Dynamic Contextualized Word Embeddings

Valentin Hofmann[*‡], **Janet B. Pierrehumbert**[†*], **Hinrich Schütze**[‡]

[*]Faculty of Linguistics, University of Oxford
[†]Department of Engineering Science, University of Oxford
[‡]Center for Information and Language Processing, LMU Munich
valentin.hofmann@ling-phil.ox.ac.uk

Abstract

Static word embeddings that represent words by a single vector cannot capture the variability of word meaning in different linguistic and extralinguistic contexts. Building on prior work on contextualized and dynamic word embeddings, we introduce dynamic contextualized word embeddings that represent words as a function of both linguistic and extralinguistic context. Based on a pretrained language model (PLM), dynamic contextualized word embeddings model time and social space jointly, which makes them attractive for a range of NLP tasks involving semantic variability. We highlight potential application scenarios by means of qualitative and quantitative analyses on four English datasets.

1 Introduction

Over the last decade, word embeddings have revolutionized the field of NLP. Traditional methods such as LSA (Deerwester et al., 1990), word2vec (Mikolov et al., 2013a,b), GloVe (Pennington et al., 2014), and fastText (Bojanowski et al., 2017) compute *static* word embeddings, i.e., they represent words as a single vector. From a theoretical standpoint, this way of modeling lexical semantics is problematic since it ignores the variability of word meaning in different linguistic contexts (e.g., polysemy) as well as different extralinguistic contexts (e.g., temporal and social variation).

The first shortcoming was addressed by the introduction of *contextualized* word embeddings that represent words as vectors varying across linguistic contexts. This allows them to capture more complex characteristics of word meaning, including polysemy. Contextualized word embeddings are widely used in NLP, constituting the semantic backbone of pretrained language models (PLMs) such as ELMo (Peters et al., 2018a), BERT (Devlin et al., 2019), GPT-2 (Radford et al., 2019), XLNet (Yang et al., 2019), ELECTRA (Clark et al., 2020), and T5 (Raffel et al., 2020).

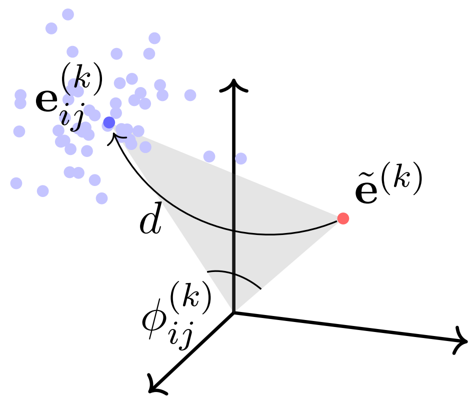


Figure 1: Dynamic contextualized word embeddings. A static embedding $\tilde{\mathbf{e}}^{(k)}$ (●) is mapped to a dynamic embedding $\mathbf{e}_{ij}^{(k)}$ (●) by a function d that takes time and social space into account. The scattered points (●) are contextualized versions of $\mathbf{e}_{ij}^{(k)}$. Variability in $\phi_{ij}^{(k)}$ indicates semantic dynamics across time and social space. The embeddings have 768 dimensions.

A concurrent line of work focused on the second shortcoming of static word embeddings, resulting in various types of *dynamic* word embeddings. Dynamic word embeddings represent words as vectors varying across extralinguistic contexts, in particular time (e.g., Rudolph and Blei, 2018) and social space (e.g., Zeng et al., 2018).

In this paper, we introduce *dynamic contextualized* word embeddings that combine the strengths of contextualized word embeddings with the flexibility of dynamic word embeddings. Dynamic contextualized word embeddings mark a departure from existing contextualized word embeddings (which are not dynamic) as well as existing dynamic word embeddings (which are not contextualized). Furthermore, as opposed to all existing dynamic word embedding types, they represent time and social space jointly.

Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, pages 6970–6984
August 1–6, 2021. ©2021 Association for Computational Linguistics

While our general framework for training dynamic contextualized word embeddings is model-agnostic (Figure 1), we present a version using a PLM (BERT) as the contextualizer, which allows for an easy integration within existing architectures. Dynamic contextualized word embeddings can serve as an analytical tool (e.g., to track the emergence and spread of semantic changes in online communities) or be employed for downstream tasks (e.g., to build temporally and socially aware text classification models), making them beneficial for various areas in NLP that face semantic variability. We illustrate application scenarios by performing exploratory experiments on English data from ArXiv, Ciao, Reddit, and YELP.

Contributions. We introduce dynamic contextualized word embeddings that represent words as a function of both linguistic and extralinguistic context. Based on a PLM, dynamic contextualized word embeddings model time and social space jointly, which makes them attractive for a range of NLP tasks. We showcase potential applications by means of qualitative and quantitative analyses.[1]

2 Related Work

2.1 Contextualized Word Embeddings

The distinction between the non-contextualized core meaning of a word and the senses that are realized in specific linguistic contexts lies at the heart of lexical-semantic scholarship (Geeraerts, 2010), going back to at least Paul (1880). In NLP, this is reflected by contextualized word embeddings that map type-level representations to token-level representations as a function of the linguistic context (McCann et al., 2017). As part of PLMs (Peters et al., 2018a; Devlin et al., 2019; Radford et al., 2019; Yang et al., 2019; Clark et al., 2020; Raffel et al., 2020), contextualized word embeddings have led to substantial performance gains on a variety of tasks compared to static word embeddings that only have type-level representations (Deerwester et al., 1990; Mikolov et al., 2013a,b; Pennington et al., 2014; Bojanowski et al., 2017).

Since their introduction, several studies have analyzed the linguistic properties of contextualized word embeddings (Peters et al., 2018b; Goldberg, 2019; Hewitt and Manning, 2019; Jawahar et al., 2019; Lin et al., 2019; Liu et al., 2019; Tenney et al., 2019; Edmiston, 2020; Ettinger, 2020; Hofmann et al., 2020; Rogers et al., 2020). Regarding lexical semantics, this line of research has shown that contextualized word embeddings are more context-specific in the upper layers of a contextualizer (Ethayarajh, 2019; Mickus et al., 2020; Vulić et al., 2020) and represent different word senses as separated clusters (Peters et al., 2018a; Coenen et al., 2019; Wiedemann et al., 2019).

2.2 Dynamic Word Embeddings

The meaning of a word can also vary across extralinguistic contexts such as time (Bybee, 2015; Koch, 2016) and social space (Robinson, 2010, 2012; Geeraerts, 2018). To capture these phenomena, various types of dynamic word embeddings have been proposed: diachronic word embeddings for temporal semantic change (Bamler and Mandt, 2017; Rosenfeld and Erk, 2018; Rudolph and Blei, 2018; Yao et al., 2018; Gong et al., 2020) and personalized word embeddings for social semantic variation (Zeng et al., 2017, 2018; Oba et al., 2019; Welch et al., 2020a,b; Yao et al., 2020). Other studies have demonstrated that performance on a diverse set of tasks can be increased by including temporal (Jaidka et al., 2018; Lukes and Søgaard, 2018) and social information (Amir et al., 2016; Hamilton et al., 2016a; Yang et al., 2016; Yang and Eisenstein, 2017; Hazarika et al., 2018; Mishra et al., 2018; del Tredici et al., 2019b; Li and Goldwasser, 2019; Mishra et al., 2019).

The relevance of dynamic (specifically diachronic) word embeddings is also reflected by the emergence of lexical semantic change detection as an established task in NLP (Kutuzov et al., 2018; Schlechtweg et al., 2018; Tahmasebi et al., 2018; Dubossarsky et al., 2019; Schlechtweg et al., 2019; Asgari et al., 2020; Pömsl and Lyapin, 2020; Pražák et al., 2020; Schlechtweg and Schulte im Walde, 2020; Schlechtweg et al., 2020). Besides dynamic word embeddings, many studies on lexical semantic change detection use methods based on static word embeddings (Kim et al., 2014; Kulkarni et al., 2015), e.g., the alignment of static word embedding spaces (Hamilton et al., 2016b). However, such approaches come at the cost of modeling disadvantages (Bamler and Mandt, 2017).

Sociolinguistics has shown that temporal and social variation in language are tightly interwoven: innovations such as a new word sense in the case of lexical semantics spread through the language community along social ties (Milroy, 1980, 1992;

[1] We make our code publicly available at `https://github.com/valentinhofmann/dcwe`.

