Time Waits for No One! Analysis and Challenges of Temporal Misalignment

Anonymous *ACL submission

Abstract

When an NLP model is trained on text data from one time period and tested or deployed on data from another, the resulting *temporal misalignment* can degrade end-task performance. In this work, we establish a suite of eight diverse tasks across different domains (social media, science papers, news, and reviews) and periods of time (spanning five years or more) to quantify the effects of temporal misalignment. Our study is focused on the ubiquitous setting where a pretrained model is optionally adapted through continued domain-specific pretraining, followed by task-specific finetuning. We establish a suite of tasks across multiple domains to study temporal misalignment in modern NLP systems. We find stronger effects of temporal misalignment on task performance than have been previously reported. We also find that, while temporal adaptation through continued pretraining can help, these gains are small compared to task-specific finetuning on data from the target time period. Our findings motivate continued research to improve temporal robustness of NLP models.[1]

1 Introduction

Changes in the ways a language is used over time are widely attested (Labov, 1994; Altmann et al., 2009; Eisenstein et al., 2014); how these changes will affect NLP systems built from text corpora, and in particular their long-term performance, is not as well understood.

This paper focuses on *temporal misalignment*, i.e., where training and evaluation datasets are drawn from different periods of time. In today's pretraining-finetuning paradigm, this misalignment can affect a pretrained language model—a situation that has received recent attention (Jaidka et al., 2018; Lazaridou et al., 2021; Peters et al., 2018; Raffel et al., 2020; Röttger and Pierrehumbert, 2021)—or the finetuned task model, or both. We suspect that the effects of temporal misalignment will vary depending on the genre or domain of the task's text, the nature of that task or application, and the specific time periods.

We focus primarily on measuring the extent of temporal misalignment on task performance. We consider eight tasks, each with datasets that span at least five years (§2.4), ranging from summarization to entity typing, a subproblem of entity recognition (Grishman and Borthwick, 1999). Notably, these task datasets span four different domains: social media, scientific articles, news, and reviews. We introduce an easily interpretable metric that summarizes the rate at which task performance degrades as function of time.

Our research questions are:

(Q_1) *how does temporal misalignment affect downstream tasks over time?*

(Q_2) *how does sensitivity to temporal misalignment vary with text domain and task?*

(Q_3) *how does temporal misalignment affect language models across domains and spans of time?*

(Q_4) *how effective is temporal adaptation, or additional pretraining on a target year, in mitigating temporal misalignment?*

We find that temporal misalignment affects both language model generalization and task performance. We find considerable variation in degradation across text domains (§3.2) and tasks (§3.1). Over 5 years, classifiers' F_1 score can deteriorate as much as 40 points (political affiliation in Twitter) or as little as 1 point (Yelp review ratings). Two distinct tasks defined on the same domain can show different levels of degradation over time.

We explore domain adaptation of a language model, using temporally selected (unannotated) data, as a way to curtail temporal misalignment (Röttger and Pierrehumbert, 2021). We find that this does not offer much benefit, especially relative

[1] Anonymized GitHub Repo

to performance that can be achieved by finetuning on temporally suitable data (i.e., from the same time period as the test data). We conclude that temporal adaptation should not be seen as a substitute for finding temporally aligned labeled data.

The evidence and benchmarks we offer motivate careful attention to temporal misalignment in many applications of NLP models, and further research on solutions to this problem.

Contributions. To facilitate the study of temporal misalignment phenomenon on downstream applications, we compile a suite of eight diverse tasks across four important language domains. We define an interpretable metric that summarizes temporal misalignment of a model on a task with timestamped data. Our experiments reveal key factors in how temporal misalignment affects NLP model performance.

2 Methodology Overview

We begin by defining the scope of our study.

2.1 Learning Pipeline

We consider a process for building an NLP model that is in widespread use by the research community, illustrated in Fig. 1. First, a (neural network) language model (LM) is pretrained on a large text collection that is not necessarily selected for topical or temporal proximity to the text of the target application (our focus is on GPT-2; Brown et al., 2020). Second, the LM is optionally adapted by continued training on a collection strategically curated for closer proximity to the target (Beltagy et al., 2019); this stage is often referred to as domain-adaptive pretraining (DAPT; Gururangan et al., 2020). Finally, the model is finetuned to minimize a task-specific loss, using labeled data representative of what the model is expected to be exposed to in testing or deployment.

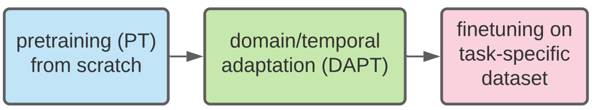

Figure 1: A typical modeling pipeline in NLP.

We study two ways in which temporal misalignment might affect the pipeline's performance as well as straightforward ways to mitigate them.

Task Shift and Temporal Finetuning The relationship between text inputs and target outputs may change over time. To the extent that this occurs, annotated datasets used to train NLP systems in the finetuning stage will become stale over time. Due to this temporal misalignment, performance will degrade after deployment, or any in evaluations that use test data temporally distant from the training data. We seek to quantify this degradation across a range of text domains and tasks.

Language Shift and Temporal Domain Adaptation Changes in language use can cause a pretrained LM, which commonly serves as the backbone for most modern NLP models, to become stale over time (Lazaridou et al., 2021), regardless of the end task. Lazaridou et al. (2021) explored *temporal adaptation*, continuing LM training on new text data. This is essentially the same procedure as DAPT, where the data is selected by time period. Their work focused on the LM alone, not downstream tasks; we consider both here.

Röttger and Pierrehumbert (2021), the closest to our work, studied temporal adaptation in conjunction to finetuning for a classification task over Reddit data. They conclude that temporal adaptation does not help any more than normal DAPT. We corroborate this work and extend it by studying a wider variety of tasks over a longer span of time periods and thus are better able to draw generalizations from our results.

We believe that the two kinds of shift—task shift and language shift—are difficult to disentangle, and we do not attempt to do so in this work. Instead, we aim to quantify the effect of temporal misalignment on a range of NLP tasks, as well as the benefits of these two strategies.

2.2 Evaluation Methodology

Our experiments are designed to measure the effect of temporal misalignment on task performance. To do so, for each task, we fix a test set within a given time period, T_{test}. We vary the time period of the training data, allowing us to interpret differences in performance as a kind of "regret" relative to the performance of a model trained on data temporally aligned with T_{test}.[2] We consider multiple different test periods for each task. We also seek to control the effect of training dataset size. We partition training data into time periods of roughly

[2]This setup avoids a confound of varying test set difficulty that we would encounter if we fixed the model and compared its performance across test datasets from *different* time periods.

the same size and always train on a single partition, keeping the training set size of each time period constant within each task. We expect that performance could be improved by accumulating training data across multiple time periods, but that would make it more difficult to achieve our research goal of quantifying the effect of temporal misalignment on performance.

2.3 Quantifying Temporal Degradation

Understanding temporal misalignment requires evaluating a model's performance across data with a range of different timestamps, which makes it difficult to compare various models in terms of their misalignment. We define a metric for temporal degradation (TD) which summarizes the overall amount of temporal misalignment on a task as a single value. In high-level terms, the TD score measures the average rate of performance deterioration (of perplexity, F_1, or Rouge-L) for each time period of misalignment between the train and evaluation sets. Higher TD scores imply greater levels of performance deterioration due to misalignment.

