# A Model of the Language Process

**Brandon Duderstadt**
Calcifer Computing
brandon@calcifercomputing.com

**Hayden Helm**
Helivan
hayden@helivan.io

## Abstract

Language is a process that changes over time as new vocabulary emerges, word meanings shift, and narratives progress. Despite this fact, most Large Language Models are trained on corpora that lack explicit temporal information, which inhibits their ability to model the language process. In this work, we introduce the Temporal Language Model 1 (TLM-1), a BERT style transformer encoder that models that language process by jointly learning to predict document contents and classify document publication dates. We also introduce a Bayesian framework for querying TLM-1 that disentangles its temporal dynamics from several sources of anachronism. Using this query framework, we demonstrate that TLM-1 effectively surfaces several sociolinguistic trends in contemporary American English and accurately detects semantic changes in word meanings. Furthermore, we perform a mechanistic analysis of TLM-1's time token embeddings, and find that they learn a curve whose geometry recovers the ordinal progression of time. We take the existence of this curve as evidence that TLM-1 is effectively learning to reconstruct temporal language dynamics.

## 1 Introduction

Language is a process (Deleuze and Guattari, 1987). As we use language to communicate, new vocabulary emerges, word meanings shift, and narratives progress. This evolution endows language with an inherent temporal structure. Traditional Large Language Models (LLMs) do not explicitly account for this temporal structure. Instead, they treat documents in their training data as if they all occurred at once.

In this report, we introduce the Temporal Language Model 1 (TLM-1), a BERT style transformer that directly models the language process by jointly learning to predict document contents and classify document publication dates. We train TLM-1 on a general-purpose monitor corpus of contemporary American English, enabling us to probe it for temporal trends in language relevant to the United States from 1990 to 2019.

To query TLM-1, we introduce a Bayesian framework that disentangles its temporal dynamics from several sources of temporal bias, including base model anachronism and query anachronism. We demonstrate that our framework can recover temporally sensitive relationships that are otherwise hidden when naïvely evaluating the model's likelihood function. Using our query framework, we investigate several sociolinguistic trends relevant to contemporary American English.

We also perform a geometric analysis of TLM-1's learned time embeddings. We show empirically that TLM-1's learned time embeddings recover a 1D curve indexed by time, which we call the temporal control curve. We argue that the existence of the temporal control curve provides evidence that TLM-1 is able to effectively reconstruct temporal language dynamics.

We conclude with a discussion of the implications of our work and possible next steps.

## 2 Background

### 2.1 Language Modeling

Early large-scale encoder models like BERT (Devlin et al., 2019) and RoBERTa (Liu et al., 2019) established the transformer architecture (Vaswani et al., 2023) and the masked language modeling (MLM) objective as the dominant pre-training paradigm for encoder models. More recent encoder variants, such as the Etin400M Model (Weller et al., 2025), provide contemporary encoder base models with improved performance and efficient scaling.

Another improvement to the transformer encoder training stack is SpanBERT (Joshi et al., 2020), which introduced span-level masking to better capture multi-token dependencies. This feature is crit-

ical for handling words that may fragment into multiple subword units at tokenization time.

### 2.2 Temporal Language Modeling

The literature on temporal language modeling overwhelmingly focuses on the task of detecting changes in word senses. This task, known as diachronic semantic change detection, has been an active area of study for over a decade. Kutuzov et al. (Kutuzov et al., 2018) provide the most recent survey paper on the field, which details the progression from classical N-Gram models (Michel et al., 2011), to neural word models requiring explicit temporal alignment (Kulkarni et al., 2014), to more sophisticated neural models that jointly learn word embeddings and temporal alignments (Bamler and Mandt, 2017; Yao et al., 2018).

TempoBERT (Rosin et al., 2022) was the first serious attempt to adapt the self supervised transformer paradigm to explicitly include document time information. It introduced the idea of jointly modeling document dates and content by prepending time tokens to documents. In the TempoBERT paper, the authors train several different, highly specific TempoBERT models for targeted tasks, such as detecting semantic change in Reddit comments about football or classifying news article publication dates. While the TempoBERT approach is promising, the narrow training data of each TempoBERT instance makes it unsuitable for generalized temporal language modeling.

## 3 Training Procedure

The Temporal Langage Model 1 (TLM-1) draws heavily on the training procedure of TempoBERT, but opts to train on a general purpose monitor corpus of American English as opposed to narrow, task specific corpora. This approach preserves the scalability and generality of BERT-style models while explicitly integrating temporal information into the pretraining loss.

Unlike previous approaches, TLM-1 does not assume time sensitivity must be engineered for a narrow domain. Instead, it treats temporal modeling as a first-class, general-purpose model capability.

### 3.1 Loss Function

The goal of TLM-1 is to create a general-purpose model for temporal language tasks by jointly modeling document contents and dates. For this purpose, we begin by considering a general form of the joint temporal-content loss function:

$$\mathcal{L}_{TC} = \lambda_1 \mathcal{L}_T + \lambda_2 \mathcal{L}_C$$

where $\mathcal{L}_T$ is a temporal modeling loss, $\mathcal{L}_C$ is a content modeling loss, and $\lambda_1$ and $\lambda_2$ are hyperparameters.

$\mathcal{L}_{TC}$ generalizes several common encoder losses. Let $S$ be the length of the sequence being encoded, $\text{MLM}_p$ be the masked language modeling objective with token mask rate $p$, and $\text{SpanMLM}_{p;l}$ be the span masked language modeling objective as presented in SpanBERT with truncated geometric distribution parameters $p$ and $l$. Under these definitions, the RoBERTa, SpanBERT, and TempoBERT loss functions are all instances of $\mathcal{L}_{TC}$, as shown in Table 1.

