Statistically Significant Detection of Linguistic Change

Vivek Kulkarni
Stony Brook University, USA
vvkulkarni@cs.stonybrook.edu

Rami Al-Rfou
Stony Brook University, USA
ralrfou@cs.stonybrook.edu

Bryan Perozzi
Stony Brook University, USA
bperozzi@cs.stonybrook.edu

Steven Skiena
Stony Brook University, USA
skiena@cs.stonybrook.edu

ABSTRACT

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.

Categories and Subject Descriptors

H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval

Keywords

Web Mining;Computational Linguistics

1. INTRODUCTION

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.

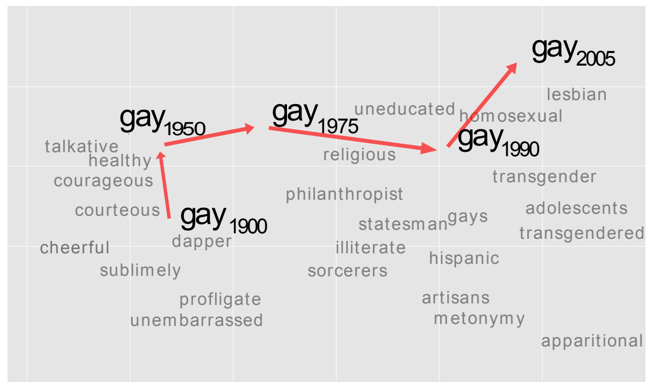


Figure 1: A 2-dimensional projection of the latent semantic space captured by our algorithm. Notice the semantic trajectory of the word `gay` transitioning meaning in the space.

In this paper, 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) 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 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

WWW 2015, May 18–22, 2015, Florence, Italy.
ACM 978-1-4503-3469-3/15/05.
http://dx.doi.org/10.1145/2736277.2741627 .

usage. The difference between frequency-based approaches and distributional methods is illustrated in Figure 2. Figure 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 storm called `Hurricane Sandy` striking the Atlantic Coast of the United States. However, only one of those words (`Sandy`) actually acquired a new meaning. Note that while the word `Hurricane` definitely experienced a surge in frequency of usage, it did not undergo any change in meaning. Indeed, using our distributional method (Figure 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:

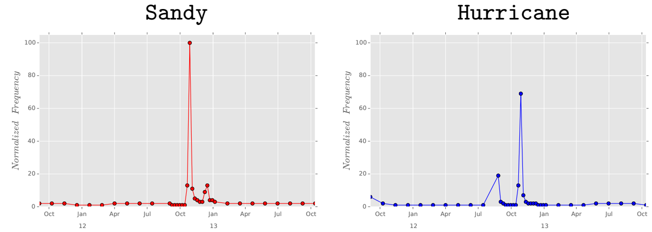
- **Word Evolution Modeling**: We study three different methods for the statistical modeling of word evolution over time. We use measures of frequency, part-of-speech tag distribution, and word co-occurrence to construct time series for each word under investigation.(Section 3)
- **Statistical Soundness**: 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 4)
- **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 6)

The rest of the paper is structured as follows. In Section 2 we define the problem of language shift detection over time. Then, we outline our proposals to construct time series modeling word evolution in Section 3. Next, in Section 4, we describe the method we developed for detecting significant changes in natural language. We describe the datasets we used in Section 5, and then evaluate our system both qualitatively and quantitatively in Section 6. We follow this with a treatment of related work in Section 7, and finally conclude with a discussion of the limitations and possible future work in Section 8.

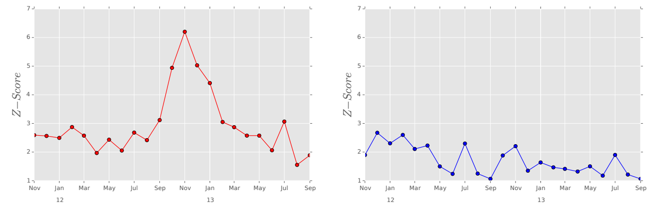
2. PROBLEM DEFINITION

Our problem is to quantify the linguistic shift in word meaning (semantic or context change) and usage across time. Given a temporal corpora $\mathcal{C}$ that is created over a time span

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



(a) **Frequency method (Google Trends)**



(b) **Distributional method**

Figure 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`.

