A Tale of Two Laws of Semantic Change: Predicting Synonym Changes with Distributional Semantic Models

Bastien Liétard and **Mikaela Keller** and **Pascal Denis**
Univ. Lille, Inria, CNRS, Centrale Lille, UMR 9189 - CRIStAL, F-59000 Lille, France
first_name.last_name@inria.fr

Abstract

Lexical Semantic Change is the study of how the meaning of words evolves through time. Another related question is whether and how lexical relations over pairs of words, such as synonymy, change over time. There are currently two competing, apparently opposite hypotheses in the historical linguistic literature regarding how synonymous words evolve: the Law of Differentiation (LD) argues that synonyms tend to take on different meanings over time, whereas the Law of Parallel Change (LPC) claims that synonyms tend to undergo the same semantic change and therefore remain synonyms. So far, there has been little research using distributional models to assess to what extent these laws apply on historical corpora. In this work, we take a first step toward detecting whether LD or LPC operates for given word pairs. After recasting the problem into a more tractable task, we combine two linguistic resources to propose the first complete evaluation framework on this problem and provide empirical evidence in favor of a dominance of LD. We then propose various computational approaches to the problem using Distributional Semantic Models and grounded in recent literature on Lexical Semantic Change detection. Our best approaches achieve a balanced accuracy above 0.6 on our dataset. We discuss challenges still faced by these approaches, such as polysemy or the potential confusion between synonymy and hypernymy.

1 Introduction

Recent years have seen a surge to model lexical semantic change (LSC) with computational approaches based on Distributional Semantic Models (DSMs) (Tahmasebi et al., 2021). While most research in this area has concentrated on developing approaches for automatically *detecting* LSC for individual words, as in the dedicated SemEval 2020 shared task (Schlechtweg et al., 2020), there has also been some work on validating or even proposing laws of semantic changes through new DSM-based approaches (Dubossarsky et al., 2015; Hamilton et al., 2016; Dubossarsky et al., 2017). Ultimately, this line of work is very promising as it can provide direct contributions to the field of historical linguistics.

In this paper, we consider two laws of semantic change that are very prominent in historical linguistics, but that have to date given rise to very little computational modeling studies. Specifically, the Law of Differentiation (LD), originally due to Bréal (1897, chapter 2), posits that synonymous words tend to take on different meanings over time; or one of them will simply disappear.[1] The same idea is also discussed in more recent work, such as Clark (1993). As an example, the verbs *spread* and *broadcast* used to be synonyms (especially in farming), but now the latter is only used in the sense of *transmit*, by means of radio, television or internet. The verbs *plead* and *beseech* are synonyms, but *beseech* is no longer used nowadays compared to *plead*. By contrast, the Law of Parallel Change (LPC),[2] inspired from the work of Stern (1921), claims that two synonyms tend to undergo the same semantic change and therefore remain synonyms. As an illustration, Stern (1921, chapter 3 and 4) describes the change of *swiftly* and its synonyms from the sense of *rapidly* to the stronger sense of *immediately*. Lehrer (1985) also observes a parallel change affecting animal terms which acquire a metaphorical sense.

These two laws are interesting under several aspects. Firstly, these laws go beyond the problem of detecting semantic change in individual words, as they concern the question of whether a lexical relationship between words, in this case synonymy, is preserved or not through time. Secondly, these laws make very strong, seemingly opposite, predictions

[1] To cite Bréal (1897): *"[S]ynonyms do not exist for long: either they differ, or one of the two terms disappears."*

[2] Name coined by Xu and Kemp (2015).

*Proceedings of the The 12th Joint Conference on Lexical and Computational Semantics (*SEM 2023)*, pages 338–352
July 13-14, 2023 ©2023 Association for Computational Linguistics

on how synonyms evolve: either their meanings diverge (under LD) or they remain close (under LPC). It is likely that both of these laws might be at work, but they possibly apply to different word classes, correspond to different linguistic or extra-linguistic factors, or operate at different time scales. A large-scale study, fueled by computational methods over large quantities of texts, would be amenable to statistical analyses addressing these questions. In this work, we focus on predicting the persistence (or disappearance) of synonymy through time, as a first step toward more complete analyses.

Prima facie, DSMs appear to provide a natural resource for constructing a computational approach for assessing the importance of these laws, as they inherently –through the distributional hypothesis– capture a notion of semantic proximity, which can be used as a proxy for synonymy. Following this idea, Xu and Kemp (2015) propose the first DSM-based method for predicting how synonymous word pairs of English evolve over time (specifically, from 1890 to 1990). This research decisively concludes that there is "evidence against the Law of Differentiation and in favor of the Law of Parallel Change" for adjectives, nouns and verbs alike (i.e., the three considered POS). However, this pioneering work suffers from some limitations that cast some doubts on this conclusion. First off, the predictions made by their approach are not checked against a ground truth, thus lacks a proper evaluation. Second, the approach is strongly biased against LD, as only pairs in which *both* words have changed are considered, excluding pairs in which differentiation may occur (e.g. in *spread/broadcast*, only the latter word changed in meaning).

This paper addresses these shortcomings by introducing a more rigorous evaluation framework for testing these two laws and evaluating computational approaches. We build a dataset of English synonyms that was obtained by combining lexical resources for two time stamps (1890 and 1990) that records, for a given list of synonym pairs at time 1890, whether these pairs are still synonymous or not in 1990. The analysis of this dataset reveals that, contra Xu and Kemp (2015) and though using the same initial synonym set, synonymous words show a strong tendency to differentiate in meaning over time. With some variation across POS, we found that between 55 and 80% of synonyms in 1890 are no longer synonyms in 1990.

Moreover, we propose several new computational approaches[3], grounded in more recent DSMs, for automatically predicting whether synonymous words diverge or remain close in meaning over time, which we recast as a binary classification problem. Inspired by Xu & Kemp (2015), our first approach is unsupervised and tracks pairwise synchronic distances over time, computed over SGNS-based vector representations. Our second approach is supervised and integrates additional variables into a logistic regression model. This latter model achieves a balanced accuracy above 0.6 over the proposed dataset.

2 Related Work

Data-driven methods to detect LSC have gained popularity in the recent years (Tahmasebi et al., 2021), using increasingly powerful and expressive word representations, ranging from the simple co-occurrence word vectors (Sagi et al., 2012) to static word embeddings (Schlechtweg et al., 2019) and transformer-based contextualized word representations (Kutuzov et al., 2022; Fourrier and Montariol, 2022). This line of research lead to the development of shared tasks (Zamora-Reina et al., 2022; Schlechtweg et al., 2020; Rodina and Kutuzov, 2020). Most often, these tasks concern the evolution of individual words, in effect focusing on *absolute* semantic change (of words individually). In this paper, we take a different stand, considering the problem of *relative* change in meaning among pairs of words, specifically focusing on synonym pairs.

Previous work on word pairs are rare in the current LSC research landscape. A first exception is (Turney and Mohammad, 2019), who also study the evolution of synonyms. They propose a dataset to track how usage frequency of words evolve over time within a sets of synonyms, as well as a new task: namely, to predict whether the dominant (most frequent) word of a synonyms set will change or not. This task is actually complementary to the one we address in this work. While Turney and Mohammad (2019) assume the stability of most synonym pairs between 1800 and 2000, and rather investigate the dynamic inside sets of synonymous words across time, we question this alleged stability and attempt to track whether these words remain synonymous at all in this time period.