Labov, 2001; Pierrehumbert, 2012). However, most proposed dynamic word embedding types cannot capture more than one dimension of variation. Recently, a few studies have taken first steps in this direction by using genre information within a Bayesian model of semantic change (Frermann and Lapata, 2016; Perrone et al., 2019) and including social variables in training diachronic word embeddings (Jawahar and Seddah, 2019). In addition, to capture the full range of lexical-semantic variability, dynamic word embeddings should also be contextualized. Crucially, while contextualized word embeddings have been used to investigate semantic change (Giulianelli, 2019; Hu et al., 2019; Giulianelli et al., 2020; Kutuzov and Giulianelli, 2020; Martinc et al., 2020a,b), the word embeddings employed in these studies are not dynamic, i.e., they represent a word in a specific linguistic context by the same contextualized word embedding independent of extralinguistic context or are fit to different time periods as separate models.[2]

3 Model

3.1 Model Overview

Given a sequence of words $X = \left[x^{(1)}, \ldots, x^{(K)}\right]$ and corresponding non-contextualized embeddings $E = \left[\mathbf{e}^{(1)}, \ldots, \mathbf{e}^{(K)}\right]$, contextualizing language models compute the contextualized embedding of a particular word $x^{(k)}$, $\mathbf{h}^{(k)}$, as a function c of its non-contextualized embedding, $\mathbf{e}^{(k)}$, and the non-contextualized embeddings of words in the left context $X^{(<k)}$ and the right context $X^{(>k)}$,[3]

$$\mathbf{h}^{(k)} = c\left(\mathbf{e}^{(k)}, E^{(<k)}, E^{(>k)}\right). \tag{1}$$

Crucially, while $\mathbf{h}^{(k)}$ is a token-level representation, $\mathbf{e}^{(k)}$ is a type-level representation and is modeled as a simple embedding look-up. Here, in order to take the variability of word meaning in different extralinguistic contexts into account, we depart from this practice and model $\mathbf{e}^{(k)}$ as a function d that depends not only on the identity of $x^{(k)}$ but also on the social context s_i and the temporal context t_j in which the sequence X occurred,

$$\mathbf{e}_{ij}^{(k)} = d\left(x^{(k)}, s_i, t_j\right). \tag{2}$$

[2]It is interesting to notice that contextualized word embeddings so far have performed worse than non-contextualized word embeddings on the task of lexical semantic change detection (Kaiser et al., 2020; Schlechtweg et al., 2020).

[3]Some contextualizing language models such as GPT-2 (Radford et al., 2019) only operate on $X^{(<k)}$.

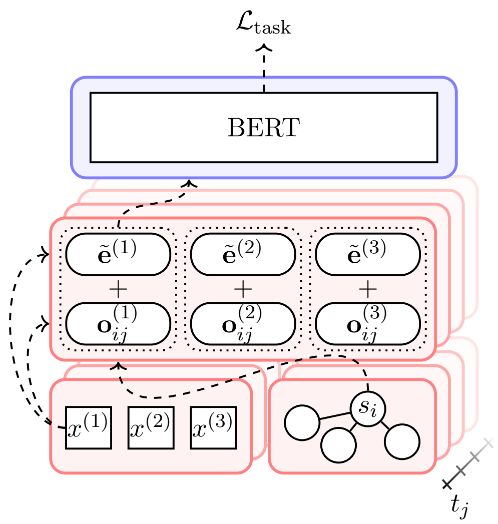


Figure 2: Model architecture. Words are mapped to dynamic embeddings by the parts of the dynamic component ( ), which are then contextualized by the contextualizer ( ). The output of the contextualizer is used to compute the task-specific loss $\mathcal{L}_{\text{task}}$.

Dynamic contextualized word embeddings are hence computed in two stages: words are first mapped to dynamic type-level representations by d and then to contextualized token-level representations by c (Figures 1 and 2). This two-stage structure follows work in cognitive science and linguistics that indicates that extralinguistic information is processed before linguistic information by human speakers (Hay et al., 2006).

Since many words in the core vocabulary are semantically stable across social and temporal contexts, we place a Gaussian prior on $\mathbf{e}_{ij}^{(k)}$,

$$\mathbf{e}_{ij}^{(k)} \sim \mathcal{N}\left(\tilde{\mathbf{e}}^{(k)}, \lambda_a^{-1}\mathbf{I}\right), \tag{3}$$

where $\tilde{\mathbf{e}}^{(k)}$ denotes a non-dynamic representation of $x^{(k)}$. Combining Equations 2 and 3, we write the function d as

$$d\left(x^{(k)}, s_i, t_j\right) = \tilde{\mathbf{e}}^{(k)} + \mathbf{o}_{ij}^{(k)}, \tag{4}$$

where $\mathbf{o}_{ij}^{(k)}$ denotes the vector offset from $x^{(k)}$'s non-dynamic embedding $\tilde{\mathbf{e}}^{(k)}$, which is stable across social and temporal contexts, to its dynamic embedding $\mathbf{e}_{ij}^{(k)}$, which is specific to s_i and t_j. The distribution of $\mathbf{o}_{ij}^{(k)}$ then follows a Gaussian with

$$\mathbf{o}_{ij}^{(k)} \sim \mathcal{N}\left(\mathbf{0}, \lambda_a^{-1}\mathbf{I}\right). \tag{5}$$

We enforce Equation 5 by including a regularization term in the objective function (Section 3.4).

3.2 Contextualizing Component

We leverage a PLM for the function c, specifically BERT (Devlin et al., 2019). Denoting with E_{ij} the sequence of dynamic embeddings corresponding to X in s_i and t_j, the dynamic version of Equation 1 becomes

$$\mathbf{h}_{ij}^{(k)} = \text{BERT}\left(\mathbf{e}_{ij}^{(k)}, E_{ij}^{(<k)}, E_{ij}^{(>k)}\right). \quad (6)$$

We also use BERT, specifically its pretrained input embeddings, to initialize the non-dynamic embeddings $\tilde{\mathbf{e}}^{(k)}$, which are summed with the vector offsets $\mathbf{o}_{ij}^{(k)}$ (Equation 4) and fed into BERT.

Using a PLM for c has the advantage of making it easy to employ dynamic contextualized word embeddings for downstream tasks by adding a task-specific layer on top of the PLM.

3.3 Dynamic Component

We model the vector offset $\mathbf{o}_{ij}^{(k)}$ as a function of the word $x^{(k)}$, which we represent by its non-dynamic embedding $\tilde{\mathbf{e}}^{(k)}$, as well as the social context s_i, which we represent by a time-specific embedding $\mathbf{s}_{ij}$. We use BERT's pretrained input embeddings for $\tilde{\mathbf{e}}^{(k)}$.[4] We combine these representations in a time-specific feed-forward network,

$$\mathbf{o}_{ij}^{(k)} = \text{FFN}_j\left(\tilde{\mathbf{e}}^{(k)} \,\|\, \mathbf{s}_{ij}\right), \quad (7)$$

where $\|$ denotes concatenation. To compute the social embedding $\mathbf{s}_{ij}$, we follow common practice in the computational social sciences and represent the social community as a graph $\mathcal{G} = (\mathcal{S}, \mathcal{E})$, where $\mathcal{S}$ is the set of social units s_i, and $\mathcal{E}$ is the set of edges between them (Section 4). We use a time-specific graph attention network (GAT) as proposed by Veličković et al. (2018) to encode $\mathcal{G}$,[5]

$$\mathbf{s}_{ij} = \text{GAT}_j\left(\tilde{\mathbf{s}}_i, \mathcal{G}\right). \quad (8)$$

We initialize $\tilde{\mathbf{s}}_i$ with node2vec (Grover and Leskovec, 2016) embeddings.

To model the temporal drift of the dynamic embeddings $\mathbf{e}_{ij}^{(k)}$, we follow previous work on dynamic word embeddings (Bamler and Mandt, 2017; Rudolph and Blei, 2018) and impose a random walk prior over $\mathbf{o}_{ij}^{(k)}$,

$$\mathbf{o}_{ij}^{(k)} \sim \mathcal{N}\left(\mathbf{o}_{ij'}^{(k)}, \lambda_w^{-1}\mathbf{I}\right), \quad (9)$$

with $j' = j - 1$. This type of Gaussian process is known as Ornstein-Uhlenbeck process (Uhlenbeck and Ornstein, 1930) and is commonly used to model time series (Roberts et al., 2013). The random walk prior enforces that the dynamic embeddings $\mathbf{e}_{ij}^{(k)}$ change smoothly over time.