$\mathcal{S}$, 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 at time t. In Section 3, 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?

3. TIME SERIES CONSTRUCTION

Constructing the time series is the first step in quantifying the significance of word change. Different approaches capture various aspects of a 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 6.

3.1 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

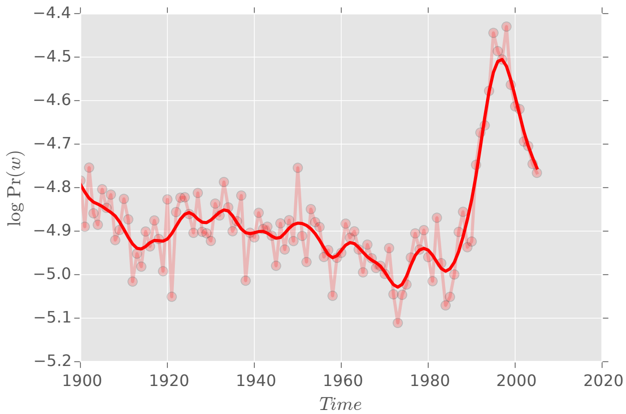


Figure 3: Frequency usage of the word **gay** over time, observe the sudden change in frequency in the late 1980s.

which are used in research to predict economical and public health changes [7, 9]. 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:

$$\mathcal{T}_t(w) = \log \frac{\#(w \in \mathcal{C}_t)}{|\mathcal{C}_t|}, \quad (1)$$

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 3. Observe the sudden jump in late 1980s of the word **gay** in frequency.

3.2 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 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 technology company (Figure 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, we calculate the divergence between the POS distributions in both snapshots. We construct the time series as follows:

$$\mathcal{T}_t(w) = JSD(Q_0, Q_t) \quad (2)$$

where JSD is the Jenssen-Shannon divergence [21].

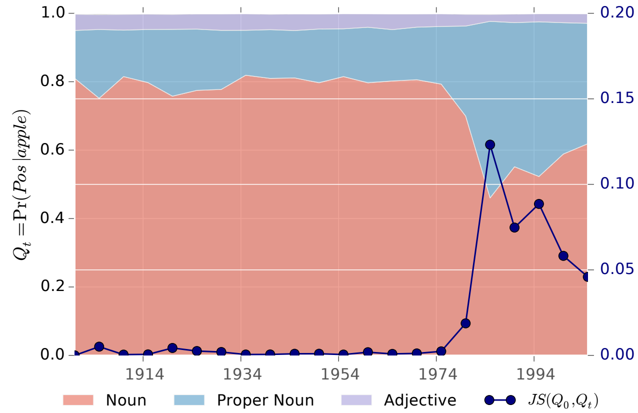


Figure 4: Part of speech tag probability distribution of the word **apple** (stacked area chart). Observe that the "Proper Noun" tag has dramatically increased in 1980s. The same trend is clear from the time series constructed using Jenssen-Shannon Divergence (dark blue line).

Figure 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.

3.3 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*) [5] have enabled the scalable learning of such models. We use a variation of these models [28] 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 1).

In this section we present our distributional approach in detail. Specifically 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.

3.3.1 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 word 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

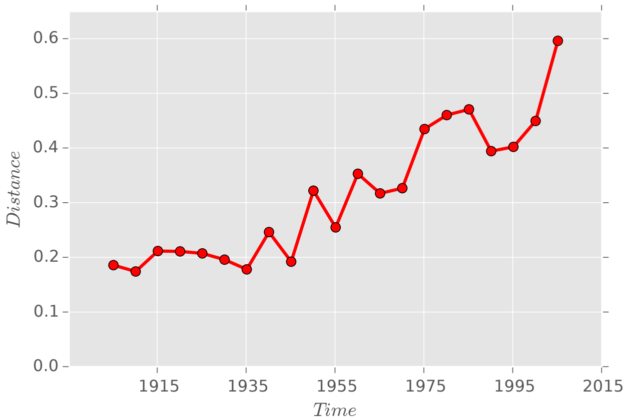


Figure 5: Distributional time series for the word `tape` over time using word embeddings. Observe the change of behavior starting in the 1950s, which is quite apparent by the 1970s.