[3]The code used to run experiments in this paper can be found at https://github.com/blietard/synonyms-semchange

Another distinctive motivation of our work is in the empirical, large-scale evaluation of two proposed laws of semantic change, originating from historical linguistics. Previous work investigating laws of semantic change with DSMs include Dubossarsky et al. (2015) and Hamilton et al. (2016), who measured semantic change of words between 1800 and 2000 and attempted to draw statistical laws of semantic change from their observations. Later, Dubossarsky et al. (2017) contrasted these observations and showed that even if these effects may be real, it may be to a lesser extent.

The closest work to the current research is the study of Xu and Kemp (2015), as they already focus on the two laws of Differentiation (LD) and Parallel Change (LPC). Their main motivation was to automatically measure, using DSMs, which of the two laws was predominant between 1890 and 1999. To study which of the two laws actually operates, they focus on word pairs that (i) are synonyms in the 1890s and (ii) where both words changed significantly in meaning between 1890 and the 1990s. First, they represent words as probability distributions of direct contexts, using normalized co-occurrence count vectors. Then, they measure the (synchronic) semantic proximity of words by computing the Jensen-Shannon Divergence between the corresponding distributions. Semantic change in a word is quantified by comparing its semantic space neighborhoods in the 1890s and in the 1990s. Finally, for every selected synonymous pair, they pick a control word pair that has a smaller divergence in the 1890s than the associated synonyms. At a later time in the 1990s, if the divergence for the synonyms is larger than that for the control pair, they decide these synonyms have undergone LD, otherwise they predict LPC. Ultimately, they found that most pairs (around 60%) have undergone LPC, which would be the dominant law.

The pioneering work of Xu and Kemp (2015) faces a number of shortcomings. First, their restriction to synonymous pairs in which both words changed mechanically excludes certain cases of LD (i.e., where one one word has changed), thus introducing an artificial bias against LD. Moreover, they often select near-synonyms as controls (e.g. *instructive* and *interesting*) because they constrain control pairs to be *closer* in divergence in the 1890s than the associated synonym pairs. Furthermore, and more importantly, Xu and Kemp (2015) did not compare their predictions to any ground-truth and there is no evaluation of the reliability of their method. Finally, their choice of word representations is not among the State-of-the-Art for static methods.

In this paper, we consider all synonymous pairs, thus avoiding the bias against LD. We propose different approaches that we compare to Xu and Kemp (2015)'s control pairs, and we provide results obtained with more recent distributional semantic models. Most importantly, we propose a complete evaluation framework to benchmark the different methods, something missing in this prior work.

3 Problem Statement

Our overarching goal is to develop new computational approaches that are able to automatically predict which pairs of synonymous words underwent LD or LPC. These predictions could be used as a first step towards providing a more refined and statistically meaningful analysis of the two laws. An important milestone towards developing such an approach is to compare it to some ground truth. Otherwise, there is no way to assess whether statistics obtained for LD or LPC are indeed reliable, a problem faced by Xu and Kemp (2015).

Unfortunately, there is no existing large-scale resource that records instances of LD/LPC, beyond a handful of examples found in research papers and textbooks in historical linguistics. What exists however are historical lists of synonyms, which we can compare to obtain some form of ground truth. This forces us to consider a slightly different methodological framework, focusing on a more constrained prediction task, namely to detect pairs of synonyms at time $T1$ that have remained synonymous or that are no longer synonymous at time $T2(> T1)$.

3.1 Formalization

Let us denote $W^{(T)}$ the set of words (or vocabulary) for a given language (say English) at time T. As language evolves through time, vocabularies at two times $T1$ and $T2$ need not have the exact same extensions: e.g., a word w in $W^{(T1)}$ might not be in $W^{(T2)}$ (i.e., w has disappeared). Making a simplistic, idealized assumption, let $\mathcal{C}$ be a mostly atemporal and exhaustive discrete set of concepts, and denote $M_w^{(T)} \subset \mathcal{C}$ the meaning of word w at time T. The definition of $M_w^{(T)}$ as a set allows homonymy and/or polysemy to be accounted for.

Given these notations, we have that $u \in W^{(T)}$

and $v \in W^{(T)}$ are synonyms at a time T if $M_u^{(T)} \cap M_v^{(T)} \neq \emptyset$. We understand that the study of LD / LPC implies to track (i) the change of $M_u^{(T)}$ and $M_v^{(T)}$ over time, (ii) the evolution of $M_u^{(T)} \cap M_v^{(T)}$ and (iii) the very persistence of both words in vocabularies $W^{(T)}$ between $T1$ and $T2$. Discussion about formalizing LD and LPC under those conditions can be found in appendix A.1.

3.2 Task Formulation: Tracking Synonyms Change

The presented formulation, though very idealized, should make it clear that the development of a computational system that attempts to directly predict LD and LPC, and even the construction of an evaluation benchmark for evaluating such a system, are very challenging tasks. First, the initial synonym set selection presupposes, not only that one has access to a list of synonyms at $T1$ and $T2$, but also that one can reliably predict LSC in one of the two words from $T1$ to $T2$; unfortunately, LSC is still an open problem for current NLP models. Second, one typically does not have meaning inventories or automatic systems (e.g. WSD systems) for mapping words to their meanings at different time stamps. Finally, even tracking the disappearance of words through time is not trivial, as it ideally requires full dictionaries at different time stamps.

Given these limitations, we suggest to narrow down our target problem to the task of predicting, for a given pair of synonymous words (u, v) at $T1$, whether (u, v) are still synonymous or not at $T2$. Stated a little more formally, we are concerned with the following binary classification problem:

$$f : \mathcal{S}^{(T1)} \rightarrow \{\text{"Syn"}, \text{"Diff"}\}$$

$$(u, v) \mapsto f((u, v)) = \begin{cases} \text{"Syn" if } (u, v) \in \mathcal{S}^{(T2)} \\ \text{"Diff" otherwise} \end{cases}$$

where $\mathcal{S}^{(T)}$ is a set of synonymous word pairs at time T, "Syn" indicates that words (u, v) that were synonymous at $T1$ remain synonymous at $T2$, while "Diff" signals that they are no longer synonymous at $T2$. This simpler problem leads to a more operational evaluation procedure, which does not require access to $M_u^{(T^*)}$ and $M_v^{(T^*)}$, but only to lists of synonyms $\mathcal{S}^{(T1)}$ and $\mathcal{S}^{(T2)}$. See Section 4 for presentation of such procedure. It should be clear that predicting which synonym pairs remain ("Syn") or cease to be synymoms ("Diff"), will provide some information about LPC and LD, although the mapping between the two problems is not one-to-one. Even if "Diff" covers pretty well LD, a pair that is still synonymous at $T2$ could either be a case of LPC (their shared meaning changed the same way for both words) or a pair of words that simply have not changed in meaning at all (or at least that their shared meaning is unchanged).

Now turning to designing a computational system that detects "Syn" vs. "Diff", a natural question that emerges is whether current DSMs, commonly used for detecting LSC in individual words, are able to capture synonym changes. More specifically, our main hypothesis will be that one can reliably track the evolution of synonymous pairs through their word vector representations at $T1$ and $T2$. This approach will be instantiated into different unsupervised and supervised models in Section 5.