Let $S_{t' \to t}$ indicate the performance a model trained on timestep t' data and evaluated on timestep t. We define $\mathrm{D}(t' \to t)$ as:

$$\mathrm{D}(t' \to t) = -\left(S_{t' \to t} - S_{t \to t}\right) \times \mathrm{sign}(t' - t).$$

$\mathrm{D}(t' \to t)$ is a modified difference in performance between two models.[3] Fig. 2 illustrates D as a function of consecutive training time periods.

We find a line of best fit for $\mathrm{D}(t' \to t)$ for all t' using least-squares regression. The slope of this line is $\mathrm{TD}(t)$, the TD score for evaluation time period t. The final TD score is the average of the $\mathrm{TD}(t)$ across all evaluation time periods t. Further details can be found in Appendix A.

2.4 Domains, Tasks, and Datasets

We describe the eight tasks and four domains used for this study. Three (out of eight) of the tasks are newly defined in this work, and all tasks required nontrivial postprocessing; we will release the corresponding datasets or postprocessing scripts publicly at publication [4]. We provide examples and detailed statistics in Table 5 of Appendix C.

[3]Without the modification, a task with degradation would have have positive performance gaps both $t' > t$ and $t' > t$; the function would not be monotone and the rate of change would be harder to approximate. The modification yields a simpler visual understanding of the deviations over time.

[4]Anonymized GitHub Repo

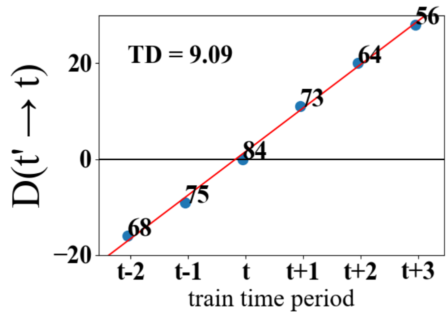

Figure 2: An example calculation of the TD score for a particular timestep t (discussed in Section 2.3). The plotted markers represent $D(t' \to t)$ (y-axis) as a function of train time period t' (x-axis). The annotated numbers on each blue dot are the raw evaluation scores $S_{t' \to t}$, not to be confused with the y values. The red line is the line of best fit and its slope is the TD score for evaluation timestep t. In this example, we would expect to see, on average, 9.09 points of deterioration for each year of misalignment. The final TD score is averaged between all evaluation timesteps.

Domain 1: Twitter Social media platforms like Twitter have been mined to study aspects of language change over time, such as the introduction or diffusion of new words (Eisenstein et al., 2014; Tamburrini et al., 2015; Wang and Goutte, 2017). We collect unlabeled data for domain adaptation by extracting a random selection of 12M tweets, spread semi-uniformly from 2015 till 2020.[5] We experiment with two tasks on Twitter data:

Political affiliation classification (**POLIAFF**) We collect English tweets dated between 2015 and 2020 from U.S. politicians with a political affiliation (*Republican* or *Democrat*). We omit any politician who changed parties over this time period or identified as independent. We consider the downstream task of detecting political affiliations, i.e., given a text of a single tweet we predict the political alignment of its author at the time of the tweet. This task can be useful for studies that involve an understanding of ideologies conveyed in text (Lin et al., 2008; Iyyer et al., 2014).

Named entity type classification (**TWIERC**) We use the Twitter NER dataset from Rijhwani and Preoţiuc-Pietro (2020). The dataset contains tweets dated from 2014 to 2019, each annotated with the mentions of named entities and their types (*Person*, *Organization*, or *Location*). We consider the task of typing a given mention, which is a subproblem of named entity recognition.

[5]Collected via the Twitter API.

Domain 2: Scientific Articles Scientific research produces vast amounts of text with great potential for language technologies (Wadden et al., 2020; Lo et al., 2020); it is expected to show a great deal of variation over time as ideas and terminology evolve. For adaptation to this domain, we collect unlabeled data from science documents available in Semantic Scholar's corpus,[6] which yields 650k documents, spread over a 30-year period (Ammar et al., 2018). For this domain, we study two tasks:

Mention type classification (**SciERC**) We use the *SciERC* dataset from Luan et al. (2018) which contains entity-relation annotations for computer science paper abstracts for a relatively wide range of years (1980s to 2019). We subdivide the annotated data into time periods with roughly equal-sized numbers of papers (1980–1999, 2000–2004, 2005–2009, 2010–2016). The task is to map a mention of a scientific concept to a type (*Task*, *Method*, *Metric*, *Material*, *Other-Scientific-Term*, or *Generic*).

AI venue classification (**AIC**) We also examine temporal misalignment on the task of identifying whether a paper was published in AAAI or ICML. We group the data into roughly equal-sized time periods (2009–2011, 2012–2014, 2015–2017, and 2018–2020). This task is, loosely, a proxy for topic classification and author disambiguation applications (Subramanian et al., 2021).

Domain 3: News Articles News articles make up a significant part of the data commonly used to train LMs (Dodge et al., 2021). News articles convey current events, suggesting strong temporal effects on topic. For adaptation, we use 9M articles from the Newsroom dataset (Grusky et al., 2018), ranging from 2009–2016.[7] We experiment with three tasks on news articles:

Newsroom summarization (**NewSum**) The Newsroom dataset provides a large quantity of high-quality summaries of news articles (Grusky et al., 2018). We group articles by years for this task (2009–2010, 2011–2012, 2013–2014, 2015–2016). Note that this task, unlike the other tasks considered here, is not a document classification task.

Publisher classification (**PubCls**) The Newsroom dataset also provides metadata, such as publication source. We take the documents published by the 3 most prolific publishers (Fox News, New York Times, and Washington Post) and train models to classify documents among them. We bin the years (2009–2010, 2011–2012, 2013–2014, 2015–2016). This task is a proxy for applications that seek to infer fact provenance (Zhang et al., 2020). We note that, unlike in our other tasks, we downsample to ensure that the labels are equally balanced.

Media frames classification (**MFC**) "Framing" often refers to the emphasis or deemphasis of different social or cultural issues in the media's presentation of the news (Entman, 1983). Card et al. (2015) provide a dataset of news articles annotated with framing dimensions. We predict the primary frame of a document, treating the problem as a 15-way classification task. We bin by timestamp (2009–2010, 2011–2012, 2013–2014, 2015–2016).

Domain 4: Food Reviews Food and restaurant reviews have been widely studied in NLP research. We considered this domain as a possible contrast to those above, expecting less temporal change. Using data from the Yelp Open Dataset,[8] we consider one task:

Review rating classification (**YelpCls**) This is a conventional sentiment analysis task, mapping the text of a review to the numerical rating given by its author (Pang et al., 2002; Dave et al., 2003). We partition the data by year (2013 to 2019) and ensure that each timestep has a roughly equal amount of reviews.

3 Empirical Results and Analysis

In this section, we summarize our experimental analysis, resulting from more than 500 experiments. In our experiments, we primarily explore the effect of temporal misalignment on GPT2 (Brown et al., 2020), a PLM often used for generation.[9] We report the macro F_1 score for classification tasks and *Rouge-L* (Lin, 2004) for NewSum.

We first focus on quantifying temporal misalignment in end tasks. As a preliminary analysis, we investigate how the marginal distribution over labels changes over time. We then study how temporal misalignment affects performance of GPT2 models in downstream tasks with temporal finetuning (Q_1,Q_2). We find that the amount of performance degradation can vary by task; in some cases the

[6] https://api.semanticscholar.org/corpus/

[7] https://lil.nlp.cornell.edu/newsroom

[8] https://www.yelp.com/dataset

[9] In our preliminary results, we found that BERT, RoBERTa, and GPT2 models showed similar patterns.

degradation can be severe.