3.4 Model Training

The combination with BERT makes dynamic contextualized word embeddings easily applicable to different tasks by adding a task-specific layer on top of the contextualizing component. For training the model, the overall loss is

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{task}} + \mathcal{L}_{\text{prior}_a} + \mathcal{L}_{\text{prior}_w}, \quad (10)$$

where $\mathcal{L}_{\text{task}}$ is the task-specific loss, and $\mathcal{L}_{\text{prior}_a}$ and $\mathcal{L}_{\text{prior}_w}$ are the regularization terms that impose the anchoring and random walk priors on the type-level offset vectors,

$$\mathcal{L}_{\text{prior}_a} = \frac{\lambda_a}{K}\sum_{k=1}^{K}\|\mathbf{o}_{ij}^{(k)}\|_2^2 \quad (11)$$

$$\mathcal{L}_{\text{prior}_w} = \frac{\lambda_w}{K}\sum_{k=1}^{K}\|\mathbf{o}_{ij}^{(k)} - \mathbf{o}_{ij'}^{(k)}\|_2^2. \quad (12)$$

It is common practice to set $\lambda_a \ll \lambda_w$ (Bamler and Mandt, 2017; Rudolph and Blei, 2018). Here, we set $\lambda_a = 10^{-3} \cdot \lambda_w$, which reduces the number of tunable hyperparameters. We place the priors only on frequent words in the vocabulary (Section 5.1), taking into account the observation that the vocabulary core constitutes the best basis for dynamic word embeddings (Hamilton et al., 2016b).

4 Data

We fit dynamic contextualized word embeddings to four datasets with different linguistic, social, and temporal characteristics, which allows us to investigate factors impacting their utility. Each dataset $\mathcal{D}$ consists of a set of texts (e.g., reviews) written by a set of social units $\mathcal{S}$ (e.g., users) over a sequence of time periods $\mathcal{T}$ (e.g., years). Furthermore, the social units are connected by a set of edges $\mathcal{E}$ within a social network $\mathcal{G}$. Table 1 provides summary statistics of the four datasets.

ArXiv. ArXiv is an open-access distribution service for scientific articles. Recently, a dataset of all papers published on ArXiv with corresponding metadata was released.[6] For this study, we

[4] We also tried to learn separate embeddings in the dynamic component, but this led to worse performance.

[5] We also tried a model with a feed-forward network instead of graph attention, but it consistently performed worse.

[6] `https://www.kaggle.com/Cornell-University/arxiv`

		Linguistic		Social						Temporal			
Dataset	$\|\mathcal{D}\|$	Unit	$\mu_{\|X\|}$	Unit	$\|\mathcal{S}\|$	$\|\mathcal{E}\|$	μ_d	μ_π	ρ	Unit	$\|\mathcal{T}\|$	t_1	$t_{\|\mathcal{T}\|}$
ArXiv	972,369	Abstract	118.10	Subject	535	5,165	19.34	3.48	.036	Year	20	[01/]2001	[10/]2020
Ciao	269,807	Review	684.68	User	10,880	129,900	18.20	3.65	.002	Year	12	[05/]2000	[09/]2011
Reddit	915,663	Comment	43.50	Subreddit	5,728	61,796	23.99	4.69	.005	Month	8	09/2019	04/2020
YELP	795,661	Review	151.59	User	5,203	223,254	45.17	2.83	.009	Year	10	[01/]2010	[12/]2019

Table 1: Dataset statistics. $|\mathcal{D}|$: number of data points; $\mu_{|X|}$: average number of tokens per text; $|\mathcal{S}|$: number of nodes in network; $|\mathcal{E}|$: number of edges; μ_d: average node degree; μ_π: average shortest path length between two nodes; ρ: network density; $|\mathcal{T}|$: number of time points; t_1: first time point; $t_{|\mathcal{T}|}$: last time point. In cases where years are the temporal unit, we also provide the first and last month included in the data.

use ArXiv's subject classes (e.g., `cs.CL`) as social units and extract the abstracts of papers published between 2001 and 2020 for subjects with at least 100 publications in that time.[7] To create the network, we measure the overlap in authors between subject classes as the Jaccard similarity of corresponding author sets, resulting in a similarity matrix $\mathbf{S}$. Based on $\mathbf{S}$, we define the adjacency matrix $\mathbf{G}$ of $\mathcal{G}$, whose elements are

$$G_{ij} = \lceil S_{ij} - \theta \rceil, \tag{13}$$

i.e., there is an edge between subject classes i and j if the Jaccard similarity of author sets is greater than θ. We set θ to 0.01.[8]

Ciao. Ciao is a product review site on which users can mark explicit trust relations towards other users (e.g., if they find their reviews helpful). A dataset containing reviews covering the time period from 2000 to 2011 has been made publicly available (Tang et al., 2012).[9] We use the trust relations to create a directed graph. Since we also perform sentiment analysis on the dataset, we follow Yang and Eisenstein (2017) in converting the five-star rating range into two classes by discarding three-star reviews and treating four/five stars as positive and one/two stars as negative.

Reddit. Reddit is a social media platform hosting discussions about a variety of topics. It is divided into smaller communities, so-called subreddits, which have been shown to be highly conducive to linguistic dynamics (del Tredici and Fernández, 2018; del Tredici et al., 2019a). A full dump of public Reddit posts is available online.[10] We retrieve all comments between September 2019 and April 2020, which allows us to examine the effects of the rising Covid-19 pandemic on lexical usage patterns. We remove subreddits with fewer than 10,000 comments in the examined time period and sample 20 comments per subreddit and month. For each subreddit, we compute the set of users with at least 10 comments in the examined time period. Based on this, we use the same strategy as for ArXiv to create a network based on user overlap.

YELP. Similarly to Ciao, YELP is a product review site on which users can mark explicit friendship relations. A subset of the data has been released online.[11] We use the friendship relations to create a directed graph between users. Since we also use the dataset for sentiment analysis, we again discard three-star reviews and convert the five-star rating range into two classes.

The fact that the datasets differ in terms of their social and temporal characteristics allows us to examine which factors impact the utility of dynamic contextualized word embeddings. We highlight, e.g., that the datasets differ in the nature of their social units, cover different time periods, and exhibit different levels of temporal granularity. We randomly split all datasets into 70% training, 10% development, and 20% test. We apply stratified sampling to make sure the model sees data from all time points during training. See Appendix A.1 for details about data preprocessing.

5 Experiments

5.1 Embedding Training

We fit dynamic contextualized word embeddings to all four datasets, using BERT$_{\text{BASE}}$ (uncased) as the contextualizer and masked language modeling as the training objective (Devlin et al., 2019), i.e., we

[7] We treat subject class combinations passing the frequency threshold (e.g., `cs.CL&cs.AI`) as individual units.

[8] We tried other values of θ, but the results were similar.

[9] `https://www.cse.msu.edu/~tangjili/trust.html`

[10] `https://files.pushshift.io/reddit/comments`

[11] `https://www.yelp.com/dataset`

	ArXiv		Ciao		Reddit		YELP	
Model	Dev	Test	Dev	Test	Dev	Test	Dev	Test
DCWE	3.521	3.513	5.920	5.902	9.480	9.596	4.717	4.720
CWE	3.523	3.530	5.922	5.910	9.580	9.555	4.714	4.723

Table 2: Masked language modeling perplexity on the four datasets (lower is better). DCWE: dynamic contextualized word embeddings; CWE: contextualized word embeddings. The better score per column (highlighted in gray) is underlined if it is significantly ($p < .01$) better as shown by a Wilcoxon signed-rank test.

add a language modeling head on top of BERT.[12] To estimate the goodness of fit, we measure masked language modeling perplexity and compare against finetuned (non-dynamic) contextualized word embeddings, specifically $\text{BERT}_{\text{BASE}}$ (uncased). See Appendix A.2 for details about implementation, hyperparameter tuning, and runtime.

Dynamic contextualized word embeddings (DCWE) yield fits to the data similar to and (sometimes significantly) better than non-dynamic contextualized word embeddings (CWE), which indicates that they successfully combine extralinguistic with linguistic information (Table 2).[13]

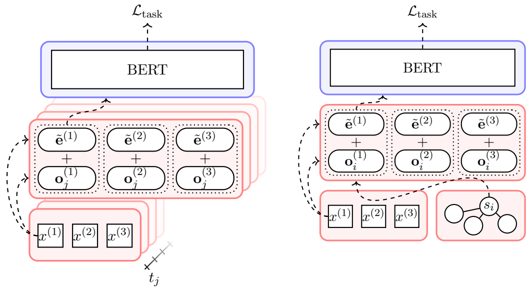
5.2 Ablation Study

To examine the relative importance of temporal and social information for dynamic contextualized word embeddings, we perform two experiments in which we ablate social context and time (Figure 3). In social ablation (SA), we train dynamic contextualized word embeddings where the vector offset depends only on word identity and time, not social context, keeping the random walk prior between subsequent time slices. In temporal ablation (TA), we use one social component for all time slices. See Appendix A.3 for details about implementation, hyperparameter tuning, and runtime.

Temporal ablation has more severe consequences than social ablation (Table 3). On Ciao, the social component does not yield better fits on the data at all, which might be related to the fact that many users in this dataset only have one review, and that its social network has the lowest density as well as the smallest average node degree out of all considered datasets (Table 1).

[12] For a given dataset, we only compute dynamic embeddings for tokens in BERT's input vocabulary that are among the 100,000 most frequent words. For less frequent tokens, we input the non-dynamic BERT embedding.

