Semantic Change Detection With Gaussian Word Embeddings

Arda Yüksel, Berke Uğurlu, and Aykut Koç, *Senior Member, IEEE*

***Abstract*—Diachronic study of the evolution of languages is of importance in natural language processing (NLP). Recent years have witnessed a surge of computational approaches for the detection and characterization of lexical semantic change (LSC) due to the availability of diachronic corpora and advancing word representation techniques. We propose a Gaussian word embedding (w2g)-based method and present a comprehensive study for the LSC detection. W2g is a probabilistic distribution-based word embedding model and represents words as Gaussian mixture models using covariance information along with the existing mean (word vector). We also extensively study several aspects of w2g-based LSC detection under the SemEval-2020 Task 1 evaluation framework as well as using Google N-gram corpus. In the Sub-task 1 (LSC binary classification) of the SemEval-2020 Task 1, we report the highest overall ranking as well as the highest ranks for the two (German and Swedish) of the four languages (English, Swedish, German and Latin). We also report the highest Spearman correlation in the Sub-task 2 (LSC ranking) for Swedish. Our overall rankings in the LSC classification and ranking sub-tasks are 1^{st} and 7^{th}, respectively. Qualitative analysis has also been presented.**

***Index Terms*—Diachronic embeddings, semantic change computation, semantic change detection, lexical semantic change, diachronic NLP, word embeddings, word2gauss.**

I. Introduction

LANGUAGES evolve with time since cultural and linguistic effects alter meanings of words in a semantic space [1]. Diachronic study of semantic change explores changes in word meanings. Advancements in diachronic corpus compiling and computational technologies allow natural language processing (NLP) based approaches to gain importance in this field [2]–[10]. Also, the knowledge created by studies in computational linguistics to understand languages makes NLP applications better performing [3], [5], [6], [9], [11]. Performance improvements are obtained for query systems and information retrieval, [6], social computing, [11], and culturomics, [3].

Semantic change comes in fast- or slow-paced natures. A contemporary example of fast changes is the word `corona` after the 2020 Worldwide Covid pandemic. Once stood for a circle or disc in astronomy, `corona`'s dominant sense has oriented towards its disease-related sense in the perspective of society. Identifying and categorizing these alterations can allow language models to detect current cultural impacts and trends [3], [11], and words with multiple meanings can be represented more accurately by diachronic knowledge, [12]. Computational methods to detect semantic change can help in information retrieval, question&answering applications and NLP-based historical studies, [8]. Additionally, the recent developments in internet technologies and increase in social media usage have accelerated the change of language, [13], stressing the importance of studying semantic change to develop better NLP algorithms.

Theoretical foundations of semantic change have been provided by [1], [14], [15] from a linguistic perspective. A type of categorizing semantic change depends on the size of the time span between two corpora used to train computational semantic models. The granularity of time span is crucial in identifying the difference between *socio-cultural* (fast-paced) and *linguistically-motivated* (slow-paced) semantic shifts, [3], [9], [16], [17]. In addition to this categorization, [9] analyzed differences between cultural and linguistic causes. [14] proposed two important types of semantic change, namely *semantic broadening* and *semantic narrowing*. While semantic broadening (also called widening) indicates that the word's meaning expanded to cover a more generic meaning, semantic narrowing is the opposite. Notable examples are observed in the changes from Old English to New English forms, [2]. The word `dog` once corresponded to merely a specific breed of dogs where `docga` in Old English stood for dogs in general. Thus, `dog` has been semantically broadened and `docga` has stopped being used. A thorough treatment and coverage of several aspects of semantic change can be found in [12], [18], [19] and the references therein.

Lexical semantic change (LSC) detection (or semantic change computation) has been a widely popular topic with the developments of word representation techniques, [20]–[22] and availability of large historical datasets such as the Corpus of Historical American English (COHA) [23], Google N-gram [3], [24], the Helsinki Corpus [25], Twitter data [17], and news datasets, [26]. Word embeddings have enabled the deployment of computational approaches in LSC detection and in diachronic studies in general [27]. Tracking the changes of word embeddings across different time periods opens a way to evaluate semantic change qualitatively and quantitatively.

Following the interest in LSC detection, several comprehensive survey papers addressing semantic change

Manuscript received May 14, 2021; revised August 16, 2021; accepted September 10, 2021. Date of publication October 20, 2021; date of current version November 6, 2021. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Jing Huang. *(Corresponding author: Aykut Koç.)*

The authors are with the Department of Electrical and Electronics Engineering, Bilkent University, Ankara 06800, Turkey, and also with UMRAM, Bilkent University, Ankara 06800, Turkey (e-mail: rd.yuksel07@gmail.com; brkugrl96@gmail.com; aykut.koc@bilkent.edu.tr).

This article has supplementary downloadable material available at https://doi.org/10.1109/TASLP.2021.3120645, provided by the authors.

Digital Object Identifier 10.1109/TASLP.2021.3120645

and its applications to high-level NLP tasks have recently been emerged, [12], [18], [19]. These surveys highlight the importance of LSC detection as it contributes to other sub-fields in NLP and present the topic from both NLP-based and computational linguistic-based viewpoints.

For the quantitative diachronic study, it is apparent that having ground truth results or human-annotated data is essential to evaluate LSC detection performances. In the SemEval-2020 Task 1 Challenge (SemEval) [27], annotated corpora in various languages and standardized evaluation methodologies have been published. SemEval has become the main dataset and evaluation framework used in the field and it provides two sub-tasks for LSC detection. Sub-task 1 is the LSC binary classification (LSC-binary) and Sub-task 2 is the LSC ranking (LSC-ranking), both with annotated ground truth data. In the LSC-binary, the aim is to decide whether words in a target word list are semantically changed or not. For the LSC-ranking, one needs to quantify levels of semantic change and rank words accordingly. Additionally, Google N-gram Corpus is heavily utilized due to its capacity of representing various periods, despite not being annotated. The mainline of diachronic research in NLP focuses on detection of semantic change [2], [9], [16], [17], [28], [29].

In LSC detection, methodologies need to diachronically keep track of words through their representations in semantic spaces by using word embeddings to make deductions, [20], [21], [30]. One of the contemporary word embedding techniques represents words with not only embedding vectors corresponding to single points in the semantic space but with probability distributions around these context vectors. To this end, [31] proposed the Gaussian word embedding model (*word2Gauss* or shortly *w2g*) in which a word is represented as a multi-dimensional Gaussian distribution with a mean vector and a covariance matrix around the mean. To deal with polysemy, [32] extended w2g and proposed *word2Gaussian mixture* model (w2gm) where they represented words with a mixture of Gaussians, each corresponding to different senses of a word. In w2g models, the mean resembles the vector output of conventional vector-based word embeddings. Variance, on the other hand, stands for the size of semantic specificity/generality of a particular sense (like a generic `animal` and specific `dog`). Variance of w2g embeddings can also be interpreted as the ambiguity or uncertainty of word meaning as well as its semantic coverage. Variances hold quantitative values correlated with the semantic breadth of a sense or meaning within semantic space. Thus, variance can provide crucial information regarding semantic change since it occurs not only as a drift of meaning within semantic space but also as a narrowing or broadening of semantic coverage.

The idea of using w2g in LSC detection is present in SemEval [33]. As part of their work, [33] used mean vectors of w2g embeddings in an attempt to classify and rank semantic changes. However, instead of deploying variances from w2g, [33] directly uses the normalized frequencies of words. Therefore, despite being inspired by w2g, [33]'s model utilizes only mean vectors that are equivalent to regular embeddings. Moreover, results of the SemEval indicate that [33] underperformed with ranks of 21^{st} and 19^{th} among 21 participants for LSC-binary and LSC-ranking, respectively [27].

In this manuscript, we present a comprehensive study and propose a methodology that deploys w2g to study semantic change both in the context of the SemEval-2020 Task 1 Challenge and beyond. Our contributions include the successful demonstration of a w2g-based LSC detection method that uses w2g's in their full-form (both mean and variance). We report three language-specific highest scores (two in LSC-binary and one in LSC-ranking) and the overall highest score for LSC-binary of the SemEval-2020 Task 1 Challenge. In both sub-tasks, our proposed methodology and models extract the potential of w2g models so that we can reach better rankings. Beyond working on the SemEval, we also study LSC detection based on w2g using the Google Books N-gram corpora. To this end, we remodel the architecture of w2g, which is originally designed for accepting only words as inputs, and proposed a model accepting n-gram training. We exhaustively study several alternatives for stages in our proposed methodology and provide an in-depth treatment of w2g-based LSC detection. Both quantitative and qualitative analyses are presented.

The rest of the manuscript is organized as follows. In Section II, related work is given. The datasets and corpora are presented in Section III. Section IV presents details of our proposed methods. Experimental procedures and results are given in Section V. Finally, we conclude in Section VI.

II. Related Work

Our scope needs treatment of two distinct topics. First, previous work on semantic change is discussed. The second part explores related work on word embeddings with a focus on probability distribution-based approaches.

A. Semantic Change

The reasons of semantic change and its categories are extensively studied in linguistics [1], [12], [14], [15], [18], [19]. [15] cited socio-cultural, linguistic, and psychological causes. According to [34], semantic change is due to the alterations in collocational patterns. Before the emergence of NLP-based computational approaches, the task of classifying semantic change was initiated in linguistics by [14]. A categorization where the extension of a sense is identified as widening (or broadening) and the decrease is named as narrowing was also proposed [14]. The scheme devised in [14] is referred to as a guideline for semantic change categorization. [35] defined three more categories: *word sense evolution* which corresponds to gain or loss of meaning of an existing word, *term-to-term evolution* which corresponds to the creation of a new synonymous word that can co-exist with the original, and *emergence of new terms* which corresponds to words describing newly-emergent concepts. Semantic change categorization in the context of semantic broadening/narrowing was studied in [2]. [17] proposed another categorization focusing on sense *splits*, *births*, *joints* and *deaths*. *Metaphorical*, *metonymical* and *novel sense* changes are also present [7].

Without large and annotated datasets for ground truth, computational studies are limited to only qualitative measures, [36].