4 Evaluation Dataset

This section presents a dataset designed to track the evolution of English synonymous word pairs between two time stamps $T1$ and $T2$, with $T2 > T1$. Specifically, the two time periods considered are the 1890's decade ($T1$) and the 1990's decade ($T2$). For extracting synonymous pairs in the 1890's (noted $\mathcal{S}^{(T1)}$), we use Fernald's *English Synonyms and Antonyms* (Fernald, 1896) as Xu and Kemp (2015) did. Pairs were selected based on a set of specific target words (see appendix A.7). As shown in Table 1, we obtain $1,507$ adjective pairs, $2,689$ noun pairs and $1,489$ verb pairs. To assess whether these word pairs are still synonyms in the 1990's, we use WordNet (Fellbaum and Princeton, 2010), as this lexical database was originally constructed in 1990's. Thus, WordNet provides us with $\mathcal{S}^{(T2)}$. Specifically, we considered that a pair of words/lemmas $(u, v) \in \mathcal{S}^{(T1)}$ are still synonymous if they point to at least one common *synset* in WordNet.

The construction of this dataset relies on two crucial hypotheses, which seem reasonable to make. First, both lexical resources rely on the same definition of synonymy. Second, $\mathcal{S}^{(T2)}$ meets some exhaustivity criterion, in the sense that $(u, v) \in \mathcal{S}^{(T1)}$ not appearing in $\mathcal{S}^{(T2)}$ should indicate that u and v are no longer synonymous at $T2$, and not be due to a lack of coverage of the resource (i.e., a false negative). WordNet is assumed to be exhaustive enough, as we checked that every word involved in at least one synonymous pair has its own entry in

Synonyms pairs	ADJ	NN	VERB	All
Synonyms at $T1$	1507	2689	1489	5685
& synonyms at $T2$	202	347	311	860
& synonyms at $T2$(%)	13.4	12.9	20.9	15.1
& hypernyms at $T2$	0	858	398	1256
& hypernyms at $T2$(%)	0.0	31.9	26.7	22.1
& hyp. at $T2$ (1) (%)	0.0	23.2	22.5	16.9
& hyp. at $T2$ (2) (%)	0.0	6.9	3.5	4.1
& hyp. at $T2$ (3) (%)	0.0	1.4	0.5	0.8

Table 1: Numbers of synonymous pairs extracted from Fernald (1896) ($T1$) displayed by POS, and numbers of those that are also considered as synonyms or hypernyms/hyponyms in WordNet ($T2$) For hypernyms, we detail the proportions of hypernym/hyponym pairs that are separated by 1, 2 or 3 nodes in the WordNet graph.

WordNet's database.

Table 1 provides some detailed statistics on the evolution of synynomous pairs between decades 1890's and 1990's, overall and for different parts of speech. A first observation on these datasets is that the proportion of pairs that are still synonyms at $T2$ ("*Syn*") is globally 15.1%. This implies that most synonymous pairs underwent differentiation. While it does not provide information about how change happened between $T1$ and $T2$ for the remaining 84.9%, it's a clue that the Law of Differentiation should be a dominant phenomenon among synonyms.

We exploit the structure of the WordNet database to analyze the different cases of "*Diff*". WordNet includes lexical relations of hyper-/hypo-nymy (e.g., *seat*/*bench*) as well as holo-/mero-nymy (e.g., *bike*/*wheel*) and antonymy (e.g., *small*/*large*) defined over synsets[4]. Note that the hyper-/hypo-nymy relation does not exist in WordNet among adjectives. Among nouns and verbs, we observe that around 30% of pairs that were synonyms at $T1$ are in an hyper-/hypo-nymy relation at $T2$ and two third of them are direct hypernyms in WordNet (their synsets are direct parent/child) indicating the preservation of a very close semantic link. For a further depiction of the dataset in terms of distance in WordNet's graph, see Figure 3 in appendix A.4.

One cannot entirely exclude that $\mathcal{S}^{(T1)}$ includes some hyper-/hypo-nyms as synonyms. However, even if we extend the notion of synonymy at $T2$ to include these cases, we would have only around 45% of all pairs still considered synonyms among nouns and verbs. This indicates that "Diff" largely remains the most common phenomenon with an estimated proportion between 55% and 80%. This finding contradicts the experimental results reported by Xu and Kemp (2015) with their computational approach (only 40% of differentiation).

In lack of additional indication that some of these hyper-/hypo-nym cases at $T2$ are indeed synonyms, or that they may also have been hyper-/hypo-nym at $T1$, we decided to still consider them as instances of "Diff". Another argument for this decision is precisely that there are well-known reported cases of lexical semantic changes in which the meaning of a particular word in effect "widens" to denote a larger subset (i.e., becomes an hypernym): this is the case of *dog* in English that used to denote a specific breed of dogs (Traugott and Dasher, 2001).

5 Approaches

This section presents two classes of computational approaches, unsupervised and supervised, for predicting whether pairs of synonyms at $T1$ remain synonyms ("Syn") or cease to be so ("Diff") at a later time $T2$. Common to all of these approaches is that they are based on two time-aware DSMs, one for each time stamp.

5.1 Time-aware DSMs

Inspired by work on LSC, we rely on separate DSMs for each time stamp $T1$ and $T2$, respectively yielding vector spaces $V^{(T1)}$ and $V^{(T2)}$ encoding the (possibly changing) word meanings at $T1$ and $T2$. Thus, for each synonym pair (u, v), we have two pairs of vectors : $(\mathbf{u}^{(T1)}, \mathbf{v}^{(T1)}) \in V^{(T1)} \times V^{(T1)}$ and $(\mathbf{u}^{(T2)}, \mathbf{v}^{(T2)}) \in V^{(T2)} \times V^{(T2)}$.

Specifically, we use pre-computed SGNS (Mikolov et al., 2013) from Hamilton et al. (2016) trained on the *English* part of the GoogleBooks Ngrams dataset[5] for every decade between 1800 and 2000 and extract $V^{(T1)}$ (1890) and $V^{(T2)}$ (1990). For any word $w \in W$ and any time period T, $\mathbf{w}^{(T)} \in V^{(T)}$ is a single 300 dimensional vector. We ensure synonymy is accurately reflected by checking that synonym pairs have a smaller cosine distance than non-synonymous pairs for both time periods, as in Figure 4 of appendix A.5.

Traditional DSM-based approaches for detecting LSC are based on self-similarities over time for a given word. For instance, for a given time

[4]As we did for synonyms, we assume that two words w_1 and w_2 are instances of one of these relations R if R holds for one of their corresponding synset pair.

[5]https://storage.googleapis.com/books/ngrams/books/datasetsv3.html

interval $(T1, T2)$, they compute for each word w an individual *Diachronic Distance*, noted here $DD^{(T1,T2)}(w)$. Cosine distance is often used (recall in appendix A.2).

There is no obvious distance for comparing *pairs* of word vectors, but one can instead rely on comparing the pairwise word vector distance at each time stamp T; we call this *Synchronic Distance* (denoted SD). The two types of distances for two time stamps $T1$ and $T2$ are described in Figure 1. Our unsupervised method, proposed in Sec. 5.2 directly exploit the idea of tracking different types of SD through time, while Sec. 5.3 presents a supervised approach that combines both SD and DD.

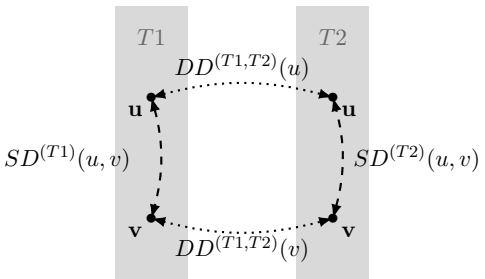

Figure 1: Pairs of word embeddings at 2 time periods and associated diachronic and synchronic distances.