[13] Statistical significance is tested with a Wilcoxon signed-rank test (Wilcoxon, 1945; Dror et al., 2018).



(a) Social ablation (b) Temporal ablation

Figure 3: Models for ablation study. In social ablation, the vector offset only depends on word identity and time, not social context. In temporal ablation, there is only one social component for all time slices.

	ArXiv		Ciao		Reddit		YELP	
Model	Dev	Test	Dev	Test	Dev	Test	Dev	Test
DCWE	3.521	3.513	5.920	5.902	9.480	9.596	4.717	4.720
SA	3.517	3.515	5.919	5.899	9.620	9.631	4.725	4.723
TA	3.534	3.541	5.924	5.931	9.598	9.612	4.726	4.734

Table 3: Masked language modeling perplexity on the four datasets in ablation study (lower is better). DCWE: dynamic contextualized word embeddings; SA: social ablation; TA: temporal ablation. The best score per column (highlighted in gray) is underlined if it is significantly ($p < .01$) better than the second-best score as shown by a Wilcoxon signed-rank test.

5.3 Qualitative Analysis

Do dynamic contextualized word embeddings indeed capture interpretable dynamics in word meaning? To examine this question qualitatively, we define as $\text{sim}_{ij}^{(k)}$ the cosine similarity between the non-dynamic embedding of $x^{(k)}$, $\tilde{\mathbf{e}}^{(k)}$, and the dynamic embeddings of $x^{(k)}$ given social and temporal contexts s_i and t_j, $\mathbf{e}_{ij}^{(k)}$,

$$\text{sim}_{ij}^{(k)} = \cos \phi_{ij}^{(k)}, \tag{14}$$

where $\phi_{ij}^{(k)}$ is the angle between $\tilde{\mathbf{e}}^{(k)}$ and $\mathbf{e}_{ij}^{(k)}$ (Figure 1).[14] To find words with a high degree of variability, we compute the standard deviation of $\text{sim}_{ij}^{(k)}$ based on all s_i and t_j in which a given word $x^{(k)}$ occurs in the data,

$$\sigma_{\text{sim}}^{(k)} = \sigma\left(\{\text{sim}_{ij}^{(k)} \,|\, (x^{(k)}, s_i, t_j) \in \mathcal{D}\}\right), \tag{15}$$

where we take the development set for $\mathcal{D}$. Looking at the top-ranked words according to $\sigma_{\text{sim}}^{(k)}$, we observe that they exhibit pronounced

[14] In cases where $x^{(k)}$ is split into several WordPiece tokens by BERT, we follow previous work (Pinter et al., 2020; Sia et al., 2020) and average the subword embeddings.

	Context for $\text{sim}_{ij}^{(k)} > \mu_{\text{sim}}^{(k)}$		Context for $\text{sim}_{ij}^{(k)} < \mu_{\text{sim}}^{(k)}$	
Word	Extralinguistic	Linguistic	Extralinguistic	Linguistic
"isolating"	`r/SAHP` 12/19	*It's really hard to explain to other people how isolating and exhausting being a SAHP can be.*	`r/Asthma` 03/20	*I wish I knew if I'd had covid so that I could stop self isolating and instead volunteer in my community.*
"testing"	`r/VJoeShows` 04/20	*Testing a photocell light fixture during the day is easy when you know how. This is what this DIY video is about.*	`r/vancouver` 03/20	*Testing is not required if a patient has no symptoms, mild symptoms, or is a returning traveller and is isolating at home.*

Table 4: Examples of dynamics in word meaning during the Covid-19 pandemic. The table lists example words with top-ranked values of $\sigma_{\text{sim}}^{(k)}$, i.e., they exhibit a high degree of extralinguistically-driven semantic dynamics.

extralinguistically-driven semantic dynamics in the data. For Reddit, e.g., many of the top-ranked words have experienced a sudden shift in their dominant sense during the Covid-19 pandemic such as "isolating" and "testing" (Table 4). Social and temporal contexts in which the sense related to Covid-19 is dominant have smaller values of $\text{sim}_{ij}^{(k)}$ (i.e., the cosine distance is larger) than the ones in which the more general sense is dominant. Such short-term semantic shifts, which have attracted growing interest in NLP recently (Stewart et al., 2017; del Tredici et al., 2019a; Powell and Sentz, 2020), can result in lasting semantic narrowing if speakers become reluctant to use the word outside of the more specialized sense (Anttila, 1989; Croft, 2000; Robinson, 2012; Bybee, 2015).

Thus, the qualitative analysis suggests that the dynamic component indeed captures extralinguistically-driven variability in word meaning. In Sections 5.4 and 5.5, we will demonstrate by means of two example applications how this property can be beneficial in practice.

5.4 Exploration 1: Semantic Diffusion

We will now provide a more in-depth analysis of social and temporal dynamics in word meaning to showcase the potential of dynamic contextualized word embeddings as an analytical tool. Specifically, we will analyze how changes in the dominant sense of a word diffuse through the social networks of ArXiv and Reddit. For ArXiv, we will examine the deep learning sense of the word "network". For Reddit, we will focus on the medical sense of the word "mask". We know that these senses have become more widespread over the last few years (ArXiv) and months (Reddit), but we want to test if dynamic contextualized word embeddings can capture this spread, and if they allow us to gain new insights about the spread of semantic associations through social networks in general.

To perform this analysis, let $r_{ij}^{(k,k')}$ be the rank of $x^{(k')}$'s embedding among the N nearest neighbors of $x^{(k)}$'s embedding, given social and temporal contexts s_i and t_j. We then define as

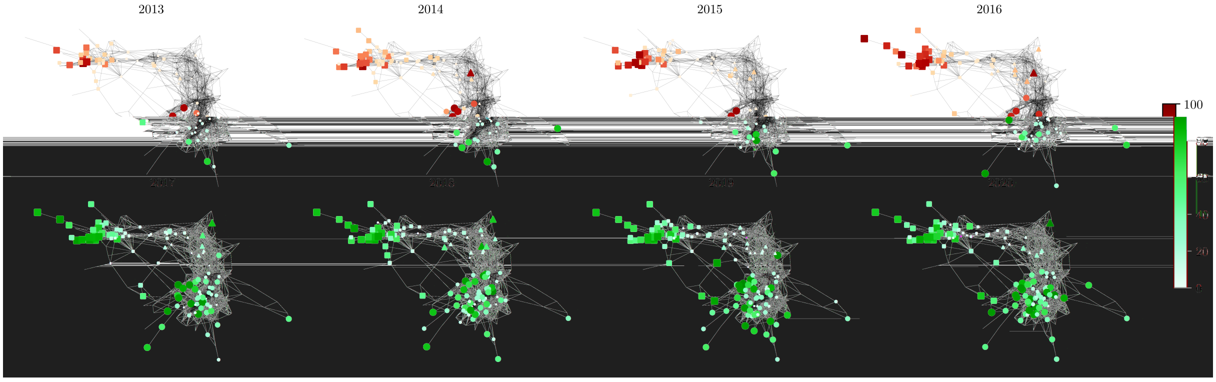
$$\hat{r}_{ij}^{(k,k')} = N - r_{ij}^{(k,k')} + 1 \tag{16}$$

a semantic similarity score between $x^{(k)}$ and $x^{(k')}$. $\hat{r}_{ij}^{(k,k')}$ is maximal when $x^{(k')}$'s embedding is closest to $x^{(k)}$'s embedding. We set $\hat{r}_{ij}^{(k,k')} = 0$ if $x^{(k')}$ is not among the N nearest neighbors of $x^{(k)}$. We set $N = 100$.

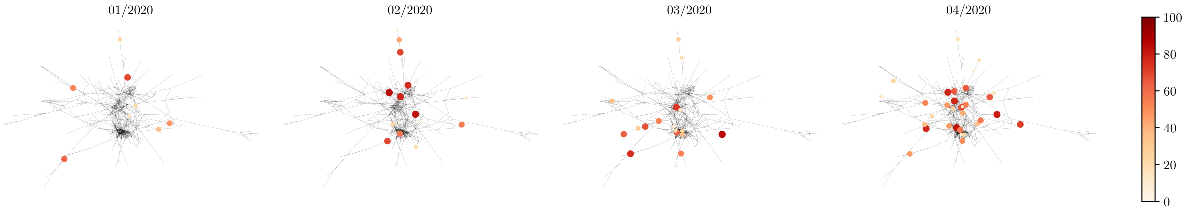
Using $\hat{r}_{ij}^{(k,k')}$, we measure dynamics in the semantic similarity between "network" and "learning" (representing the deep learning sense of "network") as well as "mask" and "vaccine" (representing the medical sense of "mask"). For all social and temporal contexts in which "network" and "mask" occur, we compute $\hat{r}_{ij}^{(k,k')}$ between their socially and temporally dynamic embeddings on the one hand and time-specific centroids of "learning" and "vaccine" averaged over social contexts on the other, employing contextualized versions of the dynamic embeddings.[15] In cases where "network" or "mask" occur more than once in a certain social and temporal context, we take the mean of $\hat{r}_{ij}^{(k,k')}$.

