Investigating Natural Language Interactions in Communities

Kelvin Luu

A dissertation
submitted in partial fulfillment of the
requirements for the degree of

Doctor of Philosophy

University of Washington
2022

Reading Committee:
Noah A. Smith, Chair
Hannaneh Hajishirzi
Sheng Wang

Program Authorized to Offer Degree:
Computer Science and Engineering

University of Washington

Abstract

Investigating Natural Language Interactions in Communities

Kelvin Luu

Chair of the Supervisory Committee:
Noah A. Smith
Computer Science and Engineering

In this work, we investigate language use in communities. First, we study how authors of scientific texts explain how their paper relates to another. We propose a new task, relationship explanation generation for scientific texts, by using in-line citation text as a source of evidence. We introduce a new model, SCIGEN, empowered by pretrained language models. We perform extensive human evaluation and discuss potential shortcomings of our system's generations.

Second, we describe work on investigating how NLP models degrade over time due to the dynamic nature of most communities. We find evidence that static models, like SCIGEN, do not generalize temporally. Based on this finding, we investigate how model performance deterioration over time differs across several tasks and domains. We discover that models with temporally misaligned train and testing sets can suffer from large amounts of performance degradation.

Finally, we show how elevating temporal characteristics in communities allows us to study particular social phenomena. Namely, we explore the problem of quantifying persuasive skill over time. Using data from an online debate forum, we construct a model of debater skill based on on the Elo ranking model and incorporate historical lingusitic data. In order to estimate skill, we frame our prediction task to *forecast* the outcome of a debate.

Though this work considers a wide range of NLP applications, it is unified by the idea that language

emerges from ever-changing communities of people engaging with each other. Our findings demonstrate how the temporal dynamics and social underpinnings of language can inform NLP research and practice.

Acknowledgements

The work in this thesis could only be completed due to my amazing mentors, friends, and colleagues.

To my advisor, Noah A. Smith: On top of his invaluable professional, personal, and writing advice, he has been incredibly supportive of me. I strongly believe that, with any other advisor, I would not have finished. Thank you for continuing to have faith in me and being stubborn enough to see my PhD through.

To my committee, Hannaneh Hajishirzi, Sheng Wang, and Jevin West: I thank them for being so flexible and helpful when planning Generals in the midst of the pandemic. They were very patient with my many e-mails and last-minute requests. I also thank them for their feedback, which has strengthened this thesis.

To Lillian Lee, Chenhao Tan, and Igor Labutov: these were the individuals who set me on the research path while I was an undergraduate. Their mentoring early on showed me how rewarding research could be and led me to pursue NLP research.

To the Seattle NLP community: I have been so lucky to have worked in an institution and city where some of the brightest minds in NLP. The work presented here is based on work written with so many brilliant people who have come to Seattle to do research. Special thanks to my co-authors: Noah Smith, Chenhao Tan, Rik Koncel-Kedziorski, Kyle Lo, Xinyi Wu, Isabel Cachola, Daniel Khashabi, Suchin Gururangan, and Karishma Mandyam.

To my former and fellow members of Noah's Ark: all of you have done so much for me from revising my papers or talks to giving recommendations on what shows to watch next. Thanks to Sofia Serrano, Lingpeng Kong, Yangfeng Ji, Chenhao Tan, Hao Peng, Jungo Kasai, Nikita Haduong, Tal August, Suchin Gururangan, Jungo Kasai, Alisa Liu, Ofir Press, Phillip Keung, Tao Yu, Yizhong Wang, Nikkos Pappas, and Rik Koncel-Kedziorski for the advice, random mid-Ark meeting slack messages, and thoughtful discussions we had. Special thanks to those in the ark who joined at the same time too: Phoebe Mulcaire, Lucy Lin,

Maarten Sap, Rahul Nadkarni, and Elizabeth Clark. I could not have begun graduate school with a better set of colleagues.

To many of my fellow students at UW: The ideas and work presented here have been influenced by others in the UW NLP community as well (there are far too many to name). I am also grateful for the conversations and fun (either grabbing coffee or leftover food) with my officemates: Elizabeth Clark, Mandar Joshi, Julian Michael, Srini Iyer, and Ofir Press.

To my parents, Thomas and Jenny Luu: An especially large thanks for raising me. They have sacrificed much for my education and provided so much for me from the beginning and continue to do so.

To Xing: Who has supported me unconditionally through rough times - close and far.

DEDICATION

To mom and dad.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Interactions between members of a community provide a rich source of data for studying various language use patterns. In today's world, many of these interactions are available as natural language text. For example, there are 2.1 million daily users generating over 500 million tweets per day on Twitter; Wikipedia has discussions over prior revisions spanning over a decade; and even peer-reviews on scientific works in submission are publicly accessible on OpenReview. The abundant data, often signifying meaningful interactions, allows for research into various social phenomena within a community.

Such data inspires much of Natural Language Processing (NLP) research and models. Computational social scientists often apply NLP techniques on social media websites, such as Reddit or Twitter, to answer social research questions. These online communities exhibit rich, naturally occurring interactions in text and offer ways to study social phenomena such as controversy, toxicity, or persuasion at scale [Hessel and Lee, 2019; Xu et al., 2021; Tan et al., 2016].

Social interactions in text empower core NLP tasks and models as well. Popular benchmarks, like Stanford Sentiment Treebank-2 or SQuAD, are built by members of online communities. The much the successes of modern NLP can be attributed to pretrained language models, such as GPT3 [Brown et al., 2020a]. These models have largely included vast amounts of social media text in their training data [Radford et al., 2019a; Brown et al., 2020a]. Consequently, many of the NLP systems deployed, for either commercial use or research, implicitly rely on social data.

Knowledge about social interactions in communities and NLP modeling and applications build on each

other. Advances in one drive innovation in the other. As described previously, social interactions have led to new NLP tasks and applications. More recently, researchers have leveraged social information in data for applications in advice giving or reasoning about common social norms[Zellers et al., 2021; Emelin et al., 2021; Jiang et al., 2021]. Gururangan et al. [2020a] showed that further pretraining is a simple, yet effective way of adding knowledge about a desired domain to a model and improve performance.

Likewise, improvements to modeling have allowed researchers to answer different social questions in communities. For example, pretrained language models such as BERT [Devlin et al., 2018] have allowed researchers to characterize linguistic variation in communities [Lucy and Bamman, 2021] and find effective strategies in science communication [August et al., 2020]. In this thesis, we suggest unique characteristics in communities lead to new NLP applications and that improvements in modeling let us uncover new knolwedge of the intricacies of a community.

One example of nuance in communities is how they evolve over time. New interests, changes in rules, or members learning to communicate can all contribute to shifts in language use [Soni et al., 2021; Jhaver et al., 2021; Danescu-Niculescu-Mizil et al., 2013]. In the latter part of this dissertation, we study temporal effects in tasks and communities. We find that integrating awareness of these temporal shifts into our models can not only give better predictive performance, but also allow scientists to tackle research questions about a community.

1.1 Overview

First, we study how computer science researchers explain the relationship of one of their papers to another. From these social interactions, we introduce a new task, relationship explanation generation for scientific texts, and operationalize it using in-line citation text as a source of evidence. We build on a pretrained language model, GPT2 [Radford et al., 2019a], to construct SciGen.

Due to limitations in input context windows of many pretrained language models, we cannot use entire scientific texts as input. Instead, we seek to find a dense representation of a scientific document that provides enough information to for SciGen to generate meaningful explanations. We experiment with various methods of selecting full sentences to use as input for scientific documents. Since explanations of scientific texts are highly technical, we perform a highly expert human evaluation and discuss potential future directions.

We note, however, that SCIGEN models only a limited view of how the scientific community explains other work for a certain time period. Indeed, we find evidence that static models like SCIGEN may suffer from performance degradation over time. Consequently, we investigate how NLP models deteriorate over time in Chapter 3. Prior work has broadly established that language use can change over time for various reasons [Labov, 2011; Altmann et al., 2009; Eisenstein et al., 2014]. However, recent research that studied language shift over time has mostly focused on a narrow set of domains or tasks [Röttger and Pierrehumbert, 2021; Cao et al., 2021; Zhang and Choi, 2021; Rijhwani and Preoţiuc-Pietro, 2020]. We consider phenomena that are characteristic of particular domains, by investigating performance degradation of NLP models for a variety of downstream tasks and text domains.

Finally, we use the temporal aspect of communities to measure persuasive skill acquisition over time. By explicitly modeling temporal linguistic features of members of an online debate forum, we quantify persuasive skill acquisition. In this project, we build a *linguistic profile* of a debater, or a summmary of their language use. We use the linguistic profiles to build on the Elo ranking model, a system for ranking participants in two-player games. Through our analysis, we are able to draw conclusions such as which types of features are correlated with skill and how more skilled users become experts over time.

Throughout this work, we consider a wide range of NLP tasks. We show, in these tasks, that knowledge about communities and modeling choices build on each other. We emphasize that social interactions play a significant role in both cases. Ultimately, a significant portion of NLP tasks, even those not covered here, may require processing natural language emerging from dynamic communities of people engaging with each other. Our findings demonstrate how temporal dynamics and social underpinnings of language can inform NLP research and practice.

.

Chapter 2

Explaining Relationships Between Scientific Texts

2.1 Introduction

The output of the world's scientists doubles roughly every nine years [Bornmann and Mutz, 2015]. Consequently, researchers must devote significant energy to quickly understand how a new piece of research fits with a rapidly changing research landscape. One avenue of research to alleviate this burden is to study how they convey their ideas in scientific text. Scientists often communicate their ideas and research to each other through scientific papers. From these papers, we see which areas are popular with researchers [Jung and Segev, 2013; Chakraborty et al., 2014] or which terms are likely to be adopted [Soni et al., 2021]. These natural language interactions in the scientific community can be useful to inspire new applications and models that help alleviate researcher burden.

Several lines of research have already studied interactions between researchers to reduce this scientist workload. Citation recommendation systems suggest references to relevant published work [McNee et al., 2002; Bhagavatula et al., 2018]. Intent classification systems help determine the type and importance of a citation in a work [Valenzuela et al., 2015; Cohan et al., 2019]. Summarization systems aim to help researchers more quickly understand the basic ideas in a piece of research [Cohan and Goharian, 2015; Yasunaga et al., 2019]. We draw inspiration from these works as well as broader challenges like explaining

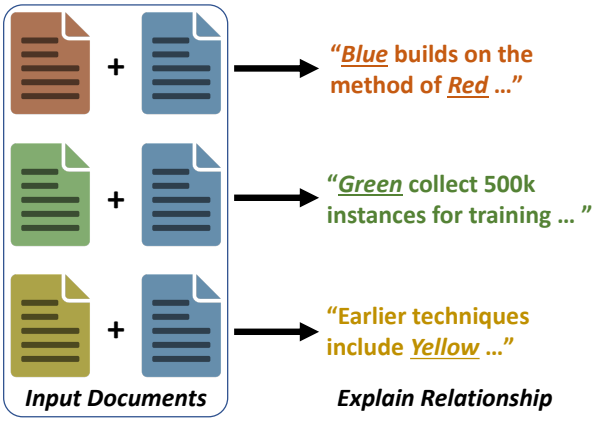

Figure 2.1: Given two scientific documents, the goal is to write the sentence describing the specific relationship between them. For a given document (in blue above), the output will vary depending the content of the other.

the connection between concurrent works or relating a new paper to those a reader is already familiar with.

Automatically describing inter-document relationships could decrease the time researchers devote to literature review. For instance, explanations for a new paper can be personalized to a particular reader by relating the new work to ones they have read before. Further, such technology could be incorporated into writing assistance systems to help less experienced or non-native writers better articulate the connection between their work and prior art. Additionally, users of citation recommendation systems can benefit from natural language explanations of recommendation system choices.

In addition to the utility of this task to scientists, it presents several interesting technical challenges. These include effectively representing the important information in a document, generating from a long-tailed technical vocabulary, and expressing the variety of connections between related scientific papers. Figure 2.1 illustrates how the same document is described differently in relation to different documents.

In this chapter we use citing sentences, the interaction we focus on, to operationalize the problem of generating natural language explanations of the relationships between two scientific papers. Authors, when citing other work, oftentimes describe how their work relates to the cited work. To this end, we use in-text citation sentences as a naturally occurring proxy explanations for how two documents relate to each other. However, we generate such sentences from general representations of document content rather than the specific in-text locations where these sentences occur, as this task formulation can better facilitate the applications described above.

We describe our contributions: we establish a new task of generating relationship explanations; introduce a novel dataset for the task; release our SciGen model for describing document relationships; and provide an extensive evaluation and analysis of machine generated technical text.

This chapter contains materials originally in Luu et al. [2021].