The TLM-1 loss is also a variant of $\mathcal{L}_{TC}$, where $\lambda = 0.5$, $\mathcal{L}_T = \text{MLM}_{p=0.9}$, $\lambda_2 = 1.0$, and $\mathcal{L}_C = \text{SpanMLM*}_{p=0.2;l=4}$. We can gain intuition for this loss function by breaking it down term by term.

TLM-1's temporal loss, $\mathcal{L}_T$, is similar to the TempoBERT temporal loss. Time tokens are added to the model's vocabulary, prepended to documents, and masked out at a rate of $0.9$. TLM-1 sets $\lambda_1 = 0.5$, which significantly upweights the loss from temporal modeling compared to TempoBERT. Empirically, we found that this was necessary for the model to perform sufficiently well on the document dating task.

TLM-1's content loss, $\mathcal{L}_C$ is similar to the SpanBERT content loss. The decision to use a span-based loss for TLM-1, as opposed to the MLM loss in TempoBERT, is purely mechanical. TempoBERT goes to great lengths to avoid splitting words that are of interest to them at query time, even going so far as to add all relevant words to their tokenizer before training begins.

The TempoBERT setup is unrealistic if we hope to build a model that is useful for general-purpose historical linguistics, as we don't know a priori what words will be of interest to users at query time. By using a span-based objective, we provide a training setup where TLM-1 regularly sees short sequences of masked tokens. This is similar to what the model will see if a word of interest gets split into multiple mask tokens at query time.

We make two further modifications to the SpanMLM loss relative to SpanBERT. First, we eliminate the span boundary objective present in SpanBERT for the sake of simplicity. We write SpanMLM* to denote the SpanBERT loss without the span boundary term. Second, we reduce

| Model | $\lambda_1$ | $\mathcal{L}_T$ | $\lambda_2$ | $\mathcal{L}_C$ |
|---|---|---|---|---|
| RoBERTa | 0.0 | N/A | 1.0 | $\text{MLM}_{p=0.15}$ |
| SpanBERT | 0.0 | N/A | 1.0 | $\text{SpanMLM}_{p=0.2;\, l=10}$ |
| TempoBERT | $\frac{1}{S}$ | $\text{MLM}_{p=0.9}$ | $\frac{S-1}{S}$ | $\text{MLM}_{p=0.15}$ |
| TLM-1 | 0.5 | $\text{MLM}_{p=0.9}$ | 1.0 | $\text{SpanMLM*}_{p=0.2;\, l=4}$ |

Table 1: Comparison of temporal ($\mathcal{L}_T$) and contextual ($\mathcal{L}_C$) loss weighting schemes across models. The RoBERTa and SpanBERT objectives can be recovered by setting the temporal loss modeling weight ($\lambda_1$) to 0. TempoBert implicitly sets $\lambda_1 = 1/S$ where $S$ is the sequence length. TLM-1 uses the SpanMLM loss without the span boundary term, which we denote SpanMLM*.

the maximum of the span length distribution from 10 to 4, which we believe more closely aligns with our goal of preparing TLM-1 to cope with words of interest that may be split into multiple tokens.

### 3.2 Dataset

We train TLM-1 on the Corpus of Contemporary American English (COCA) (Davies, 2020). COCA is an English monitor corpus that contains 1 billion words of English text in dated sequences written between 1990 and 2019. The corpus is composed of 8 different genres: spoken words, fiction, magazines, newspapers, academic texts, television and movie subtitles, blogs, and other web pages.

There are several important considerations we need to make when modeling the COCA corpus. First, all of the articles in the blog and other web page genres are sequences from 2012, resulting in a large temporal and topical imbalance. We remove these genres from the TLM-1 train set to avoid this imbalance.

Second, every copy of COCA has a "fingerprint" where a small percentage of words in the corpus are replaced with a sequence of 10 @ signs. The goal of this fingerprint is to enable the corpus author to track pirated versions of the corpus back to their original purchaser. We can model this quite naturally by tokenizing the sequences of 10 @ signs as a special token that acts like a mask token but does not receive gradients.

Finally, sequences in COCA are timestamped with yearly granularity. As a result, TLM-1 adds 30 time tokens to its vocabulary, one for each year in COCA. Every sequence in COCA is prepended with its corresponding time token during training.

### 3.3 Other Training Details

After removing the web and blog genres, the COCA corpus contains about 750 million words. This is too little data to train a reasonably sized encoder model from a random initialization; for comparison, the original BERT was trained on about 3.3 billion words. As a result, we use a similar approach to TempoBERT and fine tune a base model using our dataset. We opt for Ettin400M, a contemporary encoder architecture with strong performance and efficient scaling, as our base model.

When expanding the Ettin400M vocabulary to include time tokens, we found that the default initialization (Hewitt, 2021) for new tokens invoked by Hugging Face Transformers' (Wolf et al., 2020) resize token embeddings function lead to training instability. We were able to remedy this by initializing all time tokens to have the same embedding as the space token.

We optimized our model for 2 epochs using the AdamW (Loshchilov and Hutter, 2019) optimizer with the default parameters of $\beta_1 = 0.9$, $\beta_2 = 0.999$, and $\epsilon = 1e^{-8}$. We train our model on a single H100 in bf16 precision to accommodate a batch size of 64. We use a gradient accumulation of 8 to reduce gradient variance. We use a linear learning rate schedule that warms up to $1e^{-4}$ over 5k steps before linearly decaying back to 0.