To remedy this, efforts have been consolidated under the recent SemEval-2020 Task 1 Unsupervised LSC detection challenge, [27], where researchers assembled annotated corpora for four different languages (English, German, Swedish and Latin) and provided a standardized evaluation framework. The most notable study using human metrics before the SemEval-2020 Task 1 was performed in [16]. They examined the sense shifts in the 1960 s and 1990 s. In qualitative studies, Google N-gram corpus [3], [24] is also frequently used. Task-oriented corpora such as DURel [37], SemCor LSC [38], and WSC [39] are also used in several studies [40], [41].

Word embedding models allowed researchers to computationally analyze semantic change. One of the earliest works used the Latent Semantic Analysis (LSA) and categorized semantic changes on diverse corpora, [2]. Deployment of dense word embeddings such as word2vec, [21], and GLoVe, [22], are prominent, [42], [43]. Using word2vec, [9] developed the LSC detection framework and proposed methods to compare separately-trained embedding models. Contemporary studies also integrate BERT and other transformer architectures ([30], [44]) to LSC studies [45], [46]. In [45], BERT language model is used to evaluate semantic broadening/narrowing, where three metrics (entropy difference, Jensen-Shannon divergence and average pairwise difference) are used to categorize semantic change. An example of deploying w2g-based embeddings in a LSC detection model is also present [33] where only "word2vec-like" mean vectors are combined with normalized word frequencies. To computationally comprehend language evolution, other researchers focused on diachronic studies [17], [47]–[49] by developing different methodologies as well as trying to represent words more accurately. [49] proposed a graph-based method to determine embedding vectors by using the linear combination of their neighbors in previous periods.

When embeddings are trained independently using corpora from different time periods, due to random initializations, each word's vector in a particular time period is different than those in other periods regardless of semantic change. Thus, a linear transformation between embedding spaces is necessary to make comparisons. Being the common alignment method, the orthogonal Procrustes (OPM) [50] is used in several studies, [9], [11], [46], [51]–[56]. Other notable approaches can be listed as second-order embeddings [9], [43], [49], graph-based architectures [57], canonical correlation analysis [53], temporal referencing [56], [58]–[60] and vector initialization alignment (VIA) [33], [42], [47], [61], [62]. In [61], a method to improve [47] is proposed using incremental updating for the Skip-gram and CBOW. The above approaches are also visible in the SemEval-2020 Task 1 [27] where alignment-based methods are prominent and hold higher ranks, especially in the first sub-task, [46], [52], [53].

Cultural effects on language are also analyzed by using embeddings [9], [11], [47]. [9] analyzed the relation between polysemy and word frequencies across decades and derived two empirical laws of semantic change: the Law of Conformity and the Law of Innovation. The former states that frequent words tend to protect their positions within semantic space (i.e. tend to resist semantic changes) while the latter states that polysemous words tend to experience semantic changes. These empirical laws are utilized to identify the target and control words by [43].

Classifying semantic change requires the crucial stage of designing unsupervised threshold selection algorithms to convert continuous scores to binary decisions. Target word-based mean thresholds are proposed [43], [46], [53], where thresholds are designed as the means of distances of target words for each language separately (language-specific) or as the overall mean of the combined set of multilingual target words (cross-language). Probabilistic algorithms are applied in [56].

B. Gaussian Word Embeddings

Computational LSC detection relies on frameworks representing words quantitatively. Dense word embeddings such as word2vec and GloVe represent each word as a single word vector (or point) within a semantic space [21], [22]. Contemporary techniques favor attention and transformer architectures [63] to form word vectors such as complex neural network-based models BERT [30] and ELMo [44].

Although point embeddings have proven to be very successful in mainstream applications, they treat all words as points irrespective of the extent of their meanings and their semantic coverage. A new line of word embeddings has emerged to improve semantic modelling of words. Constructing a structured semantic space where words with different semantic coverage are represented not only with points but also with varying regions around points is originally proposed by [64]. An interesting notion that represents words with Gaussian probability distributions (w2g) rather than points is proposed by [31]. Unlike their regular counterparts, w2g embeddings carry two quantities assigned to words. The first one is the mean vector of the distribution, which is analogous to the generic word2vec [21]. The additional quantity standing for the ambiguity or semantic coverage of words is the introduction of the covariance matrix associated with the distribution. W2g's can also identify properties related to entailment relations of words, [31]. Entailment can be taken as a quantifier of the hierarchy between words such as `dog` $\models$ `animal` which implies `dog` *is an* animal, [65]. As expected, more ambiguous senses can hold greater information and thus can be taken as more generic senses with larger variances.

One of the issues of [31]'s implementation was the unwantedly increasing variance for polysemous words. Words with multiple senses can yield inevitably greater variances even though their senses might not singlehandedly carry significant semantic coverage. Certainly, addressing problems due to polysemy in word embeddings precede w2g models with the well-known contextualized word embeddings. Also, [66] and [67] can be given as examples addressing the effects of polysemy on word embeddings in general. To address the case for w2g, [32] proposed the Gaussian mixture models (GMM) instead of representing words with a single Gaussian distribution. In this model, each word embedding is composed of multiple Gaussian modes with independent means and variances so that there is flexibility in modeling different senses during training. W2g's are then composed of a weighted sum of these modes.

III. Details of Corpora

We use two diachronic corpora. The first is *the Google Books N-gram Corpus* for fiction category [24], which is in the 5-gram format with date information. The second is *the SemEval-2020 Task 1 Annotated Corpus* [27].

A. Google Books N-Gram Corpus

The Google corpus is heavily used for LSC detection since it contains multiple n-grams, which are given in alphabetical order, [9], [17], [29], [47]. Along with n-gram information, *year*, *match count*, and *volume count* are also listed. These properties are explained as the following: the *year* is the date that will be used in preprocessing procedures, *match count* is the number of occurrences, and *volume count* is the number of documents that contain the particular n-gram. In general, 5-gram version of the Google corpus is used in diachronic studies. The version we used in this work is given in [24] and contains 5-grams in the fiction category.

Based on year data, we store 5-grams in decades starting from the decade 1800-1809 up to the decade 2000-2009 with 21 sub-corpora of decades. During the preparation of multiple corpora streams, all letters are lower-cased and punctuation is removed. More details of the pre-processed sub-corpora can be found in the Supplementary Materials.

B. SemEval-2020 Task 1 Annotated Corpus

Unlike the Google corpus, the SemEval corpus is annotated and provides ground truth data with identified semantic changes. Thus, it provides a quantitative and controlled test environment to benchmark different methods. The SemEval corpus contains four languages: English, Swedish, German and Latin. For each language, two sub-corpora corresponding to two relatively distant periods (different for each language) ranging from approximately 50 to 2,000 years are provided. Token sizes for each sub-corpora are almost evenly distributed. The first and second periods are denoted by C_1 and C_2, respectively. SemEval provides two sub-tasks. The LSC-binary sub-task requires binary classification of semantically changed words while one aims to rank words by their measure of semantic change for the LSC-ranking sub-task. For each language, SemEval corpus is annotated for two sub-tasks with ground truth for certain target words. These two sets of ground truth information are named *binary results* and *graded results*. Binary results contain 0 for non-changed and 1 for changed words. Graded results are real numbers ranging from 0 to 1, where 0 means no change. Since the SemEval corpus possesses ground truth information, it is the main baseline framework used in quantitative LSC detection [42], [43]. More details can be found in the Supplementary Materials.

To study the effects of stop-words, we also generated a version of the SemEval corpus by removing stop-words and obtained two versions: original and stop-word removed. In total, there are four languages with corpora from two time periods each having original and stop-word removed versions.

IV. Methodology

Our proposed methodology divides the problem of LSC detection with w2g into three stages. First stage is the adaptation and usage of w2g models proposed by [32] for training on the Google and SemEval corpora. To train on the Google corpus, one needs to modify w2g so that it is also possible to train on n-grams. Unlike prior w2g models working on textual information, the proposed w2g model can also be trained when inputs are n-grams. To this end, we used a different kernel for n-gram training that is inspired by the n-gram-word2vec implementation [68]. In the second stage of our methodology, we apply the vector alignment procedure to the embeddings trained on corpora belonging to different time periods. We utilized the OPM in this stage. The third and the final stage is the semantic change detection with similarity measurements and thresholding. The overall illustration of our methodology can be seen in Fig. 1. We provide the details of our methodology in the following subsections.

A. Word2Gauss and Word2Gauss Mixture Models

Unlike word embeddings that represent words as vectors in semantic space, w2g models represent words by two components: mean vectors with the same functionality as regular word vectors and variance values around those means. Variances are designed to quantify the uncertainty of words (and hence semantic breadth) by assigning probability masses around the mean locations [31], [32]. Geometrically, this structure is an ellipsoid where the center is designated by the mean and the contour surface is specified by the variance. For each word, a mean vector and a covariance matrix of a Gaussian distribution are learned within a semantic space.

