for improved performance on Named Entity Recognition.

4.3.2 Domain Adaptation for Named Entity Recognition

In this section, we report the results of using our DomainDIst and DomainSense word embeddings for the task of Named Entity Recognition on Finance

and Sports Domains in the domain adaptation setting as described in Section 4.1. We also outline the baseline methods we compare to below:

Baseline methods

Since our setting is the setting of domain adaptation for Named Entity Recognition, we consider several competitive baselines for this task:

- **CoNLL-only Model**: We consider a simple baseline where we train a NER model only using CoNLL data and generic Wikipedia Embeddings without any adaptation to the target domain.
- **Feature Subsetting**: This domain adaptation method tries to penalize features which demonstrate large divergence between source and target domains [81]. It is worth noting that this models the task of NER as a classification problem and not a structured prediction problem. [3]
- **Online-FLORS**: FLORS learns robust representations of each word based on distributional features and counts, to boost performance across domains and treats the tagging problem as a classification problem. We also use a random sample of $100K$ unlabeled sentences from each domain which FLORS uses to enrich the robustness of representations learned. We consider a scalable version of FLORS [88].
- **FEMA**: Fema [89] learns low dimensional embeddings of the features used in a CRF model by using a variant of the Skipgram Model [71]. These features can be used to learn a model for sequence tagging. While they demonstrate their method on Part of Speech tagging, the method itself is general and can be applied to other tasks like Named Entity Recognition as well and provide a nice replacement for word embeddings as features in a CRF model. We used 100 dimensional FEMA embeddings in our experiment. [4]

Results and Discussion

Table 4.5 shows the performance of our methods and other baselines on Finance and Sports. First note that a CoNLL-only model without any domain adaptation results in poor performance. Domain Adaptation methods like

[3] We use the implementation of feature sub-setting for Named Entity Recognition provided by https://github.com/siqil/udaner.

[4] We use the open source implementation provided at https://github.com/yiyang-gt/feat2vec

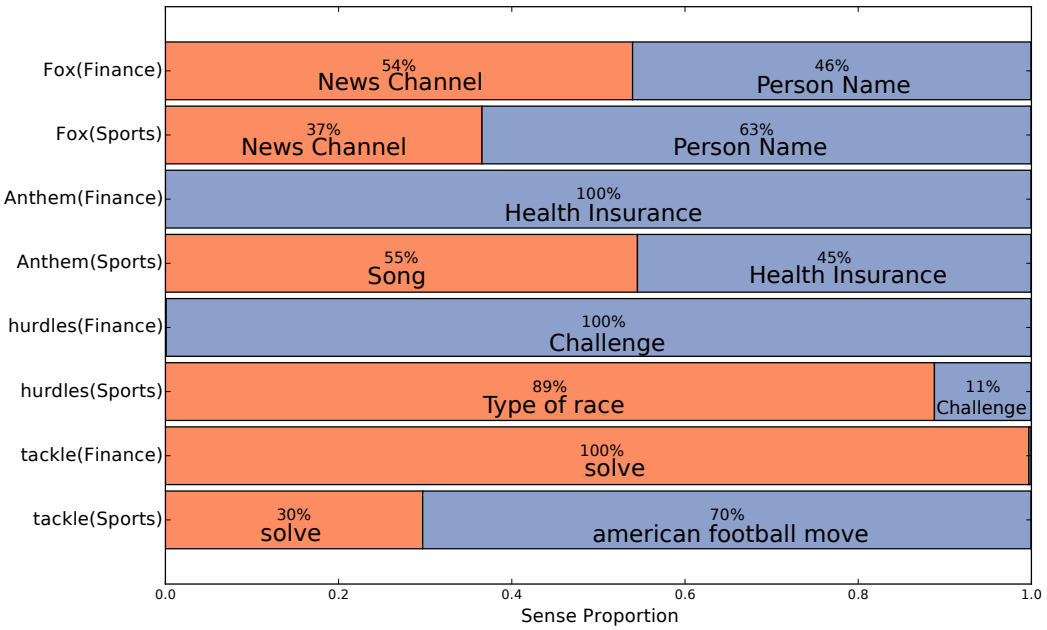


Figure 4.3: Sample set of words and their sense proportions in Sports and Finance as computed using DomainSense. Note the differences in sense usages of `Anthem`, `hurdles` and other words.

Feature Subsetting and FLORS that model Named Entity Recognition as a classification problem, rather than a sequence prediction problem perform even worse. In contrast Fema which learns dense representations of CRF features which can then be used to learn a more robust CRF model (that is more suited to domain adaptation) yields an significantly improved F1 score of 67.70 on Finance and 82.48 on Sports respectively. Empirically we observe that using DomainSense embeddings improves the performance over the ConLL only model, but does not perform as well on this task (especially in Finance). We hypothesize while decomposing a word into multiple fine-grained senses is useful to capture semantic variation, using fine-grained sense embeddings for every word results in a overly complex decision space when used for tasks like NER. Finally observe that DomainDist which learns domain specific embeddings (without explicitly decomposing words into their senses) outperforms all these methods in both domains. This superior performance results from the ability to capture useful broad domain specific differences more effectively.

In Table 4.6, we evaluate the performance of using domain specific word embeddings [5] against using just generic Wikipedia based word embeddings

[5]For brevity. we present results here only DomainDist embeddings as DomainDist

Data	Method	Finance			Sports		
		P	**R**	**F1**	**P**	**R**	**F1**
ConLL-Only	Wikipedia	51.32	54.86	53.03	74.09	68.31	71.09
ConLL + Target	Feature Subsetting	34.18	45.28	37.54	49.89	48.63	45.80
ConLL + Target	FLORS	35.75	46.78	40.53	63.48	62.18	62.82
ConLL + Target	FEMA	67.30	68.10	67.70	83.18	81.79	82.48
ConLL + Target	Wikipedia Embeddings	70.97	72.23	71.59	86.17	85.14	85.65
ConLL + Target	DOMAINSENSE embeddings	67.22	68.0	67.61	83.23	81.82	82.52
ConLL + Target	DOMAINDIST embeddings	**71.62**	**72.5**	**72.06**	**85.72**	**85.88**	**85.8**
In-domain (Upper Bound)	DOMAIN embeddings	*76.72*	*70.97*	*73.73*	*90.03*	*88.16*	*89.09*

Table 4.5: Performance of various domain adaptation methods on Named Entity Recognition in the target domains. The target domain (Target) here is one of Finance or Sports. Number of training sentences used from Finance:3219 while for Sports we use 2038 sentences. We show Precision (P), Recall (R) and F1.

α	**Finance**		**Sports**	
	Wiki	DOMAINDIST	Wiki	DOMAINDIST
0.1	61.28	**62.50**	80.18	**80.81**
0.2	66.36	**66.61**	83.61	**83.91**
0.3	67.57	**68.35**	84.42	**85.34**
0.4	69.40	**70.61**	85.49	**85.83**
0.5	71.20	**71.55**	86.30	**86.26**
0.6	71.12	**71.71**	87.08	**87.11**
0.7	72.44	72.42	87.07	**87.25**
0.8	72.99	**73.33**	88.12	**88.45**
0.9	73.59	**74.02**	88.00	**88.26**

Table 4.6: Performance of DOMAINEMBNER using DOMAINDIST Embeddings versus Wikipedia Embeddings on NER task against different proportions of training data α in target domain.

as a function of available training data α. First observe that on an average,

embeddings are the best performing embeddings.

#(Sents)	Training Data %		Finance (F1)	Sports (F1)
	Finance	Sports		
500	7.7	12.2	67.68	83.94
1000	15.5	24.5	69.00	85.53
1500	23.2	36.7	69.79	86.78
2000	31.0	49.0	70.53	87.14

Table 4.7: Performance of ActiveDomainDistNER on the target domains of Finance and Sports using DomainDist as a function of actively labeled sentences.

using DomainDist word embeddings improves the performance over using generic Wikipedia based embeddings. Observe that in general, as the amount of training data in the *target* domain increases, the advantage (gain) of using domain specific embeddings reduces. For example, when only 10% of training data is available for Finance, using domain specific word embeddings results in F1 Score gain of **1.22**(62.50-61.28). However when 90% of training data is available in Finance we get a small but still significant boost of **0.43** (74.02-73.59) in the F1 score on using domain specific word embeddings. We explain this by noting that as the proportion of training data in the target domain increases, the model is able to pick up on domain specific cues and fine-tune its decision boundary better without needing to rely too much on the domain specific cues captured by the domain specific word embeddings.

Table 4.7 shows the performance of ActiveDomainEmbNER as a function of number of sentences we sought to be actively labeled. Note that merely requiring 1500 sentences to be manually annotated, we are able to achieve close to state of art F1 performance (**69.79** on Finance and **86.78** on Sports respectively).

4.4 Related Work

Related work can be organized into two areas: (a) Socio-variational linguistics and (b) Domain Adaptation.

Socio-variational linguistics Several works study how language varies according to geography and time [32, 55, 57, 59–61, 66, 67, 85, 90–92]. Different from these studies, our work seeks to identify semantic changes in word meaning (usage) across domains with a focus on improving performance on an NLP task like NER. The methods outlined in [60, 85] are most closely related to our work. While we directly build on methods outlined by them we differentiate ourselves

from their work by explicitly modeling differences in the usage of different senses of words. While the methods outlined in [60, 85] capture domain specific differences, they do not explicitly model the fact that words have multiple senses and their usage in a domain is a mixture of different proportions over these senses which can be explicitly quantified. Finally we apply these methods to identify and analyze semantic variation in word usage across domains like Sports and Finance, highlight interesting examples of such variation prevalent across these domains with applications to Named Entity Recognition.