2.2 Related Work

The current work builds on recent research in scientific document understanding, including citation recommendation, intent categorization, and scientific document summarization. Citation recommendation systems suggest related works given a document or a span of text [McNee et al., 2002; Nallapati et al., 2008; Bhagavatula et al., 2018]. Recently, researchers have sought to categorize citations using various ontologies of citation intents. Teufel et al. [2006] develop an annotation scheme and corresponding classification model for citation functions. Valenzuela et al. [2015] seek to discern "highly influential" citations from others. Jurgens et al. [2018] use six categories including "motivation," "uses," and "future work" among others. Cohan et al. [2019] condense this ontology to just three: "background," "method," and "result comparison." Intent classification can identify relationships between documents; our relationship explanation task extends this in two ways. First, data-driven freeform generation can express a wider array of relationships compared to a manually-defined label set. Further, our task framework could be used to describe relationships between works which do not actually cite each other, such as contemporaneous works. Unlike categorization techniques, we require no task-specific annotated data as we supervise with citing sentences that are readily available in scientific documents. In practice, citation classification is used to assist in suggesting relevant works to researchers; our work complements this goal by providing rationales for the recommendation and furthering progress toward explainable AI.

Our work is also connected to a long history of research on summarizing scientific documents [Luhn, 1958; Paice, 1980]. Work in this area has mostly used used abstracts or peer reviews as targets Cachola et al. [2020]; Cohan et al. [2018]; Jaidka et al. [2017]. In particular, Pilault et al. [2020] show that using a simple extractive summary as input for abstractive summarization of scholarly texts work well. Researchers have also used citing sentences as part of the input for summarization, recognizing the explanatory power of these texts [Nakov et al., 2004; Cohan and Goharian, 2017; Yasunaga et al., 2019]. Ours is the first work to focus

on learning to express the specific relationship between two documents from such sentences.

The closest work to our own is Xing et al. [2020], who pilot a task of in-line citation generation. Their goal is a model which can insert a citing sentence into a particular context within a document. Our work, on the other hand, aims to learn from citing sentences how to describe general relationships between documents independent of particular in-document contexts. While the Xing et al. [2020] method may facilitate writing assistance, our task has applications in search and summarization. Because our task does not rely on a specific location in a document where the citation will go, solutions can be used at scale to provide users with general explanations of document relationships.

Our models rely heavily on recent advances in transfer learning in NLP. Large pretrained models such as BERT [Devlin et al., 2018] and GPT2 [Radford et al., 2019b] have made strong advances on a number of tasks [Wang et al., 2019]. It has also been shown that pretraining these models on domain-specific data further improves results on domain-specific tasks [Beltagy et al., 2019; Lee et al., 2019]. In this work, we apply that methodology by adding a pretraining phase on in-domain data before finetuning a GPT2 model toward the explanation generation task. A key challenge when using pretrained language models for document-level tasks is how to select document content to fit within the limited context window of the model, which is a major focus of our work.

2.3 Problem Definition

We aim to generate an explanation: a natural language sentence which expresses how one document relates to another. Explicit examples of such sentences are nontrivial to find in corpora, especially when annotation for a highly technical task is expensive. To this end, we use in-text citations in a scientific document to prior work as proxies for relationship explanations. We use these citing sentences as partial supervision for our task, and refer to them as "explanations."[1]

We distinguish one document as the *principal* document, from which we will draw explanations that reference the *cited* document. Let t denote an explanation drawn from principal document S, and S' denote

[1]Future work might seek to filter or systematically alter in-text citations to be more explanation-like, without otherwise changing our approach.

	total	average/doc.
documents	154K	–
tokens	813M	5.3K
unique tokens	7.1M	1.3K
explanations	622K	4.0

Table 2.1: Dataset statistics, total and per document.

S without t. Then let

$$P(t \mid S', C) \tag{2.1}$$

be the probability of t given S' and the cited document C. A good generation technique should maximize this probability across a large number of $\langle t, S, C \rangle$ triples, so that at inference time the model is able to generate a sentence t^* which accurately describes the relationship between new documents $\hat{S}$ and $\hat{C}$.

Optimizing Equation 2.1 is made easier by modern representation learning. Pretrained neural language models like GPT2 have shown strong performance when generating sentences conditioned on a context. However, existing implementations of GPT2 limit the context window to 512 or 1024 tokens, far smaller than scientific documents. In this work, we explore ways to represent the documents' content for use with language models.

Data We use English-language computer science articles and annotation from the S2ORC dataset [Lo et al., 2020a]. S2ORC is a large citation graph which includes full texts of 8.1 million scientific documents. We use 154K connected computer science articles, from which we extract 622K explanations with a single reference that link back to other documents in our corpus. We omit any sentences that cite more than one reference. We hold 5000 sentences for each of the validation and test sets. Detailed statistics can be found in Table 2.1.

Evaluation The most appropriate evaluation metric for this and many text generation tasks is human judgment by potential users of the system. Evaluating explanations of the relationships between scientific documents requires human judges with scientific expertise whose time and effort can be costly. While collecting human judgments in technical domains is relatively rare, we believe it to be an important step in evaluating our systems for this task. Thus, we conduct thorough human evaluations and analyses with expert judges. We make use of both larger scale expert evaluations yielding hundreds of judgements as well as

smaller scale, deeper evaluations where we can effect a higher degree of quality control over fewer datapoints. Further, we make use of intermediate human evaluations in the development of our models, and supplement these evaluations with automatic metrics — BLEU [Papineni et al., 2002] and ROUGE [Lin, 2004] that are established in other generation tasks.

2.4 Models

We develop several models for explaining document relationships. Following current work in neural text generation, we finetune the predictions of a large pretrained language model to our task (§ 2.4.1). In order to bring the language model into the scientific text domain, we do additional language model pretraining over full scientific texts. We also investigate approximate nearest neighbor methods to retrieve plausible human-authored explanations from the training data as a baseline (§ 2.4.2).

2.4.1 Neural Text Generation

Recent work has shown that finetuning large pretrained language models to text generation tasks yields strong results [Zellers et al., 2019]. To this end, we construct SCIGEN, a model based on GPT2 [Radford et al., 2019b], a transformer model trained on 40GB of internet text with a left-to-right language modeling objective [Vaswani et al., 2017]. We do so by finetuning the predictions of the language model to generate explanations using different expressions of the principal and cited document as context.

To finetune GPT2 architectures for text generation, it is typical to concatenate the conditioning context $X = x_1 \dots x_n$ and target sentence $Y = y_1 \dots y_m$ with a special separator token ξ^y. To adapt this technique to our task, we construct the conditioning context X from the principal and cited documents and use the explanation as Y. We take j tokens from principal document $s_1, \dots, s_j$ along with k tokens from the cited document $c_1, \dots, c_k$ (which tokens to draw from the two documents is an independent variable that we explore experimentally). We then condition the generation of explanation Y on $X = s_1, \dots, s_j, \xi^x, c_1, \dots, c_k$, where ξ^x is a token used to indicate the end of the principal document. SCIGEN is trained to predict the explanation one token at a time as described above.

At inference time, the model is provided with an unseen principal/cited document pair. An explanation of their relationship is generated one token at a time using nucleus sampling Holtzman et al. [2020]. At timestep

Figure 2.2: Overview of the construction of SciGen. We take the pretrained GPT2 and continue pretraining on scientific texts. We then finetune using data in Table 2.1.

t, output token $\hat{y}_t$ is sampled from the top 90% of the distribution $P(\hat{y}_t \mid X, \xi^y, \hat{y}_1, \ldots, \hat{y}_{t-1})$ (renormalized). The selected $\hat{y}_t$ is used to condition the prediction of subsequent tokens.

Context The primary question we investigate with the SciGen model is what kind of input is best for describing the relationship between the principal and cited documents accurately and informatively. Since models based on GPT2 have a small context window relative to the length of scientific documents, we investigate the use of abstracts, introductions, or non-citing sentences sampled from throughout the document as conditioning context. The effectiveness and description of these approaches is described in § 2.4.3.

Language Model Pretraining Prior work has shown that pretraining on in-domain data improves the performance of large language models on domain-specific tasks [Beltagy et al., 2019; Gururangan et al., 2020a]. Inspired by this, we continue pretraining the GPT2 model in the science domain to produce SciGPT2 , which we use as the underlying language model for SciGen described above. SciGPT2 starts from the standard pretrained GPT2-base model and is trained for an additional 75k gradient updates at a batch size of 64 (effectively a single epoch over 4.8 million abstracts and body paragraphs) with a language modeling objective. Figure 2.2 illustrates the process.

We observed significant improvements in the quality of SciGen outputs after replacing the underlying GPT2 language model with the domain-specific SciGPT2 model. We saw a perplexity improvement in a held-out set and, in informal inspections, qualitative improvements as well.

When using pretrained language models, text from task-specific test data cannot be guaranteed to be absent from the large task-independent corpora upon which these models are trained, which may improve model performance compared to models without this exposure. For the experiments described in this work,

we train a version of SciGPT2 only on documents appearing in the training data, so that the principal documents and target sentences in the test data are guaranteed to be unseen by the language model. We provide this and a full-corpus version of SciGPT2 as resources for future research.[2]

2.4.2 Retrieval with Approximate Nearest Neighbors

While neural text generation techniques have advanced significantly in recent years, their outputs are still inferior to human authored texts. For some tasks, it is better to retrieve a relevant human-authored text than to generate novel text automatically [Fan et al., 2018]. Is this also the case when generating explanations?

To answer this question, we use an information retrieval (IR) baseline. We adapt an approximate nearest neighbor search algorithm to find similar pairs of documents. The basic search procedure is as follows: Given a test instance input (S, C) for principal S and cited document C, we find the set $\mathbf{N}_C$, the nearest neighbors to C in the training data. For each document N_C from $\mathbf{N}_C$, let $\mathbf{N}_S$ be the set of documents that cite N_C. This means that each $N_S \in \mathbf{N}_S$ contains at least one citing sentence t' which cites N_C. We use the t' associated with the (N_S, N_C) pair from the training set which is closest to (S, C) as the explanation of their relationships, which we describe in more detail below.

We measure the closeness of two pairs of documents using the cosine distances between vector representations of their abstracts. The abstract of each document is encoded as a single dense vector by averaging the contextualized embeddings provided by the SciBERT model of Beltagy et al. [2019] and normalizing. The distance between (S, C) and neighbors (N_S, N_C) is computed as

$$\alpha \cos(S, N_S) + \beta \cos(C, N_C), \tag{2.2}$$

where α and β control the relative contribution of the two document similarities. We explore setting both α and β to 1, or tuning them to optimize BLEU on the validation data using MERT [Och, 2003].

2.4.3 Representing Documents with Sentence Selection

Methods for the related task of citation recommendation have made use of abstracts, which perhaps act as sufficient summaries of document content. Building on this, we represent the principal and cited documents

[2] https://github.com/Kel-Lu/SciGen

with the first 450 tokens of either their abstracts, introductions, or sentences randomly sampled from throughout the full document.[3] In this section, we answer two questions: 1) do neural generation models with sentence-based context outperform the IR baseline and 2) does the type of sentence-based context (abstract, introduction, sampled) matter? We answer these questions by performing both automatic and human evaluations.

2.4.4 Representing Documents with Information Extracted Contexts

We found in our work that generations using selected sentences as context can miss important phrases such as unique model or dataset names and other lower-frequency terms. We investigated using IE-based contexts, such as lists of salient words and phrases, as dense representations for our task. We report results and examples in Table 2.2 but omit details for space reasons. Please see our work, Luu et al. [2021] for more details.

2.5 Evaluation

We perform both automatic and human evaluation for our task. Automated metrics, such as BLEU or ROUGE are popular in generation tasks with reference texts since they are less expensive than human metrics Papineni et al. [2002]; Lin [2004]. However, since these metrics often rely on select texts to compare to, these automated metrics often do not cover the space of all acceptable answers. To this end, we supplement our automatic evaluation with an extensive human evaluation by experts in the NLP research community.

2.5.1 Automatic Evaluation

We compare the SciGen and IR systems using BLEU [Papineni et al., 2002] and ROUGE (specifically L; Lin, 2004). The "Sentence-based" rows of Table 2.2 show the test set performance of the IR system and the best SciGen models when provided with the different sentence-based input context combinations.

We assess statistical significance as well by bootstrapping with 1000 samples in each of 100 iterations. We find that context *does* make a difference for SciGen, and that a slight but statistically significant performance

[3]We exclude any sentence with a citation from being sampled in all conditions. This context type is also only used for the cited document and not the principal document.

	Method	Context	BLEU	Rouge-1	Rouge-2	Rouge-L
Sentence-Based	SciGen	principal abs × cited abs	9.82	10.7	0.6	8.4
		principal abs × cited intro	9.39	10.7	0.6	8.4
		principal abs × cited sample	9.60	10.7	0.7	8.5
		principal intro × cited abs	9.92	11.1	1.0	8.7
		principal intro × cited intro	9.80	11.1	1.1	8.8
		principal intro × cited sampled	9.81	10.9	0.9	8.7
	retrieval	principal abs × cited abs	9.93	14.2	0.7	9.7
		+ MERT (BLEU)	10.23	14.3	0.7	9.8
		no principal × cited abs	9.79	14.1	0.6	9.6
IE-based	SciGen	principal intro × cited tfidf	13.17	15.0	1.3	12.0
		principal abs × cited entities	13.10	14.3	0.8	11.4
		principal intro × cited entities	13.41	14.7	1.4	11.8
	+Ranking	principal intro × cited tfidf	13.50	15.5	1.6	12.3
		principal abs × cited entities	13.28	14.7	1.0	11.6
		principal intro × cited entities	13.16	15.0	1.3	11.8

Table 2.2: Automatic evaluation of generated texts for all of our systems. Our best models, the IE-based ones, are omitted for space reasons. Please see our work Luu et al. [2021] for details.