[31] introduced w2g by proposing two energy-based learning procedures, namely the expected likelihood kernel (ELK) and the Kullback-Leibler (KL) divergence. The ELK $E(f, g)$ is a symmetric similarity function that is simply an inner product of two Gaussian distributions:

$$E(f,g) = \int f(x)g(x)dx, \tag{1}$$

where f and g denote independent Gaussian distributions. In our context, they stand for two words within semantic space that corresponds to the probability space. For a given word f, Gaussian distribution $f(x)$ corresponds to:

$$\begin{aligned} f(x) &= \mathcal{N}[x; \vec{\mu}_f, \Sigma_f] \\ &= \frac{1}{\sqrt{2\pi|\Sigma_f|}} \exp(-\frac{1}{2}(x-\vec{\mu}_f)^T \Sigma_f^{-1}(x-\vec{\mu}_f)), \end{aligned} \tag{2}$$

where $\mathcal{N}[x; \vec{\mu}_f, \Sigma_f]$ is the normal distribution with mean $\vec{\mu}_f$ and covariance matrix Σ_f, and x denotes a point within semantic space. Due to the stable distribution property, the inner product of two Gaussian distributions for words f and g can be represented as another Gaussian distribution:

$$\begin{aligned} E(f,g) &= \int \mathcal{N}[x; \vec{\mu}_f, \Sigma_f]\mathcal{N}[x; \vec{\mu}_g, \Sigma_g]dx \\ &= \mathcal{N}[0; \vec{\mu}_f - \vec{\mu}_g, \Sigma_f + \Sigma_g]. \end{aligned} \tag{3}$$

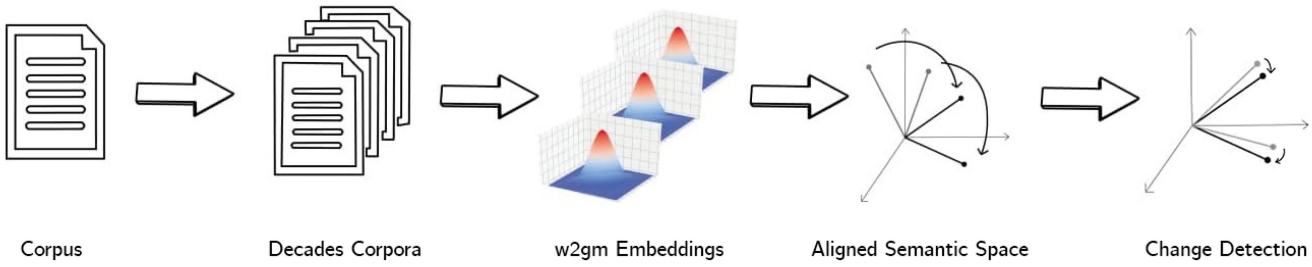


Fig. 1. Outline of the methodology for w2gm-based LSC detection.

Using the property in Eq. 3, dependence on a sample point x is negated since the origin of the semantic space replaces it.

The second energy function alternative uses the KL divergence, which is asymmetric:

$$-E(f,g) = D_{KL}(\mathcal{N}_g \,||\, \mathcal{N}_f) = \int \mathcal{N}[x;\vec{\mu}_f,\Sigma_f] log \frac{\mathcal{N}[x;\vec{\mu}_g,\Sigma_g]}{\mathcal{N}[x;\vec{\mu}_f,\Sigma_f]} dx. \quad (4)$$

Since the KL divergence is a distance measure, the energy function is represented as the negative of $D_{KL}(\mathcal{N}_g \,||\, \mathcal{N}_f)$. For the general case of a given word w, either one of the energy functions given in Eqs. 3 and 4 can be used in the learning procedure by the following energy-based max-margin objective [32]:

$$L_\theta(w,g,g') = max(0, m - logE_\theta(w,g) + logE_\theta(w,g')), \quad (5)$$

where w's positive context words g are selected from a window size l and negative context words g' are obtained by random sampling. The objective is to set the energies between positive and negative context words by at least a minimum margin m. The training is completed through mini-batch stochastic gradient descent over parameter set $\theta = \{\vec{\mu}_w, \Sigma_w\}$.

Based on w2g, [32] developed a model called word2Gaussian mixture (w2gm) that uses multimodal Gaussians to address polysemy. W2gm models polysemous words to some extent in an unsupervised way without sense-annotated corpora. In w2gm, word f is represented by:

$$f(x) = \sum_{i=1}^{K} p_i \, \mathcal{N}[x;\vec{\mu}_{f,i},\Sigma_{f,i}], \quad (6)$$

where p_i is the probability weight for sense i, and K is the maximum number of senses. The sum over weights for K senses is bounded by $\sum_{i=1}^{K} p_i = 1$. For words f and g, the ELK used in Eq. 5 becomes (in logarithm form):

$$\log E_\theta(f,g) = \log \sum_{i=1}^{K} \sum_{j=1}^{K} p_i q_j e^{\xi_{i,j}}, \quad (7)$$

where p_i's and q_j's are the probability weights of respective senses of f and g (as given in Eq. 6), respectively. $\xi_{i,j}$ is the partial log-energy [31], [32] term:

$$\xi_{i,j} = \log \mathcal{N}[0;\vec{\mu}_{f,i} - \vec{\mu}_{g,j}, \Sigma_{f,i} + \Sigma_{g,j}] = -(1/2)\log(\det(\Sigma_{f,i} + \Sigma_{g,j})) - (D/2)\log(2\pi) - (1/2)(\vec{\mu}_{f,i} - \vec{\mu}_{g,j})^T(\Sigma_{f,i} + \Sigma_{g,j})^{-1}(\vec{\mu}_{f,i} - \vec{\mu}_{g,j}), \quad (8)$$

where D denotes the embedding dimension, i.e the dimension of the semantic space and $\det$ stands for the determinant of a matrix. Eq 8 is used to analyze the relation between i^{th} and j^{th} senses of words f and g, respectively. Finally, Eq. 5 is used to learn the w2gm model. However, parameter θ should now also include the probability weights and becomes $\{\vec{\mu}_{f,i}, \Sigma_{f,i}, p_i\}$, where f is the word and i is the sense.

Two versions of the covariance matrix, spherical and diagonal, can be deployed in w2g models, [31], [32]. In spherical variant, distribution contains a single value of variance and covariance matrix is constant-diagonal. The diagonal case, however, possesses different values for each dimension, increasing the number of parameters to be learned. We adopted the spherical covariance in this study since it is generally preferred and reduces training time complexity.

B. Word2Gauss Mixture Model for N-Grams

Deploying w2g is adequate for the SemEval corpus but not suitable for the Google corpus since the latter requires n-gram training. We remodeled the architecture of w2g for n-gram training. N-grams are stored in the Google corpus as a pair of set of words (n-grams) and their occurrence counts. Treating each n-gram as context windows in ordinary co-occurrence based embedding learning, we repetitively apply the w2g learning procedure up to the number of occurrences of each n-gram. We update parameters by calculating the energy-based max-margin L after every iteration. Our proposed algorithm is presented in Algorithm 1. Inputs are *margin*, *target word*, *sets for selecting positive and context words*, *match count (the number of occurrences of n-gram)* and the energy function. In initialization, positive and negative context words are sampled from the sets N and D, where N contains words in the n-gram excluding the target word and D contains the entire vocabulary. Positive and negative logarithmic energy functions, $PosE$ and $NegE$, are calculated and the energy function L is formed accordingly. Then, for a given (word, n-gram)-pair, w2g of the target word is repeatedly updated by the number of match count. The same calculations are performed for all words within the n-gram. Finally, the overall procedure is repeated for all n-grams in the corpus. Aside from the context window, which is 10 for the SemEval and 5 for the Google n-grams, all other attributes are same across experiments.

Algorithm 1: N-gram w2g Training.

Input: m - Margin
Input: w - Target word
Input: N - Set of words in n-gram except w
Input: D - Set of words in the vocabulary
Input: E - Energy function
Input: p - Match count (number of occurrences)

g ← Select (N) // selects positive context from the remaining n-gram
g' ← Select (D) // randomly selects negative context words
count ← p // Iteration counts starts with match count
while $count > 0$ **do**
  $PosE \leftarrow log(E(w, g))$
  $NegE \leftarrow log(E(w, g'))$
  $L \leftarrow \max(0, m - PosE + NegE)$
  $Update(L, w)$ // stochastic gradient decent over parameter θ
  $count \leftarrow count - 1$
return

As an example for Algorithm 1, consider the 5-gram `analysis is often described as` which appears once in the corpus. For the word `analysis` (w), positive context words (c) will be sampled from the set N, which is `is often described as`. Note that the window size in the training procedure is set to the number of words in the original 5-gram, which is 5. The negative context word (c') can be taken from D. Starting from p (which is 1 in the particular example), w will be trained using the modified max-margin L (implemented by using the chosen energy function) until count reaches 0.

In the current work, single-word tokenization is used. However, one can also perform pre-processing that allows multi-word tokenization and then treat multi-word tokens as single tokens to form n-grams accordingly. In this case, our learning algorithm can still work without any required modifications by considering target word w which can be a multi-word token. Multi-word tokenization can increase the structural information of the corpus, and consequently contributes to the performance of the LSC detection.

C. Alignment of Semantic Vector Spaces

Diachronically trained semantic vector spaces cannot be compared directly if the models are trained separately with random initializations. Moreover, new words emerge or vanish as time passes. Thus, each semantic vector space must be aligned with each other before making bilateral comparisons.

First, a shared vocabulary, which is a set of words that are present in both corpora, needs to be created. By using mean vectors of w2g embeddings of the words in the shared vocabulary as anchors, two semantic spaces can be aligned. In this study, we utilize two versions of shared vocabulary creation, namely *Common Words (CW)* and *Frequent Words (FW)*. CWs are generated from the intersection of the sets of words used in diachronic training. For the SemEval, the intersection of words present in the originally provided two sub-corpora is used. Google corpus, on the other hand, contains 21 sub-corpora where each corresponds to a decade. CW set for the Google corpus should be constructed such that all words in the set are present at each corpus. FWs are selected based on the Law of Conformity ([9]), which dictates that meanings of frequent words tend to change less compared to infrequent ones. After obtaining normalized frequencies at each time frame, the words in the top 15% ranks are added to the list of FWs. This ratio is selected heuristically based on existing literature, [43], and thus it can be further adjusted for other LSC-tasks by fine-tuning.

After preparing shared vocabularies, we use the OPM for alignment [9], [11], [27], [51]. OPM alignment makes a basic assumption that languages as a whole do not change drastically compared to semantically changed words. This basic assumption, however, does not always hold true, especially when the two corpora are separated by immense time difference and the shared words are themselves prone to semantic change. There are also other advanced anchor word selection algorithms [69], [70] that can provide performance increases under such circumstances.