Domain Adaptation There is a long line of work on domain adaptation [77, 79–81, 84, 89, 93–95]. Most of these works can be classified based on the strategies they use as follows: (a) Instance Weighting Methods [80, 81] (b) Regularization based methods [93, 94] and (c) Representation Induction [79, 84, 89, 95]. Our method of learning domain specific word embeddings in an unsupervised manner can be placed into this final category. Finally an excellent survey of various domain adaptation algorithms for NLP is provided by [82, 83].

4.5 Conclusions

In this chapter, we proposed methods to detect and analyze semantic differences in word usage across multiple domains. Our methods explicitly capture domain specific cues by learning word embeddings from unlabeled text and scale well to large web scale data sets. Furthermore, we outline methods that leverage such domain specific linguistic variation and knowledge effectively to boost performance on NLP tasks like Named Entity Recognition on domains with scarce training data and requiring domain adaptation. Our methods not only out-perform previous competitive baselines but also require a very small number of manually annotated sentences in the target domain to achieve competitive performance. We believe our work sets the stage for new directions and further research into applications that effectively model linguistic variation across domains to improve the performance, applicability and usability of NLP systems analyzing the diverse content on the Internet.

Chapter 5

Learning Latent User Traits from Language on Social Media

Only strong personalities can endure history, the weak ones are extinguished by it.

Friedrich Nietzsche

What are the fundamental traits of people? Psychology has long tried to answer this question by deriving the latent factors that distinguish people and are stable across time and populations[96–99]. The dominant approaches to identify these traits are rooted in the lexical hypothesis [97, 100–104], which assumes that the words used by people are a window into their personality [96]. However, in practice, most approaches only utilize the lexical hypothesis indirectly, using questionnaires to ask people whether words describe themselves, rather than studying people's everyday language use directly.

Leveraging the recent growth of social media, we derive a trait model based on the everyday linguistic behavior, "in the wild". Our approach analyzes the words and phrases of tens of thousands of users and their millions of messages to infer traits. In line with trait theory [105], we seek a small number of generalizable and stable traits that capture meaningful differences between people. Our method does not rely on any hand-crafted lexica or questionnaires and it scales well to leverage the large amount of data available on social media. While some have leveraged social media and open-vocabulary techniques to assess *existing* trait models (e.g. big 5 [103], the dark triad [106]), to the best

Work described in this chapter was done in collaboration with H. Andrew Schwartz, Margaret Kern, David Stillwell, Michal Kosinski, Lyle Ungar and Steven Skiena.

of our knowledge, none have attempted to *infer* the latent traits themselves.

We seek to determine effectiveness of such *behavior-based linguistic traits* (BLTs) compared to traits derived from questionnaires. Our traits have two key goals:

- **Generalizability**: The factors need to be generalizable across a large variety of predictive tasks without being fit apriori to any particular task.
- **Endurance**: We seek factors that are stable over time and also over populations. Specifically we seek that factor scores of users over time should be correlated. Similarly the learned factors should be stable across different sub-samples of the population.

We use an extensive battery of evaluations to determine how *BLTs* align with our key goals including several predictive tasks involving psychological variables (Depression Scores) and social-demographic variables (IQ, Income and Likes) as well as tests for temporal validity and dropout-validity.

5.1 Background

In this section, we discuss related work on modeling personality traits and the role of language in modeling or predicting such traits. Most prior work revolves around two major themes: (a) Constructs and models to capture personality with an aim to gain psychological insights and (b) Predictive models that either predict personality traits from language or use personality traits to predict psychological variables. We discuss each of these themes in detail below.

5.1.1 Modeling Personality

Psychologists have long sought to characterize fundamental traits that distinguish people. In order to characterize personality traits, personality psychologists build on a lexical approach which assumes that the basic traits that distinguish people can be characterized by *words* [96, 107]. In fact the importance of language in psychology is underscored by [108] as

> Language is the most common and reliable way for people to translate their internal thoughts and emotions into a form that others can understand. Words and language, then, are the very stuff of psychology and communication.

Consequently a long line of work in psychology seeks to characterize these traits based on words that people use. The dominant approach towards

characterizing these traits is based on hand-crafted lexica and dictionaries [96, 97, 100–102, 109]. First, a lexicon of 18000 words that distinguish one person from another based on an English dictionary was proposed by [96]. Based on this lexicon, a multi-dimensional model of personality that comprised of only 35 factors was derived using a subset of 4500 words with techniques based on semantic clustering[97]. Further reduction using oblique factor analysis finally resulted in only 12 factors which were incorporated into the 16-PF (Personality factor) questionnaire [100]. Inspired by [100], several works sought to further capture/refine the dimensional structure of traits, which ultimately culminated in the well-known BIG5 factor structure [101, 102, 109, 110]. Psychologists have since developed questionnaires that capture these personality traits which users can self-report [99, 103, 111]. However questionnaires or lexicon based approaches suffer from the following drawbacks:

- **Ignore every day language**: Hand crafted lexicons that are used to infer personality traits are not reflective of every day language use and do not contain high frequency words that are used in daily language.
- **Not as generic**: The BIG5 factors claim to be fairly broad and generic thus capturing a wide variety of human behavior. However there are plenty of personality dimensions that the BIG5 do not capture effectively [112].

Recently these limitations have been addressed by proposing an open-ended approach to learning personality traits by extracting common themes from self-narrative texts [113]. First, using a data-set of 1165 open-ended self-descriptive narratives, a factor analysis is performed on the most frequently used adjectives to reveal latent factors. Finally these latent factors are shown to correlate moderately with the BIG5 factors and reveal psychologically meaningful dimensions.

5.1.2 Predictive Models of Personality

With recent advances in the evolution of social media, researchers have sought to predict the personality of users, as well as several psychological and demographic variables like satisfaction with life, depression scores etc. leveraging the massive amounts of data available on social media [114–123]. In line with the focus of this chapter, we pre-dominantly discuss work related to prediction of personality from language on social media.

Holtgraves et al. analyzed text messages of participants and showed that language use correlated well with personality and LIWC categories[115]. Sumner et al. analyzed the relationship between personality of users and their

activity of Facebook and demonstrated that their activity is significantly correlated with their personality[116]. Noting the correlation between language use and personality, Golbeck et al. leverage the the massive amount of everyday text available on social media like Twitter, and propose a method to predict personality of a user based on their posts on Twitter [117]. Similarly Iacobelli et al. conduct a large scale personality classification of bloggers and show that the best performing model is a combination of several linguistic features like stemmed bigrams, common words etc [118]. Furthermore, they show that using only a common dictionary like LIWC[108] does not perform as well on this task highlighting the need for more refined and complex linguistic features. Plank and Hovy conduct a large scale linguistic analyses of 1.2 million Tweets and propose a model to predict Myers-Briggs personality types from language [120]. Finally, recent works have shown that moving beyond words and incorporating complex features like distributed sentence representations serve as enriched features to improve the performance of the task of personality prediction [122, 123].

5.2 Materials and Methods

In this section, we describe the details of our dataset, our proposed method to learn latent factors and the design of experiments to evaluate the learned factors.

Datasets

We consider a dataset of 203, 561, 17 Facebook status messages over 152, 845 distinct users obtained using the MyPersonality application [124]. We restrict our analysis only to users who have posted more than 1000 words overall, filter out all users who claim that they are not from the US and filter out all messages that are not English[1]. Among these users, 49139 have data on age, gender and their BIG5 personality scores. A few sample messages are shown below:

- goodbye to anybody i didn't get to chill with before i left
- a relaxed mind mks u see things in a better shade whose existance were merely ignored

[1]We use the language detection tool in DLA toolkit to detect the language.

```
  • scars heal , glory fades and all we're left with are
the memories made
```

About 62.8% of the users in our data-set are female. The age distribution is skewed towards younger people with the median age of 22 years and a mean age of 25.49 years. While we believe that our learned latent factors should capture age and gender, we also investigate *residualizing* out demographic factors like age and gender as well. Finally as a pre-processing stage, all messages are tokenized and stop-words are removed. [2]

5.2.1 Factor Generation

We considered several methods to learn our latent factors. Before describing our proposed method, we discuss a few alternative methods we considered to learn latent factors.

Alternative methods

- **Latent Dirichlet Allocation (LDA)**: Here, we model each user as a document and latent factors as topics. LDA [125] then enables us to learn these factors (topics) from a large corpus of user text. Each user is then represented as a mixture over these learned factors. We use the MALLET [126] toolkit to learn LDA factors, set $\alpha = 5$ and enable hyper-parameter optimization. We optimize the hyper-parameters every 20 iterations with a burn-in of 10 iterations.

  However we noticed that all the LDA factors were negatively correlated with each other. This can be explained by noting that the factor scores obtained by LDA for each user must sum to 1. This implies that a user cannot be simultaneously high or low on several factors. This implication suggests that LDA (a probabilistic model) is not well-suited for modeling latent factors (which are not probabilistic). Furthermore this implies that any latent variable probabilistic model that models a user as a probability distribution over latent factors is not suitable for modeling traits.