Over the course of training, our model loss drops from a peak of 11.8 to a minimum of 2.8. Our final model achieves 54% top-1 content infill accuracy and 70% top-1 time token infill accuracy. (See Appendix B for detailed training curves). While we believe these metrics can be improved significantly given additional data and compute, our empirical investigations show they are sufficient for practical use of TLM-1.

Crucially, we do not claim the optimality of any part of our training procedure. Extensive, and computationally intensive, ablations would be required to make such claims.

## 4 Query Framework

The traditional query methodology for masked language models involves evaluating the probability

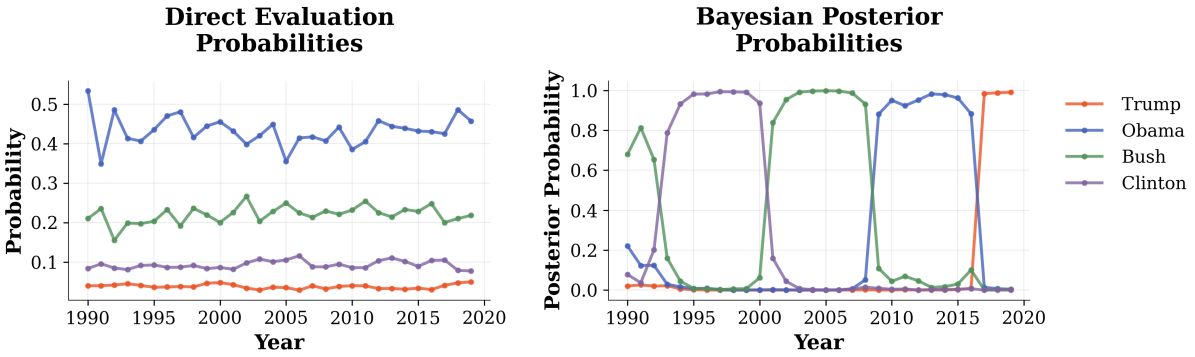


Figure 1: Visualization of the fill probabilities under direct model evaluation and the Bayesian query framework outlined in Section 4 for the context "President [MASK] made a speech today." When directly evaluating $P(F|C;T)$ using TLM-1's learned likelihood function, the anachronism in the base model overwhelms the temporal dynamics of the word President. Under the Bayesian query framework with a uniform nucleus prior over the top 4 fills, TLM-1 recovers the temporally sensitive nature of the word President (right).

of a fill $F$ when given a context $C$, or $P(F|C)$. As a result, it is tempting to query TLM-1 by directly evaluating the probability of a fill $F$ when given a context $C$ and a time $T$, or $P(F|C;T)$. However, this method of querying TLM-1 is vulnerable to several sources of anachronism. To understand why, first apply Bayes' Rule:

$$P(F|C;T) = \frac{P(T|F;C)}{P(T|C)} P(F|C). \quad (1)$$

From Eq. (1), we see that $P(F|C;T)$ depends heavily on $P(F|C)$, or the prior probability of a fill given a particular context, independent of time. Moreover, temporally imbalanced training datasets or temporally insensitive base models will have a large effect on $P(F|C)$, thereby complicating temporal analysis. For TLM-1 specifically, anachronism introduced by the temporally insensitive Etin400M base model has a distorting effect on $P(F|C)$. We can correct for this by replacing $P(F|C)$ with a more appropriate prior.

There are two main considerations when choosing a prior. First, we want to ensure that anachronism in the base model or imbalance in the training set do not overwhelm the temporal information in the fill. Second, we do not want to admit fills that have strong temporal relevance but are nonsensical in the provided context. To achieve both of these goals, we adopt the uniform nucleus prior:

$$\tilde{P}(F = f|C) = \begin{cases} 1/|\mathcal{F}| & f \in \mathcal{F} \\ 0 & \text{else} \end{cases}$$

where $\mathcal{F}$ denotes a set of feasible fills. In practice, the set of feasible fills can be selected using the the top-k fills surfaced by the empirical $P(F|C)$ distribution, or could be manually set if the user is investigating a particular phenomenon.

Figure 1 shows the direct evaluation and posterior fill probabilities for the context "President [MASK] made a speech today." Directly evaluating the TLM-1 likelihood function (left) always predicts Obama as the fill, presumably due to training-set imbalance or anachronism introduced by the Ettin base model. In contrast, the Bayesian posterior (right) with a uniform nucleus prior over the top 4 fills recovers the term of each president.

Each term in our Bayesian query framework has a natural interpretation. As previously discussed, $P(F|C)$ models the prior probability of a fill given a context, independent of temporal information.

The numerator of the Bayes Factor, $P(T|F;C)$, models the document date distribution given the complete content of the document. We estimate $P(T|F;C)$ by querying TLM-1 with a document that only has its time token masked out, and obtaining a distribution over the time token's fills.

The denominator of the Bayes Factor, $P(T|C)$, models the document date distribution when given the context alone. We estimate $P(T|C)$ by querying TLM-1 with a document that has both its time token and fill slot masked out, and obtaining a distribution over the time token's fills. $P(T|C)$ acts as a normalizing factor, enabling users of TLM-1 to query the model for different fill probabilities without worrying about anachronism introduced in the wording of the context itself.

With this query methodology at hand, we can now progress to using TLM-1 to quantitatively investigate the language process.