Embeddings, which are obtained by training on two sub-corpora belonging to time periods t and t', of words present in the shared vocabulary are stacked as columns of matrices $\mathbf{X}_{(t)}$ and $\mathbf{X}_{(t')}$, respectively. These matrices represent sub-semantic spaces and the aim is to find an orthogonal linear mapping from one of the sub-semantic spaces at t to the other one at t'. The optimal solution should perform the mapping most closely and be an orthogonal transformation to preserve inner products, which in turn preserve the semantic representation power of vectors. The corresponding orthogonal Procrustes problem can be formalized as follows [9], [50]:

$$\mathbf{Q_{opt}} = \underset{\mathbf{Q}}{\operatorname{argmin}} \mid\mid \mathbf{Q}\mathbf{X}_{(t)} - \mathbf{X}_{(t')} \mid\mid_F, \quad \text{subject to:} \quad \mathbf{Q}^T\mathbf{Q} = \mathbf{I}, \tag{9}$$

where $||\ .\ ||_F$ denotes the Frobenius norm. The optimal solution is well-known and given by $\mathbf{Q_{opt}} = \mathbf{U}\mathbf{V}^T$ where $\mathbf{U}$ and $\mathbf{V}$ are matrices holding the left and right singular vectors, respectively, of the singular value decomposition (SVD) of $\mathbf{X}_{(t')}\mathbf{X}_{(t)}^T$ [50]. $\mathbf{Q_{opt}}$ is the best rotational alignment matrix since $\mathbf{Q}^T\mathbf{Q} = \mathbf{I}$. After all vectors in the first semantic space are aligned by using rotation matrix $\mathbf{Q_{opt}}$ to the second semantic space, two independently trained w2g models can finally be compared. Note that the entire procedure is performed by using mean vectors and variance components are left intact since it is the mean vectors that designate the location of embeddings within semantic space. Finally, note that the time periods t and t' (so the granularity of time spans) on the above process can be designed to study semantic change due to socio-cultural (short-term) and linguistically-motivated (long-term) semantic shifts [18].

D. Lexical Semantic Change Detection

LSC detection can be studied quantitatively and qualitatively. Quantitative study requires ground truth results. Qualitative study, on the other hand, integrates neighborhood analysis and

visualization-based LSC detection on qualitatively well-known examples of semantic change. SemEval corpus provides ground truth while Google corpus does not.

1) Quantitative Detection: To obtain quantitative measures of semantic change, embeddings of each word in the aligned semantic spaces are compared with each other using a distance measure. To that end, we used cosine distance (CD) and Jeffreys' divergence (JD). CD is deployed between the aligned means of w2g embeddings trained for each language in the SemEval challenge. JD utilizes the KL-divergence and gives a quantitative measure taking into account both means and variances. Since JD is not bounded, it is normalized to the same range of CD. By using these two metrics, distances between two time periods for target words are obtained and semantic change is quantified.

For the LSC-ranking sub-task, semantic change values can directly be used. For the LSC-binary sub-task, however, a threshold value needs to be set to convert these values to binary decisions. Words with distances above the threshold are classified as semantically changed and below as not changed. Designing the threshold without supervision is of critical importance for the overall LSC detection performance. We will discuss several threshold selection methods in Section IV-E.

2) Qualitative Detection: To observe the performance of our proposed method on the well-known examples of semantic change, we used the common qualitative approach in which words are qualitatively compared with respect to their neighbors in the semantic space. Qualitatively studying semantic change through observing neighborhoods is frequently performed [9], [17] while there also exist other methods that look at full truly occurring sentences rather than at inferred semantic neighbors [45]. Qualitative analysis can verify the cultural impacts and recently acquired senses as well as playing a crucial role in LSC detection applications, understanding the semantic change mechanisms, and deriving intuitive comprehension for quantitative methods, especially for non-annotated corpora. In the current work, we used neighborhood analysis, which has global and local variants. Global neighborhood analysis focuses on movements of embedding vectors in semantic space between different time periods after alignment. In local neighborhood analysis, semantic change is studied indirectly through the change of neighboring words of a particular word without alignment, [71]. A global shift on an embedding vector is not important for local neighborhood analysis since locations matter only relatively.

E. Threshold Selection Methods

After two aligned semantic spaces are compared quantitatively and a distance measure is used to quantify the semantic change for each word, a threshold is needed to convert these scores to binary classification decisions of semantically changed or unchanged. There are three main methods to design the threshold: Language-specific Mean, Cross-language Mean, and Gamma Quantile thresholding. A list of reference words whose distance measures are used in the threshold design is needed. Simply, target words lists given by the SemEval itself are commonly used in the literature for that purpose. It is also possible to use a shared vocabulary list which corresponds to the set of words used during the OPM alignment in Section IV-C. [43] is the only study that deploys shared vocabulary lists used in vector space alignment to also form thresholding lists. They, however, applied this approach only to language-specific mean thresholding. To the best of our knowledge, shared vocabularies are not considered in the context of other thresholding approaches. Along with others, we also investigated shared vocabulary options of OPM such as FWs and CWs (as given in Section IV-C) to calculate thresholds. Below, we provide detailed information on threshold selection methods and vocabulary alternatives:

1) Language-Specific Mean Threshold: In language-specific thresholding, thresholds are set for each language separately. Semantic change scores for target word lists are first calculated and their averages are taken as respective threshold values. Words in the target word lists with semantic change scores exceeding the designated threshold are then categorized as semantically changed. This process is performed for each language separately and language-specific thresholds are determined. Language-specific mean thresholding, which assumes that target words with above-average scores have undergone semantic changes, is utilized in [43], [46], [53]. The usage of shared vocabularies instead of target words to calculate means is also present in [43].

2) Cross-Language Mean Threshold: Unlike language-specific mean thresholding, cross-language thresholding, proposed in [53], uses the mean of semantic change scores of words present in the combined (of all languages) target word list. The calculated mean is then used as the cross-language threshold applicable to all languages. This approach also assumes that above-average words have undergone semantic change. To the best of our knowledge, shared vocabularies used in alignment have not been used in cross-language mean thresholding in the literature.

3) Gamma Quantile Threshold: In this approach, scores of target word lists are treated as Gamma distributions [56]. 75% quantile of the distribution is advised to be selected as it divides the distribution from its right tail (the point where the cumulative probability reaches 0.75). As the reference word list, again target words are used and 75% quantile thresholds are calculated for each language separately. In the literature, shared vocabularies used as alignment word lists have not been used with Gamma quantile thresholding either.

V. Experiments and Results[1]

We present our experimental results on the SemEval-2020 Task 1 and Google corpus. In the SemEval subsection, we provide detailed experimental results from several variants of our model as we engineer it to attain higher levels of performance. We then provide results and rankings of our proposed method in the context of the SemEval LSC evaluation framework. In the Google corpus subsection, we present qualitative results by using local neighborhood analysis.

A. SemEval-2020 Task 1

SemEval-2020 Task 1 requires correctly detecting words that have experienced semantic change from a target word list, separately provided for each of the four languages present dataset.

[1]Details and Codes are Available At: [Online]. Available: https://github.com/koc-lab/lsc_w2g.

There are two LSC detection sub-tasks: binary classification (LSC-binary) and ranking (LSC-ranking).

For both sub-tasks, we independently trained w2g models on two sub-corpora of separate time periods (as explained in Section III) for each language. Two models on independent semantic vector spaces are then obtained. We also repeated this procedure with stop-words removed to study their effects. In the OPM process, CWs or FWs can be used as shared vocabulary alternatives. After the alignment, CD- and JD-based semantic change scores are calculated. With alternatives present at several parts of our pipeline, we performed experiments by using several variants and configurations of our model to engineer the best possible design. In this section, the construction of corpora, shared vocabulary and choice of distance measure will be denoted by abbreviations. The proposed variants are named starting with the prefix OURS-W2G indicating the underlying w2g. Following the prefix, shared vocabulary choice is denoted with -CW (Common Words) or -FW (Frequent Words). The naming convention continues with the indication of distance measure, -CD (Cosine Distance) or -JD (Jeffreys' Divergence). Lastly, if the training is performed after stop-words removed, -S is added. For instance, a configuration named OURS-W2G-CW-CD-S can be parsed as w2g model trained on stop-word removed corpora (-S), the cosine distance (-CD) measures are found after the common words (-CD) are used as the shared vocabulary to be used in the OPM alignment. We adopted this naming convention throughout the manuscript.

1) LSC-Ranking: In this sub-task, the aim is to find correlations between the graded ground truth information provided for a target word list and the obtained semantic change scores based on distance measures for that list. Graded ground truth data provided by [27] contain labels for each target word in the task, which are real numbers in the interval [0,1] denoting the semantic change scores. A higher value indicates a higher semantic change has occurred. Using the ground truth and acquired results, we calculate Spearman's rank-order correlation coefficients, p, which implies correlations between the ranks of target words in the experimental results and ground truth. p is in the interval $[-1, 1]$ where -1 means completely opposite ranking order while 1 means a perfect correlation with the ground truth.

Spearman correlation results of several configurations of our proposed method on LSC-ranking are tabulated in Table I. Note that we will compare our proposed method with benchmark methods later after we analyze our configurations in detail. In the LSC-ranking, CW-based alignment is a better choice compared to FW since the best performance for each language is achieved with a CW choice. Significant improvements for Swedish are observed with the usage of JD. In terms of overall performance, the best results are obtained for OURS-W2G-CW-JD and OURS-W2G-CW-CD settings. Since this sub-task directly requires to analyze magnitudes of semantic change, covariances of w2g can be used to deduce information related to semantic breadth, which is an important property as words with expanding or shrinking semantic coverage also experience semantic change even if mean locations have not been shifted much.

2) LSC-Binary: In this sub-task, one needs to convert scores of LSC-ranking to binary decisions. Same configurations for our proposed models and alignment procedures are also used for this sub-task. Thresholds are applied to scores to transform real numbers into binary classification labels. In this process, as explained in Section IV-E, three different techniques (Gamma, Language-specific Mean and Cross-language Mean thresholdings) are used with both target words and the corresponding shared vocabularies used in the alignment process for that particular setup. Our average results with several configurations are tabulated in Table II. Our detailed results for individual languages are also given in the Supplementary Materials.