We then study how temporal misalignment affects PLMs. As a first step, we analyze how vocabularies change over time in our datasets. We then experiment with (Q_3) how temporal misalignment affects upstream language modeling and (Q_4) how effective temporal adaptation, or additional pretraining on a target year, is in mitigating misalignment. We find that while PLMs are affected by misalignment, temporal domain adaptation is not enough to mitigate temporal misalignment.

Details on temporal domain adaptation and finetuning, and an extended version of our results, can be found in Appendices B and D, respectively.

3.1 Temporal Misalignment in Tasks

How much does misalignment affect task performance? We find that it depends on the task.

Label Distribution Drift We first investigate how task datasets undergo changes in the marginal distribution over labels due to time. For each task and each test period, we calculate the KL divergence between the label distributions in that period and the first test period. Full results are reported in Fig. 7 of Appendix D. In three cases, we detected notable label distribution drift: POLIAFF, AIC, and MFC.[10] In POLIAFF, Republican tweets outnumbered Democratic ones by over a 2:1 ratio in 2015, but the reverse held by 2020. This observation shows that, regardless of the properties of NLP models, the nature of many tasks changes over time, if only because the output distribution changes.

Finetuning As described in §2.4, for each task, we create training and evaluation sets associated with different time periods. We finetune GPT2 on each of the task's training sets and evaluate each on two evaluation sets. Note that there is no domain adaptation here.

Fig. 3 shows our results on downstream tasks (with no domain adaptation). To get more reliable estimates, each number in this heatmap is an average of five independent experiments with different random seeds. A summary of the fine-tuning results, in terms of TD scores (§2.3) is in Table 1 which indicates the speed of temporal degradation, for every year that the training and evaluation data diverges. Recall that this score (applied to task performance measures) summarizes the strength of the effect of temporal misalignment on the score, using evidence from across experiments.

Domain	Task (metric)	TD
Twitter	POLIAFF (F1)	7.72
	TWIERC (F1)	0.96
Science	SCIERC (F1)	1.08
	AIC (F1)	1.79
News	PUBCLS (F1)	5.46
	NEWSUM (Rouge-L)	1.38
	MFC (F1)	0.98
Reviews	YELPCLS (F1)	0.26

Table 1: Finetuned models' temporal degradation summary scores (TD; §2.3; details in Figure 3). These scores estimate how fast a model degrades as the time period of training and evaluation data diverge (higher scores imply faster degradation). We note that since we normalize by the overall time range of a task, the temporal partitions we used do have an effect on the TD scores. For example, AIC spans ten years, even though there are only four partitions.

(Q_1) Temporal misalignment degrades task performance substantially. Fig. 3, similar to earlier work (Röttger and Pierrehumbert, 2021), shows that models trained on data from the same time period as the test data perform far better than those from the past. The performance drop is most severe for POLIAFF (TD=7.72) and PUBCLS (TD=5.45).

(Q_1) Temporal misalignment has a measurable effect on most tasks. With the exception of MFC and TWIERC tasks, all tasks see an average loss of at least 1 point for each time period that the training data diverges from the test data. For datasets like SCIERC that make use of data from three decades or more, this effect could add up.

Moreover, 1 point of difference can be meaningful, especially for the summarization task where we measure Rouge-L. According to the leaderboard,[11] the best three performing models are within a point of each other in Rouge-L (Shi et al., 2019, 2021; Mendes et al., 2019). The task has a TD score of 1.38. On average, a time period of temporal misalignment results has a larger effect on performance than changing between the three best models.

(Q_1) Performance loss from temporal misalignment occurs in both directions. Another observation in Table 3 is that degradation happens in both

[10] For other tasks, it is possible that the data collection/annotation procedures suppressed label distribution changes that would be visible in data "from the wild."

[11] https://lil.nlp.cornell.edu/newsroom/index.html

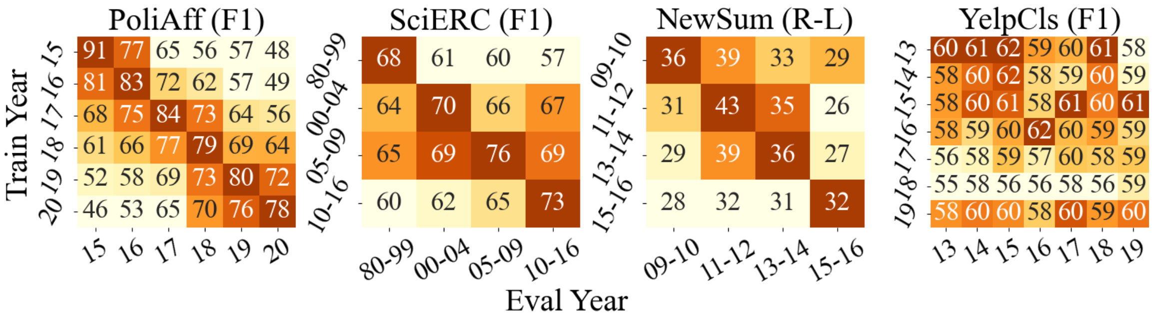

Figure 3: Temporal misalignment in finetuning affects task performance (§3.1). In all cases, higher scores are better. The heatmap is shaded per column, i.e., the darkest shade of orange in a cell means the cell has the highest score in that column. Mismatch between the the training and evaluation data result in massive performance drop. While all suffer from temporal degradation, its degree is a strong function of task definition. For example, YELPCLS shows minimal degradation. In contrast, POLIAFF shows major deterioration over time. Additional tables of our remaining tasks can be found in Appendix D.

directions (past and future). While most of the emphasis on temporal misalignment is on how to adapt our stale models/data to the present time (Dhingra et al., 2021; Lazaridou et al., 2021; Röttger and Pierrehumbert, 2021), our experiments also show that models trained on newer data can be misaligned from the past, as well. This can be important in social science applications (Abercrombie and Batista-Navarro, 2019; Soni et al., 2021), for example, where evaluation sets may come from earlier time periods than the training data. Moreover, the deterioration rates are similar in both directions.

(Q_2) **Tasks, even in the same domain, are affected differently.** Consider the two tasks of POLIAFF and TWIERC (both in the Twitter domain), with TD scores of 7.72 and 0.96, respectively. Of our 8 tasks, TWIERC, MFC, and YELPCLS are the most robust to temporal misalignment (TD scores of 0.96, 0.98 and 0.26, respectively). The high levels of variation show that temporal misalignment affects performance through labeled datasets, not just unlabeled pretraining data.

3.2 Temporal Misalignment in LMs

As LMs are widely used in modern NLP systems, it is important to inspect how robust they are to temporal misalignment. We seek to understand how temporal misalignment affects the language modeling task in our four domains and if temporal domain adaptation helps in downstream tasks.

Vocabulary Shift We first consider an extremely simple measurement of language shift: how do vocabularies change across time periods?[12] We use a similar procedure to the one Gururangan et al. (2020) used for analyzing domain similarity. Fixing a domain, we compare the (unigram) vocabularies of each pair of training sets. The vocabularies are built using the 10K most frequent terms from each time period. We note that vocabulary overlap is higher between two time periods the closer they are. Most domains see a sizeable amount of shift; however, Yelp is relatively stagnant. Fig. 4 visualizes the overlap measurement.