5.2 Unsupervised Methods

While we don't have access to $M_u^{(T)}$ and $M_v^{(T)}$, we can represent the meaning of u and v using DSM and compare them at a given time to estimate how close they are in meaning. Indeed, if $M_u^{(T)} \cap M_v^{(T)}$ changes, this should be reflected in difference of the use contexts of u and those of v, and so reflected in the distance between $\mathbf{u}^{(T)}$ and $\mathbf{v}^{(T)}$. Let

$$SD^{(T)} : W^{(T)} \times W^{(T)} \rightarrow \mathbb{R}^+$$

be a measure of **synchronic distance** between vectors representing two words. By construction of $V^{(T)}$, $SD^{(T)}(u, v)$ is smaller for words (u, v) that appear in similar contexts than for unrelated words. We assume that there exists a value δ_T such that

$$\forall (u, v) \in \mathcal{S}^{(T)},\ SD^{(T)}(u, v) \leq \delta_T.$$

This entails that for a given pair (u, v):

$$SD^{(T)}(u, v) > \delta_T \Rightarrow (u, v) \text{ are not synonyms.}$$

In this setting, one can compare the synchronic distances within $V^{(T1)}$ and with $V^{(T2)}$ and decide if the pair differentiated or stayed synonymous.

Let (u, v) be a pair of synonyms at $T1$, as such we have that $SD^{(T1)}(u, v) \leq \delta_{T1}$. If (u, v) are not synonyms at time $T2$ then $SD^{(T2)}(u, v) > \delta_{T2}$.

Combining these two inequalities, we would say that a pair of synonyms at $T1$ has differentiated at $T2$ if:

$$\underbrace{SD^{(T2)}(u, v) - SD^{(T1)}(u, v)}_{= \Delta(u, v)} > \delta_{T2} - \delta_{T1}.$$

Ideally one could imagine that the distance threshold δ_T at which, words cease to be synonyms should be independent of the time period T. Empirically however, because word embeddings are not necessarily build with an enforced scale, there might be a dilation or shrinking in the overall synchronic distances between $T1$ and $T2$. Let us assume that

$$\delta_{T2} = \delta_{T1} + \tau,\ \tau \in \mathbb{R}.$$

Our decision rule could then be rewritten as:

$$f(u, v) = \begin{cases} \text{“Diff” if } \Delta(u, v) \geq \tau \\ \text{“Syns” otherwise.} \end{cases} \quad (1)$$

This approach is shortly denoted "Δ" in section 6. It diverges from the prior work of Xu and Kemp (2015) that chooses to rely on control pairs instead of a threshold. For the sake of comparison, we implemented their method presented as "*XK controls*". It is not the full protocol presented by Xu and Kemp (2015), as (i) the experimental setting is not identical, they filtered out some synonym pairs and we didn't (ii) we use SGNS word representations and cosine distance instead of normalized co-occurrence counts and Jensen-Shannon Divergence. Schlechtweg et al. (2019) provided a longer comparison between word representations.

We propose a statistically-grounded criterion to set the value for the threshold τ. Since the meaning of most words is expected to remain stable[6], we argue that most pairwise distances should remain stable as well. We can then estimate the dilation between the representations in the two time periods by the average gap between the synchronic distances of words.

$$\tau = \frac{1}{|W|^2} \sum_{(w_1, w_2) \in W \times W} \Delta(w_1, w_2) \quad (2)$$

[6]Intuitively, someone in 2023 can still understand writings published in the 1890s in their original text, like books from Charles Dickens or Arthur Conan Doyle.

In practice, we experiment with two different types of synchronic distances between words. The first is the cosine distance (see A.2). That is:

$$SD^{(T)}(u,v) = \text{cos-dist}(\mathbf{u}^{(T)}, \mathbf{v}^{(T)}).$$

We shortly denote it "SD(cd)". Another measure of semantic proximity is based on the shared word neighborhood between the two vectors u and v:

$$SD^{(T)}(u,v) = \text{jaccard-dist}(\mathcal{N}_k^{(T)}(u), \mathcal{N}_k^{(T)}(v)),$$

with $\mathcal{N}_k^{(T)}(w)$ being the set of the k-nearest neighbors of the point representing w in the vector space at time T, and *jaccard-dist* being the Jaccard distance (see appendix A.2). This measure is ranged between 0 and 1, and we denote it "SD(nk)".

5.3 Supervised Methods

Approaches described so far use the labels in the dataset ("Syn" and "Diff") only for evaluation purposes. But one can also use part of the available data to learn a *supervised* classifier to predicts these labels. Concretely, for most of these models, we trained Logistic Regression (LR) models[7]

Synchronic Distances Combination In our unsupervised approach, we compute $SD^{(T1)}$ and $SD^{(T2)}$ and their difference, denoted Δ. This quantity is then compared to a fixed threshold τ. We propose to investigate two supervised approaches stemming from this: (i) simply tune τ and (ii) use a LR model to learn the optimal weighting in the linear combination of the two distances. This latter model is called "LR SD".

Accounting for Individual Change Most works about computational approaches to LSC focus on detecting the change of a single word (Tahmasebi et al., 2021), using a diachronic distance, which we noted $DD^{(T1,T2)}(w)$, across time periods $T1$ and $T2$ for individual words w.

In addition to synchronic distances, we input diachronic distances as features for a LR model. The resulting classifier (LR SD+DD) uses the 4 distances represented in Figure 1 as variables: self-similarities across time periods (*DD*s), and a distance measure within pairs for each of both time stamps (*SD*s). Similarly to synchronic distances defined in Sec. 5.2, we try two definitions of DD. First, we compare sets of neighbors at $T1$ and $T2$:

$$\text{DD}(w) = \text{jaccard-dist}(\mathcal{N}_k^{(T1)}(w), \mathcal{N}_k^{(T2)}(w)).$$

[7]Implemented with the *scikit-learn* library for Python[8].

We also compute the cosine distance between $\mathbf{w}^{(T1)}$ and $\mathbf{w}^{(T2)}$ after aligning the vector space $V^{(T2)}$ to $V^{(T1)}$ using Orthogonal Procrustes (Hamilton et al., 2016; Schlechtweg et al., 2019, 2020). Denoting $\mathbf{w}_{align}^{(T2)}$ the vector $\mathbf{w}^{(T2)}$ after alignement with Orthogonal Procrustes, we have:

$$\text{DD}(w) = \text{cos-dist}(\mathbf{w}^{(T1)}, \mathbf{w}_{align}^{(T2)}).$$

Using Distances and Frequencies A final step of this process is to add word frequencies for both words at both time periods, as there exist links between usage frequency and semantic change Zipf (1945). We could observe whether adding explicit frequency information helps retrieving discriminatory clues that could be missed by using only distributional representations.

Word frequencies were estimated from the Corpus of Historical American English (COHA) list,[9] which has the advantage to be genre-balanced. As variables for both words and both periods to feed our model, we try to add either raw occurrences counts (indicated by "+FR"), either grouped frequency counts ("+FG"). The procedure to create such groups is described in appendix A.6.

All Features For the sake of comparison to previous models, we evaluate LR models that take as input an implementation of each of these features (SD + DD + frequency); and an even larger model (called "LR multi.") that reunites *all* described implementations of *SD*, *DD* and frequencies.