Upon inspecting our results, the best thresholding method is the Gamma quantile for all languages except Latin. However, regarding the overall model performance, CW-aligned models surpassed FW-aligned ones as can be seen in Table II. This result implies that methodology is prone to fine-tuning for FW-alignment and thus one setting for a specific language can deteriorate performances of others. In other words, if one needs to focus on a specific language, FW is a better choice whereas CW is superior when one model is to be used across several languages. As we mentioned previously in Section IV-C, there are other methods to create shared vocabulary lists by compiling anchor words, [69], [70]. SemEval corpora sometimes contains higher time gaps between the aligned periods (e.g. Latin) and thus more advanced anchor word selection algorithms as given in [69], [70] can increase performance further. The removal of

TABLE I
LSC-RANKING SUB-TASK RESULTS (SPEARMAN CORRELATION) FOR LANGUAGES AND THEIR AVERAGE

Proposed Models	**Ave.**	**Eng.**	**Ger.**	**Lat.**	**Swe.**
OURS-W2G CW-CD	**0.396**	**0.399**	**0.381**	0.218	0.586
OURS-W2G CW-CD-S	0.267	0.285	0.250	0.244	0.288
OURS-W2G FW-CD	0.370	0.363	0.339	**0.271**	0.505
OURS-W2G FW-CD-S	0.300	0.230	0.300	0.236	0.433
OURS-W2G CW-JD	0.384	0.368	0.363	0.194	**0.610**
OURS-W2G CW-JD-S	0.309	0.272	0.250	0.204	0.509
OURS-W2G FW-JD	0.378	0.368	0.313	0.267	0.562
OURS-W2G FW-JD-S	0.304	0.184	0.293	0.216	0.523

TABLE II
LSC-BINARY ACCURACY (%) RESULTS ON AVERAGE FOR OUR PROPOSED MODELS WITH SEVERAL THRESHOLDING (GAMMA, LANGUAGE-SPECIFIC (LC) AND CROSS-LANGUAGE (CL)) AND WORD LIST ALTERNATIVES (TARGET WORDS ('T') AND SHARED WORDS ('S')) USED IN THRESHOLDING. BEST AVERAGE SCORE IS EMBOLDENED

Proposed Models OURS-	Gamma	LC	CL
	T / S	T / S	T / S
W2G CW - CD	59.68 / 60.50	58.03 / 60.18	58.05 / 60.65
W2G CW - CD - S	**65.20** / 60.33	57.23 / 64.20	60.08 / 61.80
W2G FW - CD	64.15 / 62.28	58.83 / 59.53	60.43 / 58.90
W2G FW - CD - S	64.48 / 61.78	54.50 / 57.38	61.20 / 60.58
W2G CW - JD	62.33 / 59.75	58.83 / 60.33	59.08 / 61.53
W2G CW - JD - S	**65.20** / 59.65	56.43 / 63.05	59.35 / 63.20
W2G FW - JD	63.58 / 62.80	58.20 / 58.05	60.93 / 59.93
W2G FW - JD - S	63.13 / 60.63	55.80 / 55.63	54.38 / 59.15

TABLE III
LOCAL CONFIGURATIONS FOR THE LSC-BINARY

Languages	Main Model (OURS-)	Thresholding	Word List
English	W2G-FW-CD-S	Gamma	Target
German	W2G-CW-JD-S	LS	Shared
Latin	W2G-FW-CD-S	LS	Shared
Swedish	W2G-FW-CD-S	CL	Shared

stop-words (-S) improves the performance in the Gamma thresholding with mixed results observed for the language-specific and cross-language thresholding. JD is favored in German. When shared vocabularies are used in threshold calculations, we obtained better performances for individual languages except for English. The usage of shared vocabulary is better for German, Latin and Swedish especially with language-specific and cross-language mean thresholding. Stop-word removal increased the performance regardless of the distance metric used. However, distance metric choice does not correlate with an increase in performance. To sum up, best performing configurations are OURS-W2G-CW-CD-S and OURS-W2G-CW-JD-S with both having 65.2% accuracy with Gamma thresholding when the threshold is set based on target words. Although the performance increase occurs language-wise when shared word list is used, the overall performance is higher for target word based thresholding.

3) Benchmarking With SemEval Algorithms: Having analyzed several possible configurations of our proposed method, we now compare our models with the algorithms present in the SemEval-2020 Task 1. There are alternative configuration methods in LSC detection, language-specific and cross-language configurations similar to the mean thresholding options. For simplicity, we will refer to the language-specific and cross-language configurations as "local" and "global," respectively. In the local configuration, algorithm parameters are chosen separately for each language whereas a single configuration is used for all languages in the global configurations. We denote local configuration by W2G-LOC_CNFG. This configuration selects the best performing settings among its variants for each language separately. For the LSC-ranking sub-task, W2G-LOC_CNFG is OURS-W2G-CW-CD for English and German, OURS-W2G-FW-CD for Latin and OURS-W2G-FW-JD for Swedish, as deduced from Table I. For the LSC-binary sub-task, W2G-LOC_CNFG is configured as follows: OURS-W2G-FW-CD-S with Gamma thresholding with target words implementation for English; OURS-W2G-FW-CD-S with Cross-Language (CL) mean thresholding with shared words implementation for Swedish; OURS-W2G-FW-CD-S with Language-Specific (LS) mean thresholding with shared words implementation for Latin; OURS-W2G-CW-JD-S with LS mean thresholding implemented with shared vocabulary list for German (selected among best performers). The above configurations are listed in Table III.

We refer to our two alternative global configurations as "Global" and "Global+". The first one, denoted by W2G-GLO_CNFG, for each sub-task separately, a single best-performing configuration is constructed and applied to all four languages with the objective of obtaining high performance on average at the expense of individual language-specific performance. In the LSC-ranking sub-task, our W2G-GLO_CNFG corresponds to OURS-W2G-CW-CD. For the LSC-binary sub-task, among two models with equal performance, we chose OURS-W2G-CW-JD-S with Gamma threshold with target word list implementation. Second global configuration is denoted by W2G-GLO+_CNFG where an overall single best-performing configuration is constructed and applied to all four languages for both sub-tasks. Our W2G-GLO+_CNFG corresponds to OURS-W2G-FW-CD with Gamma threshold with target word list implementation for LSC-binary.

TABLE IV
SEMEVAL LSC-BINARY ACCURACY RESULTS (%) [27]

Models	Ave.	Eng.	Ger.	Lat.	Swe.
OURS-LOC_CNFG	**69.7**	64.9	**79.2**	57.5	**77.4**
UWB [53]	68.7	62.2	75.0	**70.0**	67.7
Life-Language [43]	68.6	70.3	75.0	55.0	74.2
Jiaxin&Jinan [56]	66.5	64.9	72.9	**70.0**	58.1
RPI-TRUST [52]	66.0	64.9	75.0	50.0	74.2
OURS-GLO_CNFG	65.2	59.5	77.1	50.0	74.2
OURS-GLO+_CNFG	64.2	59.5	72.9	50.0	74.2
UG Student Intern [46]	63.9	56.8	72.9	55.0	71.0
DCC [58]	63.7	64.9	66.7	52.5	71.0
NLP@IDSIA [72]	63.7	62.2	62.5	62.5	67.7
JCT [73]	63.6	64.9	68.8	50.0	71.0
Skurt [74]	62.9	56.8	56.2	67.5	71.0
Discovery Team [55]	62.1	56.8	68.8	55.0	67.7
BASELINE-BC [27]	61.3	59.5	68.8	52.5	64.5
TUE [75]	61.2	56.8	58.3	65.0	64.5
Entity [42]	59.9	67.6	66.7	47.5	58.1
IMS [62]	59.8	54.1	68.8	55.0	61.3
cs2020 [54]	58.7	59.5	50.0	57.5	67.7
UiO-UvA [76]	58.7	54.1	64.6	45.0	71.0
NLPCR [77]	58.4	**73.0**	54.2	45.0	61.3
BASELINE-BM [27]	57.6	56.8	64.6	35.0	74.2
CBK [78]	55.4	56.8	62.5	47.5	54.8
RANDOM [59]	55.4	48.6	47.9	47.5	**77.4**
UoB [79]	52.6	56.8	47.9	57.5	48.4
UCD [60]	52.1	62.2	50.0	35.0	61.3
RIJP [33]	51.1	54.1	50.0	55.0	45.2
BASELINE-BF [27]	43.9	43.2	41.7	65.0	25.8

By using our local and global configurations, we tabulate our results along with results of all SemEval-2020 Task 1 participants in Tables IV and V, where algorithms are listed in descending order with respect to their overall average performance. We also consider three baseline algorithms among others to show important baseline performance levels: Baseline Count (BC), Baseline-Majority (BM) and Baseline-Frequency (BF). Baseline models are implemented by [27] as benchmarks for the SemEval. BM assigns each word 0 (unchanged), since

TABLE V
SEMEVAL LSC-RANKING RESULTS (SPEARMAN) [27]

Models	Ave.	Eng.	Ger.	Lat.	Swe.
UG Student Intern [46]	**0.53**	0.42	**0.73**	0.41	0.55
Jiaxin&Jinan [56]	0.52	0.33	0.72	0.44	0.59
cs2020 [54]	0.50	0.38	0.70	0.40	0.54
UWB [53]	0.48	0.37	0.70	0.25	0.60
Discovery Team [55]	0.44	0.36	0.60	0.46	0.34
RPI-TRUST [52]	0.43	0.29	0.52	**0.46**	0.50
OURS-LOC_CNFG	0.42	0.40	0.38	0.27	**0.61**
OURS-GLO_CNFG	0.40	0.40	0.38	0.22	0.59
OURS-GLO+_CNFG	0.37	0.36	0.34	0.27	0.51
Skurt [74]	0.37	0.21	0.66	0.40	0.23
IMS [62]	0.37	0.30	0.66	0.10	0.43
UiO-UvA [76]	0.37	0.14	0.70	0.37	0.28
Entity [42]	0.35	0.25	0.50	0.30	0.36
RANDOM [59]	0.30	0.21	0.34	0.25	0.39
NLPCR [77]	0.29	**0.44**	0.45	0.15	0.11
JCT [73]	0.25	0.01	0.51	0.42	0.08
UCD [60]	0.23	0.31	0.22	0.07	0.34
CBK [78]	0.23	0.06	0.40	0.34	0.14
Life-Language [43]	0.22	0.30	0.21	-0.02	0.39
NLP@IDSIA [72]	0.19	0.03	0.18	0.25	0.32
BASELINE-BC [27]	0.14	0.02	0.22	0.36	-0.02
UoB [79]	0.10	0.11	0.22	-0.02	0.10
TUE [75]	0.09	-0.16	0.39	0.18	-0.06
RIJP [33]	0.09	0.16	0.10	0.07	0.03
DCC [58]	-0.08	-0.22	0.01	0.02	-0.15
BASELINE-BF [27]	-0.08	-0.22	0.01	0.02	-0.15

unchanged class is the majority in the task. BC uses count-based word representations of target words with respect to the shared words present in two diachronic corpora and calculates semantic change scores using CD. Finally, BF calculates the normalized frequencies for a target word at each corpus. The absolute distance between the normalized frequencies is used as a distance measure. Being a binary decision, BM is obviously only applicable to LSC-binary but not to LSC-ranking. For the results of participants, we refer the published results as provided by [27]. The highest scores on each language and average are emboldened. Rankings of our two configurations are also presented in Table VI.