Temporal Domain Adaptation We next apply DAPT to GPT2: for each domain, we continue pretraining and then evaluate perplexity. We consider how the perplexity varies with the (mis)alignment between the DAPT training data and the evaluation data. We measure the TD score, which summarizes how much performance is affected by temporal misalignment (now applied to perplexity). The results of temporal domain adaptation are in Fig. 5.

(Q_3) **Domains are a major driver of temporal misalignment in LMs.** Consistent with Lazaridou et al. (2021), Fig. 5 shows degradation of LM due to temporal misalignment; it further shows considerable variation by text domain. Twitter changes most rapidly, and food reviews are much slower. This observation is consistent with past work on language change in social media (Stewart and Eisenstein, 2018; Eisenstein et al., 2014). To the extent that a LM's practical usefulness is associated with its fit to new data, researchers and practitioners should understand the temporal dynamics of their target text domains and plan LM updates accordingly.

Joint Effects of Temporal Adaptation and Finetuning As discussed in §2, continued pretraining

[12]This can be understood as a model-free way to measure covariate shift for NLP tasks that take text as input.

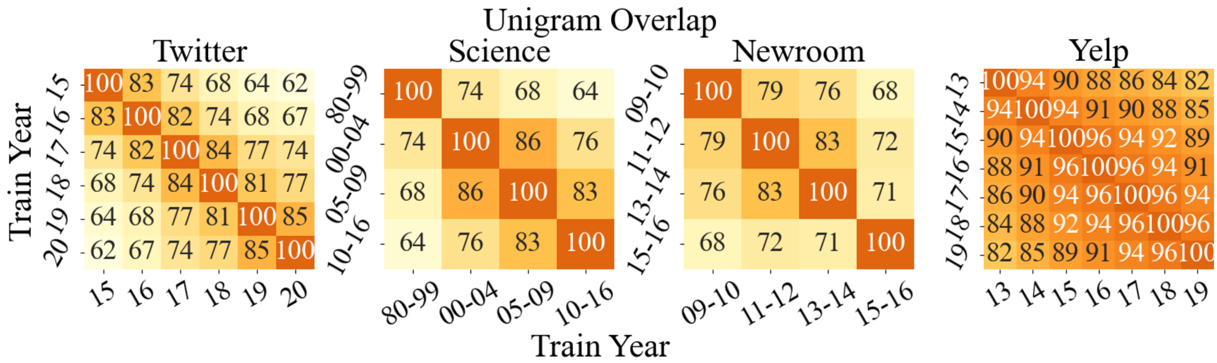

Figure 4: Vocabulary overlap between time periods, over a subset of our tasks' datasets. Each cell shows the % overlap between the vocabularies of two time periods.

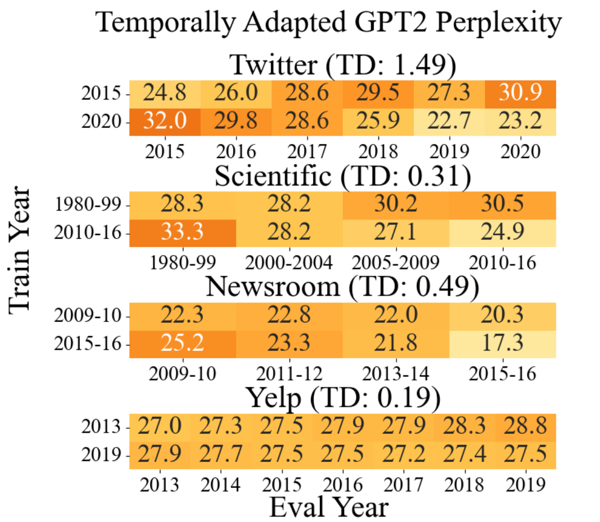

Figure 5: Perplexity of GPT2 after adaptive pretraining on temporally-selected data in different domains (lower is better). The TD score (in parentheses) estimates the expected perplexity rise (i.e., degradation) for every time period of misalignment between evaluation and training times. Degradation follows the expected pattern, but the magnitude varies by domain.

of an LM on in-domain text has been shown to improve task performance. Our prior results show that both downstream tasks and language modeling are affected by temporal misalignment. Can temporal domain adaptation help mitigate the effects of misalignment in downstream tasks?

Here we consider how the time period of the data selected for continued pretraining affects task performance. For each task's evaluation set, we apply DAPT twice: once with the earliest available time period's unannotated data and once with the latest's. We then finetune and evaluate on data from the same time periods as in the earlier experiment.

(Q_4) **Temporal adaptation does <u>not</u> overcome degradation from temporally misaligned labeled data.** In Table 2, we see small performance gains from temporal domain adaptation on LMs, and in some cases it is harmful. These observations underscore the importance of the labeled data; adjustments to the LM alone do not yet appear sufficient to mitigate the effects of temporal misalignment. In contrast to temporal domain adaptation, which does not mitigate temporal misalignment's effects, finetuning on temporally-updated labeled data is more effective.

This can be observed in each task-specific subtable of in Table 2: the top-left and bottom-right quadrants (fine-tuning on time-stamp that is used for evaluation) generally lead to higher scores.

4 Limitations and Future Work

We provided a well-controlled suite of experiments to study the effects of temporal misalignment on model performance. However, the setup has some drawbacks. For example, we expect that models trained on data accumulated across multiple time periods would perform well (Lazaridou et al., 2021; Röttger and Pierrehumbert, 2021; Jin et al., 2021).

We chose the time periods in our study so that they would each have sufficient and consistent training data sizes. However, amounts of data in a particular domain or task will fluctuate over time. Moreover, the rate of language use change may not be uniform. Future work may want to define their time periods with these two considerations in mind.

Our findings indicate that temporal misalignment's effects depend heavily on the task. Though not studied here, the same issues may arise in annotation efforts; consider, for example, recent work on controversy (Zhang et al., 2018) and social norms (Xu et al., 2021; Zhou et al., 2021) likely hinges on constructs that may be time sensitive. Annotations that are temporally misaligned with the

Domain (Task) ↓	Finetune Year ↓	Evaluation → Pretrain ↓	2015	2020
Twitter (PoliAff) *F1*	2015	Default	91.4	48.4
		Default → 2015	92.2	47.5
		Default → 2020	90.9	50.8
	2020	Default	45.8	78.0
		Default → 2015	47.2	76.9
		Default → 2020	44.2	78.3

Domain (Task) ↓	Finetune Year ↓	Evaluation → Pretrain ↓	1980-1999	2010-2016
Scientific (SciERC) *F1*	1980-1999	Default	67.9	57.2
		Default → 1980-1999	73.2	66.4
		Default → 2010-2016	73.7	66.8
	2010-2016	Default	60.3	72.5
		Default → 1980-1999	63.4	75.0
		Default → 2010-2016	64.8	76.0

Domain (Task) ↓	Finetune Year ↓	Evaluation → Pretrain ↓	2009-2010	2015-2016
News (NewsSum) *Rouge-L*	2009-2010	Default	36.4	29.0
		Default → 2009-2010	36.4	29.1
		Default → 2015-2016	36.1	28.9
	2015-2016	Default	27.8	31.8
		Default → 2009-2010	28.2	31.8
		Default → 2015-2016	27.8	31.6

Domain (Task) ↓	Finetune Year ↓	Evaluation → Pretrain ↓	2014	2019
Food Reviews (Yelp) *F1*	2009-2010	Default	58.6	58.3
		Default → 2014	63.3	60.1
		Default → 2019	60.2	62.3
	2015-2016	Default	58.3	58.3
		Default → 2014	60.2	62.3
		Default → 2019	60.8	62.3

Table 2: Combination of temporal adaptation and finetuning (§3.2) on our tasks. The row labeled "Default" corresponds to a model that has not been adapted (uses the default pretraining). The models with temporal domain adaptation are shown in rows labeled "Default → y" and each is comparable to the "Default" row above it. The color coding is proportional to the magnitude of the performances of each task (darker shade of orange indicate higher scores). It can be observed that temporal finetuning has a greater impact than temporal pretraining. Each quadrant of 3 for each task, indicating the same finetune and evaluation years, but different pretraining conditions, are mostly uniform. In contrast, we notice a sharper difference in performance when varying the finetuning year (comparing the quadrants vertically).

original data being annotated may be anachronistic.

