

# Statistically Significant Detection of Linguistic Change

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

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

Rami Al-Rfou  
Stony Brook University, USA  
ralrfou@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; I.2.6 [Artificial Intelligence]: Learning

## General Terms

Web Mining, Computational Linguistics, Time Series Modeling, Change Point Detection

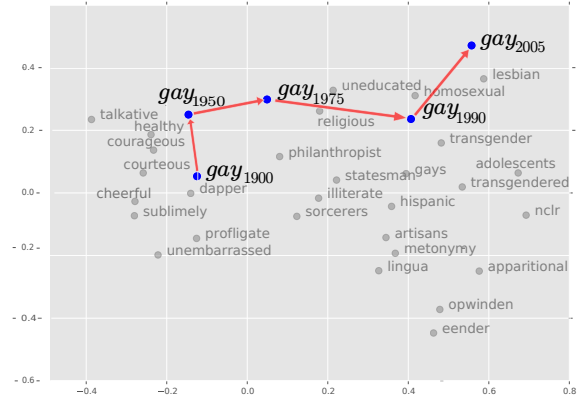


Figure 1: A 2-dimensional projection of the latent semantic space captured by our algorithm. Notice the semantic trajectory of the word *gay* transitioning meaning in the space.

## 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.

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). Ini-

tially, *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. Figure 2a shows the frequencies of two words, *Sandy* (red), and *Hurricane* (blue) as a percentage of search queries according to Google Trends<sup>1</sup>. 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. 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

<sup>1</sup><http://www.google.com/trends/>

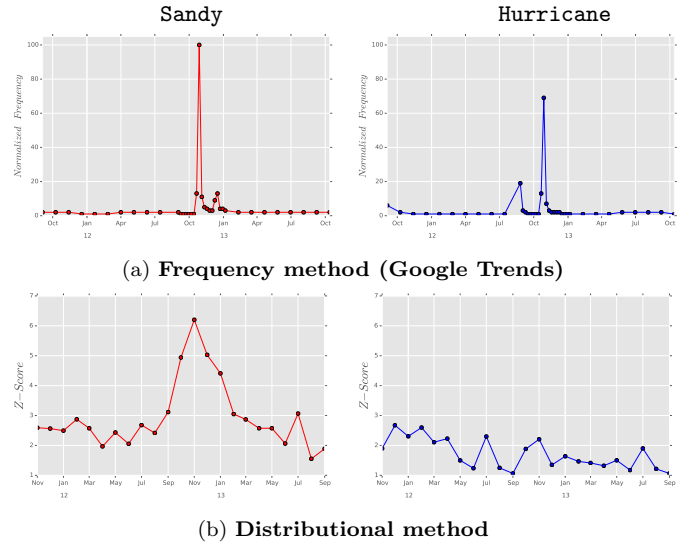


Figure 2: Comparison between Google Trends and our method. Observe how Google Trends shows spikes in frequency for both *Hurricane* (blue) and *Sandy* (red). Our method, in contrast, models change in usage and detects that only *Sandy* changed its meaning and not *Hurricane*.

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 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 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 statically 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 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

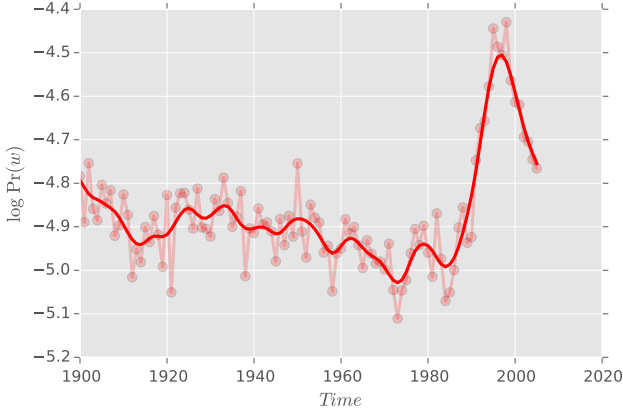


Figure 3: Frequency usage of the word **gay** over time, observe the sudden change in frequency in the late 1980s.