	Specific	Correct	S&C	*agr*
SciGen	72.3	64.0	55.0	*70.5*
IR	74.8	46.3	40.0	*77.5*
Gold	81.4	72.1	68.0	*83.8*
agreement	*69.8*	*71.4*	*63.1*	

Table 2.3: Human evaluation of SciGen (intro × abs) and IR (abs × abs) systems compared with gold explanations in percent. S&C represents those that were both specific and correct. All differences significant at $p < 0.01$ except SciGen vs. IR specific.

improvement comes from using the introduction of the principal document rather than the abstract.[4] We do not, however, find enough evidence to reject the null hypothesis that any particular representation of the cited document's content (abstract, intro, or random sample) yields any significant difference in performance.

We find that using the introduction of the principal document paired with the abstract of the cited document performs best, and so we select these for human evaluation. The IR systems perform well, obtaining slightly better scores in some settings. We choose the MERT-optimized version for human evaluation.

[4] $p < 0.01$ after Bonferroni correction.

2.5.2 Human Evaluation

We conduct a human evaluation to determine, given a particular pair of principal and cited abstracts, how *correct* and *specific* the generated explanation of their relationship is. By "correct" we mean: does the explanation correctly express the factual relationship between the principal and cited documents? Because generic explanations such as "This work extends the ideas of Chomsky and Halle (1968)", while possibly factual, do not express a detailed understanding of the documents' relationship, we ask judges whether the explanation describes a specific relationship between the two works. An explanation can be specific even it is incorrect.

We compare the *principal intro* $\times$ *cited abs* SCIGEN setting against the tuned IR system. For calibration, we also elicit judgments for the gold explanations extracted from principal documents along with the correct principal and cited abstracts. In all three cases, we ensure that the principal document appeared in the ACL anthology to ensure annotator expertise.

To ensure no annotator sees the output of more than one system on each datapoint, we randomly select 50 datapoints for each system (*principal intro* $\times$ *cited abs*, IR, and Gold explanations) from the subset of our test data whose principal documents appear in the ACL anthology. Each judge is given 15 datapoints for each of the specificity and correctness qualities. Judges are shown a table of datapoints and asked to mark whether each meets ("Yes") or fails to meet ("No") the condition. Judges are permitted to label "?" or skip examples they feel uncertain about or unqualified to judge, which we ignore. In total we solicit 37 NLP researchers and collect over 800 judgments, with over 100 for each system/quality dimension combination.

Table 2.3 shows the percentage of "yes" judgments versus the total of "yes" and "no" judgements for each system/quality combination, along with pairwise agreement rates. Gold texts received the highest scores for all dimensions of text quality from the evaluators as well as the highest agreement rate. We can also see that IR systems tend to produce incorrect explanations more often than not.

The SCIGEN system performs quite well in this analysis, with a majority of outputs deemed correct. We observe a larger difference in specificity between SCIGEN and gold texts, indicating that SCIGEN, like many neural text generation systems, often generates vague and generic sentences. These generations tended to be vacuous such as "(CITED) This work is an extension of the paper." Specificity is key for future downstream applications such as automated literature review and will need to be improved for those tasks.

	Correct
random cited	45.8
random principal	46.9
both random	17.6

Table 2.4: Correctness judgements of incorrect citing sentences (percentages).

Validity of Human Judgments

To test the validity of the human judgments in Section 2.5.2, we conduct an additional human evaluation of gold explanations paired with different kinds of mismatched inputs: (1) the correct principal document and a random cited document, (2) the correct cited document but a random principal document (3) random principal and cited documents selected from ACL anthology. Conditions 1 and 2 allow us to see whether human judges accept sentences which align with only one or the other of the input documents; condition 3 provides a lower bound. We collect 107 human evaluations of correctness across these conditions, again allowing annotators to skip datapoints they are unsure of. The results, shown in Table 2.4, indicate that human judges will sometimes accept a explanationas long as one of the principal or cited documents is correct, but at a lower rate than seen in Table 2.3 when both documents are correct. We note that both papers in the mismatched cases are drawn from the ACL anthology, meaning there is some amount of topical coherence in their pairing. There is no indication from this experiment that either the principal or cited document is a stronger influence on a judge's correctness decision, although a larger sample size might make a clear determination.

2.5.3 Discussion and Limitations

Error Analysis Example system outputs for selected test datapoints are shown in Table 2.5. The first example illustrates a case where the model identifies a correct relationship between the two documents. In this instance, they both use the pinyin representation for Chinese characters in their transliteration models.

Output 2 demonstrates a failure of the explanation generation system. The principal document deals with the topic of discourse relations, the automatic identification of which is a long-standing machine learning task. However, this particular document is an analysis paper, and does not involve any training.

1	Principal:	*A Syllable-based Name Transliteration System*
	Cited:	*A Joint Source-Channel Model for Machine Transliteration*
	SCIGEN:	Following *Cited* , Chinese characters are considered as Pinyin sequence.
2	Principal:	*Recovering discourse relations: Varying influence of discourse adverbials*
	Cited:	*The Benefits of a Model of Annotation*
	SCIGEN:	The two text collections provided by *Cited* were used for training, and the other two text collections were used for evaluation.
3	Principal:	*Coreference Resolution for Swedish and German using Distant Supervision*
	Cited:	*Collective Cross-Document Relation Extraction Without Labelled Data*
	SCIGEN:	It is one of the most widely used distant supervision techniques and is inspired by techniques proposed by *Cited*.
4	Principal:	*Neural Text Generation in Stories Using Entity Representations as Context*
	Cited:	*Delete, Retrieve, Generate: A Simple Approach to Sentiment and Style Transfer*
	SCIGEN:	The authors of *Cited* proposed a model that combines neural generation with user interaction to create an object-centric reading experience.

Table 2.5: Example explanations. The given texts are the document titles and the SCIGEN outputs. In the last example, the two documents **do not** cite each other.

Output 3 is an example of a "Not specific (but not incorrect)" case. Here again the explanation generated by SCIGEN is topical, dealing with the concept of "distant supervision" that is key to both input documents. However, this sentence fails to capture the specific use that the principal makes of the research described in cited document.

It is notable that the SCIGEN model usually outputs syntactically correct and topical explanations, even given the difficulty of the vocabulary in this domain. This is consistent with many recent findings using domain-specific language models.

Low Probability Examples The final example, output 4, showcases potential for our system to explain concurrent work. The generated text summarizes the *cited* and implies that *principal* will build on that work. However, selected papers are both concurrent generation papers published in the same venue and do not cite each other. This appears to be a weakness in using citation sentences as proxies for relationship explanations. Citations of contemporaneous work occur less frequently, so these types of sentences appear less often in training. Similarly, relationship explanations between papers with more distant connections (e.g., "multi-hop" in the citation graph) are missing in our training data.

In addition to missing some relationships, not all citation sentences are useful as explanations. As pointed out by other work, citation sentences can often be simple summaries of the cited work Qazvinian and Radev

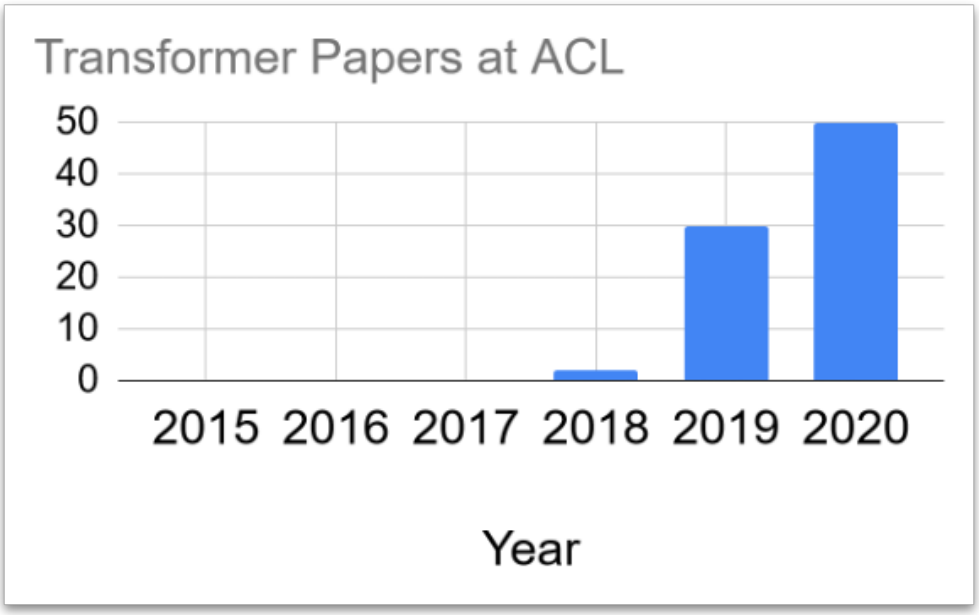

Figure 2.3: The number of times "Transformers" appear in the titles of papers published at the Assocation of Computational Lingustics, an NLP conference. We note that there is an increase of transformer papers in 2019, after their introduction.

[2008]; Cohan and Goharian [2017]. Alternatively, they can be too specific to be useful, as seen in Output 1, where a higher-level summary might be more useful. Future work could focus on curating better training sets for our task.

Modeling Only a Snapshot Like for many other models, we train SciGen on a singular dataset. Consequently, SciGen only captures a *snapshot* of the scientific community for a specific time period. However, communities in practice are not static; researchers typically focus on areas that are new or interesting to the community at large. For example, consider transformer models in the NLP community. As we see in Figure 2.3, we see a large increase in published papers on transformer models after their introduction. In Chapter 4, we will find other ways to natural language interactions that are sensitive to time.

Moreover, snapshot models of communities do seem to degrade. To demonstrate this deterioration, we take SciGPT2 and evaluate it for perplexity on documents from 2017 and 2021. As seen in Figure 2.4, the SciGPT2 model performs notably worse in language modeling perplexity when tested on articles from 2021, two years after SciGPT2 was trained. We will later discuss in Chapter 3 how this phenomenon extends to downstream tasks and is not unique to the research community.

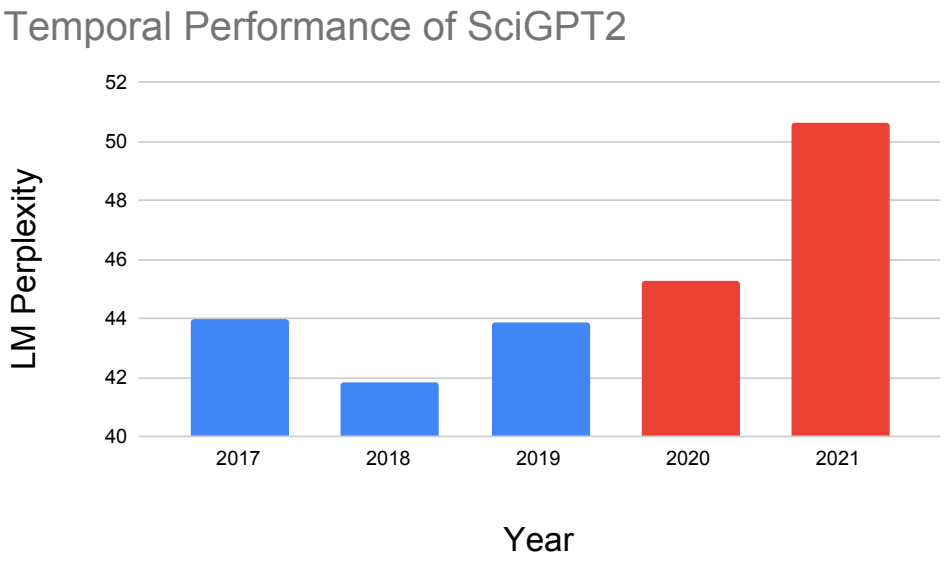

Figure 2.4: Language Modeling perplexity of SciGPT2 over scientific documents from 2017-2021. Lower is better. The red bars indicate the papers that came out after SciGPT2 was trained. These documents are out of distribution and SciGPT2 performs worse on these articles.

2.6 Summary

In this chapter, we described a task of explaining the relationship between two scientific texts and its connections to facilitating researcher productivity. We employed a large, publicly available dataset of scientific documents to train a domain-adapted left-to-right language model for use in text generation applications and beyond. We explored a collection of techniques for representing document content including using abstracts, introductions, and sampled sentences across the entire document. We conducted thorough human and automatic evaluations to determine the relative strengths of each representation for expressing document relationships in natural language text.