An opportunity for future exploration is in the context of real-world events with sudden changes such as COVID-19 pandemic (Cao et al., 2021) or political changes, which influence tasks such as question answering (Dhingra et al., 2021; Zhang and Choi, 2021).

Continual learning, which allows models to learn from a continuous stream of data, could also be one way to mitigate temporal misalignment. Most prior work in this space has focused on continual learning in PLMs (Gururangan et al., 2021; Jin et al., 2021) or learning disparate tasks (de Masson d'Autume et al., 2019; Huang et al.). Future work may investigate continual learning algorithms for tasks that change over time.

While we found that task-specific finetuning is more effective than temporal adaptation, new labeled data can be expensive. Ways to characterize or detect changes in a task could be helpful in efficiently updating datasets (Lu et al., 2019; Webb et al., 2018). Future work can also treat dataset maintenance as an optimization problem between the cost and gains of annotating new data (Bai et al., 2021).

5 Conclusion

Changes in language use over time, and how language relates to other quantities of interest in NLP applications, has clear effects on the performance of those applications. We have explored how temporal misalignment between training data—both data used to train LMs and annotated data used to finetune them—affects performance across a range of NLP tasks and domains, taking advantage of datasets where timestamps are available. We compile these datasets as a benchmark for future research as well. We also introduced a summary metric, TD score, that makes it easier to compare models in terms of their temporal misalignment.

Our experiments revealed considerable variation in temporal degradation accross tasks, more so than found in previous studies (Röttger and Pierrehumbert, 2021). These findings motivate continued study of temporal misalignment across applications of NLP, its consideration in benchmark evaluations,[13] and vigilance on the part of practitioners able to monitor live system performance over time.

Notably, we observed that continued training of LMs on temporally aligned data does not have much effect, motivating further research to find effective temporal adaptation methods that are less costly than ongoing collection of annotated/labeled datasets over time.

[13] Indeed, for benchmarks where training and testing data *are* aligned, our findings suggest that measures of performance may be in some cases inflated.

References

Gavin Abercrombie and Riza Theresa Batista-Navarro. 2019. Semantic change in the language of uk parliamentary debates. In *Proceedings of the 1st International Workshop on Computational Approaches to Historical Language Change*.

Eduardo G Altmann, Janet B Pierrehumbert, and Adilson E Motter. 2009. Beyond word frequency: Bursts, lulls, and scaling in the temporal distributions of words. *PLOS one*, 4(11):e7678.

Waleed Ammar, Dirk Groeneveld, Chandra Bhagavatula, Iz Beltagy, Miles Crawford, Doug Downey, Jason Dunkelberger, Ahmed Elgohary, Sergey Feldman, Vu Ha, et al. 2018. Construction of the literature graph in semantic scholar. In *NAACL*.

Fan Bai, Alan Ritter, and Wei Xu. 2021. Pre-train or annotate? domain adaptation with a constrained budget. *arXiv preprint arXiv:2109.04711*.

Iz Beltagy, Kyle Lo, and Arman Cohan. 2019. Scibert: A pretrained language model for scientific text. In *EMNLP*.

Tom B Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *arXiv preprint arXiv:2005.14165*.

Ivy Cao, Zizhou Liu, Giannis Karamanolakis, Daniel Hsu, and Luis Gravano. 2021. Quantifying the effects of COVID-19 on restaurant reviews. In *Proceedings of the International Workshop on Natural Language Processing for Social Media*, Online. Association for Computational Linguistics.

Dallas Card, Amber E. Boydstun, Justin H. Gross, Philip Resnik, and Noah A. Smith. 2015. The media frames corpus: Annotations of frames across issues. In *ACL*.

Kushal Dave, Steve Lawrence, and David M. Pennock. 2003. Mining the peanut gallery: opinion extraction and semantic classification of product reviews. In *WWW '03*.

Cyprien de Masson d'Autume, Sebastian Ruder, Lingpeng Kong, and Dani Yogatama. 2019. Episodic memory in lifelong language learning. In *nips*.

Bhuwan Dhingra, Jeremy R. Cole, Julian Martin Eisenschlos, Daniel Gillick, Jacob Eisenstein, and William W. Cohen. 2021. Time-aware language models as temporal knowledge bases. *CoRR*, abs/2106.15110.

Jesse Dodge, Maarten Sap, Ana Marasovic, William Agnew, Gabriel Ilharco, Dirk Groeneveld, and Matt Gardner. 2021. Documenting the english colossal clean crawled corpus. *arXiv preprint arXiv:2104.08758*.

Jacob Eisenstein, Brendan O'Connor, Noah A Smith, and Eric P Xing. 2014. Diffusion of lexical change in social media. *PloS one*, 9(11).

Robert M. Entman. 1983. Framing: Toward clarification of a fractured paradigm. *Journal of Communications*.

Ralph Grishman and Andrew Borthwick. 1999. A maximum entropy approach to named entity recognition.

Max Grusky, Mor Naaman, and Yoav Artzi. 2018. Newsroom: A dataset of 1.3 million summaries with diverse extractive strategies. In *NAACL*.

Suchin Gururangan, Mike Lewis, Ari Holtzman, Noah A. Smith, and Luke Zettlemoyer. 2021. Demix layers: Disentangling domains for modular language modeling. *CoRR*, abs/2108.05036.

Suchin Gururangan, Ana Marasović, Swabha Swayamdipta, Kyle Lo, Iz Beltagy, Doug Downey, and Noah A Smith. 2020. Don't stop pretraining: Adapt language models to domains and tasks. In *ACL*.

Yufan Huang, Yanzhe Zhang, Jiaao Chen, Xuezhi Wang, and Diyi Yang. Continual learning for text classification with information disentanglement based regularization. In *ACL*.

Mohit Iyyer, Peter Enns, Jordan Boyd-Graber, and Philip Resnik. 2014. Political ideology detection using recursive neural networks. In *ACL*.

Kokil Jaidka, Niyati Chhaya, and Lyle Ungar. 2018. Diachronic degradation of language models: Insights from social media. In *ACL*.

Xisen Jin, Dejiao Zhang, Henghui Zhu, Wei Xiao, Shang-Wen Li, Xiaokai Wei, Andrew Arnold, and Xiang Ren. 2021. Lifelong pretraining: Continually adapting language models to emerging corpora.

W. Labov. 1994. Principles of linguistic change.

Angeliki Lazaridou, Adhiguna Kuncoro, Elena Gribovskaya, Devang Agrawal, Adam Liska, Tayfun Terzi, Mai Gimenez, Cyprien de Masson d'Autume, Sebastian Ruder, Dani Yogatama, et al. 2021. Pitfalls of static language modelling. *arXiv preprint arXiv:2102.01951*.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Proc. of Text Summarization Branches Out*.