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 which are used in research to predict economical and public health changes [8, 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 technical company (Figure 4). To leverage this syntactic knowledge, we annotate our corpus

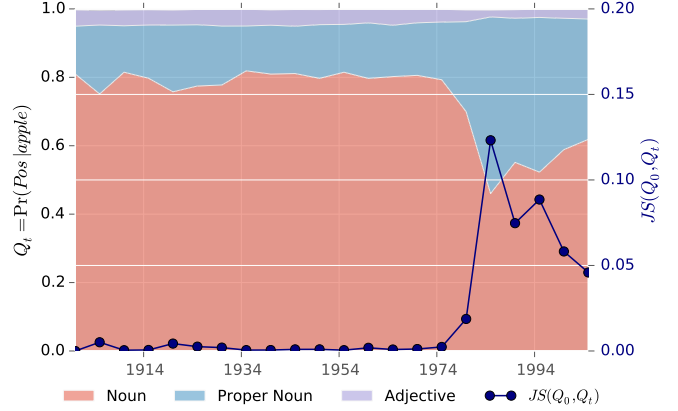


Figure 4: Part of speech tag probability distribution of the word **apple** (stacked area chart). Observe that the “Proper Noun” tag has dramatically increased in 1980s. The same trend is clear from the time series constructed using Jensen-Shannon Divergence (dark blue line).

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.

Specifically, we construct the time series as follows:

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

where JSD is the Jensen-Shannon divergence [23].

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 [15]. 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*) [6] 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).

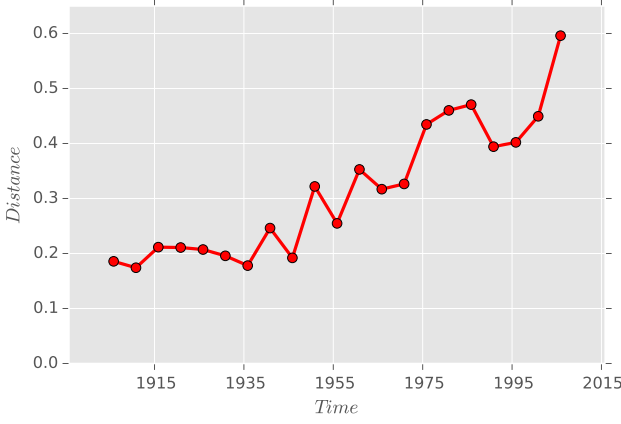


Figure 5: Distributional time series for the word **tape** over time using word embeddings. Observe the change of behavior starting in the 1950s, which is quite apparent by the 1970s.

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.

### 3.3.1 Learning embeddings

Given a time snapshot  $\mathcal{C}_t$  of the corpus, our goal is to learn  $\phi_t$  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:

$$\Pr(w_j | \mathbf{w}_i) = \frac{\exp(\mathbf{w}_j^T \mathbf{w}_i)}{\sum_{\mathbf{w}_k \in \mathcal{V}} \exp(\mathbf{w}_k^T \mathbf{w}_i)} \quad (3)$$

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  $m$  (Equation 4).

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

Notice that the normalization factor that appears in Equation 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 [32, 33]. 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 [7], as follows:

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

where  $\alpha$  is the learning rate. We calculate the derivatives of the model using the back-propagation algorithm. [35]. We use the following measure of training convergence:

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

where  $\phi^k$  is the model parameters after epoch  $k$ . We calculate  $\rho$  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<sup>2</sup>. 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  $10^{-5}$  [29].

### 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 co-ordinate 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  $\phi_t$  to a word  $w$  to be  $k\text{-NN}(\phi_t(w))$ . We seek to learn a linear transformation  $\mathbf{W}_{t' \rightarrow t}(w) \in \mathbb{R}^{d \times d}$  that maps a word from  $\phi_t$  to  $\phi_{t'}$  by solving the following optimization problem:

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

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 initial time snapshot  $\phi_0$  using a linear mapping (Eq. 7). This unification of coordinate system 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  $\phi_t(w) \mathbf{W}_{t \rightarrow 0}(w)$  and  $\phi_0(w)$  as the following:

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

Figure 5 shows the time series obtained using word embeddings for **tape**, which underwent a semantic change in the