Chapter 3

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

3.1 Introduction

In Chapter 2, we modeled how authors of scientific texts explain their work in relation to other, prior work. Our model, SciGen, was essentially a snapshot of the scientific community at a particular time period. However, active communities are generally not static, as implied in the discussion section of the previous chapter (Section 2.5.3). Prior work has widely attested that language changes over time in various ways [Labov, 2011; Altmann et al., 2009; Eisenstein et al., 2014]; these changes have also been studied in the context of language shift in communities [Stewart and Eisenstein, 2018b; Soni et al., 2021]. We now focus on general language change over a variety of communities, domains, and tasks. Specifically, we focus on how these changes affect the long-term performance of NLP systems built from text corpora.

This chapter 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. However, these works tend to focus on a narrow set of domains or tasks. Prior work in computational social science has shown that communities have distinctive characteristics and show great

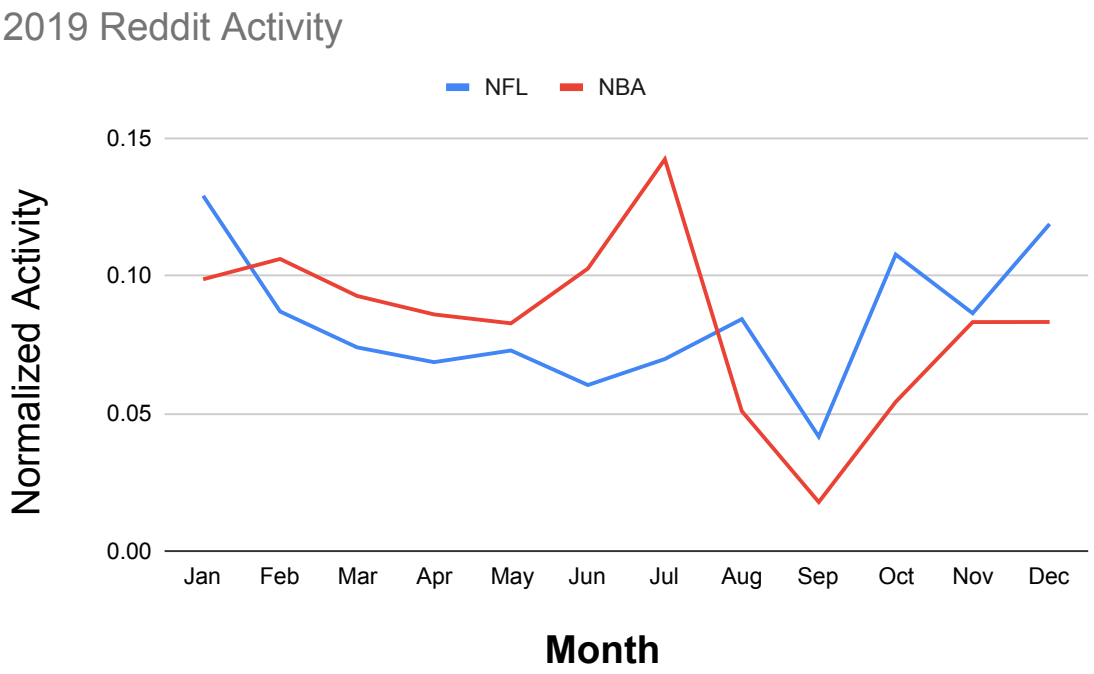

Figure 3.1: Normalized activity over the R/NBA and R/NFL subreddits. We note that the peaks for the two occur at different points of time: NBA peaks in June and July where playoffs and the beginning of the offseason occur while NFL peaks in December and January, near the end of the season.

amounts of linguistic variation [Lucy and Bamman, 2021; Zhang et al., 2017b]. For example, Figure 3.1 shows that even two sports-related communities on social media website *reddit.com* have peaks at different points in the year. We likewise suspect that the effects of temporal misalignment will vary depending on the community 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 (§3.4), ranging from summarization to entity typing, a subproblem of entity recognition [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:

- **(Q1)** *how does temporal misalignment affect downstream tasks over time?*
- **(Q2)** *how does sensitivity to temporal misalignment vary with text domain and task?*
- **(Q3)** *how does temporal misalignment affect language models across domains and spans of time?*

- **(Q4)** *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 (§3.5.2) and tasks (§3.5.1). Over five 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 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. This chapter includes content originally published in Luu et al. [2022].

3.2 Methodology Overview

We begin by defining the scope of our study.

3.2.1 Learning Pipeline

We consider a process for building an NLP model that is in widespread use by the research community, illustrated in Fig. 3.2. 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., 2020b). 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., 2020b). 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.

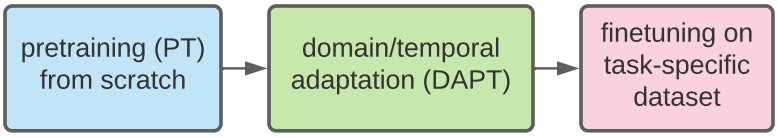

Figure 3.2: A typical modeling pipeline in NLP.

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.

3.3 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}.[1] 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 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.

3.3.1 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 expected speed of model degradation due to 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 timestep of difference between that the train and evaluation sets. Higher TD scores imply greater levels of performance deterioration due to misalignment.

Let t be the time period of the training data and t' the time period of the evaluation data. 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 timestep t' data and evaluated on timestep t. We define $D(t' \to t)$ as:

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

In other words, $D(t' \to t)$ is a modified difference in performance between a aligned and misaligned

[1] 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.

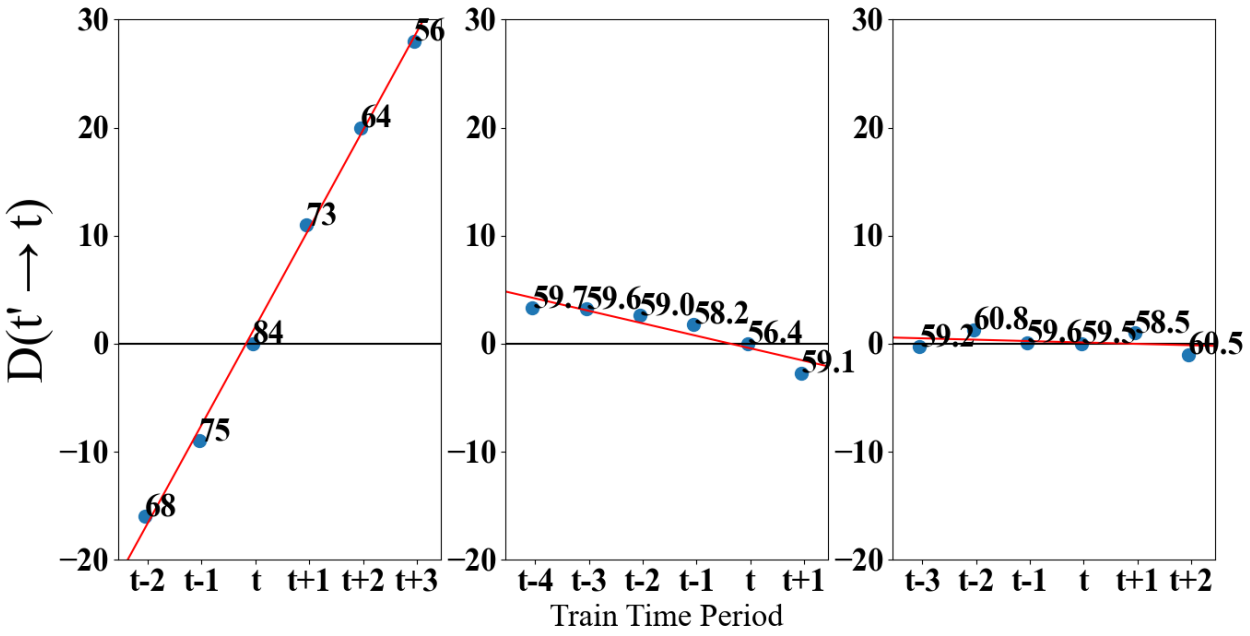

Figure 3.3: Three example calculations of the TD score (left from POLIAFF and the center and right from YELPCLS). The annotated numbers are the raw evaluation scores $S_{t' \to t}$ and the plotted markers represent the modified differences $D(t' \to t)$ discussed in Section 3.3.1. For a particular plot, 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.

models. The modification[2] ensures that, as performance deteriorates, D increases, regardless of the direction of time between t and t'.

We find a line of best fit for $D(t' \to t)$ for all t' using least-squares regression. The slope of this line is TD(t), the TD score for evaluation time period t. Formally, the temporal degradation (TD) score for a fixed evaluation timestamp t for models trained on a set of timestamps $\mathcal{T}$ is defined as:

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

where $\bar{t} = \text{avg}_{t \in \mathcal{T}} t'$ and $\bar{D} = \text{avg}_{t \in \mathcal{T}} D(t' \to t)$. 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.

[2]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.

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 3.1: The tasks from four domains studied in this paper, with examples. See Section 3.4 for more details.

Fig. 3.3 shows three examples of TD scores from our experiments[3]. 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 suitable to be approximated by a line. The final TD score is the average of the TD(t) across all evaluation time periods t

3.4 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 provide examples and detailed statistics in Table 3.1.

[3] POLIAFF (the first) and YELPCLS (the latter two); more information on these tasks can be found in Section 3.4

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.[4] 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.

Domain 2: Scientific Articles Scientific research produces vast amounts of text with great potential for language technologies [Wadden et al., 2020; Lo et al., 2020b]; 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,[5] 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

[4] Collected via the Twitter API.
[5] *https://api.semanticscholar.org/corpus/*

(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.[6] 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,

[6] *https://lil.nlp.cornell.edu/newsroom*

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,[7] 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.

3.5 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., 2020b], a LM often used for generation.[8] 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 degradation can be severe.

We then study how temporal misalignment affects LMs. As a first step, we analyze how vocabularies change over time in our datasets. We then experiment with (Q3) how temporal misalignment affects upstream language modeling and (Q4) how effective temporal adaptation, or additional pretraining on a target year, is in mitigating misalignment. We find that while LMs are affected by misalignment, temporal domain adaptation is not enough to mitigate temporal misalignment.

[7] *https://www.yelp.com/dataset*

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

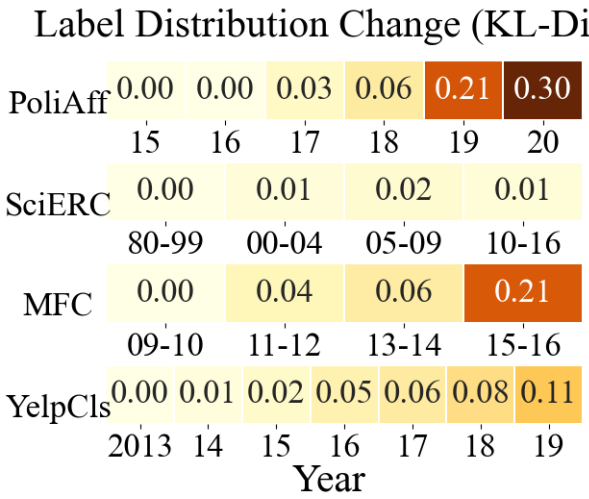

Figure 3.4: KL divergence between label distributions over time for a subset of tasks. For each cell, we compare the distribution of labels to that of the first time period; e.g., the 2017 POLIAFF cell contains the KL-divergence between the label distributions of POLIAFF in 2017 and 2015. While most tasks see little change over time, POLIAFF and MFC see a large shift.

3.5.1 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. 3.4. In three cases, we detected notable label distribution drift: POLIAFF, AIC, and MFC.[9] 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 §3.4, 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. 3.5 shows our results on downstream tasks (with no domain adaptation). To get more reliable

[9]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."

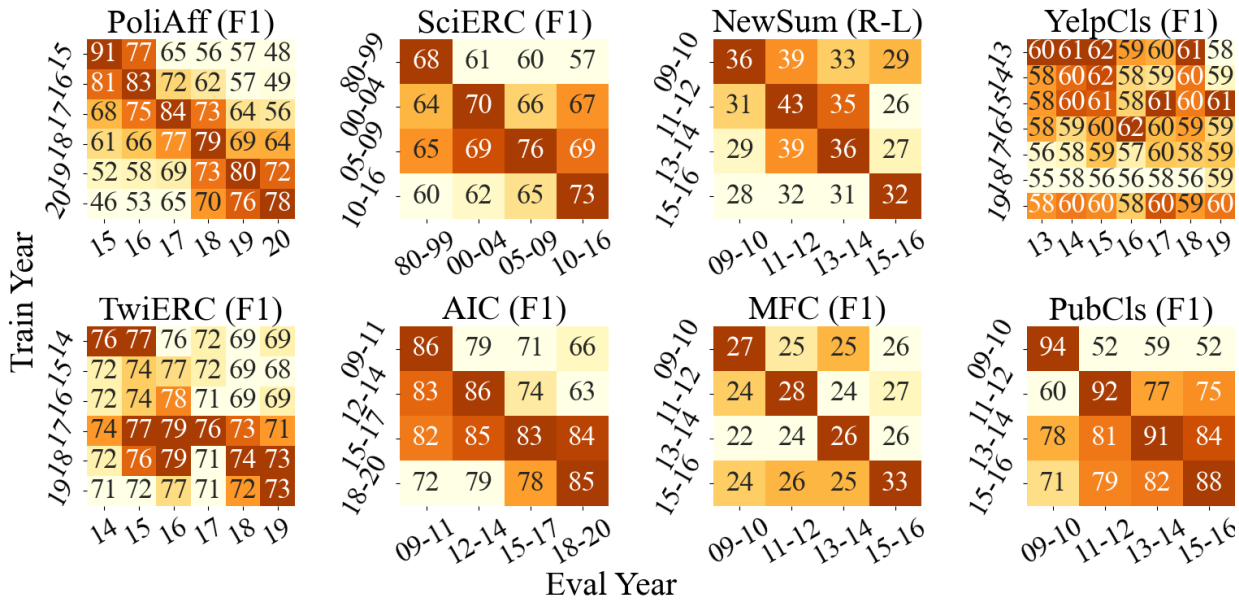

Figure 3.5: Temporal misalignment in finetuning affects task performance (§3.5.1). In all cases, higher scores are better. The heatmap is shaded per column, i.e., the darkest shade of orange in a cell means the cell has the highest score in that column. Mismatch between the the training and evaluation data can result in massive performance drop; degree varies by task. For example, YELPCLS, MFC, and TWIERC show minimal degradation. In contrast, POLIAFF and NEWSUM major deterioration over time.