The dynamics of $\hat{r}_{ij}^{(k,k')}$ reflect how the changes in the dominant sense of "network" and "mask" spread through the social networks (Figure 4). For "network", we see that the deep learning sense was already present in computer science and physics in 2013, where neural networks have been used since the 1980s. It then gradually spread from these two epicenters, with a major intensification after 2016. For "mask", we also see a gradual diffusion, with a major intensification after 03/2020.

[15] We average the first six layers of the contextualizer since they have been shown to contain the core of lexical and semantic information (Vulić et al., 2020).



(a) $\hat{r}_{ij}^{(k,k')}$ for “network” and “learning” in ArXiv



(b) $\hat{r}_{ij}^{(k,k')}$ for “mask” and “vaccine” in Reddit

Figure 4: Spread of changes in the dominant sense through the social network. The figure shows dynamics in $\hat{r}_{ij}^{(k,k')}$, a score for semantic similarity between 0 (no similarity) and 100 (very similar), for “network” and “learning” in ArXiv as well as “mask” and “vaccine” in Reddit. The different node shapes in the ArXiv network represent the three major ArXiv subject classes: computer science (square), mathematics (triangle), and physics (circle). For “network”, the change towards the deep learning sense spread gradually from computer science and physics. For “mask”, the change towards the medical sense also spread gradually, with a major intensification after 03/2020.

On what paths do new semantic associations spread through the social network? In complex systems theory, there are two basic types of random motion on networks: random walks, which consist of a series of consecutive random steps, and random flights, where step lengths are drawn from the Lévy distribution (Masuda et al., 2017). To probe whether there is a dominant type of spread for the two examples, we compute for each time slice t_j what proportion of nodes that have $\hat{r}_{ij}^{(k,k')} > 0$ for the first time at t_j (i.e., the change in the dominant sense has just arrived) are neighbors of nodes that already had $\hat{r}_{ij}^{(k,k')} > 0$ before t_j. This analysis shows that random walks are the dominant type of spread for “network”, but random flights for “mask” (Figure 5). Intuitively, it makes sense that a technical concept such as neural networks spreads through the direct contact of collaborating scientists rather than through more distant forms of reception (e.g., the reading of articles). In the case of facial masks, on the other hand, the exogenous factor of the worsening Covid-19 pandemic and the accompanying publicity was a driver of semantic dynamics irrespective of node position.

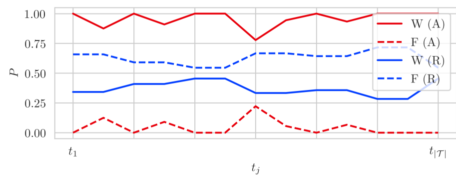


Figure 5: Types of semantic diffusion in ArXiv (A) and Reddit (R). The figure shows for each time t_j the probability that a node having the new sense for the first time is the neighbor of a node that already had it previously (walk, W) as opposed to cases where none of its neighbors had it previously (flight, F).

5.5 Exploration 2: Sentiment Analysis

As a second testbed, we apply dynamic contextualized word embeddings on a task for which social and temporal information is known to be important (Yang and Eisenstein, 2017): sentiment analysis. We use the Ciao and YELP datasets and train dynamic contextualized word embeddings by adding a two-layer feed-forward network on top of $\text{BERT}_{\text{BASE}}$ (uncased) and finetuning it for the task of sentiment classification.[16] We again compare

[16]We finetune directly on sentiment analysis without prior finetuning on masked language modeling.

	Ciao		YELP	
Model	Dev	Test	Dev	Test
DCWE	.894	.896	.969	.968
CWE	.889	.890	.967	.966

Table 5: F1 score on sentiment analysis (higher is better). DCWE: dynamic contextualized word embeddings; CWE: contextualized word embeddings. The better score per column (highlighted in gray) is underlined if it is significantly ($p < .01$) better as shown by a McNemar's test for binary data.

against contextualized word embeddings, specifically $\text{BERT}_{\text{BASE}}$ (uncased), which is finetuned without the dynamic component. See Appendix A.4 for details about implementation, hyperparameter tuning, and runtime.

Dynamic contextualized word embeddings achieve slight but significant improvements over the already strong performance of non-dynamic BERT (Table 5).[17] This provides further evidence that infusing social and temporal information on the lexical level can be useful for NLP tasks.

6 Conclusion

We have introduced dynamic contextualized word embeddings that represent words as a function of both linguistic and extralinguistic context. Based on a PLM, specifically BERT, dynamic contextualized word embeddings model time and social space jointly, which makes them advantageous for various areas in NLP. We have trained dynamic contextualized word embeddings on four datasets and showed that they are capable of tracking social and temporal variability in word meaning. Besides serving as an analytical tool, dynamic contextualized word embeddings can also be of benefit for downstream tasks such as sentiment analysis.

Acknowledgements

This work was funded by the European Research Council (#740516) as well as the Engineering and Physical Sciences Research Council (EP/T023333/1). The first author was also supported by the German Academic Scholarship Foundation and the Arts and Humanities Research Council. We thank the anonymous reviewers for their detailed and extremely helpful comments.

[17]Statistical significance is tested with a McNemar's test for binary data (McNemar, 1947; Dror et al., 2018)

References

Silvio Amir, Byron C. Wallace, Hao Lyu, Paula Carvalho, and Mário J. Silva. 2016. Modelling context with user embeddings for sarcasm detection in social media. In *Conference on Computational Natural Language Learning (CoNLL) 20*.

Raimo Anttila. 1989. *Historical and comparative linguistics*. John Benjamins, Amsterdam.

Ehsaneddin Asgari, Christoph Ringlstetter, and Hinrich Schütze. 2020. EmbLexChange at SemEval-2020 task 1: Unsupervised embedding-based detection of lexical semantic changes. In *International Workshop on Semantic Evaluation (SemEval) 2020*.

Robert Bamler and Stephan Mandt. 2017. Dynamic word embeddings. In *International Conference on Machine Learning (ICML) 34*.

Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2017. Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, 5:135–146.

Joan Bybee. 2015. *Language change*. Cambridge University Press, Cambridge, UK.

Kevin Clark, Minh-Thang Luong, Quoc V. Le, and Christopher D. Manning. 2020. ELECTRA: Pre-training text encoders as discriminators rather than generators. In *International Conference on Learning Representations (ICLR) 8*.

Andy Coenen, Emily Reif, Ann Yuan, Been Kim, Adam Pearce, Fernanda Viégas, and Martin Wattenberg. 2019. Visualizing and measuring the geometry of BERT. In *Advances in Neural Information Processing Systems (NeurIPS) 33*.

William Croft. 2000. *Explaining language change: An evolutionary approach*. Pearson, Harlow, UK.

Scott Deerwester, Susan T. Dumais, George Furnas, Thomas Landauer, and Richard Harshman. 1990. Indexing by latent semantic analysis. *Journal of the American Society for Information Science*, 41(6):391–407.

Marco del Tredici and Raquel Fernández. 2018. The road to success: Assessing the fate of linguistic innovations in online communities. In *International Conference on Computational Linguistics (COLING) 27*.

Marco del Tredici, Raquel Fernández, and Gemma Boleda. 2019a. Short-term meaning shift: A distributional exploration. In *Annual Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL HTL) 2019*.

Marco del Tredici, Diego Marcheggiani, Sabine Schulte im Walde, and Raquel Fernández. 2019b. You shall know a user by the company it keeps: Dynamic representations for social media users in nlp.

In *Conference on Empirical Methods in Natural Language Processing (EMNLP) 2019*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Annual Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL HTL) 2019*.

Jesse Dodge, Suchin Gururangan, Dallas Card, Roy Schwartz, and Noah A. Smith. 2019. Show your work: Improved reporting of experimental results. In *Conference on Empirical Methods in Natural Language Processing (EMNLP) 2019*.

Rotem Dror, Gili Baumer, Segev Shlomov, and Roi Reichart. 2018. The hitchhiker's guide to testing statistical significance in natural language processing. In *Annual Meeting of the Association for Computational Linguistics (ACL) 56*.

Haim Dubossarsky, Simon Hengchen, Nina Tahmasebi, and Dominik Schlechtweg. 2019. Time-out: Temporal referencing for robust modeling of lexical semantic change. In *Annual Meeting of the Association for Computational Linguistics (ACL) 57*.

Daniel Edmiston. 2020. A systematic analysis of morphological content in BERT models for multiple languages. In *arXiv 2004.03032*.