<sup>2</sup><https://github.com/piskvorky/gensim>



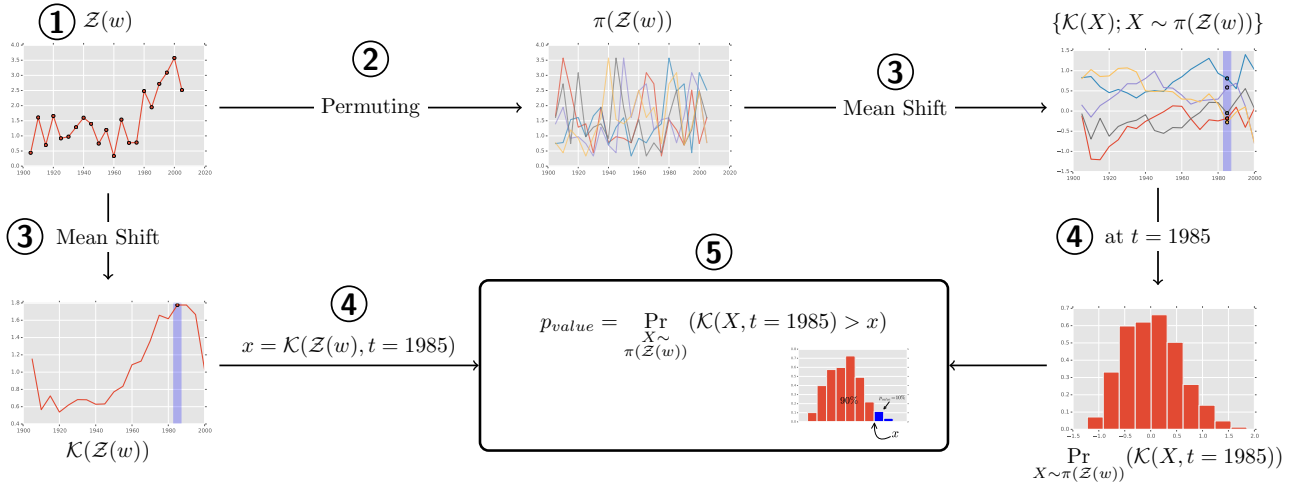


Figure 6: Our change point detection algorithm. In Step ①, we normalize the given time series  $\mathcal{T}(w)$  to produce  $\mathcal{Z}(w)$ . Next, we shuffle the time series points producing the set  $\pi(\mathcal{Z}(w))$  (Step ②). Then, we apply the mean shift transformation ( $\mathcal{K}$ ) on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

---

**Input:**  $\mathcal{T}(w)$ : Time series for the word  $w$ ,  $B$ : Number of bootstrap samples,  $\gamma$ : Z-Score threshold

**Output:**  $ECP$ : Estimated change point,  $p$ -value: Significance score.

```

// Preprocessing
1:  $\mathcal{Z}(w) \leftarrow \text{Normalize } \mathcal{T}(w)$ .
2: Compute mean shift series  $\mathcal{K}(\mathcal{Z}(w))$ 
// Bootstrapping
3:  $BS \leftarrow \emptyset$  {Bootstrapped samples}
4: repeat
5:   Draw  $P$  from  $\pi(\mathcal{Z}(w))$ 
6:    $BS \leftarrow BS \cup P$ 
7: until  $|BS| = B$ 
8: for  $i \leftarrow 1, n$  do
9:    $p\text{-value}(w, i) \leftarrow \frac{1}{B} \sum_{P \in BS} [\mathcal{K}_i(P) > \mathcal{K}_i(\mathcal{Z}(w))]$ 
10: end for
// Change Point Detection
11:  $C \leftarrow \{j | j \in [1, n] \text{ and } Z_j(w) \geq \gamma\}$ 
12:  $p\text{-value} \leftarrow \min_{j \in C} p\text{-value}(w, j)$ 
13:  $ECP \leftarrow \text{argmin}_{j \in C} p\text{-value}(w, j)$ 
14: return  $p\text{-value}$ ,  $ECP$ 

```

---

1950's 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.

There exists an extensive body of work on change point detection in time series [1, 3, 40]. Our approach models the time series based on the *Mean Shift* model described in [40]. 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 6.

Given a time series of a word  $\mathcal{T}(w)$ , we first normalize the time series. We calculate the mean  $\mu_i = \frac{1}{|V|} \sum_{w \in V} \mathcal{T}_i(w)$  and standard deviation  $Var_i = \frac{1}{|V|} \sum_{w \in V} (\mathcal{T}_i(w) - \mu_i)^2$  across all words. Then, we transform the time series into a *Z-Score* series as follows:

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

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* [40]. Let  $S = \mathcal{Z}_1(w), \mathcal{Z}_2(w), \dots, \mathcal{Z}_n(w)$  represent the time series. We model  $S$  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  $\mu_i$  is the mean and  $\epsilon_i$  is the random error at time  $i$ . We also assume that the errors  $\epsilon_i$  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:

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

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.<sup>3</sup>

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

<sup>3</sup>This is similar to the CUSUM based approach used for detecting change points which is also based on mean shift model.

statistical significance of observing a mean shift at time point  $j$ , we use bootstrapping [14] (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  $P$ , 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 [39]). 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  $\gamma$  (we typically set  $\gamma$  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, 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 [27]. 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 [16, 24, 25].

