Using Google Books Ngram in Detecting Linguistic Shifts over Time

Alaa El-Ebshihy, Nagwa El-Makky and Khaled Nagi
Dept. of Computer and Systems Engineering, Faculty of Engineering, Alexandria University, Egypt

Keywords: Linguistic Shift, Semantic Change, Google Books Ngram, FastText, Time Series Analysis, Computational Linguistics.

Abstract: The availability of large historical corpora, such as Google Books Ngram, makes it possible to extract various meta information about the evolution of human languages. Together with advances in machine learning techniques, researchers recently use the huge corpora to track cultural and linguistic shifts in words and terms over time. In this paper, we develop a new approach to quantitatively recognize semantic changes of words during the period between 1800 and 1990. We use the state-of-the-art FastText approach to construct word embedding for Google Books Ngram corpus for the decades within the time period 1800-1990. We use a time series analysis to identify words that have a statistically significant change in the period between 1900 and 1990. We conduct a performance evaluation study to compare our approach against related work, we show that our system is more robust against morphological language variations.

1 INTRODUCTION

With the evolution of natural languages over time, some words gain new meanings, some are being newly developed, while others disappear. This gradually affects the way the words are being used. As a result, the idea of automatic detection of semantic change of words gained considerable interest in recent research (Hamilton et al., 2016).

Semantic change of a word is defined as *change of one or more meanings of the word in time* (Lehmann, 2013). Developing automatic techniques for identifying changes to word meanings over time is beneficial to various natural language processing operations. For instance, it helps in information retrieval and question answering systems, since time-related information would increase the precision of query disambiguation and document retrieval.

The presence of large historical corpora, such as Google Books Ngram (Lin et al., 2012), makes it possible to track such linguistic shifts. Together with the advances in machine learning techniques, the interests of researchers to develop computational strategies for identifying and quantifying changes in languages raised significantly (Kim et al., 2014). Interesting work investigate changes by the analysis of word frequencies (Gulordava and Baroni, 2011; Mihalcea and Nastase, 2012; Sang, 2016). Others use distributional and Neural Language models (Gulordava and Baroni, 2011; Kim et al., 2014; Hamilton et al., 2016; Frermann and Lapata, 2016; Liao and Cheng, 2016).

In our work, we develop an approach for measuring semantic change using *word embeddings* and *time series* analysis. Word embedding is a representation of a word to a low dimensional real-valued vector (Levy et al., 2015). Whereas, the science of time series analysis includes the usage of statistical techniques to extract meaningful information from time series under investigation (Chatfield, 2016).

In our work, we utilize an enhanced word embedding approach, namely, FastText (Bojanowski et al., 2017), to train vector models for Google Books Ngram for decades between 1800 and 1990. The trained word vectors are inputed to construct and analyze a time series using a technique mentioned in (Kulkarni et al., 2015), for the time period between 1900 and 1990, to identify statistically significant changed words. Finally, we evaluate our models by comparing our approach against the Skip-Gram with Negative Sampling (SGNS) model of Word2Vec used in (Hamilton et al., 2016) using multiple evaluation techniques and show that we propose an approach that is robust to morphological language variation and presence of noisy data.

The rest of the paper is organized as follows. In Section 2, we discuss some of previous work in linguistic change. An overview of the proposed ap-

El-Ebshihy, A., El-Makky, N. and Nagi, K.
Using Google Books Ngram in Detecting Linguistic Shifts over Time.
DOI: 10.5220/0007188703320339
In *Proceedings of the 10th International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management (IC3K 2018) - Volume 1: KDIR*, pages 332-339
ISBN: 978-989-758-330-8

proach is given in Section 3. In Section 4, we describe the details of the experimental setup. A quantitative evaluation of the proposed approach is presented in Section 5. A discussion on the results is given in Section 6. Finally, we conclude the paper and present direction of possible future work in Section 7.