Wei-Hao Lin, Eric P. Xing, and Alexander Hauptmann. 2008. A joint topic and perspective model for ideological discourse. In *ECML/PKDD*.

Kyle Lo, Lucy Lu Wang, Mark Neumann, Rodney Kinney, and Daniel S Weld. 2020. S2orc: The semantic scholar open research corpus. In *ACL*.

Jie Lu, Anjin Liu, Fan Dong, Feng Gu, João Gama, and Guangquan Zhang. 2019. Learning under concept drift: A review. *IEEE Transactions on Knowledge and Data Engineering*.

Yi Luan, Luheng He, Mari Ostendorf, and Hannaneh Hajishirzi. 2018. Multi-task identification of entities, relations, and coreference for scientific knowledge graph construction. In *EMNLP*.

Afonso Mendes, Shashi Narayan, Sebastião Miranda, Zita Marinho, André F. T. Martins, and Shay B. Cohen. 2019. Jointly extracting and compressing documents with summary state representations. In *NAACL*.

Bo Pang, Lillian Lee, and Shivakumar Vaithyanathan. 2002. Thumbs up? sentiment classification using machine learning techniques. In *EMNLP*.

Matthew Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In *NAACL*.

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

Shruti Rijhwani and Daniel Preoţiuc-Pietro. 2020. Temporally-informed analysis of named entity recognition. In *ACL*.

Paul Röttger and Janet B Pierrehumbert. 2021. Temporal adaptation of bert and performance on downstream document classification: Insights from social media. *arXiv preprint arXiv:2104.08116*.

Tian Shi, Yaser Keneshloo, Naren Ramakrishnan, and Chandan K. Reddy. 2021. Neural abstractive text summarization with sequence-to-sequence models. *ACM Transactions on Data Science*.

Tian Shi, Ping Wang, and Chandan K. Reddy. 2019. LeafNATS: An open-source toolkit and live demo system for neural abstractive text summarization. In *NAACL*.

Sandeep Soni, Lauren Klein, and Jacob Eisenstein. 2021. Abolitionist networks: Modeling language change in nineteenth-century activist newspapers. *arXiv preprint arXiv:2103.07538*.

Ian Stewart and Jacob Eisenstein. 2018. Making "fetch" happen: The influence of social and linguistic context on nonstandard word growth and decline. In *EMNLP*.

Shivashankar Subramanian, Daniel King, Doug Downey, and Sergey Feldman. 2021. S2and: A benchmark and evaluation system for author name disambiguation. *arXiv preprint arXiv:2103.07534*.

Nadine Tamburrini, Marco Cinnirella, Vincent AA Jansen, and John Bryden. 2015. Twitter users change word usage according to conversation-partner social identity. *Social Networks*, 40:84–89.

David Wadden, Shanchuan Lin, Kyle Lo, Lucy Lu Wang, Madeleine van Zuylen, Arman Cohan, and Hannaneh Hajishirzi. 2020. Fact or fiction: Verifying scientific claims. In *EMNLP*.

Yunli Wang and Cyril Goutte. 2017. Detecting changes in twitter streams using temporal clusters of hashtags. In *Proceedings of the Events and Stories in the News Workshop*.

Geoffrey I. Webb, Loong Kuan Lee, Bart Goethals, and François Petitjean. 2018. Analyzing concept drift and shift from sample data. *Data Mining and Knowledge Discovery*.

Albert Xu, Eshaan Pathak, Eric Wallace, Suchin Gururangan, Maarten Sap, and Dan Klein. 2021. Detoxifying language models risks marginalizing minority voices. In *NAACL*.

Justine Zhang, Jonathan Chang, Cristian Danescu-Niculescu-Mizil, Lucas Dixon, Yiqing Hua, Dario Taraborelli, and Nithum Thain. 2018. Conversations gone awry: Detecting early signs of conversational failure. In *ACL*.

Michael J.Q. Zhang and Eunsol Choi. 2021. SituatedQA: Incorporating extra-linguistic contexts into QA. *EMNLP*.

Yi Zhang, Zachary Ives, and Dan Roth. 2020. "who said it, and why?" provenance for natural language claims. In *ACL*.

Xuhui Zhou, Maarten Sap, Swabha Swayamdipta, Yejin Choi, and Noah A. Smith. 2021. Challenges in automated debiasing for toxic language detection. In *EACL*.

Supplementary Material

A A Metric for Temporal Degradation

Let t be the time period of the training data and t' the time period of the evaluation data.[14] We aim to summarize the general effect of temporal misalignment (the difference between t and t') on task performance, in an interpretable way that is comparable across tasks.

Let $S_{t'\to t}$ indicate the performance a model trained on timestamp t' data and evaluated on the timestamp t. Let

$$\mathrm{D}(t' \to t) = -\left(S_{t'\to t} - S_{t\to t}\right) \times \mathrm{sign}(t' - t),$$

In other words, $\mathrm{D}(t' \to t)$ is a modified difference in performance between a aligned and misaligned models. The modification ensures that, as performance deteriorates, D increases, regardless of the direction of time between t and t'.

Our temporal degradation (TD) score for a fixed evaluation timestamp t for models trained on a set of timestamps $\mathcal{T}$ is defined as:

$$\mathrm{TD}(\mathcal{T} \to t') = \left| \frac{\sum_{t\in\mathcal{T}} \left(\mathrm{D}(t' \to t) - \bar{D}\right)(t - \bar{t})}{\sum_{t\in\mathcal{T}} (t - \bar{t})^2} \right|,$$

where $\bar{t} = \mathrm{avg}_{t\in\mathcal{T}} t'$ and $\bar{D} = \mathrm{avg}_{t\in\mathcal{T}} \mathrm{D}(t' \to t)$. This metric is the *slope* of a line fitting the the performance change of models trained on a variety of timestamps, when evaluated on a fixed timestamp. It can be interpreted as the average rate of performance deterioration per time period.

Fig. 6 shows three examples of TD scores from PoliAff(the first) and YelpCls(the latter two). These illustrate cases with and without temporal sensitivity. In practice, most examples with deterioration showed a linear trend and thus the rate of degradation was suitible to be approximated by a line. The final TD score is averaged over all evaluation years $\mathcal{T}'$.

$$\mathrm{TD} = \frac{\sum_{t\in\mathcal{T}'} \mathrm{TD}(\mathcal{T} \to t)}{n}$$

B Details of Model Development

Training Details for Temporal Adaptation We train GPT2 over each domain and timestamp for k steps using Huggingface's implementation of GPT2. Hyperparameter details can be seen in Table 3.

[14] See examples in Fig. 3.

Hyperparameter	DAPT Assignment
Number of steps	10k
Batch size	32
Maximum learning rate	5e-05
Adam Epsilon	1e-08
Adam Beta	0.9. 0.999
Block size	1024

Table 3: Hyperparameters for temporal adaptation across the four domains.

Hyperparameter	Cls. Assign	Summ. Assign
Number of Epochs	50	10
Batch size	32	8
Max learning rate	2e-05	2e-05
Adam Epsilon	1e-08	1e-08
Adam Beta	0.9. 0.999	0.9. 0.999
top p (sampling)	-	0.05
top k	-	20
temperature	-	1
max length	-	512

Table 4: Hyperparameters for temporal finetuning across the eight tasks.