Kawin Ethayarajh. 2019. How contextual are contextualized word representations? comparing the geometry of BERT, ELMo, and GPT-2 embeddings. In *Conference on Empirical Methods in Natural Language Processing (EMNLP) 2019*.

Allyson Ettinger. 2020. What BERT is not: Lessons from a new suite of psycholinguistic diagnostics for language models. *Transactions of the Association for Computational Linguistics*, 8:34–48.

Lea Frermann and Mirella Lapata. 2016. A Bayesian model of diachronic meaning change. *Transactions of the Association for Computational Linguistics*, 4:31–45.

Dirk Geeraerts. 2010. *Theories of lexical semantics*. Oxford University Press, Oxford, UK.

Dirk Geeraerts. 2018. *Ten lectures on cognitive sociolinguistics*. Brill, Leiden.

Mario Giulianelli. 2019. *Lexical semantic change analysis with contextualised word representations*. University of Amsterdam, Amsterdam.

Mario Giulianelli, Marco del Tredici, and Raquel Fernández. 2020. Analysing lexical semantic change with contextualised word representations. In *Annual Meeting of the Association for Computational Linguistics (ACL) 58*.

Yoav Goldberg. 2019. Assessing BERT's syntactic abilities. In *arXiv 1901.05287*.

Hongyu Gong, Suma Bhat, and Pramod Viswanath. 2020. Enriching word embeddings with temporal and spatial information. In *Conference on Computational Natural Language Learning (CoNLL) 24*.

Aditya Grover and Jure Leskovec. 2016. node2vec: Scalable feature learning for networks. In *International Conference on Knowledge Discovery and Data Mining (KDD) 22*.

William Hamilton, Kevin Clark, Jure Leskovec, and Dan Jurafsky. 2016a. Inducing domain-specific sentiment lexicons from unlabeled corpora. In *Conference on Empirical Methods in Natural Language Processing (EMNLP) 2016*.

William Hamilton, Jure Leskovec, and Dan Jurafsky. 2016b. Diachronic word embeddings reveal statistical laws of semantic change. In *Annual Meeting of the Association for Computational Linguistics (ACL) 54*.

Jennifer Hay, Paul Warren, and Katie Drager. 2006. Factors influencing speech perception in the context of a merger-in-progress. *Journal of Phonetics*, 34(4):458–484.

Devamanyu Hazarika, Soujanya Poria, Sruthi Gorantla, Erik Cambria, Roger Zimmermann, and Rada Mihalcea. 2018. CASCADE: Contextual sarcasm detection in online discussion forums. In *International Conference on Computational Linguistics (COLING) 27*.

John Hewitt and Christopher D. Manning. 2019. A structural probe for finding syntax in word representations. In *Annual Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL HTL) 2019*.

Valentin Hofmann, Janet B. Pierrehumbert, and Hinrich Schütze. 2020. DagoBERT: Generating derivational morphology with a pretrained language model. In *Conference on Empirical Methods in Natural Language Processing (EMNLP) 2020*.

Renfen Hu, Shen Li, and Shichen Liang. 2019. Diachronic sense modeling with deep contextualized word embeddings: An ecological view. In *Annual Meeting of the Association for Computational Linguistics (ACL) 57*.

Kokil Jaidka, Niyati Chhaya, and Lyle H. Ungar. 2018. Diachronic degradation of language models: Insights from social media. In *Annual Meeting of the Association for Computational Linguistics (ACL) 56*.

Ganesh Jawahar, Benoit Sagot, and Djamé Seddah. 2019. What does BERT learn about the structure of language? In *Annual Meeting of the Association for Computational Linguistics (ACL) 57*.

Ganesh Jawahar and Djamé Seddah. 2019. Contextualized diachronic word representations. In *International Workshop on Computational Approaches to Historical Language Change 1*.

Jens Kaiser, Dominik Schlechtweg, and Sabine Schulte im Walde. 2020. OP-IMS @ DIACR-Ita: Back to the roots: SGNS+OP+CD still rocks semantic change detection. In *Evaluation Campaign of Natural Language Processing and Speech Tools for Italian (EVALITA) 7*.

Yoon Kim, Yi-I Chiu, Kentaro Hanaki, Darshan Hegde, and Slav Petrov. 2014. Temporal analysis of language through neural language models. In *Workshop on Language Technologies and Computational Social Science*.

Diederik P. Kingma and Jimmy L. Ba. 2015. Adam: A method for stochastic optimization. In *International Conference on Learning Representations (ICLR) 3*.

Peter Koch. 2016. Meaning change and semantic shifts. In Päivi Juvonen and Maria Koptjevskaja-Tamm, editors, *The lexical typology of semantic shifts*, pages 21–66. De Gruyter, Berlin.

Vivek Kulkarni, Rami Al-Rfou, Bryan Perozzi, and Steven Skiena. 2015. Statistically significant detection of linguistic change. In *The Web Conference (WWW) 24*.

Andrey Kutuzov and Mario Giulianelli. 2020. UiO-UvA at SemEval-2020 task 1: Contextualised embeddings for lexical semantic change detection. In *International Workshop on Semantic Evaluation (SemEval) 2020*.

Andrey Kutuzov, Lilja Øvrelid, Terrence Szymanski, and Erik Velldal. 2018. Diachronic word embeddings and semantic shifts: A survey. In *International Conference on Computational Linguistics (COLING) 27*.

William Labov. 2001. *Principles of linguistic change: Social Factors*. Blackwell, Malden, MA.

Chang Li and Dan Goldwasser. 2019. Encoding social information with graph convolutional networks for political perspective detection in news media. In *Annual Meeting of the Association for Computational Linguistics (ACL) 57*.

Yongjie Lin, Yi C. Tan, and Robert Frank. 2019. Open sesame: Getting inside BERT's linguistic knowledge. In *Analyzing and Interpreting Neural Networks for NLP (BlackboxNLP) 2*.

Nelson F. Liu, Matt Gardner, Yonatan Belinkov, Matthew Peters, and Noah A. Smith. 2019. Linguistic knowledge and transferability of contextual representations. In *Annual Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL HTL) 2019*.

Jan Lukes and Anders Søgaard. 2018. Sentiment analysis under temporal shift. In *Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis (WASSA) 9*.

Matej Martinc, Petra Kralj Novak, and Senja Pollak. 2020a. Leveraging contextual embeddings for detecting diachronic semantic shift. In *International Conference on Language Resources and Evaluation (LREC) 12*.

Matej Martinc, Syrielle Montariol, Elaine Zosa, and Lidia Pivovarova. 2020b. Capturing evolution in word usage: Just add more clusters? In *The Web Conference (WWW) 29*.

Naoki Masuda, Mason A. Porter, and Renaud Lambiotte. 2017. Random walks and diffusion on networks. *Physics Reports*, 716-717:1–58.

Bryan McCann, James Bradbury, Caiming Xiong, and Richard Socher. 2017. Learned in translation: Contextualized word vectors. In *Advances in Neural Information Processing Systems (NIPS) 31*.

Quinn McNemar. 1947. Note on the sampling error of the difference between correlated proportions or percentages. *Psychometrika*, 12(2):153–157.

Timothee Mickus, Denis Paperno, Mathieu Constant, and Kees van Deemter. 2020. What do you mean, BERT? Assessing BERT as a distributional semantics model. In *Society for Computation in Linguistics (SCiL) 3*.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013a. Efficient estimation of word representations in vector space. In *arXiv 1301.3781*.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013b. Distributed representations of words and phrases and their compositionality. In *Advances in Neural Information Processing Systems (NIPS) 26*.

James Milroy. 1992. *Linguistic variation and change: On the historical sociolinguistic of English*. Blackwell, Oxford, UK.

Lesley Milroy. 1980. *Language and social networks*. Blackwell, Oxford, UK.

Pushkar Mishra, Marco del Tredici, Helen Yannakoudakis, and Ekaterina Shutova. 2018. Author profiling for abuse detection. In *International Conference on Computational Linguistics (COLING) 27*.

Pushkar Mishra, Marco del Tredici, Helen Yannakoudakis, and Ekaterina Shutova. 2019. Abusive language detection with graph convolutional networks. In *Annual Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL HTL) 2019*.

Daisuke Oba, Naoki Yoshinaga, Shoetsu Sato, Satoshi Akasaki, and Masashi Toyoda. 2019. Modeling personal biases in language use by inducing personalized word embeddings. In *Annual Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL HTL) 2019*.

Hermann Paul. 1880. *Principien der Sprachgeschichte*. Tübingen, Niemeyer.

Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. GloVe: Global vectors for word representation. In *Conference on Empirical Methods in Natural Language Processing (EMNLP) 2014*.