2 RELATED WORK

Various studies are made to quantitatively measure the diachronic change in language.(Kim et al., 2014) use word2vec to obtain vector representation for Google Books Ngram fiction corpus and find the words that significantly changed between 1900 and 2009. A similar approach is used by (Kulkarni et al., 2015) to model the meaning shift of words over the last century. They present three different approaches known as; *Frequency*, *Syntactic* and *Distributional* approaches, to construct time series for words. In the *Frequency* approach, they use the log probability of the word at specified time t to construct the time series. In order to construct the time series for the *Syntactic* approach, they make use of the probability distribution for POS (Part Of Speech) tags of words at each time snapshot. Whereas in the *Distributional* approach, they train Word2Vec embeddings with Skip-Gram model and use the trained vectors in time series construction. Using time series constructed from each approach, they are able to detect statistically significant change of word semantics.

The authors of (Gulordava and Baroni, 2011) utilize the Local Mutual Information (LMI) (Evert, 2008) method to construct co-occurrence matrix of words and detect semantic change of words from 1960s and 1990s. (Hamilton et al., 2016) introduce a procedure to measure semantic change using three different word embedding approaches PPMI (Positive Point-wise Mutual Information), SVD (Singular Value Decomposition), Skip-Gram with Negative Sampling (SGNS) model of Word2vec on six historical corpora of four languages; English, French, German and Chinese. They use Google Books Ngram as source for each of English, French and German languages and the COHA (Corpus of Historical American English) (Mark, 2010) corpus as another source for English language. As a result of the analysis, they present two novel laws of semantic change known as; the *law of conformity* (the rate of semantic change is inversely proportional with word frequency) and the *law of innovation* (polysemous words have are more subjected to semantic changes).

Some other approaches based on distributional methods to calculate semantic shifts are described in (Sagi et al., 2011; Xu and Kemp, 2015).

(Wijaya and Yeniterzi, 2011) apply K-means clustering and Topics-Over-Time (TOT) model (Wang and McCallum, 2006) to detect the evolution of words by determining the movement of word from one cluster to another. Additionally, (Lau et al., 2012) apply topic modeling in word sense induction using a given target word. Novel senses are identified based on the inconsistency between two given time periods. (Liao and Cheng, 2016) present another approach for determining linguistic shift using word embedding and DBSCAN (Ester et al., 1996) with an approximate nearest neighbor method to cluster vector of words and analyze polysemy.

Other approaches use Bayesian models to develop tasks in lexical semantics and diachronic word change (Brody and Lapata, 2009; Séaghdha, 2010; Ritter et al., 2010; Frermann and Lapata, 2016). Others (Mihalcea and Nastase, 2012) use supervised learning approach and word context to examine the word change usage in three different epochs (1800, 1900, 2000).

Several researchers study the evolution of words in different languages. (Takamura et al., 2017) use Skip-Gram to build a word vector model for semantic shift in Japanese loanwords by mapping the Japanese loanwords vectors to the corresponding English vectors and measure the cosine similarity between the Japanese word and English word. (Sang, 2016) present two approaches based on relative frequency of words to discover neologisms and archaisms in German Corpus. (Kulkarni et al., 2016) extend the work in (Kulkarni et al., 2015) by proposing approach to identify regional variation of word usage.

As shown, much of work is done to formulate linguistic shift detection task using distributional models. However, the existence of new word embedding techniques (e.g. FastText), poses an interesting direction of research to develop new approaches that are robust to morphological language variation and rare senses of words.

3 APPROACH

Figure 1 shows an overall view of the processes of the proposed approach. We construct a distributional time series to detect semantic changes of words. We follow the approach in (Kulkarni et al., 2015), but we use FastText as a word embedding method. We learn word embeddings vectors for the Google Books Ngram corpus, align the embedding spaces to a joint semantic space, and then use words' displacement in this semantic space to construct a distributional time series.

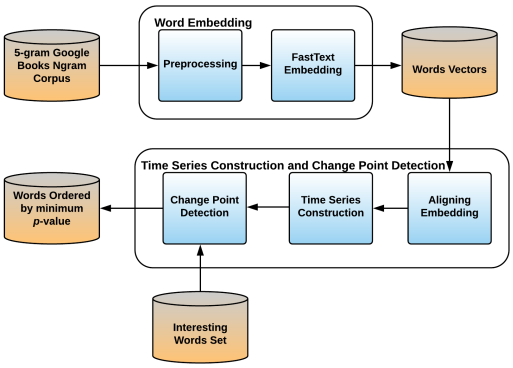

Figure 1: Overview of proposed approach.