- **Singular Value Decomposition (SVD)**: Let $\mathcal{U}$ be the set of users and $\mathcal{V}$ be a finite set of vocabulary terms corresponding to this set of users. We construct M a term-user matrix and compute a low-rank

[2] We also only restrict ourselves to users younger than 65 years.

approximation of M using SVD which factors M into three matrices: U, Σ and V^*. Formally the matrix M is approximated as:

$$M \approx U_{|\mathcal{V}|\times k}\Sigma_{k\times k}V^*_{k\times|\mathcal{U}|}$$

Note that U is a matrix that represents the loading of each word onto the latent factors and V^* represents the factor scores for each user [3].

Finally, it is possible to obtain more interpretable factors by rotating the basis matrix U (also called the factor loading matrix) which will typically result in more interpretable factors[4].

While SVD does not have the drawbacks of LDA, we observed that a more general version of dimensionality reduction known as "factor analysis (FA)" demonstrates better empirical performance on predictive tasks and motivates FA as our proposed method to capture traits.

Proposed Method: Factor Analysis (FA)

Factor Analysis (FA) seeks to represent a set of variables as linear combinations of a small number of latent factors and has a long history of use in psychology. Formally, given a matrix M, factor analysis seeks to learn latent factors F and a loading matrix L such that:

$$M = LF + E \tag{5.1}$$

where E represents an error matrix. While SVD seeks to learn factors that account for all of the variance, FA is more general and learns factors that account for the common variance but allows for some residual variance not explained by the latent factors. Therefore we investigate using Factor Analysis (FA) to learn latent user factors by applying FA on the User-Term matrix. Finally, we also investigated various rotations of the loading matrix L to obtain potentially more interpretable factors[5].

Figure 5.1 shows word clouds corresponding to the most and least correlated words for each of the factors learned using FA. Observe that the factors capture both emotion words like `life, heart, happiness, love` (see FA:F1(+)) and non-emotion words like `week-end, school, work, tomorrow, tonight`

[3] Σ represents how much of the underlying variance is explained by each factor. Also SVD computes the *best* rank-k approximation to M.

[4] We use promax-equamax as the default rotation method in all our analyses

[5] We preferred FA over SVD as our method to learn factors as it is more general and showed better predictive power empirically.

(see FA:F1(-)).In summary, these observations suggest that our factors capture a variety of behavioral cues including demographic variables.

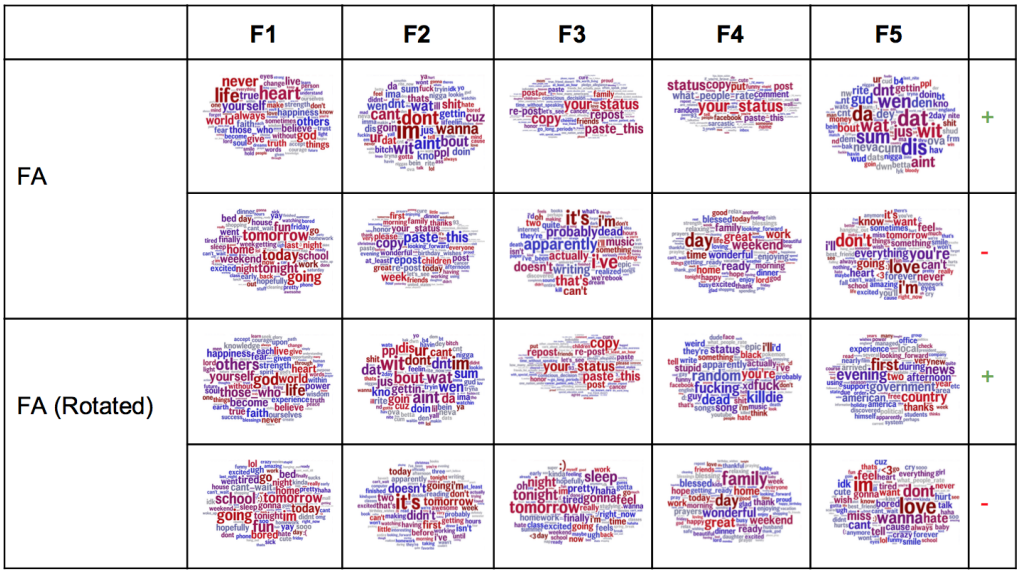


Figure 5.1: Word clouds showing the most/least correlated words for each FA factor as obtained using Differential Language Analysis. Note the presence of both non-emotion words (`home, weekend, tonight`) in FA:F1(-) as well as emotion words (`heart, love, life`) in FA:F1(+) suggesting the wide range of behavior captured by these factors.

We also explore how our factors inter-relate as well as how they relate with the BIG5. We used Pearson Product-Moment Correlation Coefficients (Pearson's R) to quantify the relationship of our factors against the BIG5 factor scores over the same set of users. If our factors do capture some aspects of personality, then we expect to see slight or moderate correlation with the BIG5 factors. Factors which are generalizable will display slight to moderate correlation structure with existing factor structures like BIG5.

Figure 5.2 shows the correlations of our learned factors with the BIG5 factors. Observe the diagonal which suggests a moderate correlation of BLT's with the Big5 (for eg. factor `F1` obtained using FA correlates well with OPENNESS and `F2` correlates well with extra-version).

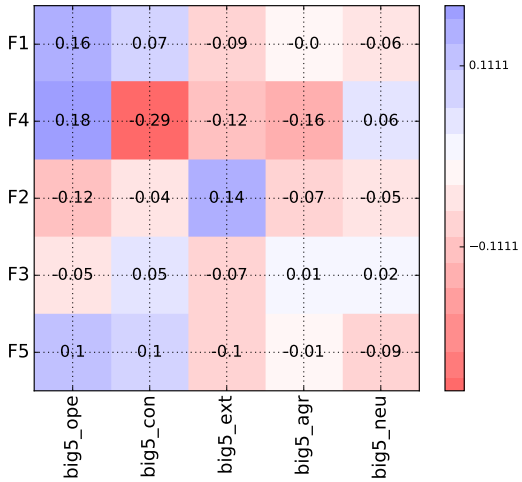


Figure 5.2: Correlation structure of learned factors using FA with BIG5 with rotation. One implication of performing a rotation (Promax-equamax) is that rotated loadings are sparse and potentially more interpretable (the factors have been re-arranged to highlight the diagonal).

5.3 Evaluation

In this section, we present methods to evaluate our factors comprehensively. Our evaluations broadly seek to quantify two aspects of our learned traits: *generalizability* and *endurance*. To that end, our evaluations are multi-fold which we discuss below:

Predictive Validity

Predictive validity seeks to measure the *generalizability* of the learned factors by measuring their predictive performance on a number of tasks. We group these evaluations into two categories as follows:

Questionnaire/Survey Outcomes

- LIFE SATISFACTION (SWL): We predict the satisfaction with life (SWL) score of users which was obtained through the 5-item Satisfaction with Life scale questionnaire.
- DEPRESSION RATINGS: For each user, we obtain their depression ratings by administering a 20-item questionnaire as specified by Center for Epidemiological Studies-Depression (CESD). We then use our inferred factors to evaluate how well we can predict these ratings.

- BIG5QUESTIONS: We consider Questions $21 - 100$ in the personality questionnaire the users answered when they need to obtain their BIG5 scores. The task then is to predict the user's response to these 80 questions[6].

5.3.1 Behavioral/Economic Outcomes

- FRIENDSIZE: In this task, we want to predict the number of friends a user has on his social network. This is a regression task for which we report our performance using Pearson's Product Moment Correlation (Pearson's R).

- INCOME: Here, we want to predict the income of a user based on his language use on social media. Specifically, we predict the logarithm of the income using our inferred factors. This is also a regression task for which we report our performance using Pearson's R.

- INTELLIGENCE QUOTIENT (IQ): We predict the IQ of a user based on his language use on social media.

- LIKES: We predict a small set of broad categories that a user likes. These broad categories are categories like ROCK MUSIC BANDS, GAMING or HOBBIES. We obtain these broad categories of likes as follows: We are given a matrix N of Users and what fine-grained categories they like on Facebook. We consider only the top 10000 likes by popularity. We then cluster the users based on their likes use a Non-negative matrix factorization (NMF) to reduce the dimensionality of N to obtain about a small number of like clusters (we compute 20 clusters). As an illustration, we show below one such broad level cluster which corresponds to music bands in the metal genre of music:

```
• Disturbed • System of a Down  • Linkedin Park
• Slipknot • Avenged Sevenfold • Breaking Benjamin
• Bullet for my Valentine • Metallica • Korn
```

Learning Predictive Models We primarily use linear models for both classification and regression. For regression tasks we use Linear Regression

[6]We consider only Questions $21 - 100$ and ignore the first 20 questions since all users answered the first 20 questions and these were directly used to compute their BIG5 scores which would significantly be an advantage for BIG5.

with $L2$ penalty (Ridge Regression) while for classification tasks, we train a Logistic Regression classifier. We restrict ourselves to linear models to ensure our models are interpretable and reveal the inherent predictive power of the factors. We set our hyper parameters using a grid search and use cross-validation. We report our results as the mean performance over 10 different random splits of training and test data (to also quantify the variance).

5.3.2 Test/Retest Validity

In this section, we evaluate the endurance of the learned factors over time and sub-populations by conducting a test-retest experiment. Our experimental procedure is as follows:

Test/Retest over Time Our experimental procedure for evaluating the stability of our factors over time is as follows:

1. Split the entire corpus into two parts: (a) a training portion used for learning a model to infer factors and (b) a held-out test portion on which we will apply the learned model to infer factors on this held out set. We use 75% of the corpus for training and the remaining 25% for testing. We further divide each user's posts into several time periods (6 months apart).