### *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.

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

Table 1: Summary of our datasets

### *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 ...

## 6. EXPERIMENTS

In this section, we apply our method for each dataset presented in Section 5 and identify words that have changed usage over time. We describe the results of our experiments below.

### 6.1 Time Series Analysis

As we shall see in Section 6.4, 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 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, 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

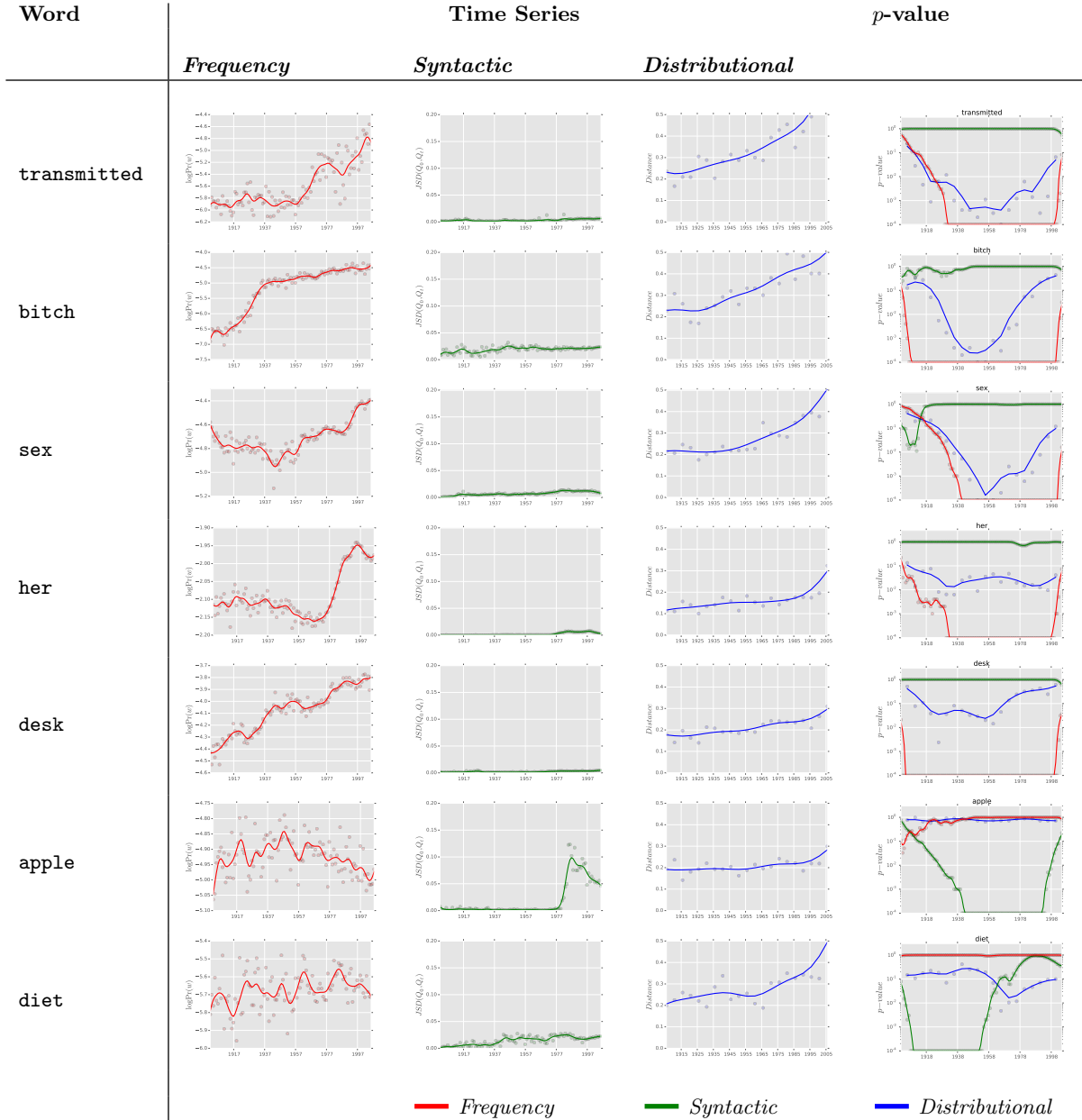


Table 2: Comparison of our different methods of constructing linguistic shift time series on the Google Books Ngram Corpus. The first three columns represent time series for a sample of words. The last column shows the *p*-value as generated by our point detection algorithm for each method.