3.1 Word Embedding

Word embeddings techniques are used to map each word to a low dimensional vector (Levy et al., 2015). The word vectors provide good representation for words semantics. Since, most of the existing approaches learn word vectors by collecting information about word context.

In order to train the embedding model, we utilize the state-of-the art FastText (Bojanowski et al., 2017) word embedding with Skip-Gram model. FastText is a word embedding method that enriches word2vec by taking into account word morphology. Morphology is modeled by considering sub-word information. This approach allows for reliable representation of rare words.

Word2vec with Skip-Gram is introduced by (Mikolov et al., 2013). It is built on the assumption that words appearing in similar context have similar meaning (Harris, 1954). In Skip-Gram model, each word in the corpus is used to predict a window of surrounding words. To optimize the trained word embeddings, stochastic gradient descent and back propagation are used. The hidden layer represents the words embedding model. Assuming, we have a training corpus given by a sequence of words $w_1, w_2, ..., w_T$ such that T is the number of words, the objective of the Skip-Gram model is to maximize the following log-likelihood:

$$\frac{1}{T}\sum_{t=1}^{T}\sum_{c\in\mathcal{C}_t}\log p(w_c|w_t) \quad (1)$$

where $\mathcal{C}_t$ is the set of indices of context words surrounding w_t.

FastText is similar to word2vec except that FastText makes use of character N-grams of variable length to enrich word vectors with sub-word information. Each word is represented by the sum of the vector representation of its N-grams. Thus, we obtain the following scoring function:

$$s(w,c) = \sum_{g\in\mathcal{G}_w} z_g^T v_c \quad (2)$$

where $\mathcal{G}_w$ is the set of N-grams that appear in the word w and z_g is the vector representation of N-gram g (Bojanowski et al., 2017).

There are several advantages of this structure that are demonstrated by the authors of FastText (Bojanowski et al., 2017):

- Using sub-word information makes the words representation robust to morphological variations in language. Unlike other models that take the tokens as words and ignore the internal structure of the words.
- Its ability to handle rare words by generating reliable embeddings for unseen words in the training data from the sum of the vectors of the word character N-grams.
- It is proved that it is superior in syntactic tasks. Thus, the syntactic structure can be identified by the bag of N-grams without depending on using words in similar context.

3.2 Time Series Construction and Change Point Detection

Using the trained word vectors, we construct time series for words using the approach mentioned in (Kulkarni et al., 2015). The process for constructing and analysis time series is summarized in the following steps:

- Aligning Embedding.
- Time Series Construction.
- Change of Point Detection.

3.2.1 Aligning Embedding

First, word vectors are aligned with respect to the final snapshot (last year) in order to map vectors to the same space. Using a piecewise linear regression model, a linear transformation is learned to map a word from the embedding space ϕ_t to ϕ_n, through minimizing the following function.

$$\underset{t\mapsto n}{\mathbf{W}}(w) = \sum_{w_i\in k-NN(\phi_t(w))} \|\phi_t(w_i)\mathbf{W} - \phi_n(w_i)\|_2^2 \quad (3)$$

where $k-NN(\phi_t(w))$ is the set of k nearest words of the word w in the embedding space $\phi_t(w)$, $\phi_t(w)$ and $\phi_n(w)$ is the embedding space of word w at time t and the final snapshot respectively (Kulkarni et al., 2015).

3.2.2 Time Series Construction

An assumption is set that a word may be subjected to linguistic shift, if the alignment model failed to align a word. Then, the displacement can be calculated between the initial time point and each point to construct the distributional time series, as follows:

$$\mathcal{T}_t(w) = 1 - \frac{(\phi_t(w)\mathbf{W}_{t\mapsto n}(w))^T(\phi_0(w)\mathbf{W}_{0\mapsto n}(w))}{\|\phi_t(w)\mathbf{W}_{t\mapsto n}(w)\|_2\|\phi_0(w)\mathbf{W}_{0\mapsto n}(w)\|_2} \quad (4)$$

where $\phi_0(w)$, $\phi_n(w)$ and $\phi_t(w)$ are the embedding space of a word w for the initial time, the final snapshot and at time t respectively. $\mathbf{W}_{0\mapsto n}(w)$ and $\mathbf{W}_{t\mapsto n}(w)$ are the linear transformation that maps a word w from $\phi_0(w)$ to $\phi_n(w)$ and from $\phi_t(w)$ to $\phi_n(w)$ respectively.