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 (§3.3.1) is in Table 3.2 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.

Temporal misalignment degrades task performance substantially. Fig. 3.5, 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).

(Q1) Temporal misalignment has a measurable effect on most tasks. Half of our 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.

Domain	Task (metric)	TD	r
Twitter	POLIAFF (F1)	7.72	0.98
	TWIERC (F1)	0.96	0.74
Science	SCIERC (F1)	0.67	0.93
	AIC (F1)	1.79	0.93
News	PUBCLS (F1)	5.46	0.85
	NEWSUM (Rouge-L)	1.38	0.91
	MFC (F1)	0.98	0.86
Reviews	YELPCLS (F1)	0.26	0.30

Table 3.2: Finetuned models' temporal degradation summary scores (TD; §3.3.1; details in Figure **??**). 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. We also show the correlation coefficient, r, that measures the strength of a linear relationship (0 meaning no correlation, 1 being perfectly correlated). In all cases but Yelp, the degree of degradation has a moderate correlation with the distance between the training and evaluation years ($r > 0.5$, $p < 0.05$). We use the Wald test with the null hypothesis that the slope is 0.

Moreover, 1 point of difference can be meaningful, especially for the summarization task where we measure Rouge-L. According to the leaderboard,[10] 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.

(Q1) Performance loss from temporal misalignment occurs in both directions. Another observation in Fig. 3.5 is that degradation happens in both 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. Weak performance in older texts has been noted in NLP for historical documents [Yang and Eisenstein, 2016; Han and Eisenstein, 2019]. However, our findings indicate deterioration can occur sooner—just a few years rather than decades or centuries.

(Q2) 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,

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

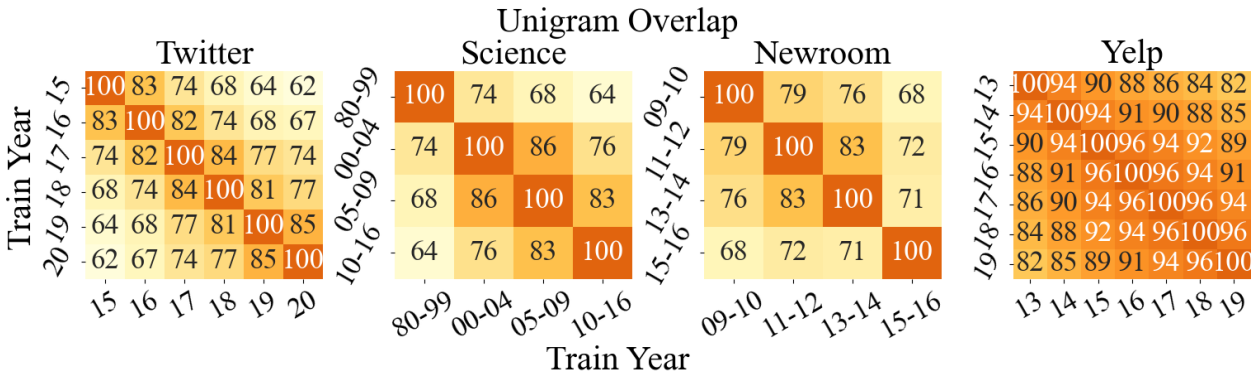

Figure 3.6: Vocabulary overlap between time periods, over a subset of our tasks' datasets. Each cell shows the % overlap between the vocabularies of two time periods.

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.

3.5.2 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?[11] We use a similar procedure to the one Gururangan et al. [2020b] 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. 3.6 visualizes the overlap measurement. Table 3.3 shows the correlation between model performance and the word overlap.

Temporal Domain Adaptation Researchers have studied the broader problem of distributional shift [Shimodaira, 2000; Zhang et al., 2013]. The NLP community has historically tackled these problems via

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

Domain	Task (metric)	Pearson's r
Twitter	POLIAFF (F_1)	0.84
	TWIERC (F_1)	0.51
Science	SCIERC (F_1)	0.72
	AIC (F_1)	0.79
News	PUBCLS (F_1)	0.65
	NEWSUM (Rouge-L)	0.72
	MFC (F_1)	0.80
Reviews	YELPCLS (F_1)	0.14

Table 3.3: Pearson r correlation coeffecients between the word overlap and performance of each task.

domain adaptation [Jiang and Zhai, 2007; Daumé III, 2007; Gururangan et al., 2020b]. Taking inspiration from these approaches, we next apply DAPT to GPT2, treating each time period as a domain: for each time period, 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. 3.7.

(Q3) Domains are a major driver of temporal misalignment in LMs. Consistent with Lazaridou et al. [2021], Fig. 3.7 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, 2018a; 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 §3.2, continued pretraining 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

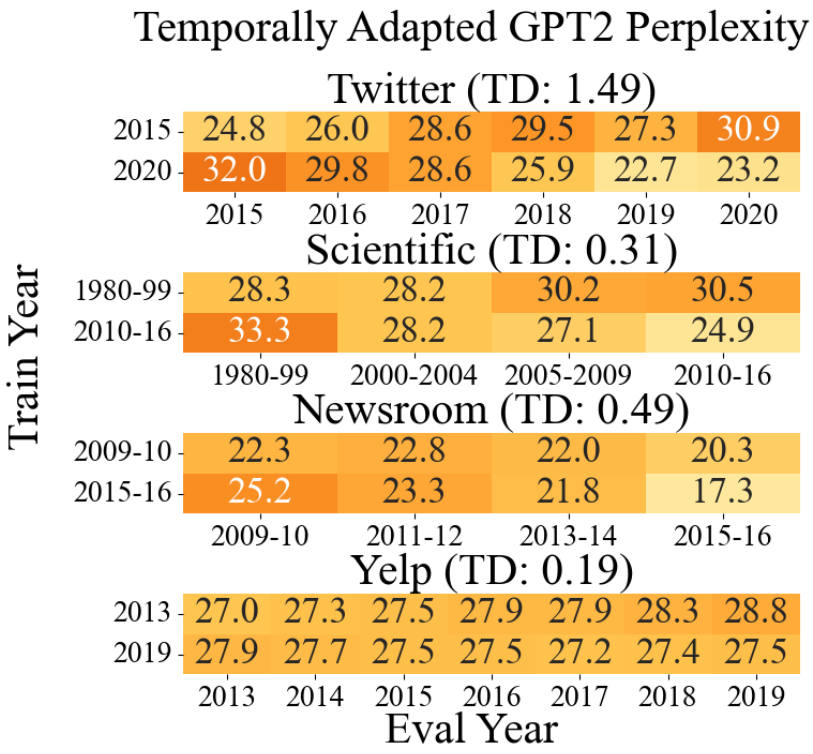

Figure 3.7: Perplexity of GPT2 after adaptive pretraining on temporally-selected data in different domains (lower is better). The TD score (in parentheses) estimates the expected perplexity rise (i.e., degradation) for every time period of misalignment between evaluation and training times. Degradation follows the expected pattern, but the magnitude varies by domain.

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.

(Q4) Temporal adaptation does not overcome degradation from temporally misaligned labeled data. In Table 3.4, 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 sub-table of in Table 3.4: the top-left and bottom-right quadrants (fine-tuning on time-stamp that is used for evaluation) generally lead to higher scores.

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
Scienctific (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			ɔefault → 1980-199	73.2	66.4
		Default → 2018-2020	88.4	**86.0**			ɔefault → 2010-201	**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			ɔefault → 1980-199	63.4	75.0
		Default → 2018-2020	86.8	79.4			ɔefault → 2010-201	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			efault → 2009-201	**95.4**	54.0
		Default → 2015-2016	29.8	30.0			ɔefault → 2015-201	**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**			efault → 2009-201	80.4	**90.7**
		Default → 2015-2016	**32.7**	**41.9**			ɔefault → 2015-201	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 3.4: Combination of temporal adaptation and finetuning (§3.5.2) 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 indicates 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).

3.6 Discussion and Limitations

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. Time periods should be selected 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 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].

Extensive work has been done on modeling and detecting lexical semantic change, or how words evolve in meaning [Hamilton et al., 2016; Rudolph and Blei, 2018; Gonen et al., 2020]. Techniques and intuition from this body of work may be useful in finding solutions to mitigate degradation due to misalignment. We believe that this phenomenon is an important aspect of temporal misalignment, but leave disentangling semantic shifts from other, perhaps task-related factors, for future work.

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 LMs [Jin et al., 2021] or learning disparate tasks [de Masson d'Autume et al., 2019; Huang et al., 2021]. Future work may investigate continual learning algorithms for tasks that change over time.

Our results showed that straightforward domain adaptation was unable to mitigate the effects of temporal misalignment. Recent work in language modeling has elevated the importance of domains by using a mixture

of domains [Gururangan et al., 2021] or giving domains a hierarchical structure [Chronopoulou et al., 2021]. More sophisticated approach to domains, in line with these works, could lead to temporally robust models.

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].

3.7 Summary

Changes in language use over time, and how language relates to other quantities of interest in NLP applications, have clear effects on the performance of those applications. We have explored how temporal misalignment between evaluation data and 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 across tasks, more so than found in previous studies [Röttger and Pierrehumbert, 2021]. These findings support our intuition: each task has unique characteristics based on the domain, community, or task definition and are affected by temporal misalignment at differing rates. Consequently, practitioners should consider the task or community they are modeling when making design decisions.

Moreover, our results motivate continued study of temporal misalignment across applications of NLP, its consideration in benchmark evaluations,[12] 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.

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

Chapter 4

Measuring Persuasive Skill Over time

4.1 Introduction

Natural language interactions between members of a community tend to shift over time, adopting new phrases or centering discussion on new ideas. As discussed in Chapter 3, temporal misalignment between test and training sets can lead to severe performance degradation over time. While these findings place more responsibility on practitioners, we also find that using knowledge of social interaction in communities can motivate novel work. In this case, we focus on integrating change over time in our models. Prior work has shown that members learn how to integrate into the community more effectively by adopting new jargon [Danescu-Niculescu-Mizil et al., 2013] or changing behaviors [Jurgens et al., 2017]. Likewise, researchers have modeled user knowledge and improvement in language acquisition [Settles et al., 2020] or in online reviews [McAuley and Leskovec, 2013].

In this chapter, we explore skill estimation of individual members of a community of debaters and how their language evolves with skill over time via natural language interactions. We find that online debate communities offer an opportunity to investigate *persuasive skill*. These communities feature users who participate in multiple debates over their period of engagement. Such debates involve two parties who willingly and formally present divergent opinions before an audience. Unlike other media of persuasion, such as letters to politicians, there is a clear signal, a win or loss, indicating whether or not a debater was successful against the adversary. This work aims to quantify the skill level of each debater in an online community and

also investigates which factors contribute to expertise.

While persuasion has generated interest in the natural language processing community, most researchers have not tried to quantify the persuasiveness of a particular speaker. Instead, they estimate how persuasive a *text* is, using linguistic features such as the author's choice of wording or how they interact with the audience [Tan et al., 2014, 2016; Althoff et al., 2014; Danescu-Niculescu-Mizil et al., 2012]. Previous research has also established that debaters' content and interactions both contribute to the success of the persuader [Tan et al., 2016; Zhang et al., 2016; Wang et al., 2017]. These works have found textual factors that contribute to a debater's success, but they do not emphasize the role of the individual debater.

There has been some recent work on studying individual debaters. Durmus and Cardie [2019] analyze users and find that a user's success and improvement depend on their social network features. We take another approach by estimating each user's skill level in each debate they participate in by considering their debate history. These estimates reveal features correlated with skill and the importance of particular debates over time.