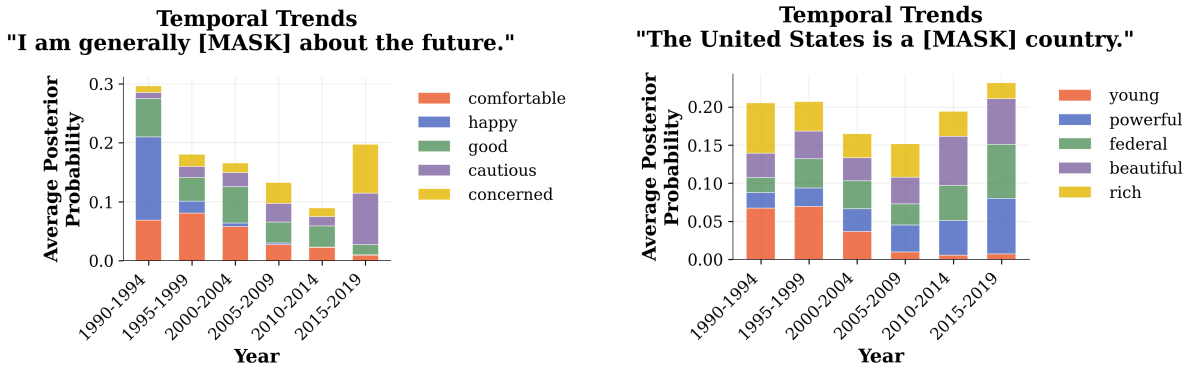


Figure 2: Visualizations of the temporal change in TLM-1's posterior probabilities for the contexts: "I am generally [MASK] about the future" (left), and "The United States is a [MASK] country" (right). These posterior trajectories reflect several known trends in United States culture from 1990 through 2019, including an aging population base, an expansion of federal power, and a population that is increasingly pessimistic about the future.

## 5 Investigating the Language Process

When using TLM-1 to quantify the language process, we must remember that all linguistics is corpus linguistics. By this, we mean that the language process captured by TLM-1 is fundamentally an artifact of the corpus it was trained on; therefore, we must be careful about extrapolating TLM-1 results to populations whose language may not be captured in COCA. Despite this, we believe that COCA is a sufficiently broad and complete monitor corpus for enabling TLM-1 to capture several trends relevant to United States culture from 1990 to 2019.

### 5.1 The Long Arc

A natural place to begin our investigation of the language process is surfacing posteriors that have an approximately monotonic trend over the entire time period covered by the corpus. We refer to this style of investigation as "Long Arc" investigation, as it captures the most slow-moving but persistent sociolinguistic trends.

Take, for example, the phrase "I am generally [MASK] about the future." Understanding how the posterior distribution of this phrase changes over time could reveal important information about the future facing sentiment of the corpus population. To do this, we compute a posterior over mask fills using a top-25 uniform nucleus prior for each year and compute the absolute correlation between each fill's posterior share trajectory and time. Figure 2 (left) visualizes the fills with the 5 highest absolute correlations with time.

From Figure 2 (left), we see the probability that attitudes about the future are described as "happy," "good," or "comfortable" decreases starkly over time. Accordingly, we see the probability that attitudes about the future are described as "concerned" or "cautious" increases starkly over time. Based on this, we conclude that the general sentiment towards the future in the United States has became more negative over the period from 1990 to 2019.

TLM-1 also enables us to query how the language used to describe entities of interest changes over time. Consider the same procedure as above, but applied to the phrase "The United States is a [MASK] country," as shown in Figure 2 (right).

We see several key narratives reflected in this posterior. First, we see a stark decrease in the probability that the United States is described as a "young" country, reflecting the fact that the United States is facing a rapidly aging population (FRED, 2025).

Second, we see a stark increase in the probability that the United States is described as a "federal" country. This could indicate a broad trend toward centralized and expanded federal power, perhaps through the increasing scope of the executive order (Peterson, 2019).

Third, we see a shift away from describing the United States as "rich" and toward describing it as "powerful." This could indicate that the source of US power is shifting away from economic dominance and toward force projection. We speculate that a potential cause of this could be the increasingly precarious U.S. federal debt situation.

Overall, we believe that these investigations demonstrate how TLM-1 can be used to perform "Long Arc" analysis on temporal corpora.