2. We learn a model to infer factors for each user using the training set.

3. We now infer factors for users in each time period of the test set.

4. We report the correlations of factors inferred over these users across time points.

If our factors are enduring over time, we expect to see good correlations of our factors over different points in time.

Dropout Reliability Here, we evaluate the endurance of the learned factors to different sample populations of users. Ideally, the learned factors should not be too dependent on whether an arbitrary set of users are present in the training data. We now quantify the sensitivity of the learned factors to the presence (or absence) of users as follows:

1. We randomly dropout 20% users from our training data before we learn our factors.

2. We repeat step 1, a large number of times (we use 100 times in our experiments).

3. We now infer factor scores on the fixed held out test set for each of the 100 models learned.

4. We consider the factor scores inferred from each pair of models (i, j):

   - We use the Hungarian Algorithm to infer the best alignment of factor scores as measured by correlations.
   - We compute the mean correlation between the aligned factor scores and report it.

5. We do this for each model pair and compute the distribution of scores observed.

5.4 Results and Discussion

5.4.1 Predictive Validity

Table 5.1 shows the performance of our predictive models on the two broad categories of predictive tasks without residualizing out demographics.[7] First, we discuss results for the social/demographic outcomes. Table 5.1a shows our performance on these outcomes using FA factors. Observe that on these outcomes, our language based factors outperform questionnaire based factors like BIG5 (highlighted with COLOR in Table 5.1a). For example, on the task of predicting LIKES, FA with 5 factors outperforms the baseline by an increase of 7% (60.11 - 52.6). It is worth emphasizing here that the task of predicting LIKES is essentially 20 different classification tasks and is inherently a hard task. Consequently an improvement of 7% on this task is very promising. Similarly, observe that FA based factors consistently outperforms baselines on the tasks of predicting INCOME and IQ. Also note that adding age and gender as co-variates improves predictive performance (compare FA5+DEMOG with FA5).

Now we turn our attention to results on questionnaire based factors as shown in Tables 5.1b. First, note that FA based factors perform competitively with BIG5 on the task of BIG5QUESTIONS where BIG5 has an inherent advantage since these questions are correlated with BIG5 scores by design. Note also that on the tasks of SWL and DEPRESSION, language based factors (FA) do not

[7] We show corresponding results residualizing out demographics in the Supporting Information section. We also show results for 10 and 30 factors here.

perform as well as BIG5. We hypothesize two reasons for this under-performance: (a) Language based factors do not capture very strong psychological variables like depression etc. very well and (b) Questionnaire based methods are subject to shared method variance which is manifested by a higher correlation of these variables with respect to the BIG5.

In Tables 5.2 and 5.3 and we also show the best and worst (a) LIKES and (b) BIG5 items, ranked according to their predictive performance by BLT's. Observe that BLT's perform the best on LIKES which are not too generic or too specific. For example, BLT's can predict very well whether users like COUNTRY MUSIC (see LIKE 8 in Table 5.3a). BLT's show poor performance on very generic LIKEs which almost all users might like (for example: YOUTUBE, FACEBOOK see Table 5.3b). Similarly we observe that BLT's perform best on BIG5 question items which are highly correlated with language like *"have a rich vocabulary"* (see Table 5.2a) and perform poorly on items that measure psychological dimensions like *"Waste my time"* (see Table 5.2b).

We conclude by emphasizing that the traits learned using FA are *not apriori tuned* to any particular predictive task, and yet perform competitively with traits derived from questionnaires in predicting a variety of outcomes and even outperform questionnaire based traits on behavioral outcomes like INCOME and IQ, thus underscoring the generalizability of these traits.

5.4.2 Test/Retest Validity

Figure 5.3 shows the factor correlations at future points in time with the factor scores at the initial point ($t = 0$) over a common set of users in a test-retest setting. Observe that the factor scores in future time periods demonstrate moderate to good correlation with the factor scores at the initial time point (t=0). Also observe that over time, even though the correlation of factor scores with the initial time decreases as expected, the strength of correlation is still > 0.3. These results suggest that our inferred factors demonstrate stability across time and are thus stable traits.

For dropout validity, we computed that the mean correlation among factors obtained over multiple runs where a random sample of 20% were dropped before learning the factors. We used the Hungarian algorithm to infer the mapping between factors across multiple runs. We observe a high correlation (> 0.90)among corresponding factors across multiple runs.

Both of these observations suggest that the method and the factors that we infer are stable across subsets of populations and indicate endurance across time and sub-populations.

Method	FriendSize	Income	IQ	Likes
Demog	0.052	0.283	0.162	55.5
Big5	0.183	0.037	0.179	52.6
Big5+Demog	0.192	0.278	0.269	56.9
FA5	0.125	0.362	0.361	60.11
FA5 + Demog	0.148	0.375	0.423	61.86

(a) Behavioral/Economic Outcomes: We show mean Pearson's R over 10 random train-test splits for FriendSize, Income and IQ while for Likes we show the mean area under the curve (AUC) over all 20 categories. Language based factors (FA) perform competitively and even outperform questionnaire based factors (big5) as highlighted in color.

Method	Big5Questions	SWL	Depression
Demog	0.072	0.053	0.103
Big5	0.178	0.486	0.407
Big5+Demog	0.191	0.524	0.424
FA5	0.178	0.165	0.293
FA5 + Demog	0.186	0.207	0.227

(b) Questionnaire based outcomes: We show mean Pearson R over 10 random train-test splits. Language based factors (FA5) do not outperform big5.

Table 5.1: Predictive performance on Social media tasks and Questionnaire based tasks for factors without residualization of age and gender. Demog indicates that age and gender were also added as co-variates to learn predictive models.

5.5 Conclusion

In this chapter, we proposed a method based on **factor analysis** to infer latent personality traits from every-day language use of users on social media. While sociologists and psychologists have long studied personality through questionnaires, our method infers latent factors from social media – a medium that allows access to large sample sizes, unprompted access to user's thoughts, emotions and language and a data-driven approach enabling an analysis at a scale that was previously unprecedented. We demonstrate the efficacy and utility of our learned traits by evaluating them on several dimensions, show that these traits are *generalizable* with good predictive power, *enduring* and therefore are useful for a variety of tasks. We believe that this work will set future directions for large scale analysis of social media text to test psychological theories and hypotheses, enabling psychologists to gain insights into people by observing their everyday behavior at scale.

QNO	Question	R
54	*Am not interested in theoretical discussions* (O-)	0.230
71	*Have a rich vocabulary* (O+)	0.224
64	*Have difficulty understanding abstract ideas* (O-)	0.222
51	*Tend to vote for liberal political candidates* (O+)	0.220
90	*Am filled with doubts* (N+)	0.215

(a) List of top 5 questions our factors best predict responses to.

QNO	Question	R
28	*Waste my time* (C-)	0.094
43	*Talk to a lot of different people at parties* (E+)	0.133
29	*Dont talk a lot* (E-)	0.135
88	*Find it difficult to get down to work* (C-)	0.139

(b) List of bottom 5 questions our factors worst predict the responses to.

Table 5.2: List of Big5 questions on which our factors perform the best and the worst at predicting the responses. In particular observe that we can predict very well using BLT's responses to questions which have strong associations with language like "*whether one has rich vocabulary or not*". Note that BLT's do not perform as well on psychological questions like *"Waste my time"*.

Supporting Information

Here, we also provide supporting results based on our analysis of BLT's. Table 5.4 shows the performance of our 10 and 30 factor BLT's (FA10 and FA30) on both categories of predictive tasks outlined. As baselines, we use an equivalent set of questionnaire based factors namely the 10 aspects based scores and the 30 facets based scores. Observe once again that BLT's consistently outperform questionnaire based models on behavioral/demographic outcomes but do not perform as well on questionnaire based outcomes. Similar results are also obtained with demographics residualized (see Tables 5.5 and 5.6).

LIKENO	LIKE	AUC
8	*Lady Antebellum, Tim McGraw, NCIS, Kenny Chesney, Country music, Jason Aldean, Walmart, Carrie Underwood, George Strait, Family Feud*	71.04
12	*Glowsticks, Finding Nemo, Being Hyper!, DORY*	68.50
11	*Lil Wayne, Drake, Eminem,T.I., Nicki Minaj, Jersey Shore,Trey*	68.50
10	*I redo high fives if they weren't good enough the first time ,Why do we have to be quiet during a fire drill? Will the fire hear us?*	68.05
14	*The Beatles, Pink Floyd, The Doors, Radiohead, Queen, Nirvana*	65.85

(a) Top 5 Best Likes for prediction.

LIKENO	LIKE	AUC
3	*I hate when im yelling at someone and i mess up what im saying, I would take a bullet for u.. Not the head but like in the leg or something*	50.73
0	*YouTube, Facebook, Oreo, Skittles, Coca-Cola, Adam Sandler, Starburst, Starbucks, Music, Toy Story*	51.46
16	*After an arguement I think about clever things I should have said, Your in a good mood, one little thing happens and BAM.... Bad mood.*	54.23
4	*I love days in class when all we do is chill and talk the whole time Get real. No one's going to form a single line if the building's on FIRE.*	54.49
1	*When I was little I liked building forts out of pillows and blankets, I like when my scissors glide through the paper so I don't have to cut.*	54.78

(b) Bottom 5 worst Likes for prediction.