We extend the Elo [1978] rating system, a model designed for rating players in two-player games, for the debate setting using linguistic features. We construct a family of models based on Elo and decompose them into two design choices that align with two questions we hope to answer: 1) the features we use and 2) how we might choose to aggregate those features from past debates.

To validate our skill estimates, we introduce a **forecasting** task (§4.4). Previous work predicted the winner of a debate using the text of the current debate. In contrast, we aim to predict winners using our skill estimates *before* the debate (ignoring the current debate's content). This design ensures that we are modeling skill of the debater, as inferred from past performance, not the idiosyncrasies of a particular debate.

We also investigate the predictive power of our estimates through an analysis of the results (§4.6). We show that our full model outperforms the baseline Elo model, approaching the accuracy of an oracle that *does* use the text of the current debate. Moreover, we find that not all past debates are equally useful for prediction: more recent debates are more indicative of the user's current level of expertise. This adds support to our conjecture that individual debaters tend to improve through the course of their time debating. Finally, we track the linguistic tendencies of each debater over the course of their debating history. We show that several features correlated with high skill, such as length of their turns, increase over time for the best debaters, but

stay static for those with less skill. These findings give further evidence that debaters improve over time.

This chapter contains work originally published in [Luu et al., 2019].

4.2 Data

We use both debate and user skill data from the *Debate.org* dataset introduced by Durmus and Cardie [2018]. Any registered user on the website can initiate debates with others or vote on debates conducted by others.

4.2.1 Mechanism of `Debate.org`

Registered users can create a debate under a topic of their choosing. The person initiating the debate, called the **instigator**, fixes the debate's number of rounds (2–10) and chooses the category (e.g., politics, economics, or music) at the start of the debate. The instigator then presents an opening statement in the first round and waits for another user, the **contender**, to accept the debate and write another opening statement to complete the first round.[1] We define a debater's **role** as being either the instigator or contender.

To determine the debate winner, other *Debate.org* users vote after the debate ends. In this phase, voters mark who they thought performed better in each of seven categories (see Figure 4.1).[2] This phase can last between 3 days and 6 months, depending on the instigator's choice at the debate's creation. After this period, the debater who received the most points wins the debate. We record in our dataset the textual information, participants, and voting records for each debate.

4.2.2 Definition of Winning

The *Debate.org* voting system lets us define a "win" in various ways (see Figure 4.1). In this study, we would like to model how convincing each debater is. One approach might be based on Oxford-style debates like the IQ2 dataset from Zhang et al. [2016], in which scores are based on the number of audience members who changed their minds as a result of the debate. We found here that the majority of voters do not deviate from their stances before the debate, and most voters tend to vote for who they already agree with. Therefore,

[1] While many debaters use their first round to make an opening statement, some use it only to propose and accept debates. If the first turn in an n-round debate is under 250 words, we merge each debater's first two turns and treat the debate as an $(n-1)$-round debate.

[2] Another system of voting lets voters choose who they thought performed better over the entire debate. While these appear in the dataset, we do not use them in this paper.

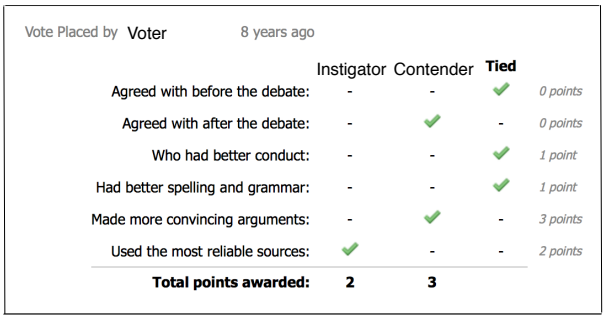

Figure 4.1: An example of the *Debate.org* voting system.

	#Users	#Debates
Completed Debates	42,424	77,595
Completed & Convincing	21,753	29,209
Full Filtered	1,284	4,486

Table 4.1: Description of the *debate.org* dataset from Durmus and Cardie [2018] and the filtered datasets. **Full Filtered** is a subset of **Completed & Convincing** that requires that participants of each debate have engaged in five or more debates. We use **Full Filtered** for the remainder of our analysis.

we count the number of times each debater was rated as more convincing to a voter *despite* presenting a viewpoint that the voter disagreed with before the debate. The debater with the higher count of such votes is considered the "winner" in the remainder of this paper.

4.2.3 Dataset Statistics

From an unfiltered set of 77,595 debates, we remove all debates where a user forfeits, that lack a winner (§4.2.2), or that do not have a typical voting style with seven categories, leaving 29,209 completed debates.

We record the number of debates that each user completes (see Figure 4.2, left). We find that the quantity of debates per user follows a heavy-tailed distribution where most users do not participate in more than one debate. For the remainder of the work, we focus on the 1,284 (out of 42,424 total users) who have completed five or more of the debates described above. This leaves us with 4,486 debates where the participants have completed at least five debates. See Table 4.1.

We also record the number of votes that each debate attracted (see Figure 4.2, right). This number also follows a heavy-tailed distribution.

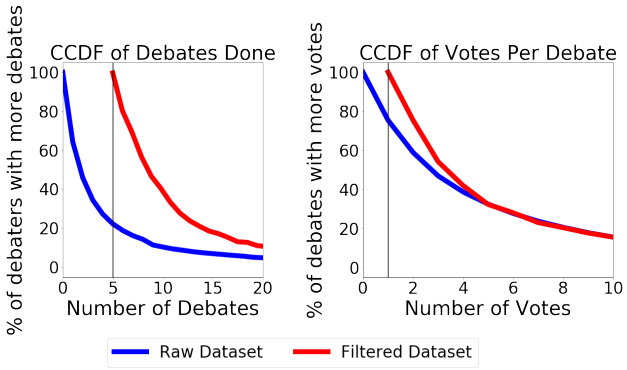

Figure 4.2: Complementary cumulative distribution functions (1 − CDF) for the total number of debates a user finished. The blue line tracks only debaters who engaged in and successfully concluded at least one debate, while the red line tracks users who have finished at least five debates in the filtered dataset. The right plot similarly shows the complementary cumulative distribution for the number of votes given per debate for all debates and the filtered dataset.

4.3 Expertise Estimation

In order to explore debater expertise and discover what contributes to a user's expertise over their time on *Debate.org*, we begin with a conventional approach to skill estimation, the Elo rating system, which serves as both a baseline and the basis for our final model.

4.3.1 Elo Model

Elo originated as a ranking system for chess players; it has been adapted to other domains, such as video games[Herbrich et al., 2007]. It is one of the standard methods to rate players of a two-player, zero-sum game [Elo, 1978].[3] Elo assigns positive integer-valued scores, typically below 3,000, with higher values interpreted as "more skill." The difference in the scores between two debaters under a logistic model is used as an estimate of the probability each debater will win. For example, consider a debate between A and B. A has an Elo rating of $R_A = 1900$, and B has an Elo rating of $R_B = 2000$. Using the Elo rating system, p_A, the probability that A wins is[4]

$$p_A = \frac{1}{1 + 10^{0.0025(R_B - R_A)}} = \frac{1}{1 + 10^{0.25}} \approx 0.36 \tag{4.1}$$

[3]The Elo model is a special case of the Bradley-Terry model [Bradley and Terry, 1952].

[4]The base of the exponent, 10, and the multiplicative factor on the difference $R_A - R_B$, 0.0025, are typically used in chess.

Ratings are updated after every debate, with the winner (equal to A or B) gaining (and the loser losing) $\Delta = 32(1 - p_W)$ points. (32 is an arbitrary scalar; we follow non-master chess in selecting this value.)

Note that the magnitude of the change corresponds to how unlikely the outcome was. While the Elo ratings traditionally take only a win or loss as input, there have been adjustments to account for the magnitude of victory. One such method would be to use the score difference between the two players to adjust the Elo gain [Silver, 2015]. If we let S_A and S_B be scores for A and B respectively the modified gain Δ' is

$$\Delta' = \log(|S_A - S_B| + 1) \times \Delta \tag{4.2}$$

Under this model, we represent a user's history and skill level as a single scalar, i.e., their Elo rating. The Elo system ignores all other features, which include the style a debater uses in the debates and the content of their argument. We therefore view this model as a baseline and extend it.

4.3.2 Do Debaters Get Better Over Time?

Our initial data analysis uses the Elo model to investigate whether users improve over time at all in the first place. An increase in Elo score can be seen as the rating system merely becoming more accurate with another sample or an actual increase in the player's skill. We wish to show that there is no fixed rating for a user, but rather that the rating is a moving target. A counterhypothesis is then that the users' skill levels do not change despite their activity on *Debate.org*. If true, then we would expect their final Elo rating to be also indicative of their skill level at the beginning of their *Debate.org* activity.

To test this hypothesis, we first define an **upset** as a debate where $p_W < \tau$, that is, where the winner of the debate was estimated (under Elo) to win with low probability (set using threshold τ). If users tend to have static skill levels, then participating in many debates does not affect debaters' skill, but simply provides more samples for measuring their skill. It follows that we would expect upsets to occur at the same rate early and late in each one's career given debaters' static skill. In this analysis, we calculate p_A and p_B for each debate using A's and B's *final* Elo ratings, which we take to be the most accurate estimate of their (presumed static) skill levels.[5]

[5] In a forecasting analysis like the one in §4.4, this would be inappropriate, as it uses "future" information to define upsets. This notion of upset is not used in the forecasting task.

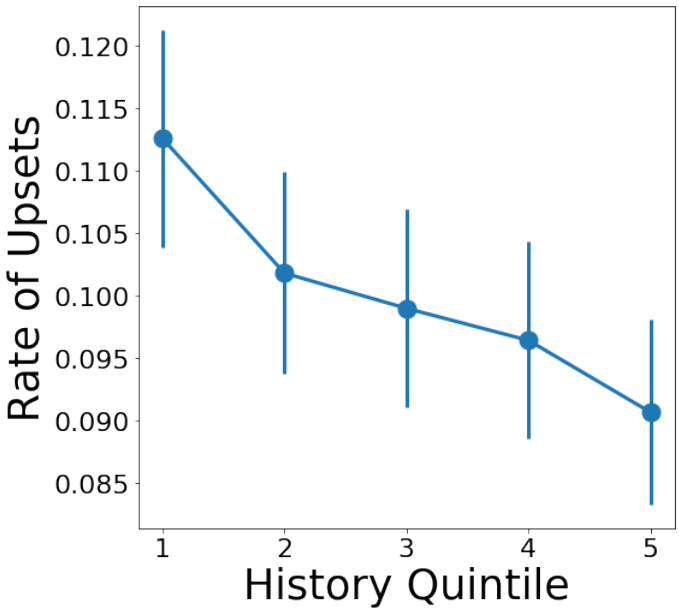

Figure 4.3: Upset rates (aggregated across users) across history quintiles, $\tau = 0.45$. The error bars represent the 95% confidence intervals.

To operationalize "early" and "late," we divide a user's debates into quintiles by time, comparing the upset rate in different quintiles. In this analysis, we only consider users with at least ten debates ($N = 4420$).[6]

We see a downward trend in the upset rate (Figure 4.3). In particular, the first and last quintiles show a statistically significant difference under a paired t-test ($p < .001$), meaning that a user's final Elo score is not a good measure of skill at earlier times. We take this finding as suggestive that users of *Debate.org* adapt as they participate in more debates.

4.4 Predicting Expertise using Earlier Debates

Our aim is to estimate a debater A's persuasive ability after observing a series of debates they participated in (denoted $d_1^A, \ldots, d_{t-1}^A$ if we are estimating ability just before the tth debate). We wish to take into account the content of those debates (not merely their outcomes, as in Elo), so as to understand what factor reveals a debater's skill levels. Drawing inspiration from Elo's interpretation as a score that can be used to predict each debater's probability of winning (Eq. 4.1), we formulate a prediction task: estimate p_A for A's tth debate, given $d_1^A, \ldots, d_{t-1}^A$ and also the opponent's debate history (which might be of a different length).

[6]Unlike in the rest of our analysis, we do not require the opponents of the users to have also participated in at least ten debates.

By observing debate outcomes alongside the two participants' histories, we can estimate the parameters of such a probability model. Elo provides a baseline; rather than opaque scores associated with individual users at different times, we seek to *explain* the probability of winning through linguistic features of past debate content.

Unlike previous work [Zhang et al., 2016; Potash and Rumshisky, 2017; Tan et al., 2018], we do not use the content of the *current* debate (d_t^A) to predict its outcome; rather, we forecast the outcome of the debate as if the debate has not yet occurred. In the remainder of this section, we discuss features of past debates and ways of aggregating them.

4.4.1 Incorporating a Linguistic Profile into Elo

Elo scores are based entirely on wins and losses; they ignore debate content. We seek to incorporate content into expertise estimation by using linguistic features. If we modify the exponential base in Equation 4.1 from 10 to e, we can view Elo probabilities (e.g., p_A) as the output of a logistic regression model with one feature (the score difference, $R_A - R_B$) whose weight is 0.0025; that is, $p_A = \sigma(0.0025 \cdot (R_A - R_B))$. It is straightforward to incorporate more features, letting

$$p_A = \sigma\left(\mathbf{w} \cdot (\boldsymbol{R}_A - \boldsymbol{R}_B)\right) \tag{4.3}$$

where $\mathbf{w}$ is a vector of weights and $\boldsymbol{R}_U$ is user U's "profile," a vector of features derived from past debates. In this work, the linguistic profiles are designed based on extant theory about the linguistic markers of persuasion, and the vectors are *weighted averages* of features derived from earlier debates.