Training Details for Temporal Finetuning We use Huggingface's implementation of GPT2 for finetuning for both the classification and summarization tasks. We train on Quadro RTX 800 GPUs. See Table 4 for details.

C Data Collection

We describe the postprocessing and data collection in greater detail. Table 5 depicts examples and detailed statistics for our task. All data released is intended for non-commercial use.

PoliAff We acquire a list of U.S. politician names and Twitter handles[15]. One of the authors manually annotated if the politician was a Republican or Democrat. In addition, one volunteer double checked to ensure correctness. We throw away any politician who changed parties between 2015 and 2020, any independents, and anyone suspended by Twitter (e.g., RealDonaldTrump).

[15] https://files.pushshift.io/twitter/US_PoliticalTweets.tar.gz

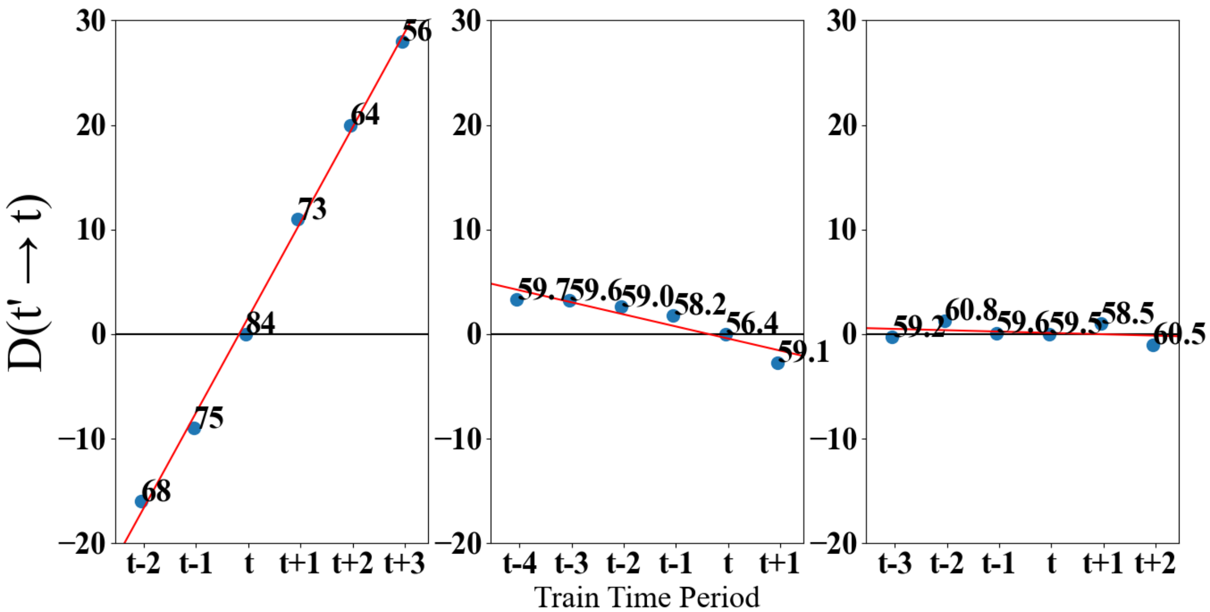

Figure 6: Three example calculations of the TD score (left from POLIAFF and the center and right from YELP-CLS). The annotated numbers are the raw evaluation scores $S_{t'\to t}$ and the plotted markers represent the modified differences $D(t' \to t)$ discussed in Section 2.3. For a particular plot, the red line is the line of best fit and its slope is the TD(t) score for evaluation timestep t. The final TD score is averaged between all evaluation timesteps for the particular task.

Domain	Task	Time Range	Size	Example
Twitter	political affiliation classification	2015-2019	120k	**Input:** *History will note that Trump didn't merely fiddle while the planet burned but tried to throw the Arctic National W...* **Output:** *Democrat (vs Republican)*
	entity type classification	2014-2019	8k	**Input:** *entity: Finola, tweet: Two 64-year olds enjoying their first birthday together in 40+ years. My twin sister, Finola, and I.* **Output:** *Person*
Science	mention type classification	1980-2016	8k	**Input:** *mention: deep Long Short-Term Memory (LSTM) subnetwork, abstract: In this paper, we study the problem of online action detection from the streaming skeleton data .... by leveraging the merits of the deep Long Short-Term Memory (LSTM) subnetwork, the proposed model ...* **Output:** *Method*
	venue classification	2009-2020	16k	**Input:** *Rank K Binary Matrix Factorization (BMF) approximates a binary matrix by the product of two binary matrices of lower rank, K...* **Output:** *AAAI (vs ICML)*
News	media frame classification	2009-2016	20k	**Input:** *You think you have heard the worst horror a gun in the wrong hands can do, and then this.You think there could not have been anywhere more tragic for it to happen...* **Output:** *Gun Control (15 possible frames)*
	publisher classification	2009-2016	67k	**Input:** *A Muslim woman said Sunday that her viral article explaining why she voted for Donald Trump has angered her liberal pals as well as other Muslims.* **Output:** *FoxNews (vs NYTimes or WaPost)*
	summarization	2009-2016	330k	**Input:** *The Consumer Financial Protection Bureau is demanding PayPal return \$15 million to consumers and pay a \$10 million fine for ...* **Output:** *The CFPB alleges many customers unwittingly signed up for PayPal Credit*
Food Reviews	review rating classification	2013-2019	126k	**Input:** *What a beautiful store and amazing experience! Not only the atmosphere, but the people...* **Output:** *4 (out of 5)*

Table 5: The tasks from four domains studied in this paper, with examples. See Section 2.4 for more details.

AIC We randomly sample science documents in Semantic Scholar's corpus.[16] Of those, we only keep documents that (1) are published in ICML or AAAI, (2) are classified as 'computer science' documents, and (3) have an abstract of at least 50 tokens.

Newsroom The following applies to the postprocessing and data selection for both supervised temporal finetuning and unsupervised temporal adaptation of PUBCLS and NEWSUM. We use the Newsroom dataset.[17]. We only keep articles where (1) the year in the metadata also appears in the main text and (2) no future year is mentioned in the main text.

PUBCLS We carry out additional postprocessing and ensure that each of the three labels (Fox News, New York Times, and Washington Post) have an equal distribution across years. We do so by uniform-random downsampling.

D Extended Results

We provide further results from our experiments described in Section 3.

Label Distribution Drift We measure how label distributions in task datasets change over time, as described in Section 3.1. For each task and each test period, we calculate the KL divergence between the label distribution of that period andthe first test period. Fig. 7 depicts our results.

Finetuning Results We provide the full results from our fientuning experiments in Section 3.1 in Fig. 8. These results are for downstream tasks with no domain adaptation.

Finetuning with Temporal Domain Adaptation We provide the full results from our finetuning with temporal domain adaptation in Section 3.2 in Fig. 6.