Non-linear Models As a further step increasing the model's complexity, we try to combine this full set of available variables in a non-linear fashion. We compare previous models to polynomial features (degree 2) preprocessing[10] and a SVM classifier with a Gaussian kernel.

6 Experiments

6.1 Experimental Settings

Target Words Selection We use a unique vocabulary W composed of $6,453$ adjectives, $16,135$ nouns and $10,073$ verbs. The process to select words is described in appendix A.7.

[9]https://www.ngrams.info/download_coha.asp

[10]We also try degrees higher than 2, finding no consistent improvement.

Dataset	ADJ	NN	VERB	ALL	ALL		
Evaluation metric	Balanced Accuracy				$\mathbf{F_1}$*(Syn)*	$\mathbf{F_1}$*(Diff)*	%(D)
All (*Syn*)	.50	.50	.50	.50	.48	0	0
All (*Diff*)	.50	.50	.50	.50	0	.81	100
LR F	.51	.56	.59	.55	.35	.74	75
XK controls	.52	.49	.51	.50	.33	.67	65
Δ (cd)	.50	.49	.51	.50	.27	.73	75
Δ (nk)	.48	.49	.49	.50	.32	.67	66
Δ (tuned τ)	.51	.52	.52	.51	.27	.74	79
LR SD	.60	.62	.59	.60	.48	.69	56
LR SD + DD	.61	.62	.60	.60	.48	.69	56
LR SD + F	.61	**.64**	.63	**.62**	.51	.71	57
LR SD + DD + F	**.62**	**.64**	.63	**.62**	.50	.70	57
LR multi	**.62**	**.64**	**.65**	**.62**	.51	.71	57
LR multi. poly. degree (2)	.56	.63	.62	**.62**	.50	.70	60
SVM (gaussian)	.60	**.64**	**.65**	**.62**	.50	.74	63

Table 2: Performances of the different approaches. Results are averaged over 20 random splits.

Dataset Splits For every POS tag, we have a set of word pairs that are synonymous at $T1$. We call *ALL* the dataset that comprises all pairs indistinctly of their POS. These datasets (ADJ,NN,VERB or ALL) are individually shuffled and 33% of their samples (pairs) are set aside for testing. For each dataset, a model is trained on the 66% remaining pairs and evaluated on the test part. Presented results are averaged over 20 random train/test splits.

Hyperparameters We train models with combinations of the different definitions of distances and frequency variables. Choice of synchronic distances was between SD(cd) and SD(nk) with k in $\{5, 10, 15, 20, 40, 100\}$. For DD, we tried neighborhoods with fixed size 100, like Xu and Kemp (2015), and Orthogonal Procrustes with cosine distances. For frequency, the choice is between raw counts and groups. The selected models are detailed in Appendix A.9. The ideal value for the SVM's regularization parameter is found using 5-fold cross-validation over the training set.

Evaluation Metrics We use two standard evaluation metrics: F_1 score and *Balanced Accuracy (BA)*. F_1 scores were computed for both classes, denoting it "$\mathbf{F_1}$*(Syn)*" for *Syns* and "$\mathbf{F_1}$*(Diff)*" for *Diff*. BA is defined as the average of recalls for both classes, and provide a notion of accuracy robust to class imbalance. We also display the percentage of predicted *Diff* ("%D").

Baselines The first two baselines are constant output classifiers, always predicting "*Syn*" or "*Diff*" respectively. They are expected to have a balanced accuracy of 50%, as they would be fully accurate for one class and always wrong for the other. The third baseline (*LR Frequency*) is a Logistic Regression model trained *only* with frequency variables, without any knowledge on the semantic aspect of the pair (neither *SD* or *DD*).

6.2 Results

Performances over the test parts of the different datasets are displayed Table 2.

The first observation is that, in line with the dataset's proportions, all models predict a majority of "Diff", even unsupervised ones (including our reimplementation of Xu & Kemp's control pair selection method). While our task does not directly address the question of the opposition between LD and LPC, this is an empirical clue in favor of LD, contradicting Xu and Kemp (2015). However, predicting the right amount of "Diff" does not guarantee the quality of predictions. Indeed, obtained balanced accuracies range between 0.49 and 0.65.

Considering our unsupervised methods and the Δ (tuned τ), we find no real improvement over baselines. In particular, they fail to outperform the frequency-based baseline model which performs surprisingly well. On the other hand, Logistic Regression and SVM models substantially improve

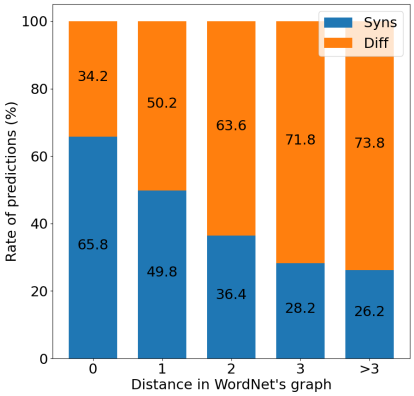

Figure 2: Proportions of predictions of the models w.r.t. the actual distance d in WordNet of *noun* pairs. Pairs with $d = 0$ are synonymous pairs in WordNet.

over the baselines, Xu & Kemp's control pairs and all Δ-based methods. Interestingly, LR SD outperforms Δ-based methods despite the fact that they rely on the same components.

The gap between baselines and models is larger for nouns and lesser for verbs. Despite these POS-specific differences, best models are consistently the ones using both SD and frequencies, while DD brings little to no improvement. This can be expected as individual changes of words seem less important on the problem of *Syn*/*Diff*. However, this factor could be used in future work to distinguish pairs of synonyms (among the *Syn* class) that did not change and pairs that went under LPC.

We observe that there is a substantial difference in F_1 scores between the two classes, $\mathbf{F_1}$*(Syn)* being lower than $\mathbf{F_1}$*(Diff)* across all models. Moreover, models with higher $\mathbf{F_1}$*(Syn)* are often found to be the ones with higher balanced accuracy, even when $\mathbf{F_1}$*(Diff)* is lower. This is likely linked to the fact that the datasets are highly imbalanced as presented in Table 1: the ground truth proportion of *Syn* never exceeds 21%. We also remark that Xu and Kemp (2015) decision rule based on control pairs also predicts a majority of *Diff*, contrarily to the results they showed. It may be because the protocol is not fully identical.

6.3 Confounding Factors

Using WordNet, we discuss two aspects that may be sources of errors when detecting a change in synonymy: polysemy and hypernymy. We study predictions of our best performing LR model on the noun dataset.

Polysemy WordNet provides us with different set of synonyms for every entry, corresponding to different senses or usages, and therefore we can measure the polysemy of a word at $T2$. We found that pairs misclassified as "Syn" tend to be those whose second term has fewer senses (6 senses on average as compared with well classified "Diff" which have 8 senses on average). Indeed, as we use static embeddings and no Word Sense Disambiguation (WSD) method, our model is subject to the complexity brought by polysemy. In a recent shared task about Lexical Semantic Change measures, best performing models are the one using WSD methods (Zamora-Reina et al., 2022). This finding highlights the importance of handling polysemy as a potential confounding factor.

Distances in WordNet In Figure 2 we display the percentage of prediction with respect to shortest distance between the two words of *noun* pairs in WordNet's graph. The distance d is the minimum number of nodes separating the two words.