Valerio Perrone, Palma, Marco, Simon Hengchen, Alessandro Vatri, Jim Q. Smith, and Barbara McGillivray. 2019. GASC: Genre-aware semantic change for Ancient Greek. In *International Workshop on Computational Approaches to Historical Language Change 1*.

Matthew Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018a. Deep contextualized word representations. In *Annual Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL HLT) 2018*.

Matthew Peters, Mark Neumann, Luke Zettlemoyer, and Wen-tau Yih. 2018b. Dissecting contextual word embeddings: Architecture and representation. In *Conference on Empirical Methods in Natural Language Processing (EMNLP) 2018*.

Janet Pierrehumbert. 2012. The dynamic lexicon. In Abigail Cohn, Cécile Fougeron, and Marie Huffman, editors, *The Oxford handbook of laboratory phonology*, pages 173–183. Oxford University Press, Oxford.

Yuval Pinter, Cassandra L. Jacobs, and Jacob Eisenstein. 2020. Will it unblend? In *Findings of Empirical Methods in Natural Language Processing (EMNLP) 2020*.

Martin Pömsl and Roman Lyapin. 2020. CIRCE at SemEval-2020 task 1: Ensembling context-free and context-dependent word representations. In *International Workshop on Semantic Evaluation (SemEval) 2020*.

James Powell and Kari Sentz. 2020. Tracking short-term temporal linguistic dynamics to characterize candidate therapeutics for COVID-19 in the CORD-19 corpus. In *Conference on Embedded Networked Sensor Systems (SenSys) 18*.

Ondřej Pražák, Pavel Přibáň, Stephen Taylor, and Jakub Sido. 2020. UWB at SemEval-2020 task 1: Lexical semantic change detection. In *International Workshop on Semantic Evaluation (SemEval) 2020*.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21:1–67.

Stephen Roberts, Michael Osborne, Mark Ebden, Steven Reece, Neale Gibson, and Suzanne Aigrain. 2013. Gaussian processes for time-series modelling. *Philosophical Transactions of the Royal Society A*, 371(1984):20110550.

Justyna Robinson. 2010. Awesome insights into semantic variation. In Dirk Geeraerts, Gitte Kristiansen, and Yves Peirsman, editors, *Advances in cognitive sociolinguistics*, pages 85–109. De Gruyter, Berlin.

Justyna Robinson. 2012. A sociolinguistic approach to semantic change. In Kathryn Allan and Justyna Robinson, editors, *Current methods in historical semantics*, pages 199–231. De Gruyter, Berlin.

Anna Rogers, Olga Kovaleva, and Anna Rumshisky. 2020. A primer in BERTology: What we know about how BERT works. In *arXiv 2002.12327*.

Alex Rosenfeld and Katrin Erk. 2018. Deep neural models of semantic shift. In *Annual Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL HLT) 2018*, pages 474–484.

Maja Rudolph and David Blei. 2018. Dynamic embeddings for language evolution. In *The Web Conference (WWW) 27*.

Dominik Schlechtweg, Anna Hätty, Marco del Tredici, and Sabine Schulte im Walde. 2019. A wind of change: Detecting and evaluating lexical semantic change across times and domains. In *Annual Meeting of the Association for Computational Linguistics (ACL) 57*.

Dominik Schlechtweg, Barbara McGillivray, Simon Hengchen, Haim Dubossarsky, and Nina Tahmasebi. 2020. SemEval-2020 task 1: Unsupervised lexical semantic change detection. In *International Workshop on Semantic Evaluation (SemEval) 2020*.

Dominik Schlechtweg and Sabine Schulte im Walde. 2020. Simulating lexical semantic change from sense-annotated data. In *International Conference on the Evolution of Language (EvoLang) 13*.

Dominik Schlechtweg, Sabine Schulte im Walde, and Stefanie Eckmann. 2018. Diachronic usage relatedness (DURel): A framework for the annotation of lexical semantic change. In *Annual Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL HLT) 2018*.

Suzanna Sia, Ayush Dalmia, and Sabrina J. Mielke. 2020. Tired of topic models? Clusters of pretrained word embeddings make for fast and good topics too! In *Conference on Empirical Methods in Natural Language Processing (EMNLP) 2020*.

Ian Stewart, Dustin Arendt, Eric Bell, and Svitlana Volkova. 2017. Measuring, predicting and visualizing short-term change in word representation and usage in VKontakte social network. In *International AAAI Conference on Weblogs and Social Media (ICWSM) 11*.

Nina Tahmasebi, Lars Borin, and Adam Jatowt. 2018. Survey of computational approaches to lexical semantic change detection. In *arXiv 1811.06278*.

Jiliang Tang, Huiji Gao, and Huan Liu. 2012. mTrust: Discerning multi-faceted trust in a connected world. In *International Conference on Web Search and Data Mining (WSDM) 5*.

Ian Tenney, Patrick Xia, Berlin Chen, Alex Wang, Adam Poliak, R. Thomas McCoy, Najoung Kim, Benjamin van Durme, Samuel R. Bowman, Dipanjan Das, and Ellie Pavlick. 2019. What do you learn from context? probing for sentence structure in contextualized word representations. In *International Conference on Learning Representations (ICLR) 7*.

George Uhlenbeck and Leonard Ornstein. 1930. On the theory of the Brownian motion. *Physical Review*, 36:823–841.

Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. 2018. Graph attention networks. In *International Conference on Learning Representations (ICLR) 6*.

Ivan Vulić, Edoardo M. Ponti, Robert Litschko, Goran Glavaš, and Anna Korhonen. 2020. Probing pretrained language models for lexical semantics. In *Conference on Empirical Methods in Natural Language Processing (EMNLP) 2020*.

Charles Welch, Jonathan Kummerfeld, Verónica Pérez-Rosas, and Rada Mihalcea. 2020a. Compositional demographic word embeddings. In *Conference on Empirical Methods in Natural Language Processing (EMNLP) 2020*.

Charles Welch, Jonathan Kummerfeld, Verónica Pérez-Rosas, and Rada Mihalcea. 2020b. Exploring the value of personalized word embeddings. In *International Conference on Computational Linguistics (COLING) 28*.

Gregor Wiedemann, Steffen Remus, Avi Chawla, and Chris Biemann. 2019. Does BERT make any sense? interpretable word sense disambiguation with contextualized embeddings. In *arXiv 1909.10430*.

Frank Wilcoxon. 1945. Individual comparisons by ranking methods. *Biometrics Bulletin*, 1(6):80–83.

Yi Yang, Ming-Wei Chang, and Jacob Eisenstein. 2016. Toward socially-infused information extraction: Embedding authors, mentions, and entities. In *Conference on Empirical Methods in Natural Language Processing (EMNLP) 2016*.

Yi Yang and Jacob Eisenstein. 2017. Overcoming language variation in sentiment analysis with social attention. *Transactions of the Association for Computational Linguistics*, 5:295–307.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Ruslan Salakhutdinov, and Quoc V. Le. 2019. XLNet: Generalized autoregressive pretraining for language understanding. In *Advances in Neural Information Processing Systems (NeurIPS) 33*.

Jing Yao, Zhicheng Dou, and Ji-Rong Wen. 2020. Employing personal word embeddings for personalized search. In *International Conference on Research and Development in Information Retrieval (SIGIR) 43*.

Zijun Yao, Yifan Sun, Weicong Ding, Nikhil Rao, and Hui Xiong. 2018. Dynamic word embeddings for evolving semantic discovery. In *International Conference on Web Search and Data Mining (WSDM) 11*.

Ziqian Zeng, Xin Liu, and Yangqiu Song. 2018. Biased random walk based social regularization for word embeddings. In *International Joint Conference on Artificial Intelligence (IJCAI) 27*.

Ziqian Zeng, Yichun Yin, Yangqiu Song, and Ming Zhang. 2017. Socialized word embeddings. In *International Joint Conference on Artificial Intelligence (IJCAI) 26*.

A Appendices

A.1 Data Preprocessing

For each dataset, we remove duplicates as well as texts with less than 10 words. For the Ciao dataset, we further remove reviews rated as not helpful. We lowercase all words. Since BERT's input is limited to 512 tokens, we truncate longer texts by taking the first and last 256 tokens.