3.2.3 Change Point Detection

By normalizing the time series, the *Mean Shift* algorithm (Taylor, 2000) and bootstrapping (Efron and Tibshirani, 1994) are used to estimate the change point, in case the word is determined to have a statistically significant change compared to other words in the corpus. The procedure to analyze and estimate the statistically significant changed points is described below.

First the time series $\mathcal{T}(w)$ is normalized to generate the Z-score time series using:

$$\mathcal{Z}_i(w) = \frac{\mathcal{T}_i(w) - \mu_i}{\sqrt{Var_i}} \quad (5)$$

where, at time snapshot i, $\mathcal{Z}_i(w)$ is the Z-score time series for the word w, μ_i and Var_i are the mean and variance across all words respectively.

Then, a mean shift series $\mathcal{K}(Z(w))$ is computed using mean shift transformation on $Z(w)$. Bootstrapping is, then, applied on the normalized time series $Z(w)$ to permute it with B bootstrap samples. Means shift is applied again to produce $\mathcal{K}(\mathcal{P})$ for each bootstrap sample $\mathcal{P}$.

By setting the null hypothesis, that there is no change in the mean, the p-value at time point i is calculated by comparing the mean shift $\mathcal{K}_i(\mathcal{P})$ and $\mathcal{K}_i(Z(w))$. And, the change point is set to the time point j with the minimum p-value score.

Finally, the words that have significantly changed with respect to other words, are determined by observing the magnitude of the difference in the Z-score that exceeds a pre-defined user threshold.

For more information about the algorithm for detecting statistically significant change point, please refer to (Kulkarni et al., 2015).

4 EXPERIMENTAL SETUP

4.1 Dataset

We use Google Books Ngrams (Lin et al., 2012) English corpus to train FastText models. The Google Books Ngrams corpus is a huge dataset formed of N-grams that are extracted from about 8 million books over five centuries. The N-grams differ in size (1-5) grams. We use 5-grams from the English dataset.

4.2 Pre-processing

For building the model, we follow the same pre-processing procedure used in (Hamilton et al., 2016). We lower-case words and remove punctuation. We restrict the vocabulary to words that occur at least 500 times spanning the time period 1800-2000. Also, we downsample the larger years (i.e. starting from 1870) to have at most 10^9 tokens as recommended by (Hamilton et al., 2016).

4.3 Parameter Setting

4.3.1 Word Embedding Hyperparameters

We construct the vector representation for the decades from 1800 to 1990 using FastText [1]. For the sake of fair comparison, we use a symmetric context window of size 4 and dimensionality of 300 for the word vectors (the same as (Hamilton et al., 2016)). In order to set other hyperparameters, we use grid search to get the best set of parameters to train FastText vectors.

We set the character n-gram minimum and maximum values to 2 and 6 respectively, the learning rate to 0.1, the epoch size to 3 and negative sampling (ns) loss. The rest of the hyperparameters are set to their default values.

4.3.2 Parameters of the Time Series Construction Module

The input to the time series construction module is

- The word vector models for each time snapshot.
- The set of words of interest that should be tracked.

We choose the set of words as the top-10000 words ordered by their average frequency over the entire time period, excluding stop words and proper nouns [2]. We use the same parameters as in (Kulkarni et al., 2015). We set the bootstrap value $B = 1000$ and Z-Score threshold γ to 1.75.