Table 5.3: List of Likes our factors perform the best and the worst at prediction. In particular observe that we can predict very well using BLT's whether users like *country music* (LIKE 8). BLT's do not perform as well on too generic likes (LIKE 0:YOUTUBE) or likes which are too specific (LIKE 3). We show the top LIKES in each cluster for interpretation.

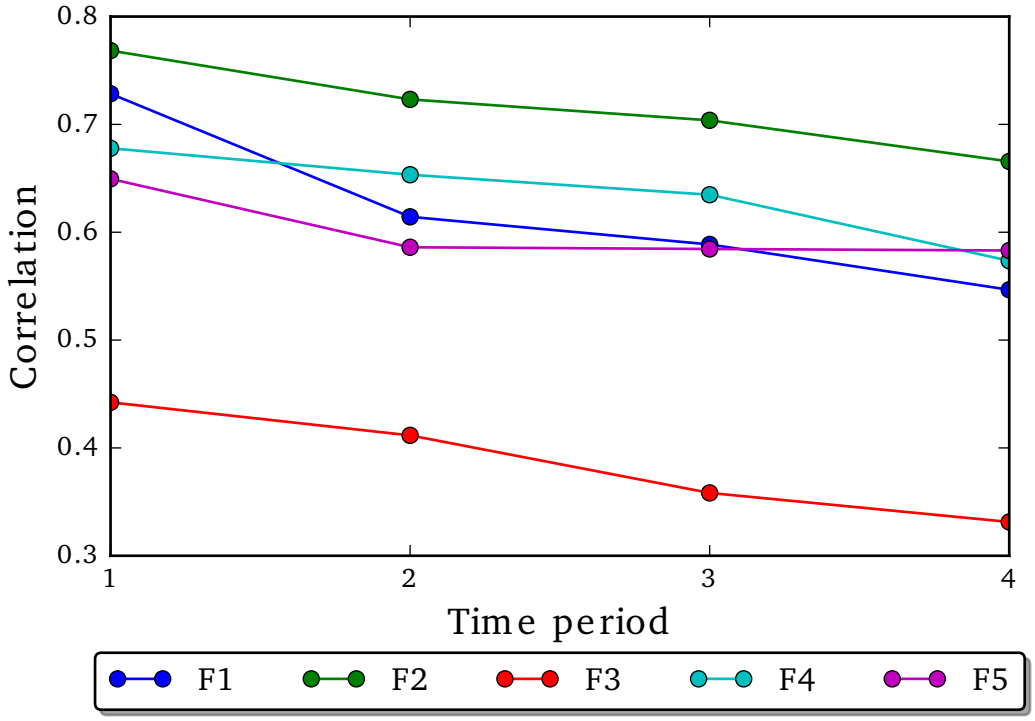


Figure 5.3: Results for test-retest validity: Correlations of factors observed over different time periods with the factors at time $t = 0$ measured over the same set of users in a test-retest setting. Observe the moderate correlation (> 0.3) even after 4 time periods indicating a degree of stability over time.

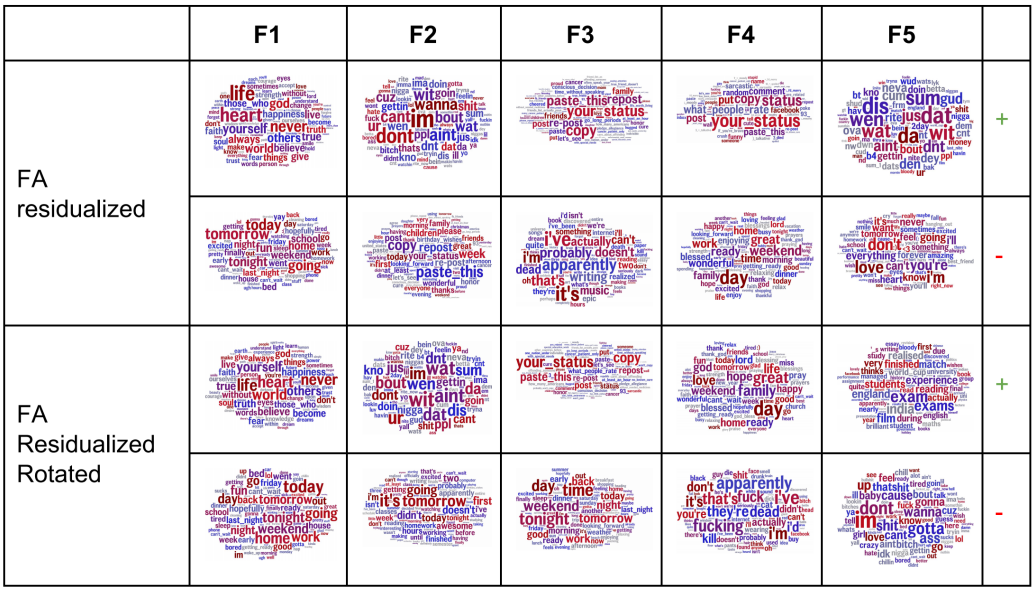


Figure 5.4: Word clouds showing the most/least correlated words for each FA factor as obtained using Differential Language Analysis with ae and gender residualized. Residualizing out demographics like age and gender appears to reveal other dimensions of variance like (geography, ethnicity) as illustrated by F5 (see row `FA residualized rotated`) that reveals a factor highlighting language use of Indians in India with words like `india, world-cup, match`.

Method	FriendSize	Income	IQ	Likes
Demog	0.052	0.283	0.162	55.50
Big5-10	0.202	0.147	0.252	53.50
Big5-10 +Demog	0.200	0.344	0.220	57.30
FA10	0.178	0.378	0.395	63.39
FA10 + Demog	0.191	0.411	0.396	64.32
Big5-30	0.244	–	0.285	56.28
Big5-30 +Demog	0.233	–	0.330	59.32
FA30	0.316	0.379	0.420	64.98
FA30 + Demog	0.329	0.398	0.459	65.76

(a) Behavioral/Economic Outcomes: We show mean Pearson's R over 10 random train-test splits for FriendSize, Income and IQ while for Likes we show the mean area under the curve (AUC) over all 20 categories. Language based factors (FA) perform competitively and even outperform questionnaire based factors as highlighted in color.

Method	Big5Questions	SWL	Depression
Demog	0.072	0.053	0.103
Big5-10	0.627	0.470	0.299
Big5-10+Demog	0.629	0.479	0.313
FA10	0.191	0.229	0.179
FA10 + Demog	0.199	0.241	0.222
Big5-30	0.766	0.583	0.464
Big5-30+Demog	0.767	0.615	0.405
FA30	0.215	0.229	0.242
FA30 + Demog	0.220	0.297	0.207

(b) Questionnaire based outcomes: We show mean Pearson R over 10 random train-test splits. Language based factors (FA) do not outperform questionnaire based factors.

Table 5.4: Predictive performance on Social media tasks and Questionnaire based tasks for factors (FA10 and FA30) without residualization of age and gender. Demog indicates that age and gender were also added as co-variates to learn predictive models.

Method	FriendSize	Income	IQ	Likes
Demog	0.052	0.283	0.162	55.50
Big5	0.183	0.037	0.179	52.60
Big5+Demog	0.192	0.278	0.269	56.90
FA5	0.140	0.285	0.353	56.33
FA5 + Demog	0.160	0.361	0.370	60.86

(a) Behavioral/Economic outcomes: We show mean Pearson's R over 10 random train-test splits for FriendSize, Income and IQ while for Likes we show the mean area under the curve (AUC) over all 20 categories. Language based factors (FA) perform competitively and even outperform questionnaire based factors as highlighted in color.

Method	Big5Questions	SWL	Depression
Demog	0.072	0.053	0.103
Big5	0.178	0.486	0.407
Big5+Demog	0.191	0.524	0.424
FA5	0.167	0.232	0.187
FA5 + Demog	0.185	0.211	0.289

(b) Questionnaire based outcomes: We show mean Pearsons R over 10 random train-test splits. Language based factors (FA) do not outperform questionnaire based factors.

Table 5.5: Predictive performance on Social media tasks and Questionnaire based tasks for factors (FA5) with residualization of age and gender. Demog indicates that age and gender were also added as co-variates to learn predictive models.

Method	FriendSize	Income	IQ	Likes
Demog	0.052	0.283	0.162	55.50
Big5-10	0.202	0.147	0.252	53.50
Big5-10 +Demog	0.200	0.344	0.220	57.30
FA10	0.176	0.309	0.291	57.85
FA10 + Demog	0.191	0.360	0.421	62.10
Big5-30	0.244	–	0.285	56.28
Big5-30 +Demog	0.233	–	0.330	59.32
FA30	0.305	0.334	0.351	59.64
FA30 + Demog	0.310	0.379	0.398	63.64

(a) Behavioral/Economic outcomes: We show mean Pearson's R over 10 random train-test splits for FriendSize, Income and IQ while for Likes we show the mean area under the curve (AUC) over all 20 categories. Language based factors (FA) perform competitively and even outperform questionnaire based factors (big5) as highlighted in color.

Method	Big5Questions	SWL	Depression
Demog	0.072	0.053	0.103
Big5-10	0.627	0.470	0.299
Big5-10+Demog	0.629	0.479	0.313
FA10	0.182	0.204	0.132
FA10 + Demog	0.197	0.200	0.198
Big5-30	0.766	0.583	0.464
Big5-30+Demog	0.767	0.615	0.405
FA30	0.197	0.241	0.127
FA30 + Demog	0.211	0.251	0.170

(b) Questionnaire based outcomes: We show mean Pearsons R over 10 random train-test splits. Language based factors (FA) perform do not outperform questionnaire based factors.