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 change-points.

## 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 (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 intro-

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

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

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

duction 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 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 [20]. 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.

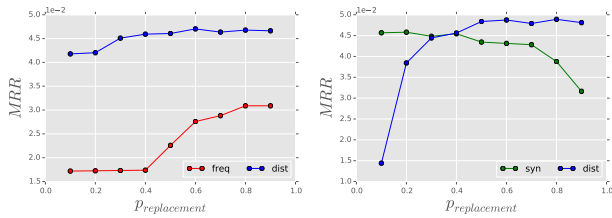
The *Syntactic* section of Table 3 shows that words like **hug** and **sink** were previously used mainly as verbs. Over time organizations and movements started using **hug** as a noun

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** became widely used as proper noun in 1989, which coincides with George H. W. Bush’s presidency in USA.

### 6.3 Cross Domain Analysis





(a) Frequency Perturbation (b) Syntactic Perturbation

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

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 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 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

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](http://mattmahoney.net/dc/text8.zip)<sup>4</sup> 20 times to model time snapshots. We tagged the Wikipedia corpora with part of speech tags using the [TextBlob tagger](http://textblob.readthedocs.org/en/dev/)<sup>5</sup>. 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<sup>6</sup>. 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 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 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 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  $p_{\text{replacement}} > 0.4$  without requiring any language specific resources or annotators.

## 7. 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.

### 7.1 Linguistic Shift

There has been a surge in the work about language evolution over time [11, 17, 18, 21, 27, 41]. Michel et al. [27] detected important political events by analyzing frequent patterns. Juola [21] 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 [11, 17, 18, 38, 41] 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. [31] 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

<sup>4</sup><http://mattmahoney.net/dc/text8.zip>

<sup>5</sup><http://textblob.readthedocs.org/en/dev/>

<sup>6</sup>NLTK Stopword List: <http://www.nltk.org/>

procedure proposed by Kim et al. [22] 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.

## 7.2 Word Embeddings

Hinton [19] proposed distributed representations (word embeddings), to learn a mapping of symbolic data to continuous space. Bengio et al. [5] 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 [4, 13, 32, 33]. Word embeddings are shown to capture fine grain structures and regularities in the data [29, 30]. Moreover, they proved to be useful for a wide range of natural language processing tasks [2, 10]. The same technique of learning word embeddings has been applied recently to learning graph representation [34].

## 7.3 Change point detection

Change Point Detection and Analysis is an important problem in the area of Time Series Analysis and Modeling. Taylor [40] describes control charts and CUSUM based methods in detail. Adams and MacKay [1] describes a Bayesian approach to Online Change Point Detection. The method of bootstrapping and establishing statistical significance is outlined in [14]. Basseville and Nikiforov [3] provides an excellent survey on several elementary change point detection techniques and time series models.

## 7.4 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. [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 (IM). Merchant [26] study the language use by teenagers in online chat forums. An excellent survey on Internet Linguistics is provided by [12] and includes linguistic analyses of social media like Twitter, Facebook or Google+.

## 8. CONCLUSIONS AND FUTURE WORK

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.

For our future work, we will focus on building real time analysis of data that link other attributes of data like geographical locations and content source to understand better the mechanism of meaning change and influential players in such change.

## Acknowledgments

We thank Andrew Schwartz for providing us access to the Twitter stream data and for insightful discussions.