w_i ($\mathbf{w}_i = \phi_t(w_i)$), we seek to maximize the probability of w_j through the following equation:

$$\Pr(w_j \mid \mathbf{w}_i) = \frac{\exp\left(\mathbf{w}_j^T \mathbf{w}_i\right)}{\sum\limits_{w_k \in \mathcal{V}} \exp\left(\mathbf{w}_k^T \mathbf{w}_i\right)} \tag{3}$$

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

$$J = \sum_{w_i \in \mathcal{C}_t} \sum_{\substack{j=i-m \\ j!=i}}^{i+m} -\log \Pr(w_j \mid \mathbf{w}_i) \tag{4}$$

Notice that the normalization factor that appears in Eq. (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 [30, 31]. 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 [6], as follows:

$$\phi_t(w_i) = \phi_t(w_i) - \alpha \times \frac{\partial J}{\partial \phi_t(w_i)}, \tag{5}$$

where α is the learning rate. We calculate the derivatives of the model using the back-propagation algorithm [34]. We use the following measure of training convergence:

$$\rho = \frac{1}{|\mathcal{V}|} \sum_{w \in \mathcal{V}} \frac{{\phi^k}^T(w)\phi^{k+1}(w)}{\|\phi^k(w)\|_2 \|\phi^{k+1}(w)\|_2}, \tag{6}$$

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 the `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} [27].

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

3.3.2 *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:

$$\underset{t' \mapsto t}{\mathbf{W}}(w) = \underset{\mathbf{W}}{\operatorname{argmin}} \sum_{\substack{w_i \in \\ k\text{-NN}(\phi_{t'}(w))}} \|\phi_{t'}(w_i)\mathbf{W} - \phi_t(w_i)\|_2^2, \tag{7}$$

which is equivalent to a piecewise linear regression model.

3.3.3 *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. 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

$$\mathcal{T}_t(w) = 1 - \frac{(\phi_t(w)\mathbf{W}_{t \mapsto n}(w))^T (\phi_0(w)\mathbf{W}_{0 \mapsto n}(w))}{\|\phi_t(w)\mathbf{W}_{t \mapsto n}(w)\|_2 \|\phi_0(w)\mathbf{W}_{0 \mapsto n}(w)\|_2} \tag{8}$$

Figure 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.

4. CHANGE POINT DETECTION

Given a time series of a word $\mathcal{T}(w)$, constructed using one of the methods discussed in Section 3, 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 [1, 3, 38]. Our approach models the time series based on the *Mean Shift* model described in [38]. 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

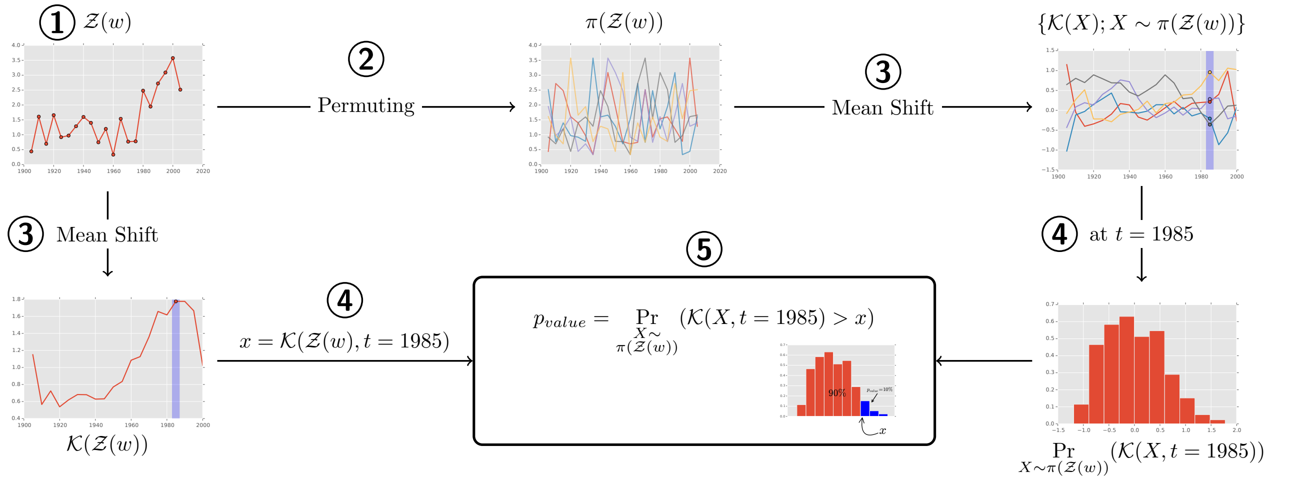


Figure 6: Our change point detection algorithm. In Step ①, we normalize the given time series $\mathcal{T}(w)$ to produce $\mathcal{Z}(w)$. Next, we shuffle the time series points producing the set $\pi(\mathcal{Z}(w))$ (Step ②). Then, we apply the mean shift transformation ($\mathcal{K}$) on both the 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).