[1] https://github.com/facebookresearch/fastText

[2] For the sake of fair comparison, we choose the same set words of interest as (Hamilton et al., 2016)

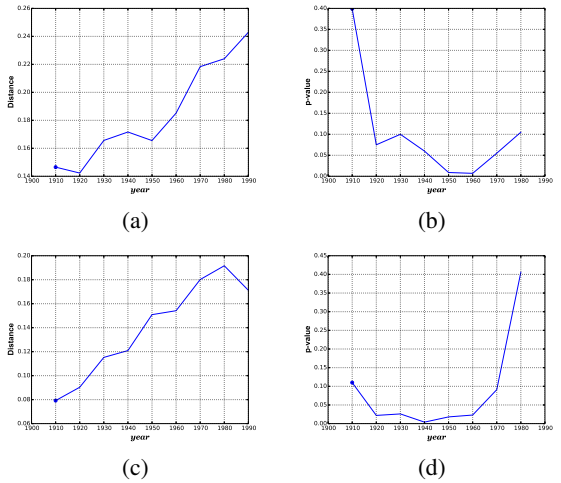

Figure 2: Time series and p-value of two examples that are detected to be statistically significant changed by using FastText word vectors and by the distributional method in (Kulkarni et al., 2015): (a) and (b) time series and p-value for the word gay and, (c) and (d) similar plots for the word plastic.

4.3.3 Time Series Analysis

We analyze the performance of the time series that is constructed from FastText word vectors using the words that are detected to be statistically significant changed in (Kulkarni et al., 2015) by their *Distributional* method.

Figure 2, shows time series constructed for two examples of these words *gay* and *plastic*, using FastText word vectors and their corresponding *p*-value. A dip in the *p*-value represents an indication of a statistically significant change in the word usage. Figure 2 ((a), (b)) shows that the word *gay* underwent a statistically significant semantic change. It began to move away from the words: *happy*, *cheerful* and *showy* around 1920, similar to the results in (Hamilton et al., 2016). On the contrary, it starts to be similar to the words: *homosexual*, *bisexual* and *lesbian* since 1960. Similar results can be obtained with the word *plastic*, where it starts to gain shift in meaning with the introduction of *Polystyrene* around 1950 (Kulkarni et al., 2015). Before that time, *plastic* was used to give the meaning of the *flexibility* physical property.

5 EVALUATION

The lack of gold standard data poses challenges on the evaluation of our approach. Therefore, we use some quantitative approaches in (Kulkarni et al., 2015; Hamilton et al., 2016), to compare the performance of our approach to the SGNS approach in (Hamilton et al., 2016). we compare the two approaches by evaluating their *synchronic* accuracy (i.e., ability to capture word similarity within individual time-periods) and their *diachronic* validity (i.e., ability to quantify semantic changes over time). We also use the reference data set in (Kulkarni et al., 2015) to evaluate the performance of time series constructed for FastText and SGNS approaches.

5.1 Synchronic Accuracy

To asses the synchronic accuracy, we use the standard modern benchmark MEN (Bruni et al., 2012) and the 1990s word vectors. We compute the Spearman correlation coefficient (Spearman, 1904) between human judgment of word similarity and cosine similarity between pairs of words.

As shown in Table 1, FastText models outperform the SGNS model, even if we don't generate vectors for out-of-vocabulary (OOV) words (i.e words that appear in the testing set but not the vocabulary that is used to train the embedding model). Since FastText exploits sub-word information, it can generate vectors for these words, which leads to further improvement in performance. This demonstrate the claim of the authors (Bojanowski et al., 2017), that adding sub-word information, improves the ability to capture word similarity.

Table 1: Synchronic accuracy results of SGNS (Hamilton et al., 2016) against results of using FastText without generating vectors for OOV words (FastText_OOV) and after generating vectors for OOV words (FastText).

Approach	Spearman Correlation (ρ)	OOV
SGNS Results (Hamilton et al., 2016)	0.649	54
FastText_OOV	0.73	54
FastText	0.741	0

5.2 Diachronic Validity

In order to measure the diachronic validity of our model, we detect known shifts using the method proposed in (Hamilton et al., 2016). In this task, we want to detect whether the approach can identify if a pair of words move closer or apart from each other in semantic space during a pre-determined time-period.

Using the set of examples (28 known historical shifts) shown in Table 2, we check if the pairwise similarity series have the correct sign on their Spearman correlation with time. Then we determine whether the shift is statistically significant at $p < 0.05$ level. From the results in Table 3, FastText is able to detect the

Table 2: Set of known historical shifts used to evaluate the diachronic validity (Hamilton et al., 2016).