Our overall average scores for LOC_CNFG are 69.7% for accuracy and 0.415 for Spearman correlation, with ranks of 1^{st} (along with language-specific 1^{st} ranks for two of four languages) and 7^{th} in LSC-binary and LSC-ranking, respectively. In LSC-binary, we reach the highest scores for German and Swedish and report the highest score for Swedish for the LSC-ranking sub-task. In both of our global configurations GLO_CNFG and GLO+_CNFG, although our rank drops to 5^{th} on average for LSC-binary, there is no performance degradation in LSC-ranking, where we maintain the rank of 7^{th}. We still

TABLE VI
RANKS OF OUR METHODS IN THE SEMEVAL

Subtask	Config.	Avg.	Eng.	Ger.	Lat.	Swe.
LSC-binary	LOCAL	1	4	1	6	1
	GLOBAL	5	11	1	15	2
	GLOBAL+	5	11	4	15	2
LSC-ranking	LOCAL	7	3	15	12	1
	GLOBAL	7	3	15	15	2
	GLOBAL+	7	5	15	12	5

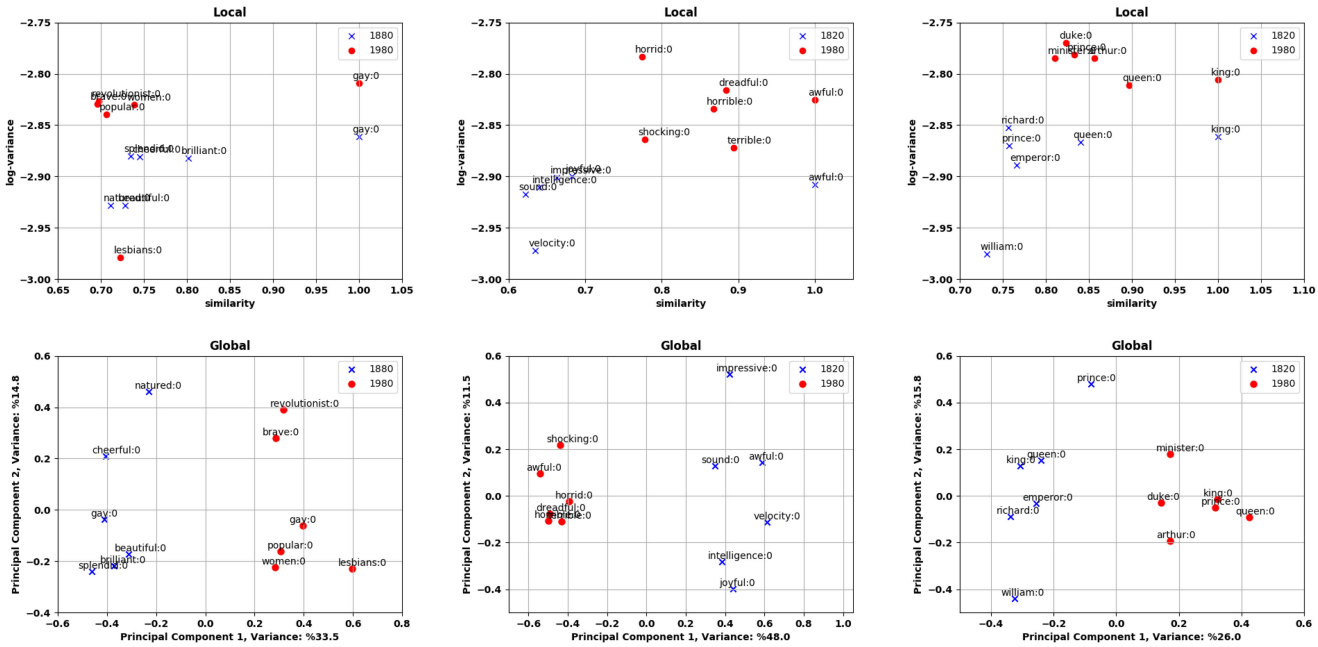
hold the highest score in German with GLO_CNFG, where we report overall 65.2% accuracy and 0.40 Spearman correlation for LSC-binary and LSC-ranking sub-tasks, respectively. We report 64.2% accuracy and 0.37 Spearman correlation with GLO+_CNFG for LSC-binary and LSC-ranking sub-tasks, respectively.

In the SemEval framework, reaching higher performance in one sub-task does not imply similar success in the other sub-task. Scores of LSC-binary sub-task are formed out of correct detection rates while the LSC-ranking requires correctly ranking words with respect to semantic changes they have experienced. Our proposed model performs better in the LSC-binary, although it also ranks considerably high in the LSC-ranking.

B. Qualitative Neighborhood Analysis on Google Corpus

We qualitatively study Google N-gram corpus by observing the local and global neighborhood changes of particular words. As in [9], [28], we use words `gay`, `awful` and `king` as examples. Local and global neighborhood change analysis for these words are presented in Fig. 2(a), 2(b) and 2(c). To analyze local and global neighborhood changes, w2g models are trained on decade sub-corpora of Google N-gram corpus separated by at least a century. For visual demonstration, dimension reduction is applied for mean components. In local neighborhood change plots, nearest neighbors of a word in two different time periods are shown with respect to similarity (across mean vectors) and log-variances. For the global neighborhood change, principal component analysis (PCA) is used after OPM alignment since the locations of target words matter in global neighborhood analysis. For visual demonstration, the first two principal components are used to visualize the nearest neighbors of a given word. The axes of plots are the first two principal components in the global case and the variance values explained by these components are also noted.

In Fig. 2(a), `gay` has the nearest neighbors `brilliant` and `splendid` in 1880. Due to semantic change, `gay` acquires a new meaning of sexual orientation which is closer to other gender identities and sexual orientations, `women` and `lesbians` in 1980. This change occurs with the removal of the original sense and thus the meanings in 1880 and 1980 are not correlated. In Fig. 2(b), the semantic change of `awful` can be clearly seen. `Awful` is related to `joyful` and `impressive` in 1820 while possessing a negative sense in 1980 as implied by its association with `horrible` and `terrible`. Semantic



(a) *gay* between 1880 and 1980 (b) *awful* between 1820 and 1980 (c) *king* between 1820 and 1980

Fig. 2. Local and global changes of `gay`, `awful` and `king` in the Google N-gram Fiction.

degeneration (or pejoration [14]) has occurred in the case of `awful`. Neighborhood analysis also allows us to demonstrate cases where semantic change is not observed. In Fig. 2(c), the local neighborhood of the embedding of `king` does not change significantly as demonstrated by appearances of `queen` and `prince` as neighbors on both time periods. Local positions of `king` are also similar (with similarity metrics 1) as given in Fig. 2(c). As a final note, we also performed neighboorhood analysis in the SemEval corpus and provide examples in the Supplementary Materials.

VI. Conclusion

In this study, we proposed a w2g-based successful solution to the LSC detection problem. We utilized w2g embeddings by using their covariance information that models the semantic coverage of words within a semantic space. Our proposed models are benchmarked with the main standardized evaluation framework of the LSC field (SemEval-2020 Task 1). We reported well-performing results indicating that w2g can be effectively used in the LSC detection.

We provided an in-depth treatment and analysis of w2g's in the context of LSC detection by extensively studying several alternatives and aspects present in the overall pipeline. Our proposed models are categorized as the local and global configurations. While local configurations focus on language-specific performance, global configurations aim to optimize overall best performance. We used CD and JD to calculate semantic change scores for quantitative measurements. Distance measures are compared against ground truth labels of the LSC-ranking subtask using Spearman correlation. OPM is used to align independently trained and randomly initialized embeddings. We also considered several thresholding methods such as *Gamma Quantile, Language-specific Mean* and *Cross-language Mean*.

In addition to the SemEval framework, we studied the Google corpus that contains n-gram information. To deploy w2g-based models on n-gram data, we developed a modified w2g model that accepts n-grams as inputs. We provided extensive results of qualitative analysis by using neighborhood analysis to further confirm the effectiveness of w2g's. Our n-gram based w2g model can work on n-grams with multi-word tokens. Pre-processing the corpus with multi-word tokenization methods and investigating its effects can be considered as a future work to improve performance and insights.

We stressed the importance of the shared vocabulary selection by utilizing two variants, FW and CW. We also noted the ongoing research in the anchor word selection procedures for alignment. When there is large time separation between diachronic corpora, the assumption of OPM starts to be violated. This increases the possibility of compiling shared vocabularies with some semantically-changed words, which are not suitable for alignment. Anchor words provide room for improvements especially under these circumstances.

Since variances of w2g's directly correlate with semantic coverage, they can be instrumental to study semantic change in utmost detail. For example, specific types of semantic change such as semantic narrowing and broadening can better be studied with w2g models. The ensemble of word vector-based distance scores and variance measures has the potential of offering better strategies to study, detect, rank and classify semantic change. With the successful introduction of w2g-based models to the diachronic NLP studies, we expect to see future efforts exploring ways to exploit word variances in LSC detection and categorization.

REFERENCES

[1] M. Hale, *Historical Linguistics: Theory and Method*. Oxford, U.K.: Blackwell, 2007.

[2] E. Sagi, S. Kaufmann, and B. Clark, "Semantic density analysis: Comparing word meaning across time and phonetic space," in *Proc. Workshop Geometrical Models Natural Lang. Semantics*, Athens, Greece: Association for Computational Linguistics, 2009, pp. 104–111.