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
1:  Z(w) ← Normalize T(w).
2:  Compute mean shift series K(Z(w))
    // Bootstrapping
3:  BS ← ∅ {Bootstrapped samples}
4:  repeat
5:    Draw P from π(Z(w))
6:    BS ← BS ∪ P
7:  until |BS| = B
8:  for i ← 1, n do
9:    p-value(w, i) ← (1/B) Σ_{P∈BS} [K_i(P) > K_i(Z(w))]
10: end for
    // Change Point Detection
11: C ← {j | j ∈ [1, n] and Z_j(w) >= γ}
12: p-value ← min_{j∈C} p-value(w, j)
13: ECP ← argmin_{j∈C} p-value(w, j)
14: return p-value, ECP
```

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 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:

$$\mathcal{Z}_i(w) = \frac{\mathcal{T}_i(w) - \mu_i}{\sqrt{Var_i}}, \tag{9}$$

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* [38]. 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:

$$\mathcal{K}(S) = \frac{1}{l-j}\sum_{k=j+1}^{l} S_k - \frac{1}{j}\sum_{k=1}^{j} S_k \tag{10}$$

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 [12] (see Figure 6 and Lines 3-10) 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 3-10). 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 [38]). 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

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

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

Table 1: Summary of our datasets

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).

5. 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 1 shows a summary of three different datasets spanning different modes of expression on the Internet: books, an online forum and a micro-blog.

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. The 5-grams include phrases like '`thousand pounds less then nothing`' and '`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 [14, 22, 23].

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. The reviews describe user's opinions of a movie, for example: '`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 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 tweet is a status message with up to 140 characters: '`I hope sandy doesn't rip the roof off the pool while we're swimming ...`'.

6. EXPERIMENTS

In this section, we apply our methods to each dataset presented in Section 5 and identify words that have changed usage over time. We describe the results of our experiments below. The code used for running these experiments is available at the first author's website.[4]

6.1 Time Series Analysis

As we shall see in Section 6.4.1, 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 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 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 signifantly in meaning, 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. 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 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* methods indicate the change in meaning of the word `diet`, it is only the *Distributional* method that detects the right point of change (as shown in Table 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 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.

6.2 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 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

[4] `http://vivekkulkarni.net`

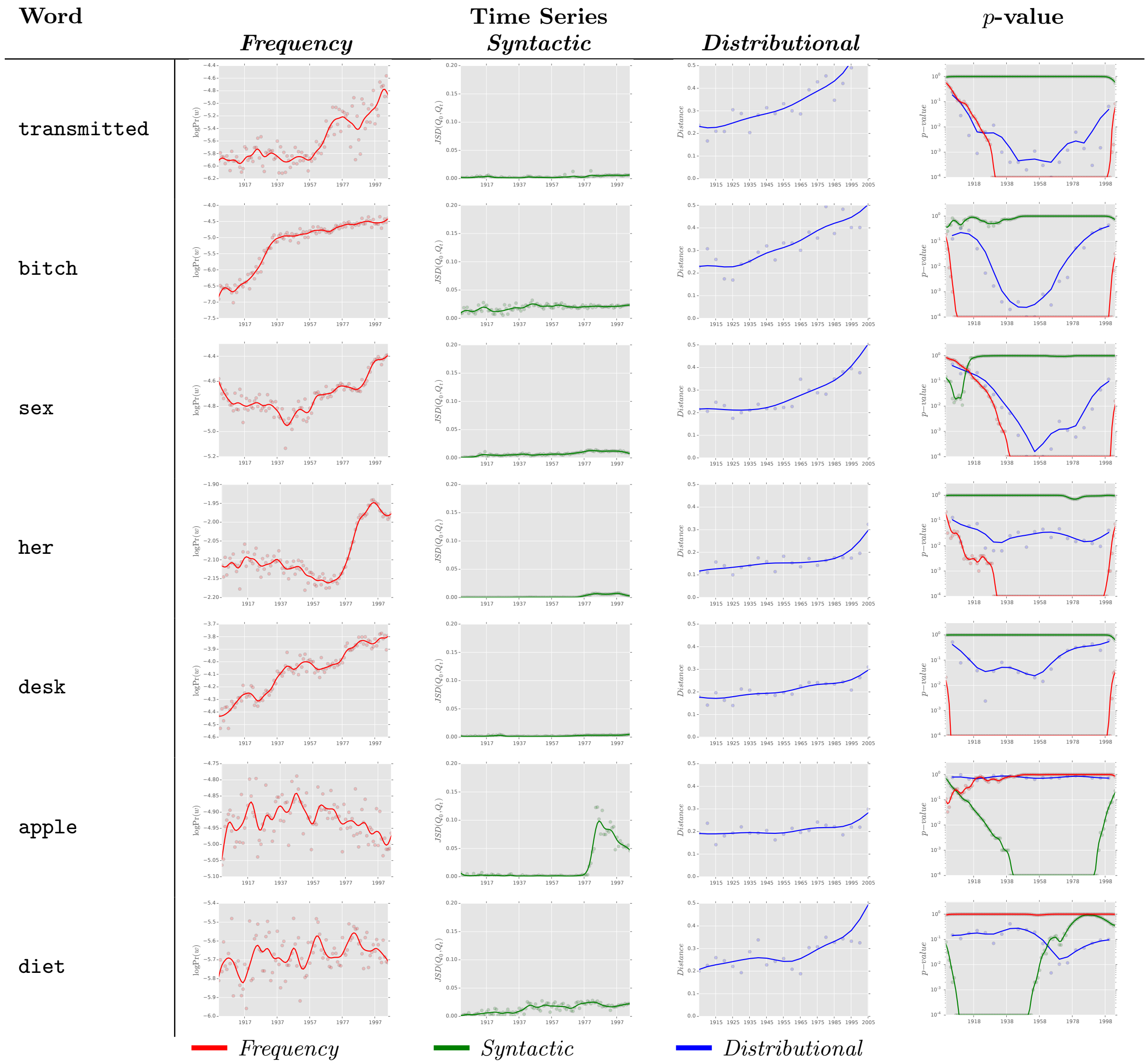


Table 2: Comparison of our different methods of constructing linguistic shift time series on the Google Books Ngram Corpus. The first three columns represent time series for a sample of words. The last column shows the p-value for each time step of each method, as generated by our change point detection algorithm.

(Figure 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 in 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 [16]. 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 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` that is widely used as a proper noun in 1989, which coincides with George H. W. Bush's presidency in USA.