Word	Moving towards	Moving away	Shift start	Source
gay	homosexual, lesbian	happy, showy	ca 1920	(Kulkarni et al., 2015)
fatal	illness, lethal	fate, inevitable	$<$ 1800	(Jatowt and Duh, 2014)
awful	disgusting, mess	impressive, majestic	$<$ 1800	(Simpson and Weiner, 1989)
nice	pleasant, lovely	refined, dainty	ca 1900	(Wijaya and Yeniterzi, 2011)
broadcast	transmit, radio	scatter, seed	ca 1920	(Jeffers and Lehiste, 1979)
monitor	display, screen	–	ca 1930	(Simpson and Weiner, 1989)
record	tape, album	–	ca 1920	(Kulkarni et al., 2015)
guy	fellow, man	–	ca 1850	(Wijaya and Yeniterzi, 2011)
call	phone, message	–	ca 1890	(Simpson and Weiner, 1989)

correct direction of shifts in all cases except for one case **(awful, majestic)**, while its performance was the same as SGNS (Hamilton et al., 2016) for measuring the significance level of shift.

Table 3: Diachronic validity performance, detecting attested shifts from Table 2, of SGNS (Hamilton et al., 2016) against results of our approach using FastText.

Approach	% Correct	% Sig.
SGNS Results (Hamilton et al., 2016)	100.0	93.8
FastText	96.4	93.8

5.3 Evaluation on a Reference Dataset

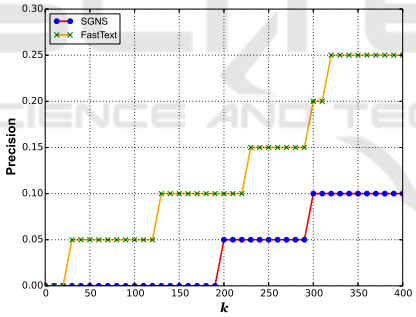

Figure 3: Performance of FastText vs SGNS on a reference dataset.

Using this method, we attempt to quantify the performance of the FastText and SGNS approaches on a reference dataset. We use the evaluation method illustrated in (Kulkarni et al., 2015). We use a data set D of 20 words, that are known as having undergone linguistic shift, collected by (Kulkarni et al., 2015) from various sources (Gulordava and Baroni, 2011; Jatowt and Duh, 2014; Kim et al., 2014; Wijaya and Yeniterzi, 2011). Then for each approach, we make a list L of words ordered by the significance score of change. After that, we calculate Precision@k (Manning et al., 2008) between L and D as follows:

$$Precision@k(L,D) = \frac{|L[1:k] \cap D|}{|D|} \qquad (6)$$

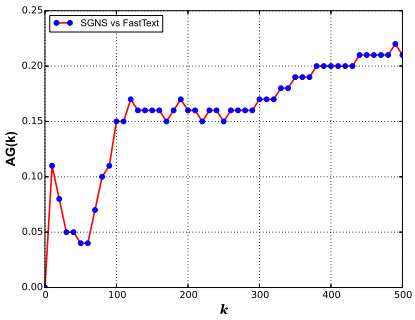

Figure 4: Method agreement of changed words between FastText and SGNS.

Figure 3 shows the performance of both approaches against the reference dataset. We can notice that as k increases, the number of relevant retrieved words increases. It is clear that FastText outperforms SGNS.

5.4 Method Agreement

To explore the agreement between FastText and SGNS approaches, we use the method suggested by (Kulkarni et al., 2015). We consider the top k words that each approach claims to be subject to linguistic shift ordered by their significance scores. Consider that we have two lists $M_F(k)$ and $M_S(k)$, we compute the agreement between the two lists, using Jaccard Similarity, as follows:

$$AG(M_F(k), M_S(k)) = \frac{|M_F(k) \cap M_S(K)|}{|M_F(k) \cup M_S(K)|} \qquad (7)$$

where $M_F(k)$ and $M_S(k)$ represent the top k word lists generated from FastText and SGNS (Hamilton et al., 2016) respectively.

Figure 4 depicts the agreement scores between both approaches for different k values. We observe that the agreement between FastText and SGNS is relatively low, which means that each approach captures different linguistic aspects. This implies that if we use a combination of both approaches, it may yield to improved results.