Table 5.6: Predictive performance on Social media tasks and Questionnaire based tasks for factors (FA10 and FA30) with residualization of age and gender. Demog indicates that age and gender were also added as co-variates to learn predictive models.

Chapter 6

Conclusions

If you thought that science was certain – well, that is just an error on your part.

Richard P. Feynman

In this thesis, we advocate for computational models that account for the rich variation in the language of the Internet and on-line social media to improve natural language understanding. We explore this perspective by proposing statistical models to track and detect variation in word semantics across multiple modalities that reveal insights into language change on the Internet and social media. We outline below detailed contributions of this work:

6.1 Summary of contributions

- We present a model to track how word semantics evolve over time by outlining methods to capture evolving word semantics as a time-series. In particular, we demonstrate how to effectively use word embeddings to capture semantic changes. We also present a method to estimate the time-point at which a word's meaning changed dominantly drawing on techniques from time-series analysis. We conduct an analysis of language change over a time period spanning more than a century,
- We present a model to track regional variation in word usage using an additive model to learn region specific word embeddings. To effectively weed out false positives, we also propose a NULL model to establish

statistical significance of observed changes. We conduct an analysis of language variation across multiple scales ranging from the 50 states in the USA to four English speaking countries.

- We present two models to track domain specific linguistic variation, one of which explicitly models the different senses of words using sense specific word embeddings. We demonstrate that detecting domain specific linguistic variation can improve performance of Named Entity Recognition in a domain adaptation setting. In particular, we improve the performance of Named Entity Recognition systems in the domains of Finance and Sports.

- Finally, we present a model to infer stable latent user traits from their daily language use on social media. We demonstrate that these latent user traits are useful for a variety of predictive tasks without any a priori tuning and are stable across time and populations.

6.2 Future Directions

In this thesis we outline models to uncover variation manifested by two focal points: (a) language and (b) users. We now outline a few research directions that are open for investigation on each of these focal points.

6.2.1 Richer Models for Linguistic Variation

The models that are presented in this thesis focus on capturing variation in word usage or semantics across multiple modalities. This line of research can be extended in the following directions:

- **Automatic construction of dictionaries**: Given that, one can detect variation in word semantics across time or geographical regions, a natural research direction is to investigate whether one can automatically generate a definition (dictionary entry) for words. This has applications to not only the field of variational linguistics but also enables richer semantic web applications that incorporate semantic variation in word usage in real time.

- **Detecting Euphemism Treadmills**: Euphemism treadmills [1] refers to a phenomenon where a word is introduced to replace an offensive

[1] http://englishcowpath.blogspot.com/2011/06/euphemism-treadmill-replacing-r-word.html

word which in turn itself becomes offensive. For example, the word `retardation` has been replaced with `mentally handicapped` which in turn has been replaced with `intellectually challenged`. Computational methods to track and detect such euphemism treadmills can reveal insights into how language is affected by cultural changes in society.

- **Detecting Variation in Phrases, Idioms and Metaphors**: While most of the methods highlighted in this thesis focus on variation in word usage/semantics language usage varies in its usage of metaphors and idioms. The idioms *"bring home the bacon"* or *"keeping up with the Joneses"* are quite specific to American culture. Similarly the *"time is like space"* metaphor has different forms influenced by culture[2] which manifests in language. One research direction therefore would be to seek to develop computational methods to infer different forms of metaphorical usage based on language analysis.

6.2.2 Modeling Users using Multi-modal Representations

In Chapter 5, we provide models and methods to infer latent user traits from their daily language on social media. However users do not generate only textual content on social media but upload images and videos as well. Furthermore, a user on social media is not a isolated entity but is usually a part of a social network. These multiple modalities other than text can capture important latent traits/dimensions of users. For example, recent research has shown correlations between a users personality and his/her profile picture choice on social media [123]. Therefore, one potential research direction is to incorporate multiple modalities like images, and social network roles while modeling users on social media. This can enable researchers to analyze human behavior differences as they manifest across multiple modalities and enable a more holistic models of users on social media with broad and far reaching applications in fields like health analytics and ad-targeting.

[2]https://www.scientificamerican.com/article/how-we-make-sense-of-time/

Bibliography

[1] Bo Han and Timothy Baldwin. Lexical normalisation of short text messages: Makn sens a# twitter. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1*, pages 368–378. Association for Computational Linguistics, 2011.

[2] Bo Han, Paul Cook, and Timothy Baldwin. Automatically constructing a normalisation dictionary for microblogs. In *Proceedings of the 2012 joint conference on empirical methods in natural language processing and computational natural language learning*, pages 421–432. Association for Computational Linguistics, 2012.

[3] Fei Liu, Fuliang Weng, and Xiao Jiang. A broad-coverage normalization system for social media language. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Long Papers-Volume 1*, pages 1035–1044. Association for Computational Linguistics, 2012.

[4] Bo Han, Paul Cook, and Timothy Baldwin. Lexical normalization for social media text. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 4(1):5, 2013.

[5] Kevin Gimpel, Nathan Schneider, Brendan O'Connor, Dipanjan Das, Daniel Mills, Jacob Eisenstein, Michael Heilman, Dani Yogatama, Jeffrey Flanigan, and Noah A Smith. Part-of-speech tagging for twitter: Annotation, features, and experiments. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies: short papers-Volume 2*, pages 42–47. Association for Computational Linguistics, 2011.

[6] Olutobi Owoputi, Brendan OConnor, Chris Dyer, Kevin Gimpel, and Nathan Schneider. Part-of-speech tagging for twitter: Word clusters and other advances. *School of Computer Science*, 2012.

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

Association for Computational Linguistics, 2011.

[8] Oren Etzioni, Michael Cafarella, Doug Downey, Ana-Maria Popescu, Tal Shaked, Stephen Soderland, Daniel S Weld, and Alexander Yates. Unsupervised named-entity extraction from the web: An experimental study. *Artificial intelligence*, 165(1):91–134, 2005.

[9] Rami Al-Rfou, Vivek Kulkarni, Bryan Perozzi, and Steven Skiena. Polyglot-ner: Massive multilingual named entity recognition. In *Proceedings of the 2015 SIAM International Conference on Data Mining*, pages 586–594. SIAM, 2015.

[10] Hyunyoung Choi and Hal Varian. Predicting the present with google trends. *Economic Record*, 88:2–9, 2012.

[11] Herman Anthony Carneiro and Eleftherios Mylonakis. Google trends: A web-based tool for real-time surveillance of disease outbreaks. *Clinical Infectious Diseases*, 49(10):1557–1564, 2009.

[12] Jianhua Lin. Divergence measures based on the shannon entropy. *Information Theory, IEEE Transactions on*, 37(1):145–151, 1991.

[13] John Rupert Firth. *Papers in Linguistics 1934-1951: Repr.* Oxford University Press, 1961.

[14] Yoshua Bengio, Aaron Courville, and Pascal Vincent. Representation learning: A review and new perspectives. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 35(8):1798–1828, 2013.

[15] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. *CoRR*, abs/1301.3781, 2013.

[16] Frederic Morin and Yoshua Bengio. Hierarchical probabilistic neural network language model. In *Proceedings of the international workshop on artificial intelligence and statistics*, pages 246–252, 2005.

[17] Andriy Mnih and Geoffrey E Hinton. A scalable hierarchical distributed language model. *Advances in neural information processing systems*, 21: 1081–1088, 2009.

[18] Léon Bottou. Stochastic gradient learning in neural networks. In *Proceedings of Neuro-Nîmes 91*, Nimes, France, 1991. EC2.

[19] David E Rumelhart, Geoffrey E Hinton, and Ronald J Williams. Learning representations by back-propagating errors. *Cognitive modeling*, 1:213, 2002.

[20] Tomas Mikolov, Ilya Sutskever, Kai Chen, Gregory S. Corrado, and Jeffrey Dean. Distributed representations of words and phrases and their compositionality. In *Advances in Neural Information Processing Systems 26: 27th Annual Conference on Neural Information Processing Systems 2013. Proceedings of a meeting held December 5-8, 2013, Lake Tahoe,*

Nevada, United States., pages 3111–3119, 2013.

[21] Wayne A. Taylor. Change-point analysis: A powerful new tool for detecting changes, 2000.

[22] Michèle Basseville and Igor V. Nikiforov. *Detection of Abrupt Changes: Theory and Application*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 1993.

[23] Ryan P. Adams and David J.C. MacKay. Bayesian online changepoint detection. Cambridge, UK, 2007.

[24] B. Efron and R. J. Tibshirani. An introduction to the bootstrap. 1971.

[25] Jean-Baptiste Michel, Yuan Kui Shen, Aviva Presser Aiden, Adrian Veres, Matthew K. Gray, The Google Books Team, Joseph P. Pickett, Dale Hoiberg, Dan Clancy, Peter Norvig, Jon Orwant, Steven Pinker, Martin A. Nowak, and Erez Lieberman Aiden. Quantitative analysis of culture using millions of digitized books. *Science*, 331(6014):176–182, 2011.

[26] Yoav Goldberg and Jon Orwant. A dataset of syntactic-ngrams over time from a very large corpus of english books. In *Second Joint Conference on Lexical and Computational Semantics (*SEM), Volume 1: Proceedings of the Main Conference and the Shared Task: Semantic Textual Similarity*, pages 241–247, Atlanta, Georgia, USA, June 2013. Association for Computational Linguistics.