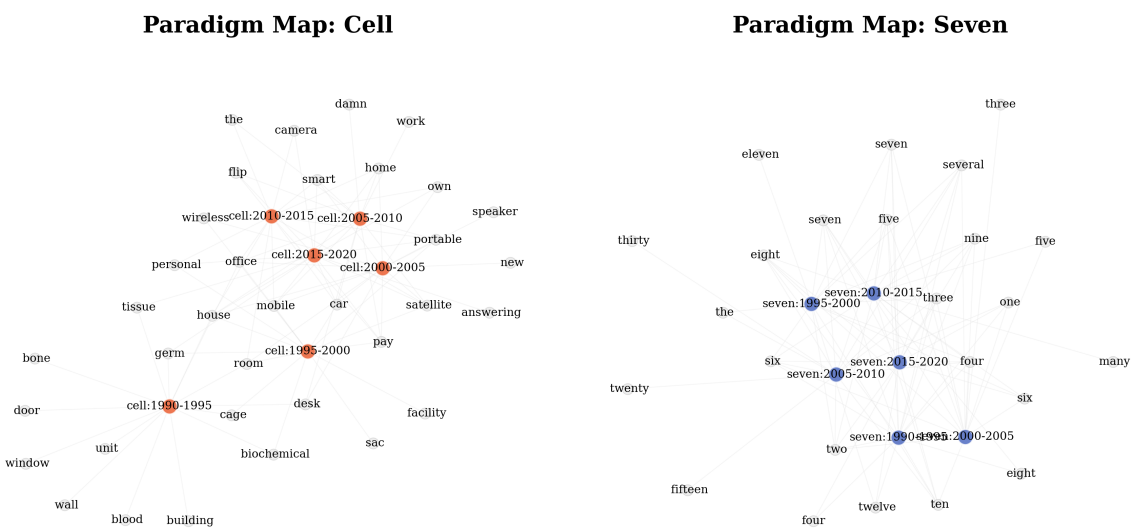


Figure 3: Visualizations of the paradigm maps for the words "cell" (left) and "seven" (right). In the 1990s, "cell" was primarily used in paradigms where alternative fills referred to biology (e.g. germ, tissue, bone) or containment (cage, building, wall). By the 2000s "cell" was being used in paradigms where alternative fills referred to phones (e.g. flip, camera, smart). This shift results in cell's paradigm map having two distinct visual communities. Compare this to the paradigm map for "seven" (right), a known semantically stable word whose paradigm map exhibits only one visual community.

### 5.2 Diachronic Semantic Change

Another application of TLM-1 is the diachronic semantic change detection task, which involves detecting if a word changes its meaning over time. We approach this task through the lens of Saussure's linguistic paradigm (Saussure, 1959). In structural linguistics, a paradigm refers to a set of words that can be substituted for a target word in a given phrase. Historically, paradigms were used to reason about how the meaning of a sentence changes as different words within the same paradigm were substituted with the target word.

In our work, we concern ourselves with determining if the target word's paradigm distribution is changing over time. We assume that a change in the paradigm distribution of a word is a sufficient condition for the semantic change of that word.

As an example, we can consider the target word "cell", which is known to have undergone a semantic change from 1990 to 2019. Before the introduction of the cell phone, the word "cell" primarily occurred in biological and physical containment contexts (e.g., the mitochondria is the powerhouse of the cell; Jean Valjean was locked away in a jail cell). As a result, the pre-2000s paradigms for the word "cell" will contain primarily biological and physical containment words. (e.g., the mitochondria is the powerhouse of the body; Jean Valjean was locked away in a jail house). After the introduction of the cell phone, the word "cell" acquired a new sense. The paradigm distribution for the word "cell" shifted as a result of this new sense, and words from its newly acquired paradigm became feasible alternative fills.

We can computationally model a word's paradigm distribution by tracking the frequency of its alternative fills using TLM-1. To do this, we begin by mining a uniform random sample of uses of a target word from the COCA corpus. Then, for each mined use, we mask the target word and compute a posterior over alternative fills using TLM-1 and a uniform nucleus prior. Next, we aggregate these posteriors into time buckets by using a uniform distribution over documents within a time bucket.

We can organize the results of this process into a matrix $M \in [0,1]^{|B|\times|F|}$, where $|B|$ is the number of time buckets, and $|F|$ is the size of the set of all feasible alternative fills.

Formally, let $d = (c, t)$ be a particular document containing a context $c$ at time $t$. Let $b_i$ denote the $i$th time bucket, and $P(D = d|B = b_i) = 1/|b_i|$ denote the uniform probability of selecting a particular document $d$ from time bucket $b_i$. Let $f_j$

denote a particular fill in $F$. Then let

$$M_{ij} = \sum_{d \in b_i} P(f_j|d)P(d|b_i).$$

The matrix $M$ can be interpreted as a weighted bipartite graph $G$, where one set of nodes corresponds to the set of time buckets $B$, another set of nodes corresponds to the set of fills $F$, and the weight of an edge $ij$ corresponds to the probability that a fill $f_j$ is substituted for our target word in time period $b_i$.

This interpretation enables us to transform questions about the change in the paradigm distribution of a target word into questions about the community structure of $G$. Intuitively, if $b_i$ and $b_j$ reside in different communities in $G$, then there is evidence that the alternative fill distribution is sensitive to the choice of time bucket. We call $G$ a paradigm graph, and write $G_{\text{target}}$ to indicate the paradigm graph for a particular target word.

We can visualize the structure of a paradigm graph using a force directed layout. We call the result of applying a force directed layout to a paradigm graph a paradigm map, and outline a detailed procedure for their generation in Appendix A.

Figure 3 shows the paradigm maps for the words "cell" and "seven". There are two distinct visual communities in the paradigm map for $G_{\text{cell}}$; one roughly centered around cell:1990-1995, and another roughly centered around cell:2015-2020. This indicates that the distribution of words that act as alternates for the word cell has changed over time, providing evidence for semantic change. Contrast this with the paradigm map for seven, which exhibits only one visual community.

We can formalize this notion of "visual communities" by computing the modularity of our paradigm graphs. Formally, let $Q : G \to [-0.5, 1)$ be the Clauset-Newman-Moore greedy modularity function (Clauset et al., 2004). Paradigm graphs with a high $Q$ exhibit more prominent community structure, indicating that their target word has undergone a more prominent change in its paradigm distribution.

### 5.3 Evaluating Semantic Change Detection

Benchmarking the semantic change detection performance of a model is challenging due to the lack of strong standardized datasets for the task (Kutuzov et al., 2018; Dubossarsky et al., 2019; Schlechtweg et al., 2020).