A.2 Embedding Training: Hyperparameters

DCWE. The hyperparameters of the contextualizer are as for $\text{BERT}_{\text{BASE}}$ (uncased). In particular, the dimensionality of the input embeddings $\tilde{\mathbf{e}}^{(k)}$ is 768. For the dynamic component, the social vectors $\mathbf{s}_{ij}$ and $\tilde{\mathbf{s}}_i$ have a dimensionality of 50. The node2vec vectors for the initialization of $\tilde{\mathbf{s}}_i$ are trained on 10 sampled walks of length 80 per node with a window size of 2. The GAT has two layers with four attention heads, respectively (activation function: $\tanh$). The feed-forward network has two layers (activation function: $\tanh$). We apply dropout

	ArXiv						Ciao						Reddit						YELP					
Model	μ	σ	n_e	l	λ_a	τ	μ	σ	n_e	l	λ_a	τ	μ	σ	n_e	l	λ_a	τ	μ	σ	n_e	l	λ_a	τ
DCWE	3.848	.307	7	3e-6	1e-1	6,756	6.794	.606	7	3e-6	1e-1	11,831	9.836	.318	7	3e-6	1e-1	4,629	5.122	.384	7	3e-6	1e-1	7,002
CWE	3.851	.305	7	3e-6	—	3,749	6.789	.589	7	3e-6	—	3,564	9.869	.274	7	3e-6	—	2,160	5.129	.384	7	3e-6	—	3,551

Table 6: Validation performance statistics and hyperparameter search details for embedding training. DCWE: dynamic contextualized word embeddings; CWE: contextualized word embeddings. The table shows the mean (μ) and standard deviation (σ) of the validation performance (masked language modeling perplexity) on all hyperparameter search trials and gives the number of epochs (n_e), learning rate (l), and regularization constant (λ_a) with the best validation performance as well as the runtime (τ) in minutes for one full hyperparameter search (28 trials for DCWE on Ciao, 14 trials for CWE on Ciao, 7 trials for DCWE and CWE on ArXiv, Reddit, and YELP).

with a rate of 0.2 after each layer of the dynamic component. The number of trainable parameters varies between models trained on different datasets due to differences in $|\mathcal{T}|$ and is 134,914,570 for ArXiv, 124,990,698 for Ciao, 120,028,762 for Reddit, and 122,509,730 for YELP. We use a batch size of 4 and perform grid search for the number of epochs $n_e \in \{1, \dots, 7\}$, the learning rate $l \in \{1 \times 10^{-6}, 3 \times 10^{-6}\}$, and the regularization constant $\lambda_a \in \{1 \times 10^{-2}, 1 \times 10^{-1}\}$, thereby also determining λ_w (Section 3.4).

CWE. All hyperparameters are as for $\text{BERT}_{\text{BASE}}$ (uncased). The number of trainable parameters is 110,104,890. We use a batch size of 4 and perform grid search for the number of epochs $n_e \in \{1, \dots, 7\}$ and the learning rate $l \in \{1 \times 10^{-6}, 3 \times 10^{-6}\}$.

For both DCWE and CWE, we tune hyperparameters except for the number of epochs on the Ciao dataset (selection criterion: masked language modeling perplexity) and use the best configuration for ArXiv, Reddit, and YELP. Models are trained with categorical cross-entropy as the loss function and Adam (Kingma and Ba, 2015) as the optimizer. Experiments are performed on a GeForce GTX 1080 Ti GPU (11GB).

Table 6 lists statistics of the validation performance over hyperparameter search trials and provides information about best hyperparameter configurations.[18] We also report the number of hyperparameter search trials as well as runtimes for the hyperparameter search.

A.3 Ablation Study: Hyperparameters

SA. Words are mapped to offsets using time-specific two-layer feed-forward networks (activation function: tanh). Both layers have a dimensionality of 768. All other hyperparameters are as for DCWE with a full dynamic component (Appendix A.2). The number of trainable parameters again varies between models trained on different datasets due to differences in $|\mathcal{T}|$ and is 133,728,570 for ArXiv, 124,279,098 for Ciao, 119,554,362 for Reddit, and 121,916,730 for YELP. We use a batch size of 4 and perform grid search for the number of epochs $n_e \in \{1, \dots, 7\}$, the learning rate $l \in \{1 \times 10^{-6}, 3 \times 10^{-6}\}$, and the regularization constant $\lambda_a \in \{1 \times 10^{-2}, 1 \times 10^{-1}\}$, thereby also determining λ_w (Section 3.4).

TA. All hyperparameters are as for DCWE with a full dynamic component (Appendix A.2), with the difference that we only use one social component (consisting of a two-layer GAT and a two-layer feed-forward network) for all time units. The number of trainable parameters is 111,345,374. We use a batch size of 4 and perform grid search for the number of epochs $n_e \in \{1, \dots, 7\}$, the learning rate $l \in \{1 \times 10^{-6}, 3 \times 10^{-6}\}$, and the regularization constant $\lambda_a \in \{1 \times 10^{-2}, 1 \times 10^{-1}\}$.

For both SA and TA, we tune hyperparameters except for the number of epochs on the Ciao dataset (selection criterion: masked language modeling perplexity) and use the best configuration for ArXiv, Reddit, and YELP. Models are trained with categorical cross-entropy as the loss function and Adam as the optimizer. Experiments are performed on a GeForce GTX 1080 Ti GPU (11GB).

Table 7 lists statistics of the validation performance over hyperparameter search trials and provides information about best hyperparameter configurations. We also report the number of hyperparameter search trials as well as runtimes for the hyperparameter search.

A.4 Sentiment Analysis: Hyperparameters

DCWE. The mid layer of the feed-forward network on top of BERT has a dimensionality of 100. All other hyperparameters are as for DCWE trained on masked language modeling (Appendix A.2).

[18] Since expected validation performance (Dodge et al., 2019) may not be correct for grid search, we report mean and standard deviation of the performance instead.

	ArXiv						Ciao						Reddit						YELP					
Model	μ	σ	n_e	l	λ_a	τ	μ	σ	n_e	l	λ_a	τ	μ	σ	n_e	l	λ_a	τ	μ	σ	n_e	l	λ_a	τ
SA	3.849	.302	7	3e-6	1e-1	4,438	6.790	.635	7	3e-6	1e-1	7,616	9.851	.282	6	3e-6	1e-1	2,699	5.127	.392	7	3e-6	1e-1	4,231
TA	3.860	.303	7	3e-6	1e-1	6,080	6.843	.782	7	3e-6	1e-1	10,343	9.871	.321	7	3e-6	1e-1	3,859	5.129	.388	7	3e-6	1e-1	6,471

Table 7: Validation performance statistics and hyperparameter search details for ablation study. SA: social ablation; TA: temporal ablation. The table shows the mean (μ) and standard deviation (σ) of the validation performance (masked language modeling perplexity) on all hyperparameter search trials and gives the number of epochs (n_e), learning rate (l), and regularization constant (λ_a) with the best validation performance as well as the runtime (τ) in minutes for one full hyperparameter search (28 trials on Ciao, 7 trials on ArXiv, Reddit, and YELP).

	Ciao						YELP					
Model	μ	σ	n_e	l	λ_a	τ	μ	σ	n_e	l	λ_a	τ
DCWE	.883	.010	4	3e-6	1e-1	8,128	.967	.003	2	3e-6	1e-1	4,373
CWE	.880	.011	5	3e-6	—	2,122	.967	.001	3	3e-6	—	2,221

Table 8: Validation performance statistics and hyperparameter search details for sentiment analysis. DCWE: dynamic contextualized word embeddings; CWE: contextualized word embeddings. The table shows the mean (μ) and standard deviation (σ) of the validation performance (F1 score) on all hyperparameter search trials and gives the number of epochs (n_e), learning rate (l), and regularization constant (λ_a) with the best validation performance as well as the runtime (τ) in minutes for one full hyperparameter search (20 trials for DCWE on Ciao, 10 trials for CWE on Ciao, 5 trials for DCWE and CWE on YELP).

The number of trainable parameters again varies between models trained on different datasets due to differences in $|\mathcal{T}|$ and is 124,445,049 for Ciao and 121,964,081 for YELP. We use a batch size of 4 and perform grid search for the number of epochs $n_e \in \{1, \dots, 5\}$, the learning rate $l \in \{1 \times 10^{-6}, 3 \times 10^{-6}\}$, and the regularization constant $\lambda_a \in \{1 \times 10^{-2}, 1 \times 10^{-1}\}$, thereby also determining λ_w (Section 3.4).

CWE. The mid layer of the feed-forward network on top of BERT has a dimensionality of 100. All other hyperparameters are as for $\text{BERT}_{\text{BASE}}$ (uncased). The number of trainable parameters is 109,559,241. We use a batch size of 4 and perform grid search for the number of epochs $n_e \in \{1, \dots, 5\}$ and the learning rate $l \in \{1 \times 10^{-6}, 3 \times 10^{-6}\}$.

For both DCWE and CWE, we tune hyperparameters except for the number of epochs on the Ciao dataset (selection criterion: F1 score) and use the best configuration for YELP. Models are trained with binary cross-entropy as the loss function and Adam as the optimizer. Experiments are performed on a GeForce GTX 1080 Ti GPU (11GB).

Table 8 lists statistics of the validation performance over hyperparameter search trials and provides information about best hyperparameter configurations. We also report the number of hyperparameter search trials as well as runtimes for the hyperparameter search.