We remark that, as expected, the model predicts more and more *Diff* as d increases. What is more interesting is that for $d = 1$ (direct hypernymy), there is still an important proportions of predicted *Syn*. This highlights that our model has difficulties to handle hypernymy and confuses it with synonymy.

7 Conclusion

In this work, we considered two contradicting laws about the semantic change of synonyms. We discussed the necessary adaptations of the problem statement for this particular type of LSC and elaborated a framework to evaluate models for this new classification problem. The use of linguistic resources from two different time periods allowed us to improve model analysis with respect to prior work on the matter. Then we proposed unsupervised and supervised approaches relying on measures of semantic change extracted or inspired by existing literature on LSC, and also leveraged the usefulness of explicit word usage frequency information. We compared these approaches in our evaluation framework, finding that distances in vector spaces from different time periods should not be considered equally. We also observed that explicit frequency information actually help distributional methods to capture the change of synonymy. Finally we discussed challenges that DSM approaches still face and opened a discussion about the interplay between hypernymy and synonymy.

Limitations

As mentioned already, the problem *Syn/Diff* does not reflect the initial question of LD/LPC. In particular, the *Syn* class of pairs that remained synonyms contains pairs that underwent LPC and pairs which shared meaning remained unchanged. The latter does not play a role in the LD/LPC dichotomy and should be discarded for deeper study of the two apparently opposite laws. Also, we restrain the study to some target words that are chosen to occur at both time periods, thus preventing us to fully measure the importance of LD. Indeed, recall that Bréal's Law of Differentiation predicts that some synonyms may disappear in the process. Thus, our *Diff* class could be considered incomplete. However, including such disappeared words would prevent the use of time-aware DSMs.

Section 3 presented synonymy as a symmetrical relation between words. However, a thesaurus like Fernald (1896) displays asymmetrical synonymy: for an entry u we have a set of synonyms $v_1, v_2, ...$ from which we extract pairs (u, v). We observe that v itself is rarely an entry of the thesaurus, and when it does, u may not appear in the list of synonyms of v. This is contradictory to WordNet's definition of synonymy that consider this relationship to be symmetrical. However, up to our knowledge, there is no lexical database (like WordNet) being also historical and that could help us ensure the notion of synonymy at both time periods is strictly the same. In the absence of such a resource, we leave potential disagreements in definition between the two linguistic resources to future investigations.

In section 4, we discussed that hyper/hypo-nymy could be misleading. We made the assumption that Fernald (1896) and Wordnet (Fellbaum and Princeton, 2010) used similar-enough notions of synonymy such that our labels *Syn/Diff* are relevant. However, thesaurus like Fernald (1896) are created as a tool for writers and authors to avoid redundancy, thus including wide lists of synonyms that include hypernyms (instead of repeating *the bench*, you could say *the seat*). In section 6.3 we showed that direct hypernymy is misleading for our model. Yet, we still miss guidelines/insights about the possibility to include some cases of hypernymy among synonyms at $T2$. Another approach would be to remove hypernyms from the source material at $T1$, which implies to automatically detect them or manually review thousands of pairs.

There are remaining factors that presented approaches do not take in account and that one could think relevant. In particular, further work could investigate the influence of pressure of words on a concept, for instance many words sharing (at least partially) a similar meaning. However, this would require access to list of senses for each word at time $T1$, which we do not have in Fernald (1896). To this extent, contextualized language models fine-tuned for the different time periods could be helpful.

Finally, because we used pre-computed SGNS embeddings on historical data binned in decade, we have no guarantee that this is the optimal setting for studying Lexical Semantic Change. Maybe different kind of changes could be observed using larger or smaller time periods, and conducting the study over a larger or a smaller time span instead of just a century.

Acknowledgements

We would like to thank the three anonymous reviewers for their helpful comments on this paper. We would also thank Anne Carlier for the thoughtful discussion about this work. This research was funded by Inria Exploratory Action COMANCHE.

References

Michel Bréal. 1897. *Essai de Sémantique*. Paris: Hachette.

Eve V. Clark. 1993. *Conventionality and contrast*, Cambridge Studies in Linguistics, page 67–83. Cambridge University Press.

Haim Dubossarsky, Y. Tsvetkov, C. Dyer, and Eitan Grossman. 2015. A bottom up approach to category mapping and meaning change. *CEUR Workshop Proceedings*, 1347:66–70.

Haim Dubossarsky, Daphna Weinshall, and Eitan Grossman. 2017. Outta control: Laws of semantic change and inherent biases in word representation models. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1136–1145, Copenhagen, Denmark. Association for Computational Linguistics.

Christiane Fellbaum and University of Princeton. 2010. Wordnet. In *About WordNet*. Princeton University.

James Champlin Fernald. 1896. *... English Synonyms and Antonyms*. Funk & Wagnalls Company.

Clémentine Fourrier and Syrielle Montariol. 2022. Caveats of measuring semantic change of cognates and borrowings using multilingual word embeddings.

In *Proceedings of the 3rd Workshop on Computational Approaches to Historical Language Change*, pages 97–112, Dublin, Ireland. Association for Computational Linguistics.

William L. Hamilton, Jure Leskovec, and Dan Jurafsky. 2016. Diachronic word embeddings reveal statistical laws of semantic change. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1489–1501, Berlin, Germany. Association for Computational Linguistics.

Andrey Kutuzov, Erik Velldal, and Lilja Øvrelid. 2022. Contextualized embeddings for semantic change detection: Lessons learned. *ArXiv*, abs/2209.00154.

Adrienne Lehrer. 1985. *The influence of semantic fields on semantic change*, pages 283–296. De Gruyter Mouton, Berlin, New York.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in Neural Information Processing Systems*, volume 26. Curran Associates, Inc.

Julia Rodina and Andrey Kutuzov. 2020. RuSemShift: a dataset of historical lexical semantic change in Russian. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 1037–1047, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Eyal Sagi, Stefan Kaufmann, and Brady Clark. 2012. *Tracing semantic change with Latent Semantic Analysis*, pages 161–183. De Gruyter Mouton, Berlin, Boston.

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 *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 732–746, Florence, Italy. Association for Computational Linguistics.

Dominik Schlechtweg, Barbara McGillivray, Simon Hengchen, Haim Dubossarsky, and Nina Tahmasebi. 2020. SemEval-2020 task 1: Unsupervised lexical semantic change detection. In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, pages 1–23, Barcelona (online). International Committee for Computational Linguistics.

Gustaf Stern. 1921. *Swift, swiftly, and their synonyms: A contribution to semantic analysis and theory*. Wettergren & Kerber.

Nina Tahmasebi, Lars Borina, and Adam Jatowtb. 2021. Survey of computational approaches to lexical semantic change detection. *Computational approaches to semantic change*, 6.

Elizabeth Closs Traugott and Richard B. Dasher. 2001. *Prior and current work on semantic change*, Cambridge Studies in Linguistics, page 51–104. Cambridge University Press.

Peter D. Turney and Saif M. Mohammad. 2019. The natural selection of words: Finding the features of fitness. *PLoS ONE*, 14.

Yang Xu and Charles Kemp. 2015. A computational evaluation of two laws of semantic change. In *Proceedings of the 37th Annual Meeting of the Cognitive Science Society, CogSci 2015, Pasadena, California, USA, July 22-25, 2015*. Cognitive Science Society.

Frank D. Zamora-Reina, Felipe Bravo-Marquez, and Dominik Schlechtweg. 2022. LSCDiscovery: A shared task on semantic change discovery and detection in Spanish. In *Proceedings of the 3rd Workshop on Computational Approaches to Historical Language Change*, pages 149–164, Dublin, Ireland. Association for Computational Linguistics.