[27] Yuri Lin, Jean-Baptiste Michel, Erez Lieberman Aiden, Jon Orwant, Will Brockman, and Slav Petrov. Syntactic annotations for the google books ngram corpus. In *Proceedings of the ACL 2012 System Demonstrations*, ACL '12, pages 169–174, Stroudsburg, PA, USA, 2012. Association for Computational Linguistics.

[28] Jason Mann, David Zhang, Lu Yang, Dipanjan Das, and Slav Petrov. Enhanced search with wildcards and morphological inflections in the google books ngram viewer. In *Proceedings of ACL Demonstrations Track*. Association for Computational Linguistics, June 2014.

[29] Robert C Atkins. Dr. atkins' diet revolution; the high calorie way to stay thin forever. 1972.

[30] Kristina Gulordava and Marco Baroni. A distributional similarity approach to the detection of semantic change in the google books ngram corpus. In *Proceedings of the GEMS 2011 Workshop on GEometrical Models of Natural Language Semantics*, pages 67–71, Edinburgh, UK, July 2011. Association for Computational Linguistics.

[31] Derry Tanti Wijaya and Reyyan Yeniterzi. Understanding semantic change of words over centuries. In *Proceedings of the 2011 International Workshop on DETecting and Exploiting Cultural diversiTy on the Social*

Web, DETECT '11, pages 35–40, New York, NY, USA, 2011. ACM.

[32] Yoon Kim, Yi-I Chiu, Kentaro Hanaki, Darshan Hegde, and Slav Petrov. Temporal analysis of language through neural language models. In *Proceedings of the ACL 2014 Workshop on Language Technologies and Computational Social Science*, pages 61–65, Baltimore, MD, USA, June 2014. Association for Computational Linguistics.

[33] Adam Jatowt and Kevin Duh. A framework for analyzing semantic change of words across time. In *Proceedings of the Joint JCDL/TPDL Digital Libraries Conference*, 2014.

[34] Paul Cook, Jey Han Lau, Michael Rundell, Diana McCarthy, and Timothy Baldwin. A lexicographic appraisal of an automatic approach for detecting new word senses. In *Electronic lexicography in the 21st century: thinking outside the paper: proceedings of the eLex 2013 conference, 17-19 October 2013, Tallinn, Estonia*, pages 49–65, 2013.

[35] Gerhard Heyer, Florian Holz, and Sven Teresniak. Change of topics over time and tracking topics by their change of meaning. In Ana L. N. Fred, editor, *KDIR 2009: Proc. of Int. Conf. on Knowledge Discovery and Information Retrieval*. INSTICC Press, October 2009.

[36] Patrick Juola. The time course of language change. *Computers and the Humanities*, 37(1):77–96, 2003.

[37] Xuri Tang, Weiguang Qu, and Xiaohe Chen. Semantic change computation: A successive approach. In Longbing Cao, Hiroshi Motoda, Jaideep Srivastava, Ee-Peng Lim, Irwin King, PhilipS. Yu, Wolfgang Nejdl, Guandong Xu, Gang Li, and Ya Zhang, editors, *Behavior and Social Computing*, volume 8178 of *Lecture Notes in Computer Science*, pages 68–81. Springer International Publishing, 2013.

[38] Sunny Mitra, Ritwik Mitra, Martin Riedl, Chris Biemann, Animesh Mukherjee, and Pawan Goyal. That's sick dude!: Automatic identification of word sense change across different timescales. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1020–1029, Baltimore, Maryland, June 2014. Association for Computational Linguistics.

[39] Geoffrey E Hinton. Learning distributed representations of concepts. In *Proceedings of the eighth annual conference of the cognitive science society*, pages 1–12. Amherst, MA, 1986.

[40] Yoshua Bengio, Holger Schwenk, Jean-Sébastien Senécal, Fréderic Morin, and Jean-Luc Gauvain. Neural probabilistic language models. In *Innovations in Machine Learning*, pages 137–186. Springer, 2006.

[41] Yoshua Bengio and J-S Senecal. Adaptive importance sampling to accelerate training of a neural probabilistic language model. *Neural*

Networks, IEEE Transactions on, 19(4):713–722, 2008.

[42] Jeffrey Dean, Greg S. Corrado, Rajat Monga, Kai Chen, Matthieu Devin, Quoc V. Le, Mark Z. Mao, MarcAurelio Ranzato, Andrew Senior, Paul Tucker, Ke Yang, and Andrew Y. Ng. Large scale distributed deep networks. In *NIPS*, 2012.

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

[44] Ronan Collobert, Jason Weston, Léon Bottou, Michael Karlen, Koray Kavukcuoglu, and Pavel Kuksa. Natural language processing (almost) from scratch. *J. Mach. Learn. Res.*, 12:2493–2537, November 2011.

[45] Rami Al-Rfou, Bryan Perozzi, and Steven Skiena. Polyglot: Distributed word representations for multilingual nlp. In *Proceedings of the Seventeenth Conference on Computational Natural Language Learning*, pages 183–192, Sofia, Bulgaria, August 2013. Association for Computational Linguistics.

[46] Bryan Perozzi, Rami Al-Rfou, and Steven Skiena. Deepwalk: Online learning of social representations. In *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '14, New York, NY, USA, August 2014. ACM.

[47] Diane J. Schiano, Coreena P. Chen, Ellen Isaacs, Jeremy Ginsberg, Unnur Gretarsdottir, and Megan Huddleston. Teen use of messaging media. In *Computer Human Interaction*, pages 594–595, 2002.

[48] Sali A. Tagliamonte and Derek Denis. Linguistic Ruin? LOL! Instant Messaging and Teen Language. *American Speech*, 83:3–34, 2008.

[49] Guy Merchant. Teenagers in cyberspace: an investigation of language use and language change in internet chatrooms. *Journal of Research in Reading*, 24:293–306, 2001.

[50] David Crystal. *Internet Linguistics: A Student Guide*. Routledge, New York, NY, 10001, 1st edition, 2011.

[51] Sali A. Tagliamonte. *Analysing Sociolinguistic Variation*. Cambridge University Press, 2006.

[52] William. Labov. *Locating language in time and space / edited by William Labov*. Academic Press New York, 1980.

[53] James Milroy. *Linguistic variation and change: on the historical sociolinguistics of English*. B. Blackwell, 1992.

[54] Walt Wolfram and Natalie Schilling-Estes. *American English: dialects and variation*, volume 20. Wiley-Blackwell, 2005.

[55] David Bamman et al. Gender identity and lexical variation in social media. *Journal of Sociolinguistics*, 2014.

[56] Gabriel Doyle. Mapping dialectal variation by querying social media. In *EACL*, 2014.

[57] Jacob Eisenstein, Brendan O'Connor, Noah A Smith, and Eric P Xing. A latent variable model for geographic lexical variation. In *EMNLP*, 2010.

[58] Jacob Eisenstein, Brendan O'Connor, Noah A Smith, and Eric P Xing. Diffusion of lexical change in social media. *PLoS ONE*, 2014.

[59] Vivek Kulkarni, Rami Al-Rfou, Bryan Perozzi, and Steven Skiena. Statistically significant detection of linguistic change. In *WWW*, 2015.

[60] David Bamman, Chris Dyer, and Noah A. Smith. Distributed representations of geographically situated language. *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 828–834, June 2014.

[61] Jacob Eisenstein, Noah A Smith, et al. Discovering sociolinguistic associations with structured sparsity. In *In ACL-HLT*, 2011.

[62] Charu C Aggarwal. *Outlier analysis*. Springer Science & Business Media, 2013.

[63] Gail M Sullivan and Richard Feinn. Using effect size-or why the p value is not enough. *Journal of graduate medical education*, 2012.

[64] Jean-Baptist du Prel, Gerhard Hommel, Bernd Röhrig, and Maria Blettner. Confidence interval or p-value?: part 4 of a series on evaluation of scientific publications. *Deutsches Ärzteblatt International*, 2009.

[65] Olutobi Owoputi, Brendan O'Connor, et al. Improved part-of-speech tagging for online conversational text with word clusters. Association for Computational Linguistics, 2013.

[66] Tom Kenter, Melvin Wevers, Pim Huijnen, et al. Ad hoc monitoring of vocabulary shifts over time. In *CIKM*. ACM, 2015.

[67] Bruno Gonçalves and David Sánchez. Crowdsourcing dialect characterization through twitter. 2014.

[68] Tim Berners-Lee, James Hendler, and Ora Lassila. The Semantic Web. *Scientific American*, 284:34–43, 2001.

[69] Igor Brigadir, Derek Greene, and Pádraig Cunningham. Analyzing discourse communities with distributional semantic models. In *ACM Web Science 2015 Conference*. ACM, 2015.

[70] Brendan O'Connor, Jacob Eisenstein, Eric P Xing, and Noah A Smith. Discovering demographic language variation. In *Proc. of NIPS Workshop on Machine Learning for Social Computing*, 2010.

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

[72] Bryan Perozzi, Rami Al-Rfou, et al. Inducing language networks from

continuous space word representations. In *Complex Networks V*. 2014.

[73] Rami Al-Rfou, Vivek Kulkarni, Bryan Perozzi, and Steven Skiena. Polyglot-ner: Massive multilingual named entity recognition. In *SDM*, 2015.