6.3 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).

	Word	ECP	p-value	Past ngram	Present ngram
Distributional better	`recording`	1990	0.0263	*to be ashamed of recording that*	*recording, photocopying*
	`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 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)

	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*
Twitter	`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 4: Sample of words detected by our *Distributional* method on Amazon Reviews and Twitter.

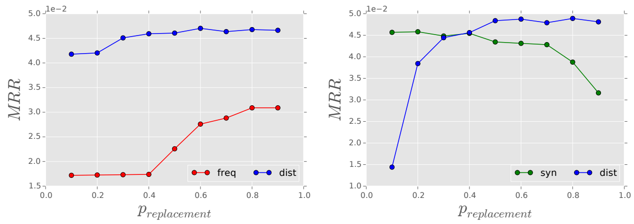
Table 4 shows results from our *Distributional* method on the Amazon Reviews and Twitter datasets. New technologies and products introduced new meanings to words like `streaming`, `ray`, `rays`, and `combo`. The word `twilight` acquired a new sense in 2009 concurrent with the release of the Twilight movie in November 2008.

Similar trends can be observed on 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 hit the east coast of USA. Similarly we see that 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.

6.4 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



(a) Frequency Perturbation (b) Syntactic Perturbation

Figure 7: Performance of our proposed methods under different scenarios of perturbation.

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.

6.4.1 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.

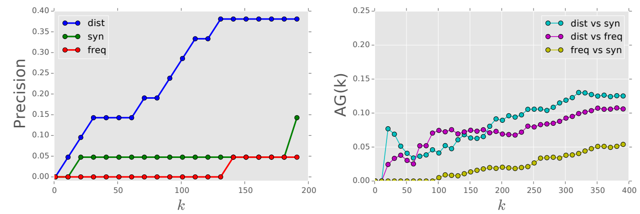
Our synthetic corpus is created as follows: First, we duplicate a copy of a Wikipedia corpus[5] 20 times to model time snapshots. We tagged the Wikipedia corpora with part of speech tags using the *TextBlob* tagger[6]. Next, we introduce changes to a word's usage to model linguistic shift. To do this, 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[7]. 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 half snapshots of our temporal corpus.

Figure 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 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

[5]`http://mattmahoney.net/dc/text8.zip`
[6]`http://textblob.readthedocs.org/en/dev/`
[7]NLTK Stopword List: `http://www.nltk.org/`



(a) Reference Dataset (b) Methods Agreement

Figure 8: Method performance and agreement on changed words in the Google Books Ngram Corpus.

the same part of speech categories. Figure 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 noting 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.

6.4.2 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 have been suggested by prior work [15, 17, 19, 39] as having undergone a linguistic change[8]. 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:

$$\text{Precision@}k(L, D) = \frac{|L[1:k] \cap D|}{|D|} \tag{11}$$

Figure 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.

6.4.3 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 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.

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

6.4.4 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:

$$AG(M_1(k), M_2(k)) = \frac{|M_1(k) \cap M_2(k)|}{|M_1(k) \cup M_2(k)|} \quad (12)$$

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

Figure 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.

7. RELATED WORK

Here we discuss the four most relevant areas of related work: linguistic shift, word embeddings, change point detection, and Internet linguistics.

Linguistic Shift: There has been a surge in the work about language evolution over time. Michel et al. [25] detected important political events by analyzing frequent patterns. Juola [18] compared language from different time periods and quantified the change. Lijffijt et al. [20] and Säily et al. [35] study variation in noun/pronoun frequencies, and lexical stability in a historical corpus. Different from these 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.