4.4.2 Features

We select features discussed in prior work as the basis for our linguistic profiles [Tan et al., 2016, 2018; Zhang et al., 2016]. We extract these measurements from each of the user's debates. For a given debate and user, we calculate these values over the rounds written by the user. For example, if we were interested in a debate by a user as the instigator, we would only calculate features from the instigator rounds of that debate (since the contender rounds of that debate were written by their opponent). Table 4.2 shows the full list of features.

Hedging with fightin' words. We introduce one novel feature for our work: hedging with fightin' words. "Fightin' words" refer to words found using a method, introduced by Monroe et al. [2008], which seeks to identify words (or phrases) most strongly associated with one side or another in a debate or other partisan discourse.[7] We are interested in situations where debaters evoke fightin' words (their own, or their opponents') with an element of uncertainty or doubt. We use each debater's top 20 fightin' words (unigrams or bigrams) as features, following Zhang et al., 2016, who found this feature useful in predicting winners of Oxford-style televised debates. We also count co-occurences of fightin' words with hedge phrases like "it could be possible that" or "it seems that." An example of this conjoined feature is found in the utterance "Could you give evidence that **supports** the idea that married couples are **more likely** to be committed to [other tasks]?" where hedge phrases are emboldened and brackets denote fightin' words (which are selected separately within each debate). We use a list of hedging cues curated by Tan et al. [2016] and derived from Hyland [1996] and Hanauer et al. [2012]. The conjoined feature is the count of the user's sentences in a debate where a fightin' word cooccurs with a hedge phrase in a sentence.

4.4.3 Aggregating Earlier Debates

Since we consider the full history of a debater when estimating their skill level, we opt to aggregate the textual features over each debate. We do so by taking a weighted sum of the feature vectors of the previous debates. We consider four weighting schemes, none of which have free parameters, to preserve interpretability. Let f be any one of the features in the linguistic profile, a function from a single debate to a scalar.

- **Exponential growth:** the most recent debates are most indicative of skill, $\sum_{i=1}^{t-1} \frac{f(d_i^A)}{2^{t-i}}$. We take this to be the most intuitive choice, experimentally comparing against the alternatives below to confirm this intuition.
- **Simple average:** each earlier debate's feature vector is weighted equally, $\frac{1}{t-1} \sum_{i=1}^{t-1} f(d_i^A)$.
- **Exponential decay:** the first debates are most indicative of skill, $\sum_{i=1}^{t-1} \frac{f(d_i^A)}{2^i}$.
- **Last only:** only the single most recent debate matters, $f(d_{t-1}^A)$ (an extreme version of "exponential growth").

[7] The method estimates log-odds of words given a side, with Dirichlet smoothing, and returns the words with the highest log-odds for each side.

Feature	Description	
Elo Score	Traditional Elo score calculated and updated. Updated traditionally, not averaged as in §4.4.3.	n/a
Length	Number of words this user uttered in the debate.	↑↑↑
Part of speech	Count of each noun, verb, adjective, preposition, adverb, or pronoun from the participant in the entire debate.	Noun:(↑↑↑) Adj:(↑↑↑)
Flesch reading ease	Measure of readability given the number of sentences in a document and the number of words in each sentence [Kincaid et al., 1975].	↑↑
Emotional words	Cues that indicate a positive or negative emotion [Tausczik and Pennebaker, 2010].	Pos:(↑↑↑) Neg:(↑↑↑)
Links	Links to external websites outside of *debate.org*. This feature operationalizes the number of sources a debater used.	
Questions	The number of questions the user asked in the debate.	↓↓↓
Quotations	The number of quotations the user included in the debate.	
Hedging	The number of phrases that soften a statement by adding uncertainty [Hyland, 1996; Hanauer et al., 2012].	
Fightin' words	The number of instances of words most strongly associated with either debater [Monroe et al., 2008].	↑↑↑
H∧FW	The number of cooccurences of hedging and fightin' words, described in §4.4.2.	↑↑

Table 4.2: Debate-level features used in estimating skill levels. Aside from Elo, the features are a part of the user's linguistic profile. The third column represents statistical significance levels in comparing winners and losers' features (independently) with Bonferroni correction: ↑ is $p < 0.05$, ↑↑ is $p < 0.01$, ↑↑↑ is $p < 0.001$.

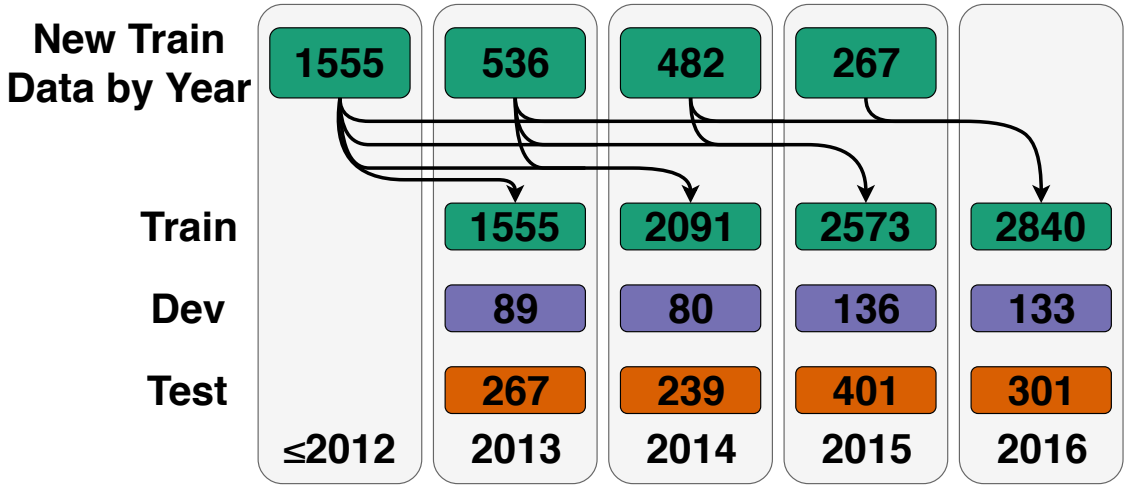

Figure 4.4: We split data into training/development/test based on year. This chart shows the number of debates in each subset of the data. We note that, for training, we use all the training debates from previous years (e.g., if we were to test on 2015, we would train using the training splits from 2012, 2013, and 2014). Each instance in this figure corresponds to a debater.

In each variation of our method, some or all of the linguistic profile features are aggregated (using one of the four weighted averages), then applied to predict debate outcomes through logistic regression. We note that if our sole aim were to maximize predictive accuracy, we might explore much richer linguistic profiles, perhaps learning word embeddings for the task and combining them using neural networks and enabling interactions between a debate's two participants' profiles.[8] In this study, we seek to estimate skill but also to understand it, so our focus remains on linear models.

4.5 Experimental Setup

We validate our skill models with a binary classification task, specifically forecasting which of two debaters will win a debate (without looking at the content of the debate). Here we remove any debates where someone forfeits or there is no winner. We then considered debates where each debater has completed at least five of the remaining debates. As discussed in §4.2.1, the winner is taken to be the debater receiving the most "more convincing argument" votes from observers who did not initially agree with them. We create four training/evaluation splits of the data, using debates from 2013, 2014, 2015, and 2016 as evaluation sets (i.e.,

[8] Indeed, in preliminary experiments we *did* explore using a recurrent neural network instead of a fixed weighted average, but it did not show any benefit, perhaps owing to the relatively small size of our dataset.

development and test) and debates prior to the evaluation year for training. Figure 4.4 shows the number of debates in each split. We note that our training sets are cumulative. For example, if we were to test on 2015 data, we would use the 2012, 2013, and 2014 training as training data.

Since we do not test on 2012 data or train on 2016 data due to the low number of debates before 2012 and after 2016, we treat the whole of 2012 as training data and 2016 as development and test. We report the accuracy for each run.

We compare several predictors:[9]

- **Full model**: our model with all features, as described in §4.4, and (except where otherwise stated) the exponential growth weighting. This model combines linguistic profiles from earlier debates with a conventional Elo score.

- **Full model with point difference**: our full model as described above, but we scale the Elo gain by the point difference as described in Equation 4.2.

- **Linguistic profile only**: our model with exponential growth weighting (except where otherwise stated), but ablating the Elo feature. This model is most similar to those found in prior literature [Zhang et al., 2016; Tan et al., 2018; Wang et al., 2017].

- **Elo**: the prediction is based solely on the Elo score calculated just before the debate. This is equivalent to ablating the linguistic profiles from our model.

- **Final Elo oracle**: the prediction is based solely on the two debaters' *final* Elo scores (i.e., using all debates from the past, present, and future).

- **Current debate text oracle**: a model that uses the linguistic profile derived just from the current debate. While this model is most similar to previous work, it is not a fair estimate of skill (since it ignores past performance). We therefore view it as another oracle.

- **Majority choice**: a baseline that always predicts that the contender will win.[10]

[9] We use ℓ_2 regularization in our models with the linguistic profile features.

[10] In this dataset, contenders win nearly 59% of the time, a fact frequently discussed in the *Debate.org* community; see *http://ddo.wikia.com/wiki/Contender_Advantage*. The contender advantage is sometimes attribted to having the "final say," or to the fact that contenders choose the instigators they wish to debate.

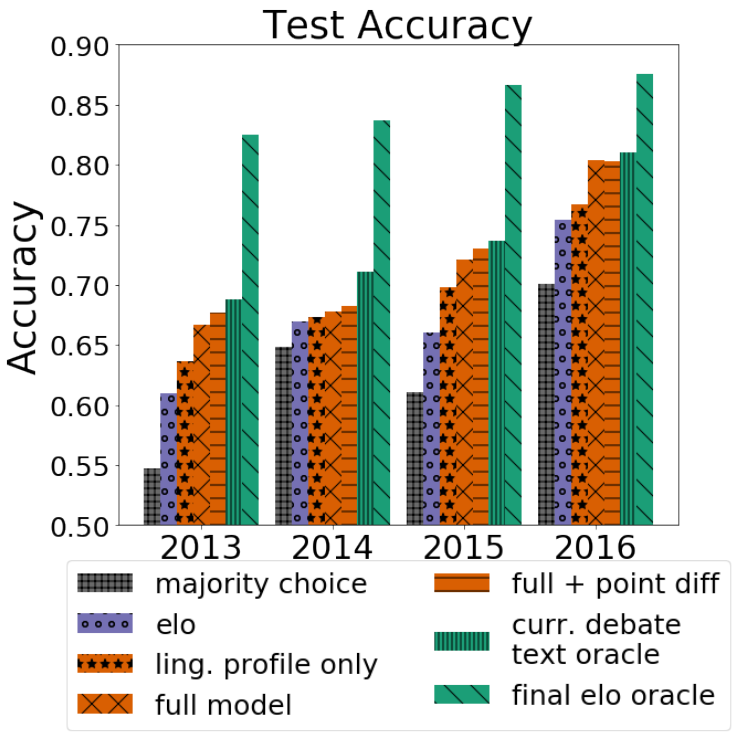

Figure 4.5: Our results for the prediction task. Our full model outperforms the Elo baseline and approaches the current debate text oracle.

4.6 Results

In this section, we first show that the expertise of a debater can be better estimated with the linguistic profile, and then analyze the contribution of different components. We further examine the robustness of our results by controlling for additional variables.

4.6.1 Prediction Performance

We first present our results with what we consider our best models, i.e., our full model (with point differences), which consists of all features and uses the exponentially growing weight.

Importance of linguistic features. We see from Figure 4.5 that the full model outperforms the standard Elo baseline. The gap between the two models suggests that the addition of the linguistic profile contributes to the performance of the model and therefore plays a useful role in skill estimation. Moreover, the linguistic profile only model shows that the linguistic profile features are not only useful, but have at least as strong

predictive power as Elo alone. By only using the linguistic features aggregated over the course of a debater's history, *without knowing winning records*, we can forecast at least as well as the Elo baseline.

Importance of multiple debates. We also note that our full model only performs slightly worse than the current debate text oracle despite the current debate text model directly observing the content of the debate. This result implies that using information from only previous debates has at least similar predictive strength to information from the debate at hand. Moreover, the large gap between the final Elo and current debate text oracles implies that a user's skill is evidenced by more than the content of a single debate. These results further demonstrate the importance of accounting for debaters' prior history.

Magnitude of victory might matter. Our full model with the point difference scaled Elo gain does roughly as well or slightly better than our normal full model. As the focus of this paper is on incorporating linguistic profiles, we use the **full model without the point difference scaling** for analyses in the rest of the paper.