George Kingsley Zipf. 1945. The repetition of words, time-perspective, and semantic balance. *The Journal of General Psychology*, 32(1):127–148.

A Appendix

A.1 Formalizing LD and LPC

In this work, we reduced the problem from finding pairs in which LD or LPC operates to a binary classification problem between pairs that remained synonymous and those who did not. To understand the need for a reduction, let us introduce some notation and definitions.

First, let us denote by $W^{(T)}$ the set of words (or vocabulary) for a given language (say English) at time T. As language evolves through time, vocabularies at two times $T1$ and $T2$ (with $T2 > T1$) need not have the exact same extensions: e.g., a word w in $W^{(T1)}$ might not be in $W^{(T2)}$ (i.e., w has disappeared) or vice versa (i.e., w is a new word). Assuming a simple, idealized denotational semantics, we will further define $\mathcal{C}^{(T)}$ as the set of discrete concepts available at time T,[11] and $M_w^{(T)} \subset \mathcal{C}$ the meaning of word w at time T. It is defined as a set to model cases of homonymy and/or polysemy. From these definitions, we can now define *synonymy* at time T between words $u \in W^{(T)}$ and $v \in W^{(T)}$ as $M_u^{(T)} \cap M_v^{(T)} \neq \emptyset$; that is, u and v do share a common meaning. Furthermore, we can define the *semantic change* from $T1$ to $T2$ in a word w as follows: $M_w^{(T1)} \neq M_w^{(T2)}$; that is, w has different sets of meanings at $T1$ and $T2$.

[11] We take $\mathcal{C}^{(T)}$ to be mostly stable over time, but new concepts might of course appear or disappear (e.g., due to techonological or cultural evolution).

Equipped with these definitions, we are now ready to formalize the two laws LD and LPC, starting with what their common scope.

First, both laws concern synonyms: they are restricted to a set of synonyms at some initial time $T1$, defined by $\mathcal{S}^{(T1)} = \{(u, v) : M_u^{(T1)} \cap M_v^{(T1)} \neq \emptyset\}$.

Second, both LD and LPC assume some individual semantic change, from $T1$ to $T2$ (with $T2 > T1$), in at least one of two synonymous words: that is, $M_u^{(T1)} \neq M_u^{(T2)}$ or (logical) $M_v^{(T1)} \neq M_v^{(T2)}$.

Given these preconditions, the application of LD implies that either:

- one of the two words has disappeared: $u \in W^{(T1)} \wedge u \notin W^{(T2)}$ or (exclusive) $v \in W^{(T1)} \wedge v \notin W^{(T2)}$,
- u and v are no longer synonymous at $T2$: $M_u^{(T1)} \cap M_v^{(T1)} = \emptyset$.

By contrast, LPC implies that words u and v remain synonymous from $T1$ to $T2$. While this could be simply stated as: $M_u^{(T2)} \cap M_v^{(T2)} \neq \emptyset$, we feel that this misses an important aspect of the law, namely that $M_u^{(T1)}$ and $M_v^{(T1)}$ should evolve in the same way:

- either by acquiring (a) new shared sense(s): $(M_u^{(T2)} - M_u^{(T1)}) \cap (M_v^{(T2)} - M_v^{(T1)}) \neq \emptyset$,
- or inversely by losing the same sense(s): $(M_u^{(T1)} - M_u^{(T2)}) \cap (M_v^{(T1)} - M_v^{(T2)}) \neq \emptyset$.

A.2 Useful definitions

Recall the definition of *cosine distance* between two vectors $\mathbf{x}$ and $\mathbf{y}$:

$$\text{cos-dist}(\mathbf{x}, \mathbf{y}) = 1 - \frac{\langle \mathbf{x}, \mathbf{y} \rangle}{\|\mathbf{x}\| \|\mathbf{y}\|}. \quad (3)$$

We also recall the definition of *Jaccard distance* between two sets A and B:

$$\text{jaccard-dist}(A, B) = 1 - \frac{|A \cap B|}{|A \cup B|}. \quad (4)$$

A.3 Xu & Kemp's control pairs

In Table 3 we display samples of word pairs selected as control pairs following Xu and Kemp (2015)'s procedure. As we can observe, for every Part-Of-Speech, a significant number of these pairs are themselves synonymous. After manually reviewing a hundred pairs for each POS tag, we estimate that the proportion of synonyms in the selected control pairs is between 20 and 40%. Synonym pairs shouldn't be used to control other synonym pairs, which may explain why our reproduction of Xu and Kemp (2015) decision rule does not perform well according to Table 2.

A.4 Distances in WordNet

In Figure 3 are displayed the distributions of distances in WordNet. The distance in WordNet between two words (u, v) is the number of nodes of the shortest path between a synset of u and a synset of v.

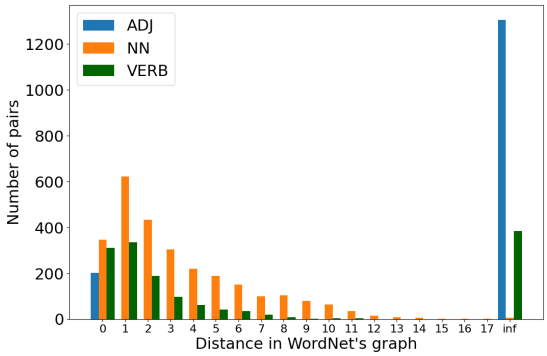

Figure 3: Distribution of shortest distances in WordNet between pairs of words that were synonymous at $T = O$. *inf* means that there is no path between the two words in WN. A distance of 0 means that they are actually synonyms, while a distance of 1 implies there is direct hypernymy.

A.5 Synonymy in our DSMs

In Figure 4 are displayed the distributions of cosine distance between word pairs at both periods. In blue are synonyms at this time (from Fernald (1896) at $T1$, and from WordNet at $T2$). In black are all possible word pairs. We observe that synonymy is indeed captured by our DSM as synonyms are significantly closer in cosine distance than other word pairs.

A.6 Frequency groups

The procedure to create a fixed number M of frequency group is the following. At a time T, the list of target words is sorted by increasing frequency, we label as group '0' the first 50% of the list. In the remaining 50%, The first half is labeled as group '1', and so on until group $M - 2$ is created. The still unlabeled words are labeled group $M - 1$, for a total of M groups. Group labels are therefore positively correlated with occurrences counts.

POS	Control pairs
ADJ	brownish/red, kindly/mild, teeming/agricultural, likeliest/meaningless, *various/heterogeneous*, *barbarous/cruel*, *abandoned/unsuccessful*, trojan/escaping, *subjective/relative*, reliable/readable.
NN	diphtheria/typhus, *muskets/pistol*, surgery/appendicitis, beech/apples, accountants/prints, commodity/substances, *cups/pots*, wife/grandmother, fool/fisherman, obstacles/multiplication.
VERB	*moan/groan*, divide/span, needed/secured, *flowed/flooded*, stall/owned, *told/asked*, *mentioned/described*, cooperate/accord, copy/filed, increased/diminished.

Table 3: Random samples of size 10 among selected control pairs. In italic are control pairs which are considered synonyms according to the definition in Section 3.1.