Gulordava and Baroni [15] propose a distributional similarity approach to detecting semantic change in the Google Book Ngram corpus between 2 time periods. Wijaya and Yeniterzi [39] study evolution of words using a topic modeling approach but do not suggest an explicit change point detection algorithm. Our work differs from the above studies by tracking word evolution through multiple time periods and explicitly providing a change point detection algorithm to detect significant changes. Mitra et al. [29] 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. [19] 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.

Word Embeddings: Bengio et al. [4] used word embeddings to develop a neural language model that outperforms traditional ngram models. These word embeddings have been shown to capture fine grain structures and regularities in the data [26, 27, 32]. Moreover, they have proved to be useful for a wide range of natural language processing tasks [2, 8, 10]. The same technique of learning word embeddings has been applied recently to graph representations [33].

Change Point Detection: Change point detection and analysis is an important problem in the area of time series analysis and modeling. Taylor [38] describes control charts and CUSUM based methods in detail. Adams and MacKay [1] presents a Bayesian approach to online change point detection. The method of bootstrapping and establishing statistical significance is outlined in [12]. Basseville and Nikiforov [3] provides an excellent survey on several elementary change point detection techniques and time series models.

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. [36] and Tagliamonte and Denis [37] study how teenagers use messaging media focusing on their usage patterns and the resulting implications on design of e-mail and instant messaging. Merchant [24] study the language use by teenagers in online chat forums. An excellent survey on Internet Linguistics is provided by Crystal [11] and includes linguistic analyses of social media like Twitter, Facebook or Google+.

8. CONCLUSIONS AND FUTURE WORK

In this work, we have proposed three approaches to model word evolution through different time series construction methods. Our computational approach then uses statistically sound change point detection algorithms to detect significant linguistic shifts. Finally, we demonstrated our method's effectiveness 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 media like Tweets and Amazon Reviews, we were able to detect recent events like storms, 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 for the fields of Semantic Search and the recently burgeoning field of Internet Linguistics.

Our future work in the area will focus on the real time analysis of linguistic shift, the creation of better resources for the quantitative evaluation of computational methods, and the effects of attributes like geographical location and content source on the underlying mechanisms of meaning change in language.

Acknowledgments

We thank Andrew Schwartz for providing us access to the Twitter data. This research was partially supported by NSF Grants DBI-1355990 and IIS-1017181, a Google Faculty Research Award, a Renaissance Technologies Fellowship and the Institute for Computational Science at Stony Brook University.

References

[1] R. P. Adams and D. J. MacKay. Bayesian online changepoint detection. Cambridge, UK, 2007.

[2] R. Al-Rfou, B. Perozzi, and S. Skiena. Polyglot: Distributed word representations for multilingual nlp. In *CoNLL*, 2013.

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

[4] Y. Bengio, H. Schwenk, et al. Neural probabilistic language models. In *Innovations in Machine Learning*, pages 137–186. Springer, 2006.