[74] Yanqing Chen, Bryan Perozzi, Rami Al-Rfou, and Steven Skiena. The expressive power of word embeddings. *arXiv preprint arXiv:1301.3226*, 2013.

[75] Radu Florian, Abe Ittycheriah, et al. Named entity recognition through classifier combination. In *Proceedings of the seventh conference on Natural language learning at HLT-NAACL 2003-Volume 4*, pages 168–171. Association for Computational Linguistics, 2003.

[76] Hai Leong Chieu and Hwee Tou Ng. Named entity recognition: a maximum entropy approach using global information. In *Proceedings of the 19th international conference on Computational linguistics-Volume 1*, pages 1–7. Association for Computational Linguistics, 2002.

[77] Rie Kubota Ando and Tong Zhang. A framework for learning predictive structures from multiple tasks and unlabeled data. *Journal of Machine Learning Research*, 6(Nov):1817–1853, 2005.

[78] Zhiheng Huang, Wei Xu, and Kai Yu. Bidirectional LSTM-CRF models for sequence tagging. *CoRR*, abs/1508.01991, 2015. URL http://arxiv.org/abs/1508.01991.

[79] John Blitzer, Ryan McDonald, et al. Domain adaptation with structural correspondence learning. In *Proceedings of the 2006 conference on empirical methods in natural language processing*, pages 120–128. Association for Computational Linguistics, 2006.

[80] Jing Jiang and ChengXiang Zhai. Instance weighting for domain adaptation in nlp. In *ACL*, volume 7, pages 264–271, 2007.

[81] Sandeepkumar Satpal and Sunita Sarawagi. Domain adaptation of conditional probability models via feature subsetting. In *European Conference on Principles of Data Mining and Knowledge Discovery*, pages 224–235. Springer, 2007.

[82] Jing Jiang. *Domain adaptation in natural language processing*. ProQuest, 2008.

[83] Qi Li. Literature survey: domain adaptation algorithms for natural language processing. 2012.

[84] Minmin Chen, Zhixiang Xu, et al. Marginalized denoising autoencoders for domain adaptation. *arXiv preprint arXiv:1206.4683*, 2012.

[85] Vivek Kulkarni, Bryan Perozzi, and Steven Skiena. Freshman or fresher? quantifying the geographic variation of internet language. *arXiv preprint arXiv:1510.06786*, 2015.

[86] Sergey Bartunov, Dmitry Kondrashkin, Anton Osokin, and Dmitry Vetrov. Breaking sticks and ambiguities with adaptive skip-gram. *arXiv preprint arXiv:1502.07257*, 2015.

[87] Naoaki Okazaki. Crfsuite: a fast implementation of conditional random fields (crfs), 2007. URL http://www.chokkan.org/software/crfsuite/.

[88] Wenpeng Yin, Tobias Schnabel, and Hinrich Schütze. Online updating of word representations for part-of-speech tagging. *arXiv preprint arXiv:1604.00502*, 2016.

[89] Yi Yang and Jacob Eisenstein. Unsupervised multi-domain adaptation with feature embeddings. 2015.

[90] Paul Cook, Jey Han Lau, Diana McCarthy, and Timothy Baldwin. Novel word-sense identification. 2014.

[91] Lea Frermann and Mirella Lapata. A bayesian model of diachronic meaning change. *Transactions of the Association for Computational Linguistics*, 2016.

[92] William L Hamilton, Jure Leskovec, and Dan Jurafsky. Diachronic word embeddings reveal statistical laws of semantic change. *arXiv preprint arXiv:1605.09096*, 2016.

[93] Theodoros Evgeniou and Massimiliano Pontil. Regularized multi–task learning. In *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 109–117. ACM, 2004.

[94] Hal Daumé III. Frustratingly easy domain adaptation. *arXiv preprint arXiv:0907.1815*, 2009.

[95] Tobias Schnabel and Hinrich Schütze. Flors: Fast and simple domain adaptation for part-of-speech tagging. *Transactions of the Association for Computational Linguistics*, 2:15–26, 2014.

[96] Gordon W Allport and Henry S Odbert. Trait-names: A psycho-lexical study. *Psychological monographs*, 47(1):i, 1936.

[97] Raymond Bernard Cattell. Description and measurement of personality. 1946.

[98] Dan P McAdams. The five-factor model in personality: A critical appraisal. *Journal of personality*, 60(2):329–361, 1992.

[99] Oliver P John and Sanjay Srivastava. The big five trait taxonomy: History, measurement, and theoretical perspectives. *Handbook of personality: Theory and research*, 2(1999):102–138, 1999.

[100] Raymond Bernard Cattell, Herbert W Eber, and Maurice M Tatsuoka. *Handbook for the sixteen personality factor questionnaire (16 PF): In clinical, educational, industrial, and research psychology, for use with all*

forms of the test. Institute for Personality and Ability Testing, 1970.

[101] Lewis R Goldberg. Language and personality: Toward a taxonomy of trait descriptive terms. *Psikoloji Çalışmaları Dergisi*, 12:1–23, 1976.

[102] Lewis R Goldberg. Language and individual differences: The search for universals in personality lexicons. *Review of personality and social psychology*, 2(1):141–165, 1981.

[103] PT Costa and RR McCrae. Neo five-factor inventory (neo-ffi).

[104] PT Costa. J, mccrae rr.(1992) revised neo personality inventory (neo-pir) and neo five-factor inventory (neo-ffi) professional manual. *Psychological assessment inventories*.

[105] Gordon W Allport. Concepts of trait and personality. *Psychological Bulletin*, 24(5):284, 1927.

[106] Adrian Furnham, Steven C Richards, and Delroy L Paulhus. The dark triad of personality: A 10 year review. *Social and Personality Psychology Compass*, 7(3):199–216, 2013.

[107] Ludwig Klages and Walter Henry Johnston. The science of character. 1933.

[108] Yla R Tausczik and James W Pennebaker. The psychological meaning of words: Liwc and computerized text analysis methods. *Journal of language and social psychology*, 29(1):24–54, 2010.

[109] John M Digman and Naomi K Takemoto-Chock. Factors in the natural language of personality: Re-analysis, comparison, and interpretation of six major studies. *Multivariate behavioral research*, 16(2):149–170, 1981.

[110] Willem K Hofstee, Boele De Raad, and Lewis R Goldberg. Integration of the big five and circumplex approaches to trait structure. *Journal of personality and social psychology*, 63(1):146, 1992.

[111] Samuel D Gosling, Peter J Rentfrow, and William B Swann. A very brief measure of the big-five personality domains. *Journal of Research in personality*, 37(6):504–528, 2003.

[112] Sampo V Paunonen and Douglas N Jackson. What is beyond the big five? plenty! *Journal of personality*, 68(5):821–835, 2000.

[113] Cindy K Chung and James W Pennebaker. Revealing dimensions of thinking in open-ended self-descriptions: An automated meaning extraction method for natural language. *Journal of research in personality*, 42 (1):96–132, 2008.

[114] Shlomo Argamon, Moshe Koppel, James W Pennebaker, and Jonathan Schler. Mining the blogosphere: Age, gender and the varieties of self-expression. *First Monday*, 12(9), 2007.

[115] Thomas Holtgraves. Text messaging, personality, and the social context. *Journal of research in personality*, 45(1):92–99, 2011.

[116] Chris Sumner, Alison Byers, and Matthew Shearing. Determining personality traits & privacy concerns from facebook activity. *Black Hat Briefings*, 11:197–221, 2011.

[117] Jennifer Golbeck, Cristina Robles, Michon Edmondson, and Karen Turner. Predicting personality from twitter. In *Privacy, Security, Risk and Trust (PASSAT) and 2011 IEEE Third Inernational Conference on Social Computing (SocialCom), 2011 IEEE Third International Conference on*, pages 149–156. IEEE, 2011.

[118] Francisco Iacobelli, Alastair J Gill, Scott Nowson, and Jon Oberlander. Large scale personality classification of bloggers. In *Affective Computing and Intelligent Interaction*, pages 568–577. Springer, 2011.

[119] Chris Sumner, Alison Byers, Rachel Boochever, and Gregory J Park. Predicting dark triad personality traits from twitter usage and a linguistic analysis of tweets. In *Machine Learning and Applications (ICMLA), 2012 11th International Conference on*, volume 2, pages 386–393. IEEE, 2012.

[120] Barbara Plank and Dirk Hovy. Personality traits on twitterorhow to get 1,500 personality tests in a week. 2015.

[121] Gregory Park, H Andrew Schwartz, Johannes C Eichstaedt, Margaret L Kern, Michal Kosinski, David J Stillwell, Lyle H Ungar, and Martin EP Seligman. Automatic personality assessment through social media language. *Journal of personality and social psychology*, 108(6):934, 2015.

[122] Fei Liu, Julien Perez, and Scott Nowson. A language-independent and compositional model for personality trait recognition from short texts. *arXiv preprint arXiv:1610.04345*, 2016.

[123] Leqi Liu, Daniel Preotiuc-Pietro, Zahra Riahi Samani, Mohsen E Moghaddam, and Lyle H Ungar. Analyzing personality through social media profile picture choice. 2016.

[124] Michal Kosinski, Sandra C Matz, Samuel D Gosling, Vesselin Popov, and David Stillwell. Facebook as a research tool for the social sciences: Opportunities, challenges, ethical considerations, and practical guidelines. *American Psychologist*, 70(6):543, 2015.

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

[126] Andrew Kachites McCallum. Mallet: A machine learning for language toolkit. 2002.