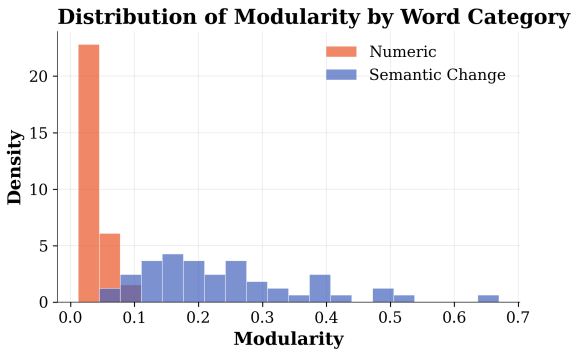


Figure 4: A visualization of the the modularity distributions of the paradigm graphs for both the numeric and semantic change set. The semantic change set' paradigm graph modularities are consistently higher than those of the numeric set.

To address this gap, we manually curate a semantic change benchmark dataset tailored to our period of interest (1990–2020). The dataset contains 70 words: 50 positive examples, which we identify as having undergone a sense shift, and 20 negative examples, which we consider semantically stable over the same interval.

We source the list of 50 positive words from a combination of the Oxford English Dictionary (Oxford, 2025) and the Collins Online Dictionary (HarperCollins, 2025). We provide this list and a short description of each word's sense change in Appendix Table 2.

For our negative examples, we exploit the fact that low-limit number words are consistently observed to be semantically stable across long time periods (Calude, 2021). As such, we use the words "one" through "twenty" as our negative list.

Figure 4 reports the modularity scores for the paradigm graphs corresponding to each word in our benchmark. The known semantically changing words achieve systematically higher modularities than the numeric words. We interpret this as evidence that TLM-1 is able to perform the semantic change detection task.

Overall, though, we reiterate the need for more extensive semantic change detection benchmarks. Based on our results, we believe that TLM-1 could support this effort by supplying soft labels for a larger annotated semantic change benchmark.

## 6 The Temporal Control Curve

To increase our understanding of TLM-1's learned time representations, we perform an analysis of its learned time token embeddings. Recall that the

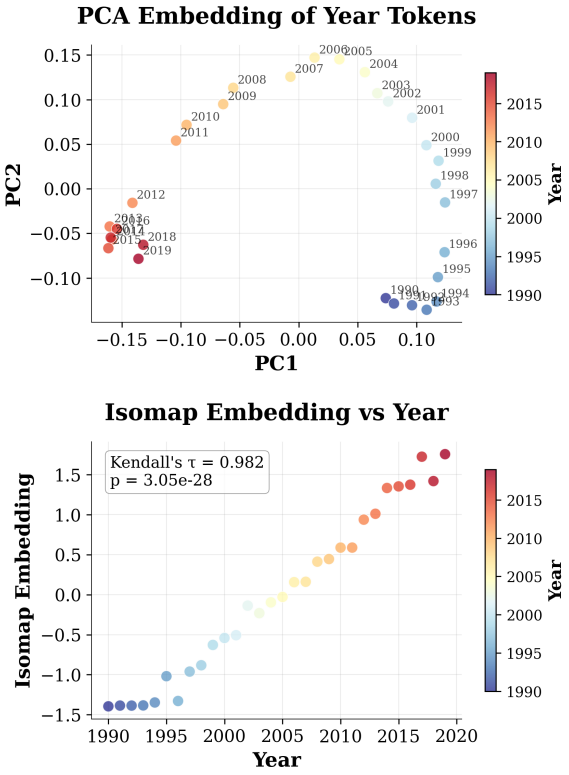


Figure 5: A visualization of the 2D PCA embedding (top) and 1D Isomap embedding (bottom) of TLM-1's learned time tokens. Both embeddings show curves whose arclength is clearly parameterized by the ordinal progression of time. The Kendall Tau between the Isomap embedding order and the year of the corresponding time bin is 0.98, which provides overwhelming evidence of an ordinal association between the Isomap curve and the progression of time.

time token embeddings were all identically initialized, and that the TLM-1 objective treats document dating as a classification problem. This means that the time token embeddings are not endowed with any *a priori* geometric structure relating to the linear progression of time.

We start our investigation by building a matrix from the rows corresponding to time tokens in TLM-1's vocabulary matrix. We call this matrix the time token embedding matrix. We use PCA to project the time token embedding matrix into 3D and visualize it in the Figure 5 (top). Visually, TLM-1 seems to have learned a geometry that organizes time tokens on a curve according to their actual temporal order.

To evaluate whether TLM-1's learned time token geometry captures an ordinal progression of time, we project the time token embedding matrix to one dimension using Isomap (Tenenbaum et al., 2000). We then conduct a Kendall–Tau test (Kendall, 1938) to measure the ordinal association between the resulting Isomap coordinates and the actual temporal ordering of the time tokens. We overwhelmingly reject ($p = 3 \times 10^{-28}$) the null hypothesis that there is no ordinal association between a time token's Isomap 1D coordinate and its actual temporal order.

Overall, we conclude based on our hypothesis test and our qualitative investigation that TLM-1's time tokens are recovering a curve whose geometry recovers the ordinal progression of time. We call this curve the "temporal control curve."

## 7 Conclusion

In this report, we introduce TLM-1, a model of the language process that learns to jointly predict document contents and classify document dates. We train TLM-1 on a general purpose monitor corpus of American English, and provide a query methodology that disentangles TLM-1's temporal dynamics from several sourcese of anachronism. This enables us to probe TLM-1 for temporal trends in language relevant to the United States from 1990 to 2019. We find that TLM-1 accurately reflects several "long arc" trends in contemporary American English and effectively surfaces semantic changes in word meanings.