[3] J.-B. Michel *et al.*, "Quantitative analysis of culture using millions of digitized books," *Science*, vol. 331, no. 6014, pp. 176–182, 2011.

[4] C. Rohrdantz, A. Hautli, T. Mayer, M. Butt, D. A. Keim, and F. Plank, "Towards tracking semantic change by visual analytics," in *Proc. 49th Annu. Meeting Assoc. Comput. Linguistics, Hum. Lang. Technol.*, Portland, OR, USA: Association for Computational Linguistics, 2011, pp. 305–310.

[5] A. Jatowt and K. Duh, "A framework for analyzing semantic change of words across time," in *Proc. IEEE/ACM Joint Conf. Digit. Libraries*, 2014, pp. 229–238.

[6] S. Mitra *et al.*, "An automatic approach to identify word sense changes in text media across timescales," *Natural Lang. Eng.*, vol. 21, no. 5, pp. 773–798, 2015.

[7] X. Tang, W. Qu, and X. Chen, "Semantic change computation: A successive approach," *World Wide Web*, vol. 19, no. 3, pp. 375–415, 2015.

[8] L. Frermann and M. Lapata, "A Bayesian model of diachronic meaning change," *Trans. Assoc. Comput. Linguistics*, vol. 4, pp. 31–45, 2016.

[9] W. L. Hamilton, J. Leskovec, and D. Jurafsky, "Diachronic word embeddings reveal statistical laws of semantic change," in *Proc. 54th Annu. Meeting Assoc. Comput. Linguistics*, Berlin, Germany: Association for Computational Linguistics, 2016, pp. 1489–1501.

[10] H. Dubossarsky, D. Weinshall, and E. Grossman, "Outta control: Laws of semantic change and inherent biases in word representation models," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, Copenhagen, Denmark: Association for Computational Linguistics, 2017, pp. 1136–1145.

[11] Z. Yao, Y. Sun, W. Ding, N. Rao, and H. Xiong, "Dynamic word embeddings for evolving semantic discovery," in *Proc. 11th ACM Int. Conf. Web Search Data Mining*, New York, NY, USA: Association for Computational Machinery, 2018, pp. 673–681.

[12] X. Tang, "A state-of-the-art of semantic change computation," *Natural Lang. Eng.*, vol. 24, no. 5, pp. 649–676, 2018.

[13] D. Crystal, *Language and the Internet*, *2nd ed*. New York, NY, USA: Cambridge Univ. Press, 2006.

[14] L. Bloomfield, *Language*. New York, NY, USA: Holt, Rinehart and Winston, 1933.

[15] K. Blank, *Historical Semantics and Cognition*. Berlin, Germany: De Gruyter Mouton, 1999.

[16] K. Gulordava and M. Baroni, "A distributional similarity approach to the detection of semantic change in the Google Books Ngram Corpus," in *Proc. Workshop Geometrical Models Natural Lang. Semantics*, Edinburgh, U.K.: Association for Computational Linguistics, 2011, pp. 67–71.

[17] 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 *Proc. 52nd Annual Meeting Assoc. Comput. Linguistics*, Baltimore, Maryland, USA: Association for Computational Linguistics, Jun. 2014, pp. 1020–1029.

[18] A. Kutuzov, L. Øvrelid, T. Szymanski, and E. Velldal, "Diachronic word embeddings and semantic shifts: A survey," in *Proc. 27th Int. Conf. Comput. Linguistics*, Santa Fe, New Mexico, USA: Association for Computational Linguistics, 2018, pp. 1384–1397.

[19] N. Tahmasebi, L. Borin, and A. Jatowt, "Survey of computational approaches to lexical semantic change," 2019, *arXiv: 1811.06278*.

[20] H. Schütze, "Automatic word sense discrimination," *Comput. Linguistics*, vol. 24, no. 1, pp. 97–123, 1998.

[21] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," 2013, *arXiv:1301.3781*.

[22] J. Pennington, R. Socher, and C. Manning, "GloVe: Global vectors for word representation," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, Doha, Qatar: Association for Computational Linguistics, 2014, pp. 1532–1543.

[23] M. Davies, "Corpus of historical American English (COHA)," Harvard Dataverse, 2015. [Online]. Available: https://doi.org/10.7910/DVN/8SRSYK

[24] Y. Lin, J.-B. Michel, E. Aiden Lieberman, J. Orwant, W. Brockman, and S. Petrov, "Syntactic annotations for the Google Books Ngram Corpus," in *Proc. ACL Syst. Demonstrations*, Jeju Island, Korea: Association for Computational Linguistics, 2012, pp. 169–174.

[25] M. Rissanen and J. Tyrkkö, "The Helsinki corpus of English texts (HC)," in *Principles and Practices for the Digital Editing and Annotation of Diachronic Data (Studies in Variation, Contacts and Change in English Series)*. Finland: Varieng, 2012, no. 14.

[26] R. C. Barranco, R. F. D. Santos, M. S. Hossain, and M. Akbar, "Tracking the evolution of words with time-reflective text representations," in *Proc. IEEE Int. Conf. Big Data*, 2018, pp. 2088–2097.

[27] D. Schlechtweg, B. McGillivray, S. Hengchen, H. Dubossarsky, and N. Tahmasebi, "SemEval-2020 Task 1: Unsupervised lexical semantic change detection," in *Proc. 14th Int. Workshop Semantic Eval.*, Barcelona, Spain: Association for Computational Linguistics, 2020, pp. 1–23.

[28] D. T. Wijaya and R. Yeniterzi, "Understanding semantic change of words over centuries," in *Proc. Int. Workshop Detecting Exploiting Cultural Diversity Social Web*, New York, NY, USA: Association Computer Machinery, 2011, pp. 35–40.

[29] V. Kulkarni, R. Al-Rfou, B. Perozzi, and S. Skiena, "Statistically significant detection of linguistic change," in *Proc. 24th Int. Conf. World Wide Web*, Republic Canton Geneva, 2015, pp. 625–635.

[30] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," in *Proc. Conf. North Amer. Chapter Assoc. Comput. Linguistics, Hum. Lang. Technol.*, 2019, pp. 4171–4186.

[31] L. Vilnis and A. McCallum, "Word representations via Gaussian embedding," in *Proc. Int. Conf. Learn. Representations*, 2015.

[32] B. Athiwaratkun and A. Wilson, "Multimodal word distributions," in *Proc. 55th Annu. Meeting Assoc. Comput. Linguistics*, Vancouver, Canada: Association for Computational Linguistics, Jul. 2017, pp. 1645–1656.

[33] R. Iwamoto and M. Yukawa, "RIJP at SemEval-2020 Task 1: Gaussian-based embeddings for semantic change detection," in *Proc. 14th Workshop Semantic Eval.*, Barcelona: International Committee on Computational Linguistics, 2020, pp. 98–104.

[34] M. Hilpert, *Germanic Future Constructions: A Usage-Based Approach to Language Change*. Amsterdam, The Netherlands: John Benjamins, 2008.

[35] S. Morsy and G. Karypis, "Accounting for language changes over time in document similarity search," *Assoc. Comput. Machinery Trans. Inf. Syst.*, vol. 35, no. 1, pp. 1–26, Sep. 2016.

[36] Y. Jiang, Y. Yu, X. Wang, and Z. Liu, "The law of semantic change of opposite compounds," in *Proc. Int. Conf. Asian Lang. Process.*, 2018, pp. 108–112.

[37] D. Schlechtweg, S. Schulte im Walde, and S. Eckmann, "Diachronic usage relatedness (DURel): A framework for the annotation of lexical semantic change," in *Proc. Conf. North Amer. Chapter Assoc. Comput. Linguistics, Hum. Lang. Technol.*, New Orleans, LA, USA: Association for Computational Linguistics, Jun. 2018, pp. 169–174.

[38] D. Schlechtweg and S. Schulte Im Walde, "Simulating lexical semantic change from sense-annotated data," in *Proc. Evol. Lang., Proc. 13th Int. Conf.*, A. Ravignani, C. Barbieri, M. Martins, M. Flaherty, Y. Jadoul, E. Lattenkamp, H. Little, K. Mudd, and T. Verhoef, Eds., 2020, p. 393.

[39] N. Tahmasebi and T. Risse, "Finding individual word sense changes and their delay in appearance," in *Proc. Int. Conf. Recent Adv. Natural Lang. Process.*, Varna, Bulgaria: INCOMA Ltd., 2017, pp. 741–749.

[40] D. Schlechtweg, A. Hätty, M. D. Tredici, and S. S. im Walde, "A wind of change: Detecting and evaluating lexical semantic change across times and domains," in *Proc. 57th Annu. Meeting Assoc. Comput. Linguistics*, Florence, Italy: Association for Computational Linguistics, 2019, pp. 732–746.

[41] D. Schlechtweg, S. Eckmann, E. Santus, S. S. im Walde, and D. Hole, "German in flux: Detecting metaphoric change via word entropy," in *Proc. 21st Conf. Comput. Natural Lang. Learn.*, Vancouver, Canada: Association for Computational Linguistics, 2017, pp. 354–367.

[42] V. Jain, "GloVeInit at SemEval-2020 Task 1: Using Glove vector initialization for unsupervised lexical semantic change detection," in *Proc. 14th Int. Workshop Semantic Eval.*, Barcelona, Spain: Association for Computational Linguistics, 2020, pp. 208–213.

[43] E. Asgari, C. Ringlstetter, and H. Schütze, "EmbLexChange at SemEval-2020 Task 1: Unsupervised embedding-based detection of lexical semantic changes," in *Proc. Int. Workshop Semantic Eval.*, 2020, pp. 201–207.

[44] M. Peters *et al.*, "Deep contextualized word representations," *Proc. Conf. North Amer. Chapter Assoc. Comput. Linguistics, Hum. Lang. Technol.*, New Orleans, LA, USA: Association for Computational Linguistics, vol. 1, pp. 2227–2237, Jun. 2018.