6 DISCUSSION

In this section, we discuss the evaluation results in Section 5. FastText outperforms SGNS in most of the cases. This is due to the following:

- Optimizing the size of N-grams, in FastText parameters, improves the semantic tasks. This agrees with the claim of (Bojanowski et al., 2017).
- As shown in the evaluation results in Section 5.1, the robustness of FastText in the presence of rare words is due to the usage of sub-word information, especially in the presence of noisy text, unlike SGNS and other embedding approaches that deal with the word as a whole entity. This agrees with the findings in (Bojanowski et al., 2017) that FastText performed similarly to SGNS in semantic tasks when using test sets with common words in English. It outperforms SGNS when using sets with some less frequent words or unseen words, which is the case with the standard modern benchmark MEN (Bruni et al., 2012) used in synchronic accuracy evaluation.
- From the findings of (Hamilton et al., 2016), one can infer that rare words semantically change at a faster rate. The ability of FastText to capture rare words helps in detecting their semantic changes.
- Exploiting sub-word information helps in overcoming the presence of messy text in large datasets such as the Google Books Ngram dataset.

It is also observed, in Section 5.4, that the agreement between FastText and SGNS is low. This can be discussed as follows:

- While the semantic information can be encoded in SGNS embedding, FastText incorporates morphological information. As a result, FastText models can extract syntactic variation as well. It has been shown by the results in (Bojanowski et al., 2017) that FastText is superior in syntactic tasks.
- The usage of character N-grams, makes FastText able to capture semantic relation between words that share common N-grams (e.g. the words behavior and behavioral) (Tissier et al., 2017).

7 CONCLUSION AND FUTURE WORK

7.1 Conclusion

In this paper, we propose an approach to computationally detect word meaning shift across time. We build our system based on the work done by (Hamilton et al., 2016) and (Kulkarni et al., 2015). We use the state-of-the-art word embedding approach, FastText, to build word representation. We then use the trained word vectors to construct and analyze time series in order to identify words that undergone statistically significant change of meaning in addition to the detection of the point of change.

We apply our proposed approach on a large historical corpus, Google Books Ngram. We use different evaluation strategies to compare the performance of our approach against its counterpart (Hamilton et al., 2016).

7.2 Future Work

As a future work, we will focus on using more quantitative evaluation methods to analyze the system performance in linguistic change task. We will also work on the creation of an improved approach to capture syntactic shifts beside semantic changes as suggested by (Kulkarni et al., 2015).

One interesting direction of research, is to detect whether the same words evolve similarly in different languages.

We can apply our approach on morphological rich languages (e.g., Arabic, Czech and German languages). This is due to the robustness of FastText to such types of languages.

REFERENCES

Bojanowski, P., Grave, E., Joulin, A., and Mikolov, T. (2017). Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, 5:135–146.

Brody, S. and Lapata, M. (2009). Bayesian word sense induction. In *Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics*, pages 103–111. Association for Computational Linguistics.

Bruni, E., Boleda, G., Baroni, M., and Tran, N.-K. (2012). Distributional semantics in technicolor. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Long Papers-Volume 1*, pages 136–145. Association for Computational Linguistics.

Chatfield, C. (2016). *The analysis of time series: an introduction*. CRC press.

Efron, B. and Tibshirani, R. J. (1994). *An introduction to the bootstrap*. CRC press.

Ester, M., Kriegel, H.-P., Sander, J., and Xu, X. (1996). A density-based algorithm for discovering clusters a density-based algorithm for discovering clusters in large spatial databases with noise. In *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining*, KDD'96, pages 226–231. AAAI Press.

Evert, S. (2008). Corpora and collocations. *Corpus linguistics. An international handbook*, 2:1212–1248.

Frermann, L. and Lapata, M. (2016). A bayesian model of diachronic meaning change. *Transactions of the Association for Computational Linguistics*, 4:31–45.

Gulordava, K. and Baroni, M. (2011). 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. Association for Computational Linguistics.

Hamilton, W. L., Leskovec, J., and Jurafsky, D. (2016). Diachronic word embeddings reveal statistical laws of semantic change. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Association for Computational Linguistics.

Harris, Z. S. (1954). Distributional structure. *Word*, 10(2-3):146–162.