Furthermore, a mechanistic analysis of TLM-1's time token embeddings reveals that they learn a curve whose geometry recovers the ordinal progression of time. The emergence of this one-dimensional "temporal control curve" within the model's embedding space provides evidence that temporal order is an implicit geometric feature of TLM-1's learned representation. We conjecture that TLMs can be used to forecast the likelihood of future language by extrapolating soft tokens from a fit of this curve.

We view TLM-1 as an important first step towards a general purpose computational model of the dynamics of the language process. Since language models are multitask learners, we conjecture that an scaled up TLM will learn to model several underlying temporal processes that lead to the production of the language in the corpus. We intend to explore this conjecture, among others, in future work.

## 8 Limitations

Despite its promising characteristics, TLM-1 has several drawbacks. TLM-1 is trained on only 750 million words from a single monitor corpus, which prevents it from taking advantage of the blessings of scale. Moreover, TLM-1 is reliant on an anachronistic base model, which complicates temporal analysis and necessitates our Bayesian query framework.

Beyond scaling, there are several concrete procedural, architectural, and data composition questions that TLM-1 does not address. Procedurally, the rate of time masking seems to be a particularly important parameter that is worth studying via ablation. We conjecture that the time token masking rate controls a tradeoff where a high masking rate improves the model's document dating ability while a low masking rate improves the model's ability to perform naive temporally sensitive fills.

Architecturally, there may be utility in exploring decoder-only variants of TLM-1. One challenge of a decoder variant, in particular, is how to integrate time masking. If the time token remains as the first token in the sequence, a decoder model's causal attention structure would prevent context information from being used to fill it when masked. Further, if the time token is not the first token in the sequence, then tokens preceding it will not receive any temporal information in their fills. As a result of these challenges, the task of integrating time tokens into a decoder variant of TLM will require careful consideration and experimentation.

From a data composition standpoint, the challenges of creating an improved dataset for TLM-2 are similar to the challenges associated with creating any monitor corpus. A corpus of sufficient size for a GPT-2- or GPT-3-class model will necessarily require heterogeneous data sources, which could lead to topical skew in the data sources if not integrated correctly. Moreover, there is inherent temporal bias in digital monitor corpora because of the increasing rate of digital data production. This creates a recency bias, as a model that spends more capacity modeling recent phenomena will perform well on the outsized share of recent data in the dataset. Combating both of these biases remains an open challenge in both the language modeling and monitor corpus construction communities.

We believe that the capabilities of an idealized TLM are well worth the effort of solving these challenges.

## References


Robert Bamler and Stephan Mandt. 2017. Dynamic word embeddings.

Andreea S. Calude. 2021. The history of number words in the world's languages — what have we learnt so far? *Philosophical Transactions of the Royal Society B: Biological Sciences*, 376(1824):20200206.

Aaron Clauset, M. E. J. Newman, and Cristopher Moore. 2004. Finding community structure in very large networks. *Physical Review E*, 70(6).

Mark Davies. 2020. *The corpus of contemporary American English*. English-Corpora. org.

Gilles Deleuze and Félix Guattari. 1987. *A Thousand Plateaus: Capitalism and Schizophrenia*. University of Minnesota Press, Minneapolis, MN.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding.

Haim Dubossarsky, Simon Hengchen, Nina Tahmasebi, and Dominik Schlechtweg. 2019. Time-out: Temporal referencing for robust modeling of lexical semantic change. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 457–470, Florence, Italy. Association for Computational Linguistics.

FRED. 2025. Fred graph: data series visualisation (graph id=1mnpj). `https://fred.stlouisfed.org/graph/?g=1MnPJ`. Accessed: 2025-11-10.

HarperCollins. 2025. Collins english dictionary. Accessed from `https://www.collinsdictionary.com/`.

John Hewitt. 2021. Initializing new word embeddings for pretrained language models. `https://www.cs.columbia.edu/~johnhew/vocab-expansion.html`.

Mandar Joshi, Danqi Chen, Yinhan Liu, Daniel S. Weld, Luke Zettlemoyer, and Omer Levy. 2020. Spanbert: Improving pre-training by representing and predicting spans.

Maurice G. Kendall. 1938. A new measure of rank correlation. *Biometrika*.

Vivek Kulkarni, Rami Al-Rfou, Bryan Perozzi, and Steven Skiena. 2014. Statistically significant detection of linguistic change.

Andrey Kutuzov, Lilja Øvrelid, Terrence Szymanski, and Erik Velldal. 2018. Diachronic word embeddings and semantic shifts: a survey. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 1384–1397, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization.

Jean-Baptiste Michel, Yuan Kui Shen, Aviva Presser Aiden, Adrian Veres, Matthew K. Gray, Joseph P. Pickett, Dale Hoiberg, Dan Clancy, Peter Norvig, Jon Orwant, Steven Pinker, Martin A. Nowak, and Erez Lieberman Aiden. 2011. Quantitative analysis of culture using millions of digitized books. *Science*, 331(6014):176–182.

Oxford. 2025. Online oxford english dictionary. Accessed from https://www.oed.com/.

Erin Peterson. 2019. Presidential power surges. https://hls.harvard.edu/today/presidential-power-surges/. Accessed: 2025-11-10.

Guy D. Rosin, Ido Guy, and Kira Radinsky. 2022. Time masking for temporal language models.

Ferdinand de Saussure. 1959. *Course in General 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.

Joshua B. Tenenbaum, Vin de Silva, and John C. Langford. 2000. A global geometric framework for nonlinear dimensionality reduction. *Science*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2023. Attention is all you need.