[45] M. Giulianelli, M. Del Tredici, and R. Fernández, "Analysing lexical semantic change with contextualised word representations," in *Proc. 58th Annu. Meeting Assoc. Comput. Linguistics*, 2020, pp. 3960–3973.

[46] M. Pömsl and R. Lyapin, "CIRCE at SemEval-2020 Task 1: Ensembling context-free and context-dependent word representations," in *Proc. 14th Workshop Semantic Eval.*, Barcelona, Spain: International Committee on Computational Linguistics, 2020, pp. 180–186.

[47] Y. Kim, Y.-I. Chiu, K. Hanaki, D. Hegde, and S. Petrov, "Temporal analysis of language through neural language models," in *Proc. ACL Workshop Lang. Technol. Comput. Social Sci.*, Baltimore, MD, USA: Association for Computational Linguistics, 2014, pp. 61–65.

[48] H. Dubossarsky, S. Hengchen, N. Tahmasebi, and D. Schlechtweg, "Time-out: Temporal referencing for robust modeling of lexical semantic change," in *Proc. 57th Annu. Meet. Assoc. Comput. Linguistics*, Florence, Italy: Association for Computational Linguistics, 2019, pp. 457–470.

[49] S. Eger and A. Mehler, "On the Linearity of Semantic Change: Investigating Meaning Variation Via Dynamic Graph Models," in *Proc. 54th Annu. Meeting Assoc. Comptu. Linguistics*, Berlin, Germany: Association for Computational Linguistics, Aug. 2016, pp. 52–58.

[50] P. H. Schönemann, "A generalized solution of the orthogonal Procrustes problem," *Psychometrika*, vol. 31, pp. 1–10, 1966.

[51] R. Bamler and S. Mandt, "Dynamic word embeddings," *Proc. 34th Int. Conf. Mach. Learn., Ser. Proc. Mach. Learn. Res.*, D. Precup and Y. W. Teh, Eds., International Convention Centre, Sydney, Australia, vol. 70, pp. 380–389, Aug. 2017.

[52] M. Gruppi, S. Adali, and P.-Y. Chen, "SChME at SemEval-2020 Task 1: A model ensemble for detecting lexical semantic change," in *Proc. 14th Workshop Semantic Eval.*, Barcelona, Spain: International Committee on Computational Linguistics, 2020, pp. 105–111.

[53] O. Pražák, P. Přibáň, S. Taylor, and J. Sido, "UWB at SemEval-2020 Task 1: Lexical semantic change detection," in *Proc. 14th Workshop Semantic Eval.*, Barcelona, Spain: International Committee on Computational Linguistics, 2020, pp. 246–254.

[54] N. Arefyev and V. Zhikov, "BOS at SemEval-2020 Task 1: Word sense induction via lexical substitution for lexical semantic change detection," in *Proc. 14th Workshop Semantic Eval.*, Barcelona, Spain: International Committee on Computational Linguistics, 2020, pp. 171–179.

[55] M. Martinc, S. Montariol, E. Zosa, and L. Pivovarova, "Discovery team at SemEval-2020 Task 1: Context-sensitive embeddings not always better than static for semantic change detection," in *Proc. 14th Workshop Semantic Eval.*, Barcelona, Spain: International Committee on Computational Linguistics, 2020, pp. 67–73.

[56] J. Zhou and J. Li, "TemporalTeller at SemEval-2020 Task 1: Unsupervised lexical semantic change detection with temporal referencing," in *Proc. 14th Workshop Semantic Eval.*, Barcelona, Spain: International Committee on Computational Linguistics, 2020, pp. 222–231.

[57] G. Recchia, E. Jones, P. J. Nulty Regan, and P. de Bolla, "Tracing shifting conceptual vocabularies through time," in *Knowledge Engineering and Knowledge Management*, P. F. Ciancarini, M. Poggi, J. H. Zhao, T. Groza, M. C. Suarez-Figueroa, M. d'Aquin, and V. Presutti, Eds. Cham, Switzerland: Springer, 2017, pp. 19–28.

[58] F. D. Zamora-Reina and F. Bravo-Marquez, "DCC-uchile at SemEval-2020 Task 1: Temporal referencing word embeddings," in *Proc. 14th Workshop Semantic Eval.*, Barcelona, Spain: International Committee on Computational Linguistics, 2020, pp. 194–200.

[59] P. Cassotti, A. Caputo, M. Polignano, and P. Basile, "GM-CTSC at SemEval-2020 Task 1: Gaussian mixtures cross temporal similarity clustering," in *Proc. 14th Workshop Semantic Eval.*, Barcelona, Spain: International Committee on Computational Linguistics, 2020, pp. 74–80.

[60] P. Nulty and D. Lillis, "The UCD-Net system at SemEval-2020 Task 1: Temporal referencing with semantic network distances," in *Proc. 14th Workshop Semantic Eval.*, Barcelona, Spain: International Committee on Computational Linguistics, 2020, pp. 119–125.

[61] H. Peng, J. Li, Y. Song, and Y. Liu, "Incrementally learning the Hierarchical Softmax Function for neural language models," in *Proc. 31st AAAI Conf. Artif. Intell.*, Association for the Advancement of Artificial Intelligence Press, 2017, pp. 3267–3273.

[62] J. Kaiser, D. Schlechtweg, S. Papay, and S. Schulte im Walde, "IMS at SemEval-2020 Task 1: How low can you go? Dimensionality in lexical semantic change detection," in *Proc. 14th Workshop Semantic Eval.*, Barcelona, Spain: International Committee on Computational Linguistics, 2020, pp. 81–89.

[63] A. Vaswani *et al.*, "Attention is all you need," in *Proc. 31st Int. Conf. Neural Inf. Process. Syst.*, Red Hook, NY, USA: Curran Associates, 2017, pp. 6000–6010.

[64] K. Erk and S. Padó, "Paraphrase assessment in structured vector space: Exploring parameters and datasets," in *Proc. Workshop Geometrical Models Natural Lang. Semantics*, USA: Association for Computational Linguistics, 2009, pp. 57–65.

[65] M. Baroni, R. Bernardi, N.-Q. Do, and C.-C. Shan, "Entailment above the word level in distributional semantics," in *Proc. 13th Conf. Eur. Chapter Assoc. Comput. Linguistics*, 2012, pp. 23–32.

[66] K. Heylen, T. Wielfaert, D. Speelman, and D. Geeraerts, "Monitoring polysemy: Word space models as a tool for large-scale lexical semantic analysis," *Lingua*, vol. 157, pp. 153–172, 2015.

[67] J. Li and D. Jurafsky, "Do multi-sense embeddings improve natural language understanding?," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, Lisbon, Portugal: Association for Computational Linguistics, 2015, pp. 1722–1732.

[68] Z. Yin, "N-gram-word2vec," 2018, Accessed: Jan. 12, 2021. [Online]. Available: https://github.com/ziyin-dl/ngram-word2vec

[69] Y. N. Lubin, J. Goldberger, and Y. Goldberg, "Aligning vector-spaces with noisy supervised lexicon," in *Proc. Conf. North Amer. Chapter Assoc. Comput. Linguistics, Hum. Lang. Tech.*, Minneapolis, MIN, USA: Association for Computational Linguistics, Jun. 2019, pp. 460–465.

[70] M. Gruppi, P.-Y. Chen, and S. Adali, "Fake it till you make it: Self-supervised semantic shifts for monolingual word embedding tasks," *Proc. AAAI Conf. Artif. Intell.*, vol. 35, no. 14, pp. 12893–12901, May 2021.

[71] W. L. Hamilton, J. Leskovec, and D. Jurafsky, "Cultural shift or linguistic drift? Comparing two computational measures of semantic change," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, Austin, TX, USA: Association for Computational Linguistics, 2016, pp. 2116–2121.

[72] V. Kanjirangat, S. Mitrovic, A. Antonucci, and F. Rinaldi, "SST-BERT at SemEval-2020 Task 1: Semantic shift tracing by clustering in BERT-based embedding spaces," in *Proc. 14th Workshop Semantic Eval.*, Barcelona, Spain: International Committee on Computational Linguistics, 2020, pp. 214–221.

[73] E. Amar and C. Liebeskind, "JCT at SemEval-2020 task 1: Combined semantic vector spaces models for unsupervised lexical semantic change detection," in *Proc. 14th Workshop Semantic Eval.*, Barcelona, Spain: International Committee on Computational Linguistics, 2020, pp. 90–97.

[74] A. Cuba-Gyllensten, E. Gogoulou, A. Ekgren, and M. Sahlgren, "SenseCluster at SemEval-2020 Task 1: Unsupervised lexical semantic change detection," in *Proc. 14th Workshop Semantic Eval.*, Barcelona, Spain: International Committee on Computational Linguistics, 2020, pp. 112–118.

[75] A. Karnysheva and P. Schwarz, "TUE at SemEval-2020 task 1: Detecting semantic change by clustering contextual word embeddings," in *Proc. 14th Workshop Semantic Eval.*, Barcelona, Spain: International Committee on Computational Linguistics, 2020, pp. 232–238.

[76] A. Kutuzov and M. Giulianelli, "UiO-UvA at SemEval-2020 task 1: Contextualised embeddings for lexical semantic change detection," in *Proc. 14th Workshop Semantic Eval.*, Barcelona, Spain: International Committee on Computational Linguistics, 2020, pp. 126–134.

[77] D. Rother, T. Haider, and S. Eger, "CMCE at SemEval-2020 task 1: Clustering on manifolds of contextualized embeddings to detect historical meaning shifts," in *Proc. 14th Workshop Semantic Eval.*, Barcelona, Spain: International Committee on Computational Linguistics, 2020, pp. 187–193.

[78] C. Beck, "DiaSense at SemEval-2020 Task 1: Modeling sense change via pre-trained BERT embeddings," in *Proc. 14th Workshop Semantic Eval.*, Barcelona, Spain: International Committee on Computational Linguistics, 2020, pp. 50–58.

[79] E. Sarsfield and H. Tayyar Madabushi, "UoB at SemEval-2020 Task 1: Automatic identification of novel word senses," in *Proc. 14th Workshop Semantic Eval.*, Barcelona, Spain: International Committee on Computational Linguistics, 2020, pp. 239–245.