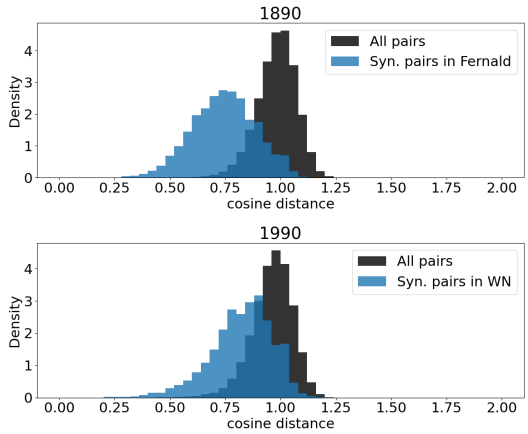

Figure 4: Distribution (as density histograms) of cosine distances between word pairs at time period $T1$ (decade 1890s) and $T2$ (decade 1990s). In blue are represented pairs of synonyms, and in black are represented all pairs of target words, without any particular constraint.

A.7 Target words selection

Among words represented in the embeddings provided by Hamilton et al. (2016), we keep only words following these three requirements. The first is to be POS-tagged as an *adjective*, a *noun* and/or as a *verb* in the COHA. For a given POS-tag among these three, the second requirement is to appear at least 3 times in every decade between 1890 and 1999. Lastly, we require words to be composed of 3 letters or more. If a word appears with multiple POS-tags in the COHA and fulfills the minimum frequency requirement with each of these tags, the same embedding is used as its representation, as Hamilton et al. (2016)'s training data aggregated POS-tags.

A.8 Unsupervised models

In Figure 5, we observe that the quantity Δ does not reflect a clear separation between *Syn* pairs and *Diff* pairs. This explains why the unsupervised methods proposed in Sec. 5.2 fail to significantly outperform baselines.

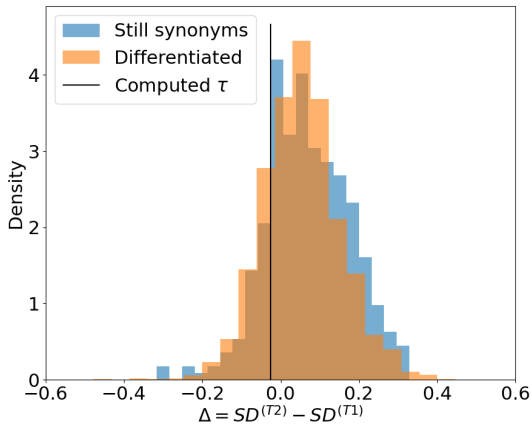

Figure 5: Histograms of the value of divergence Δ of synonymous pairs, depending whether they differentiated (orange) or stayed synonyms (blue).

In Figure 6 we show the influence of k in SD (neighbors) for the unsupervised Δ method. We see that while there is close to no change in balance accuracy, F_1 scores for both classes are more and more unbalanced as k increases, indicating a more unfair model for high values of k. This is explained by the fact that the unsupervised model predicts more *Diff* (dominant class) with higher k.

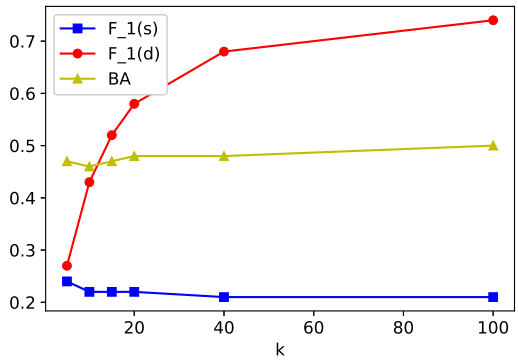

Figure 6: Unsupervised method, neighborhood-based SD, for ALL (mixed POS).

A.9 Components of selected models

Depending on the POS tag, the implementation strategies of *SD*, *DD* and frequency variables were different. Recall that these strategies were chosen given the average performances over 20 random train/test splits.

On adjectives, neighborhood based *DD* as well as raw frequency counts was found to be better than alternatives. For *SD*, cosine distance provides slightly higher performances than neighborhood-based measures, except when tuning the threshold for Δ: in this case, SD(nk) with $k = 15$ was best. Generally, for every method implying a neighborhood based SD (Δ(nk), LR multi., as well as the two non-linear models), a small/mid ranged k was preferable (between 10 and 20).

For nouns, SD(cosine distance) was also the best choice except for Δ with tuned threshold: here, SD(nk) was preferred. Overall, the best range for the value of k for neighbors-based SD was smaller (5 to 15). Frequency groups worked better than raw frequencies, while there was no difference in performance between the two definitions of DD.

Yet, for verbs, SD(nk) with $k = 40$ actually outperforms cosine distance (except for unsupervised Δ), and DD using Orthogonal Procrustes alignment and cosine distance (Hamilton et al., 2016) was actually better than the definition relying on comparisons local neighborhoods. Both types of frequency variables (raw counts and groups) worked equally well.

Finally, on the ALL dataset reuniting pairs across POS tags, raw frequencies provide better results than groups. Cosine distance is better than neighborhoods for synchronic distances, and both techniques of diachronic distances performed similarly. For models forced to use SD(nk) in addition to SD(cd), the choice of k did not really change the results.

A.10 Predictive variables in our model

In this supplementary section, we conduct a study about the role of some predictive variables in our best-performing Logistic Regression model, as potential sources of errors. The studied model uses SD with cosine-distance, both implementations of DD and raw frequency counts.

	Pred.		$y =$ Syn		$y =$ Diff	
	Syn	Diff	TS	FD	TD	FS
$SD^{(TT1)}$	**.64**	**.83**	**.62**	**.84**	**.83**	**.64**
$DD(u)$	.46	.46	.45	.47	.46	.47
$DD(v)$	**.50**	**.54**	**.48**	**.54**	**.54**	**.50**
$FG_u^{(T2)}$	2.3	2.2	2.4	2.1	2.2	2.2
$FG_v^{(T2)}$	**1.9**	**1.4**	**2.1**	**1.5**	**1.4**	**1.8**

Table 4: Average values of some variables for data subset based on the prediction of our best-performing LR model. TS,FS,TD,FD stand for True/False Syn/Diff. $FG_w^{(T)}$ stands for Frequency Groups of word w at time T. Significant difference within a pair of columns are in bold.

For a selected number of variables, we look for significant differences between well-classified pairs and pairs with wrong prediction, in both classes separately. For a given variable, we estimate if a difference is significant between the well-classified and the misclassified samples of this class using a t-test for Gaussian distributed variables, or a Mann-Whitney U test for other variables. A difference is significant if the p-value of the test is below 5%. Results are reported in table 4.

We observe significant differences of SD in pairs that are predicted as *Syn* and those predicted as *Diff* by our model, the first having a smaller SD at $T1$ than the latter. Because our model relies mostly on these SD to separate both classes, we wrongly classify *Syn* pairs whose $SD^{(TT1)}$ is close to that of *Diff*, and conversely *Diff* pairs whose $SD^{(TT1)}$ is close to that of *Syn* are misclassified. This indicates that our model still misses some subtleties that are now reflected by SD.

A similar non-separability of the distribution of "Syns" and "Diff" appears on DD and Frequency variable for the second word pair of the pair. While it seems logical for our model to behave so regarding to the definition of LD, it is a clue that our input

variables reflect noisy information that is confusing to the model. In the same idea, Kutuzov et al. (2022) remarked that recent LSC detection models tend to raise False Positive, drawing attention to the limit of current models for LSC.