Jatowt, A. and Duh, K. (2014). A framework for analyzing semantic change of words across time. In *Digital Libraries (JCDL), 2014 IEEE/ACM Joint Conference on*, pages 229–238. IEEE.

Jeffers, R. J. and Lehiste, I. (1979). *Principles and methods for historical linguistics*. MIT press.

Kim, Y., Chiu, Y.-I., Hanaki, K., Hegde, D., and Petrov, S. (2014). Temporal analysis of language through neural language models. *arXiv preprint arXiv:1405.3515*.

Kulkarni, V., Al-Rfou, R., Perozzi, B., and Skiena, S. (2015). Statistically significant detection of linguistic change. In *Proceedings of the 24th International Conference on World Wide Web*, pages 625–635. International World Wide Web Conferences Steering Committee.

Kulkarni, V., Perozzi, B., and Skiena, S. (2016). Freshman or fresher? quantifying the geographic variation of language in online social media. In *ICWSM*, pages 615–618.

Lau, J. H., Cook, P., McCarthy, D., Newman, D., and Baldwin, T. (2012). Word sense induction for novel sense detection. In *Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics*, pages 591–601. Association for Computational Linguistics.

Lehmann, W. P. (2013). *Historical linguistics: an introduction*. Routledge.

Levy, O., Goldberg, Y., and Dagan, I. (2015). Improving distributional similarity with lessons learned from word embeddings. *Transactions of the Association for Computational Linguistics*, 3:211–225.

Liao, X. and Cheng, G. (2016). Analysing the semantic change based on word embedding. In *International Conference on Computer Processing of Oriental Languages*, pages 213–223. Springer.

Lin, Y., Michel, J.-B., Aiden, E. L., Orwant, J., Brockman, W., and Petrov, S. (2012). Syntactic annotations for the google books ngram corpus. In *Proceedings of the ACL 2012 system demonstrations*, pages 169–174. Association for Computational Linguistics.

Manning, C. D., Raghavan, P., and Schütze, H. (2008). *Evaluation in information retrieval*, page 139–161. Cambridge University Press.

Mark, D. (2010). The corpus of historical american english: 400 million words, 1810–2009. *http://corpus.byu.edu/coha*.

Mihalcea, R. and Nastase, V. (2012). Word epoch disambiguation: Finding how words change over time. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Short Papers-Volume 2*, pages 259–263. Association for Computational Linguistics.

Mikolov, T., Chen, K., Corrado, G., and Dean, J. (2013). Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

Ritter, A., Etzioni, O., et al. (2010). A latent dirichlet allocation method for selectional preferences. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 424–434. Association for Computational Linguistics.

Sagi, E., Kaufmann, S., and Clark, B. (2011). Tracing semantic change with latent semantic analysis. *Current methods in historical semantics*, pages 161–183.

Sang, E. T. K. (2016). Finding rising and falling words. *LT4DH 2016*, page 2.

Séaghdha, D. O. (2010). Latent variable models of selectional preference. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 435–444. Association for Computational Linguistics.

Simpson, J. A. and Weiner, E. S. C. (1989). *The Oxford English dictionary*, volume 2. Clarendon Press Oxford.

Spearman, C. (1904). The proof and measurement of association between two things. *The American journal of psychology*, 15(1):72–101.

Takamura, H., Nagata, R., and Kawasaki, Y. (2017). Analyzing semantic change in japanese loanwords. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, volume 1, pages 1195–1204.

Taylor, W. A. (2000). Change-point analysis: a powerful new tool for detecting changes.

Tissier, J., Gravier, C., and Habrard, A. (2017). Dict2vec: Learning word embeddings using lexical dictionaries. In *Conference on Empirical Methods in Natural Language Processing (EMNLP 2017)*, pages 254–263.

Wang, X. and McCallum, A. (2006). Topics over time: a non-markov continuous-time model of topical trends. In *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 424–433. ACM.

Wijaya, D. T. and Yeniterzi, R. (2011). Understanding semantic change of words over centuries. In *Proceedings of the 2011 international workshop on DETecting and Exploiting Cultural diversiTy on the social web*, pages 35–40. ACM.

Xu, Y. and Kemp, C. (2015). A computational evaluation of two laws of semantic change. In *CogSci*.