## References

- [1] R. P. Adams and D. J. MacKay. Bayesian online change-point detection. Cambridge, UK, 2007.
- [2] R. Al-Rfou, B. Perozzi, and S. 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.
- [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 and J.-S. Senécal. Adaptive importance sampling to accelerate training of a neural probabilistic language model. *Neural Networks, IEEE Transactions on*, 19(4):713–722, 2008.
- [5] Y. Bengio, H. Schwenk, J.-S. Senécal, F. Morin, and J.-L. Gauvain. Neural probabilistic language models. In *Innovations in Machine Learning*, pages 137–186. Springer, 2006.
- [6] Y. Bengio, A. Courville, and P. Vincent. Representation learning: A review and new perspectives. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 35(8):1798–1828, 2013.
- [7] L. Bottou. Stochastic gradient learning in neural networks. In *Proceedings of Neuro-Nimes 91*, Nimes, France, 1991. EC2.
- [8] 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.
- [9] H. Choi and H. Varian. Predicting the present with google trends. *Economic Record*, 88:2–9, 2012.
- [10] R. Collobert, J. Weston, L. Bottou, M. Karlen, K. Kavukcuoglu, and P. Kuksa. Natural language processing (almost) from scratch. *J. Mach. Learn. Res.*, 12: 2493–2537, Nov. 2011.
- [11] P. Cook, J. H. Lau, M. Rundell, D. McCarthy, and T. 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.
- [12] D. Crystal. *Internet Linguistics: A Student Guide*. Routledge, New York, NY, 10001, 1st edition, 2011.
- [13] J. Dean, G. S. Corrado, R. Monga, K. Chen, M. Devin, Q. V. Le, M. Z. Mao, M. Ranzato, A. Senior, P. Tucker, K. Yang, and A. Y. Ng. Large scale distributed deep networks. In *NIPS*, 2012.
- [14] B. Efron and R. J. Tibshirani. An introduction to the bootstrap. 1971.
- [15] J. R. Firth. *Papers in Linguistics 1934-1951: Repr.* Oxford University Press, 1961.

- [16] Y. Goldberg and J. 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.
- [17] K. Gulordava and M. 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.
- [18] G. Heyer, F. Holz, and S. Teresniak. Change of topics over time and tracking topics by their change of meaning. In A. L. N. Fred, editor, *KDIR 2009: Proc. of Int. Conf. on Knowledge Discovery and Information Retrieval*. INSTICC Press, October 2009.
- [19] G. 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.
- [20] D. Immerwahr. The books of the century, 2014. URL <http://www.ocf.berkeley.edu/~immer/books1970s>.
- [21] P. Juola. The time course of language change. *Computers and the Humanities*, 37(1):77–96, 2003.
- [22] Y. Kim, Y.-I. Chiu, K. Hanaki, D. Hegde, and S. 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.
- [23] J. Lin. Divergence measures based on the shannon entropy. *Information Theory, IEEE Transactions on*, 37(1):145–151, 1991.
- [24] Y. Lin, J.-B. Michel, E. L. Aiden, J. Orwant, W. Brockman, and S. 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.
- [25] J. Mann, D. Zhang, L. Yang, D. Das, and S. 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.
- [26] 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.
- [27] J.-B. Michel, Y. K. Shen, A. P. Aiden, A. Veres, M. K. Gray, T. G. B. Team, J. P. Pickett, D. Hoiberg, D. Clancy, P. Norvig, J. Orwant, S. Pinker, M. A. Nowak, and E. L. Aiden. Quantitative analysis of culture using millions of digitized books. *Science*, 331(6014):176–182, 2011.
- [28] T. Mikolov, K. Chen, G. Corrado, and J. Dean. Efficient estimation of word representations in vector space. *CoRR*, abs/1301.3781, 2013.
- [29] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. 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.
- [30] T. Mikolov, W.-t. Yih, and G. Zweig. Linguistic regularities in continuous space word representations. In *Proceedings of NAACL-HLT*, pages 746–751, 2013.
- [31] S. Mitra, R. Mitra, M. Riedl, C. Biemann, A. Mukherjee, and P. 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.
- [32] A. Mnih and G. E. Hinton. A scalable hierarchical distributed language model. *Advances in neural information processing systems*, 21:1081–1088, 2009.
- [33] 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.
- [34] B. Perozzi, R. Al-Rfou, and S. 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.
- [35] D. E. Rumelhart, G. E. Hinton, and R. J. Williams. Learning representations by back-propagating errors. *Cognitive modeling*, 1:213, 2002.
- [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. LINGUISTIC RUIN? LOL! INSTANT MESSAGING AND TEEN LANGUAGE. *American Speech*, 83:3–34, 2008.
- [38] X. Tang, W. Qu, and X. Chen. Semantic change computation: A successive approach. In L. Cao, H. Motoda, J. Srivastava, E.-P. Lim, I. King, P. Yu, W. Nejdl, G. Xu, G. Li, and Y. Zhang, editors, *Behavior and Social Computing*, volume 8178 of *Lecture Notes in Computer Science*, pages 68–81. Springer International Publishing, 2013.
- [39] W. A. TAYLOR. Change-Point Analysis: A Powerful New Tool For Detecting Changes.
- [40] W. A. Taylor. Change-point analysis: A powerful new tool for detecting changes, 2000.
- [41] D. T. Wijaya and R. 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.