Orion Weller, Kathryn Ricci, Marc Marone, Antoine Chaffin, Dawn Lawrie, and Benjamin Van Durme. 2025. Seq vs seq: An open suite of paired encoders and decoders.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander M. Rush. 2020. Huggingface's transformers: State-of-the-art natural language processing.

Zijun Yao, Yifan Sun, Weicong Ding, Nikhil Rao, and Hui Xiong. 2018. Dynamic word embeddings for evolving semantic discovery. In *Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining*, WSDM 2018, page 673–681. ACM.

## A Paradigm Mapping Procedure

To create a paradigm map, we begin by constructing a paradigm graph for a target word with a sample size of 1000 and a uniform nucleus prior over the $k = 10$ most likely substitutions for each document. We then apply a force directed layout to the adjacency matrix of $G$ to create a visualization. We limit the visualization to the 15 most common alternative fills in each time bucket to reduce crowding. We also impose a minimum distance between nodes in the visualization to improve readability. We call the resulting visualization a paradigm map. The paradigm maps for $G_{\text{cell}}$ and $G_{\text{seven}}$ are shown in Figure 3.

| Word | Historical Sense | New Sense | Domain |
|---|---|---|---|
| platform | raised surface | digital ecosystem | Technology |
| viral | pathogenic | spreading online | Technology |
| streaming | liquid flow | online media delivery | Technology |
| tweet | bird sound | Twitter post | Technology |
| text | written words | SMS message | Technology |
| friend | companion | social media connection | Technology |
| share | divide portion | post content online | Technology |
| story | narrative | ephemeral photo/video post | Technology |
| cloud | water vapor | internet storage | Technology |
| mouse | rodent | computer device | Technology |
| tablet | stone/pill | touchscreen computer | Technology |
| app | application | mobile program | Technology |
| spam | canned meat | junk email | Technology |
| troll | folklore creature | online provocateur | Technology |
| post | mail/position | publish online | Technology |
| swipe | strike | touchscreen gesture | Technology |
| scroll | rolled manuscript | move through screen text | Technology |
| like | to enjoy | digital approval button | Technology |
| follow | trail behind | subscribe digitally | Technology |
| stream | small river | online broadcast | Technology |
| cancel | annul | socially boycott | Politics |
| woke | awake | socially aware | Politics |
| queer | odd | LGBTQ+ identity | Politics |
| ally | military partner | supporter of a group | Politics |
| gender | grammatical category | social identity | Politics |
| binary | numeric system | gender dichotomy | Politics |
| fluid | liquid | identity flexibility | Politics |
| closet | storage | secrecy of sexuality | Politics |
| out | exterior | disclose sexuality | Politics |
| climate | weather | social/political atmosphere | Politics |
| subscribe | sign | digital follow/pledge | Business |
| content | substance | digital media | Business |
| creator | originator | online producer | Business |
| drop | let fall | product/music release | Business |
| launch | set afloat | product debut | Business |
| startup | act of starting | early company | Business |
| unicorn | mythical beast | billion-dollar startup | Business |
| pivot | turn | strategy shift | Business |
| hustle | hurry | entrepreneurial grind | Business |
| deck | floor surface | presentation slides | Business |
| lit | illuminated | exciting/great | Slang |
| salty | briny | resentful | Slang |
| ghost | spirit | cut off communication | Slang |
| shade | shadow | subtle insult | Slang |
| tea | drink | gossip | Slang |
| ship | vessel | endorse relationship | Slang |
| slay | kill | succeed spectacularly | Slang |
| cringe | recoil | awkward/embarrassing | Slang |
| fire | combustion | amazing (slang) | Slang |
| goat | animal | Greatest Of All Time | Slang |

Table 2: A full list of our manually curated Semantic Change set.

## B TLM-1 Training Charts

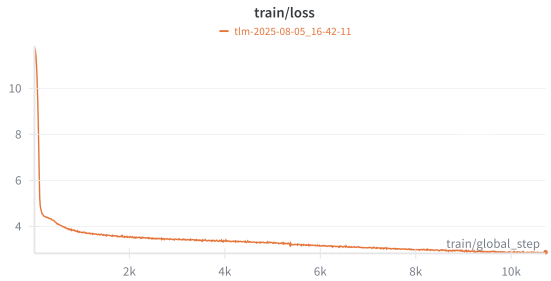


Figure 6: The TLM-1 Loss Curve.

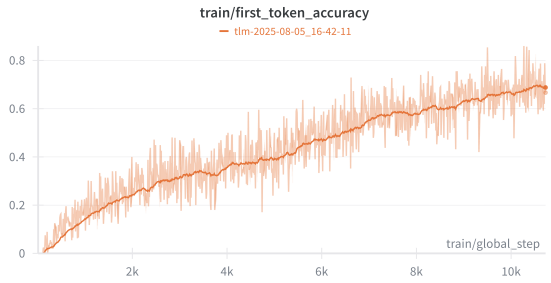


Figure 7: The TLM-1 top-1 time token accuracy curve.

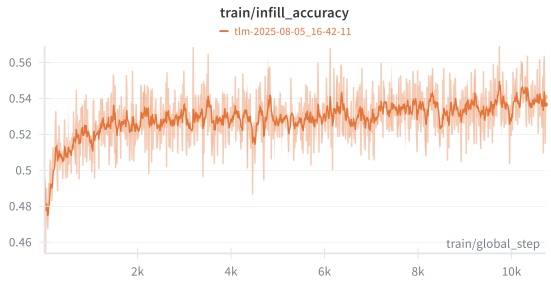


Figure 8: The TLM-1 top-1 content token accuracy curve.