[5] Y. Bengio et al. Representation learning: A review and new perspectives. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(8):1798–1828, 2013.

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

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

[8] Y. Chen, B. Perozzi, R. Al-Rfou, and S. Skiena. The expressive power of word embeddings. *CoRR*, abs/1301.3226, 2013.

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

[10] R. Collobert, J. Weston, et al. Natural language processing (almost) from scratch. *J. Mach. Learn. Res.*, 12: 2493–2537, Nov. 2011.

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

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

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

[14] Y. Goldberg and J. Orwant. A dataset of syntactic-ngrams over time from a very large corpus of english books. In **SEM*, 2013.

[15] K. Gulordava and M. Baroni. A distributional similarity approach to the detection of semantic change in the google books ngram corpus. In *GEMS*, July 2011.

[16] D. Immerwahr. The books of the century, 2014. URL `http://www.ocf.berkeley.edu/~immer/books1970s`.

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

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

[19] Y. Kim, Y.-I. Chiu, K. Hanaki, et al. Temporal analysis of language through neural language models. In *ACL*, 2014.

[20] J. Lijffijt, T. Säily, and T. Nevalainen. Ceecing the baseline: Lexical stability and significant change in a historical corpus. VARIENG, 2012.

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

[22] Y. Lin, J. B. Michel, E. L. Aiden, J. Orwant, W. Brockman, and S. Petrov. Syntactic annotations for the google books ngram corpus. In *ACL*, 2012.

[23] J. Mann, D. Zhang, et al. 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.

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

[25] J. B. Michel, Y. K. Shen, et al. Quantitative analysis of culture using millions of digitized books. *Science*, 331 (6014):176–182, 2011.

[26] T. Mikolov et al. Linguistic regularities in continuous space word representations. In *Proceedings of NAACL-HLT*, 2013.

[27] T. Mikolov et al. Distributed representations of words and phrases and their compositionality. In *NIPS*, 2013.

[28] T. Mikolov et al. Efficient estimation of word representations in vector space. *CoRR*, abs/1301.3781, 2013.

[29] S. Mitra, R. Mitra, et al. That's sick dude!: Automatic identification of word sense change across different timescales. In *ACL*, 2014.

[30] A. Mnih and G. E. Hinton. A scalable hierarchical distributed language model. *NIPS*, 21:1081–1088, 2009.

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

[32] B. Perozzi, R. Al-Rfou, V. Kulkarni, and S. Skiena. Inducing language networks from continuous space word representations. In *Complex Networks V*, volume 549 of *Studies in Computational Intelligence*, pages 261–273. 2014.

[33] B. Perozzi, R. Al-Rfou, and S. Skiena. Deepwalk: Online learning of social representations. In *KDD*, New York, NY, USA, August 2014. ACM.

[34] D. E. Rumelhart, G. E. Hinton, and R. J. Williams. Learning representations by back-propagating errors. *Cognitive modeling*, 1:213, 2002.

[35] T. Säily, T. Nevalainen, and H. Siirtola. Variation in noun and pronoun frequencies in a sociohistorical corpus of english. *Literary and Linguistic Computing*, 26(2): 167–188, 2011.

[36] D. J. Schiano, C. P. Chen, E. Isaacs, J. Ginsberg, U. Gretarsdottir, and M. Huddleston. Teen use of messaging media. In *Computer Human Interaction*, pages 594–595, 2002.

[37] S. A. Tagliamonte and D. Denis. Linguistc Ruin? LOL! Instant messaging and teen language. *American Speech*, 83:3–34, 2008.

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

[39] D. T. Wijaya and R. Yeniterzi. Understanding semantic change of words over centuries. In *DETECT*, 2011.