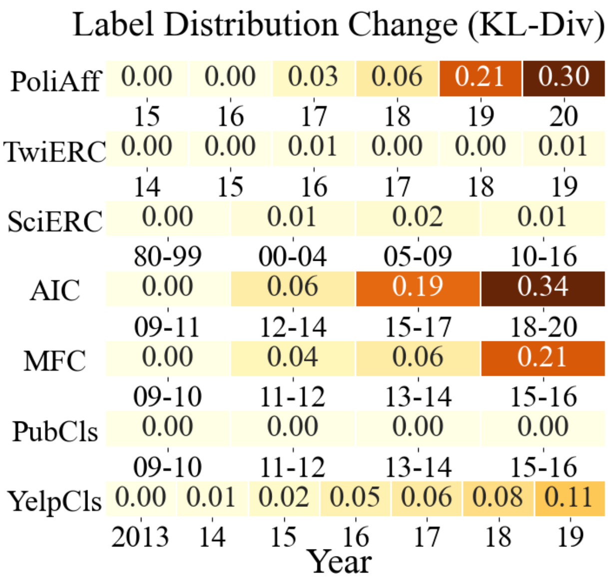

Figure 7: KL divergence between label distributions over time for all tasks. For each cell, we compare the distribution of labels to that of the first time period; e.g., the 2017 POLIAFF cell contains the KL-divergence between the label distributions of POLIAFF in 2017 and 2015. We note that while most distributions see little change over time, POLIAFF, AIC and MFC see a large shift. However, we note that POLIAFF was sensitive to temporal misalignment while MFC was not.

[16] https://api.semanticscholar.org/corpus/; licensed under an ODC-BY

[17] https://lil.nlp.cornell.edu/newsroom/

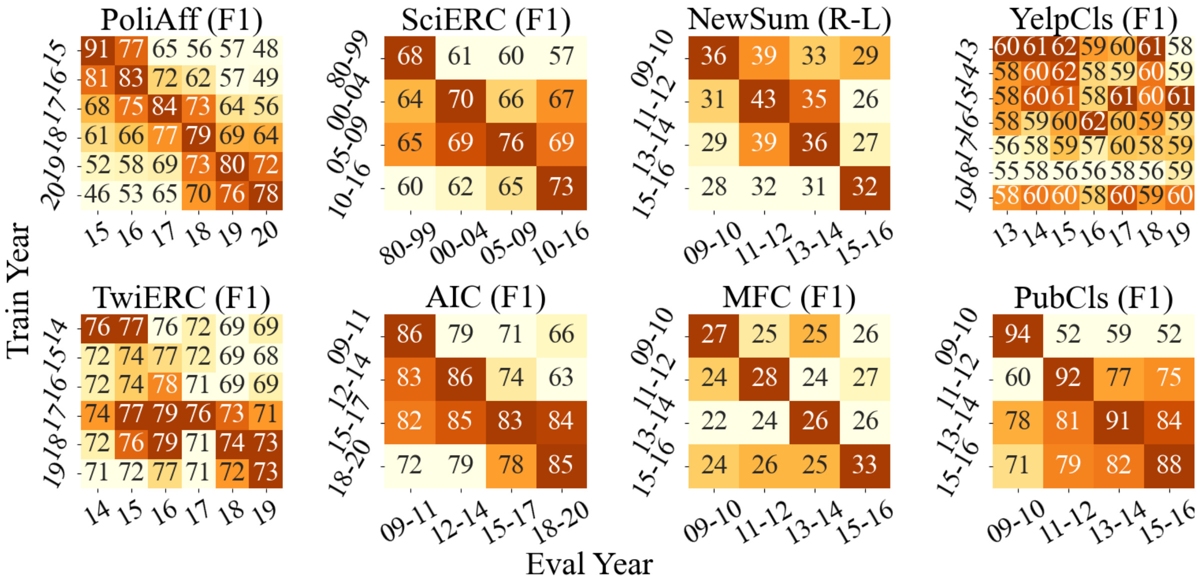

Figure 8: emporal misalignment in finetuning affects task performance (§3.1). In all cases, higher scores are better. The heatmap is shaded per column, i.e., the darkest shade of orange in a cell means the cell has the highest score in that column. Mismatch between the the training and evaluation data result in massive performance drop. While all suffer from temporal degradation, its degree is a strong function of task definition. For example, YELPCLS, MFC, and TWIERC show minimal degradation. In contrast, POLIAFF and NEWSUM major deterioration over time.

Domain (Task)	**Finetune Year**	**Evaluation → Pretrain ↓**	2015	2020	**Domain (Task)**	**Finetune Year**	**Evaluation → Pretrain ↓**	2014	2019
Twitter (PoliAff) *F1*	2015	Default	91.4	48.4	Twitter (TwiERC) *F1*	2014	Default	74.3	68.9
		Default → 2015	92.2	47.5			Default → 2014	76.1	69.6
		Default → 2020	90.9	50.8			Default → 2019	74.1	68.9
	2020	Default	45.8	78.0		2019	Default	71.0	74.6
		Default → 2015	47.2	76.9			Default → 2014	73.1	75.2
		Default → 2020	44.2	78.3			Default → 2019	73.7	75.8
Domain (Task)	**Finetune Year**	**Evaluation → Pretrain ↓**	2009-11	2018-20	**Domain (Task)**	**Finetune Year**	**Evaluation → Pretrain ↓**	1980-1999	2010-2016
Sciencitific (AIC) *F1*	2009-2011	Default	79.0	72.0	Scientific (SciERC) *F1*	1980-1999	Default	67.9	57.2
		Default → 2009-2011	94.5	68.8			Default → 1980-1999	73.2	66.4
		Default → 2018-2020	88.4	86.0			Default → 2010-2016	73.7	66.8
	2018-2020	Default	72.0	85.0		2010-2016	Default	60.3	72.5
		Default → 2009-2011	87.2	65.2			Default → 1980-1999	63.4	75.0
		Default → 2018-2020	86.8	79.4			Default → 2010-2016	64.8	76.0
Domain (Task)	**Finetune Year**	**Evaluation → Pretrain ↓**	2009-2010	2015-2016	**Domain (Task)**	**Finetune Year**	**Evaluation → Pretrain ↓**	2009-2010	2015-2016
News (MFC) *F1*	2009-2010	Default	27.0	26.0	News (PubCls) *F1*	2009-2010	Default	94.1	52.4
		Default → 2009-2010	30.6	31.8			Default → 2009-2010	95.4	54.0
		Default → 2015-2016	29.8	30.0			Default → 2015-2016	95.4	53.5
	2015-2016	Default	23.8	33.4		2015-2016	Default	71.3	88.2
		Default → 2009-2010	29.7	41.6			Default → 2009-2010	80.4	90.7
		Default → 2015-2016	32.7	41.9			Default → 2015-2016	78.7	91.1
Domain (Task)	**Finetune Year**	**Evaluation → Pretrain ↓**	2009-2010	2015-2016	**Domain (Task) ↓**	**Finetune Year ↓**	**Evaluation → Pretrain ↓**	2014	2019
News (NewsSum) *Rouge-L*	2009-2010	Default	36.4	29.0	Food Reviews (Yelp) *F1*	2013	Default	58.6	58.3
		Default → 2009-2010	36.4	29.1			Default → 2013	63.3	60.1
		Default → 2015-2016	36.1	28.9			Default → 2019	60.2	62.3
	2015-2016	Default	27.8	31.8		2019	Default	58.3	58.3
		Default → 2009-2010	28.2	31.8			Default → 2013	60.2	62.3
		Default → 2015-2016	27.8	31.6			Default → 2019	60.8	62.3

Table 6: Combination of temporal adaptation and finetuning (§3.2) on our tasks. The row labeled "Default" corresponds to a model that has not been adapted (uses the default pretraining). The color coding is proportional to the magnitude of the performances of each task (darker shade of orange indicates higher scores). We see that models that were finetuned on similar time periods performed similarly, no matter how their DAPT conditions differed.