4.6.2 Feature Ablations

We inspect the contribution of each feature by removing each one from the model. Then, for each feature, we perform a bootstrap test over the last year of data (trained on 2012–2015 data; tested on 2016). At each iteration, we sample 1000 training examples to train on, but fix the test set across iterations. We then train our full model alongside several other models, each with a feature ablated, on the sample. We track the drop in performance between our full model and each of our other models. We record the average performance over 100 iterations for comparison.

From Figure 4.6, we find that removing Elo results in the most severe drop in performance (5.8%). Ablating part-of-speech, negative emotion, and length from our model had a moderate effect on performance. Surprisingly, we find that, although the H∧FW feature is the overlap between hedge cues and fightin' words, the latter two features individually contribute very little to the performance of our model compared to H∧FW.

4.6.3 Combining Prior Debate Features

As described in §4.4.3, we explore several ways of combining features over the past debates. By inspecting how these different aggregation functions might differ in performance, we hope to find out whether or not

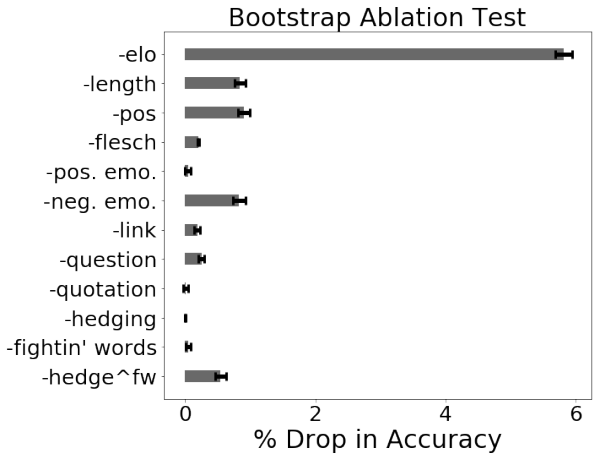

Figure 4.6: Our bootstrap on the feature ablations. We record the average drop in performance across 100 iterations and tested on the 2016 test set. Higher means a larger drop in performance.

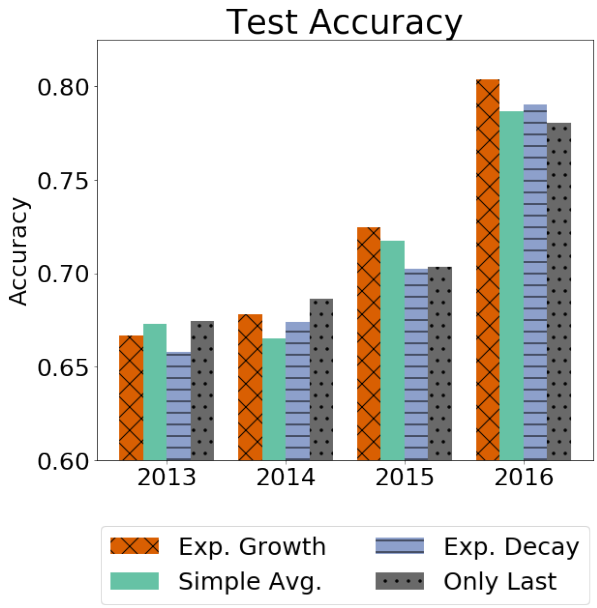

Figure 4.7: Comparison in performance for the four ways we aggregate features over time.

some debates are more important than others, if recency matters at all, and if some history is important at all. We do so by 1) giving the last debates more weight (**exponential growth**), 2) giving all weights equal weight (**simple average**), 3) giving the first debates more weight (**exponential decay**), and 4) giving all the weight to the last debate (**last debate only**).

More debates help. When using only the last debate's features and ignoring all previous debates, our performance is initially very good in the in years 2013 and 2014 (when our training sets are smaller and histories shorter). However, as debate histories become richer in the later years, last-only's performance drops in comparison to that of the growing weight aggregation. These results match our intuition: with more experienced users, considering more debates gives us a better gauge of how a debater performs and a debater's more recent debates give a better snapshot of the user's current skill level.

Later debates matter more. As hinted in our last-only results, giving the last debates more weight does best in all four years. In light of its performance compared to the simple mean and decaying weight settings, our results imply that not all debates contribute equally to a debater's skill under our model. Indeed, our results show that the most recent debates are also the most important for estimating a debater's skill rating.

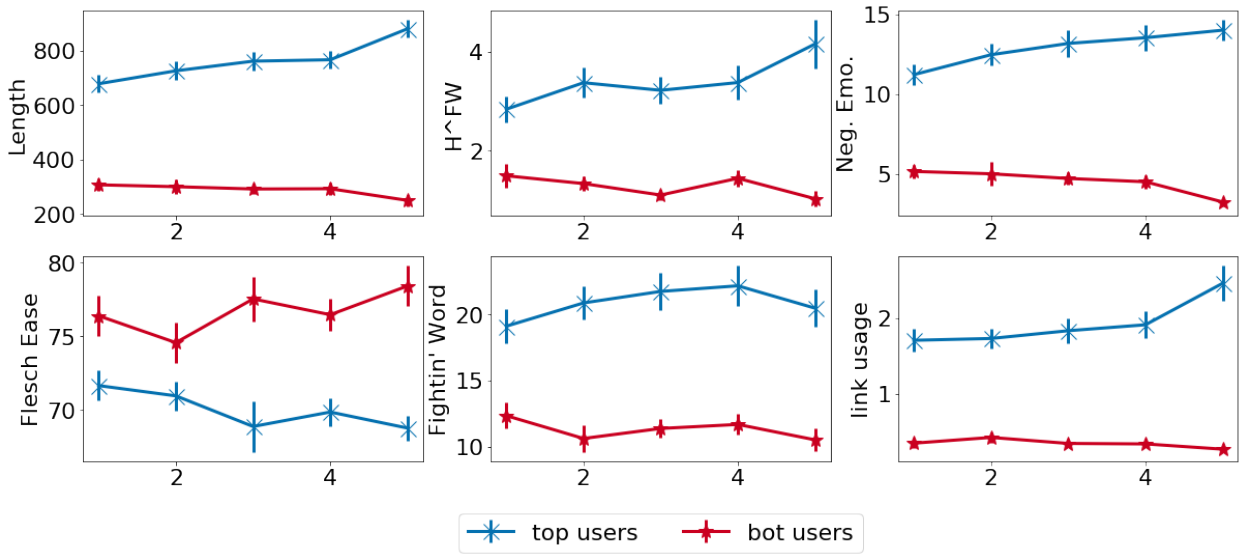

Figure 4.8: Feature measurements averaged across each history quintile. We see that there is a general trend for those who eventually become the best debaters to improve on these measures while the bottom users stagnate. The error bars represent the 95% confidence intervals.

4.7 Language Change over Time: Experts Improve and the Worst Stagnate

Our findings from §4.6.3 imply that users tend to experience some change over time and that these changes are helpful for forecasting who in a debate is more likely to win. In this section, we explore whether debaters' linguistic tendencies change over time by tracking their use of features over the course of their debate history.

We examine how language use changes over time for the best and worst users. To do so, we divide each user's debate history into quintiles. For each quintile, we average the same features we use in the linguistic profile that we use in our model (Table 4.2). We take the top 100 and bottom 100 users ranked by our model to see how the trajectories change over time.

Figure 4.8 shows that best and worst users have different linguistic preferences even from the beginning of their debating activities. The best debaters have a higher feature count in every case except for the Flesch reading ease score. However, the best users do seem to improve over time in length, H∧FW, use of emotional cues, and link use, which mostly correlates with our feature ablation in Figure 4.6 ($p < 0.05$ after Bonferroni correction). In contrast, the worst users do not seem to experience any significant change over time except for the length of their rounds and negative emotional cue use. In those cases, the worst users seem to worsen over time.

4.8 Related Work

In addition to the most relevant studies mentioned so far, our work is related to three broad areas: skill estimation, argumentation mining, and studies of online debates.

Skill estimation. Ranking player strength has been studied extensively for sports and for online matchmaking in games such as *Halo* [Herbrich et al., 2007]. The Bradley-Terry models (of which Elo is an example) serve as a basis for much of the research in learning from pairwise comparisons [Bradley and Terry, 1952; Elo, 1978]. Another rating system used for online matchmaking is Microsoft's Trueskill™ rating system [Herbrich et al., 2007], which assumes performance is normally distributed. Neural networks have recently been explored [Chen and Joachims, 2016; Menke and Martinez, 2008; Delalleau et al., 2012], incorporating player or other contextual game features from previous games at the cost of interpretability of those features.

Argumentation and persuasion. Past studies have noted the persuasiveness of stylistic effects such as phrasing or linguistic accommodation. For example, Danescu-Niculescu-Mizil et al. [2012] showed that, in a pool of people vying to become an administrator of a website, those who were promoted tended to coordinate more than those who were not. Similarly, other works define and discuss power relations over discussion threads such as emails [Prabhakaran and Rambow, 2014, 2013]. Additionally, Tan et al. [2018] explored how debate quotes are selected by news media. They found that linguistic and interactive factors of an utterance are predictive of whether or not it would be quoted. Prabhakaran et al. [2014] also studied political debates and found that a debater's tendency to switch topics correlates with their public perception. Argumentation has also been studied extensively in student persuasive essays and web discourse [Persing and Ng, 2015; Ong et al., 2014; Song et al., 2014; Stab and Gurevych, 2014; Habernal and Gurevych, 2017; Lippi and Torroni, 2016]. Most relevant to our work on how users improve over time, Zhang et al. [2017a] study how one document may improve over time through annotated revisions. While our work examines users' linguistic change across multiple debates, they focus on how a user improves a single document over multiple revisions.

Online debates. There has also been recent work in characterizing specific arguments in online settings in contrast to our focus on the debaters themselves. For example, Somasundaran and Wiebe [2009], Walker et al. [2012], Qiu et al. [2015], and Sridhar et al. [2015] built systems for identifying the *stances* users take in

online debate forums. Lukin et al. [2017] studied how persuasiveness of arguments depends on personality factors of the audience.

Other researchers have focused on annotation tasks. For example, Park and Cardie [2014] annotated online user comments to identify and classify different propositions. Hidey et al. [2017] annotated comments from the *changemyview* subreddit, a community where participants ask the community to change a view they hold. Likewise, Anand et al. [2011] annotated online blogs with a classification of persuasive tactics. Inspired by Aristotle's three modes of persuasion (ethos, pathos, and logos), their work annotates claims and premises within the comments. Habernal and Gurevych [2016] used crowdsourcing to study what makes an argument more convincing. They paired two similar arguments and asked annotators to indicate the more convincing one. This framework allowed them to study the flaws in the less convincing arguments. The annotations they produced offer a rich understanding of arguments which, though costly, can be useful as future work.

4.9 Summary

In this chapter, we used community interactions to estimate debater skill for individual members of an online debate community. We introduced a method that uses a linguistic profile derived from an individual debater's history of past debates to model their skill level as it changes over time. Using data from *Debate.org*, we formulate a forecasting task around predicting which of two debaters will be most convincing to observers predisposed to be unconvinced. We find that linguistic profiles on their own are similarly predictive to the classic Elo method, which does not parameterize skill according to attributes of a participant or their behavior, but only models wins and losses. Moreover, we show that our findings are robust to topic of debate and frequency of user activity. Further, a model combining linguistic profiles with Elo achieves predictive accuracy nearly on par with an oracle based on the content of the debate itself. A particular feature combining hedging with fightin' words is notably important in our model, and consistent with evidence that debaters improve with practice, more recent debates appear to provide better estimates of skill than earlier ones. To verify our hypothesis that users improve, we explicitly track the feature use of debaters over the course of their debating activity to show that the best users improve while the bottom users stagnate. Our approach sets the stage for future explorations on the role of a user's history profile on their future writings.

Chapter 5

Conclusion

Natural language interactions are widely available in today's world and make up a fair amount of the data that we use. In this thesis, we proposed that NLP advances and social interactions in communities build on each other.We explored two distinct communities, the scientific research community and an online debate forum, and used the natural language interactions within them to understand how communities discuss each other's work and how they develop expertise over time.

In Chapter 2, we focused on the problem of explaining scientific text in relation to another. We introduced a task where, given two documents, we would like to generate a natural language sentence that explains how one document relates to the other and operationalize the task with in-line citation sentences. We train a model based on GPT2, SciGen, for the task and investigate what types of inputs work well between random sentences, introductions, and abstracts. We perform an extensive automated and human evaluation to find that using the introduction and abstract as contexts work well. Moreover, we found that human and automated metrics tell different stories; while SciGen performed worse than an IR baseline on automated metrics, SciGen did much better on human assessments.

We also discussed how SciGen was essentially a static model that only captured a small snapshot of the computer science research community at a particular time and could perform worse as the time between test and training time periods gets larger. Chapter 3 delves deeper into this phenomenon, temporal misalignment, for a variety of downstream tasks and domains. Given the rise of pretrained language models, which also empowered SciGen, we focused on whether or not modern NLP models could generalize well

temporally. We found that these models could degrade severely due to temporal misalignment, but the degree of deterioration depends greatly on the task definition and domain. Even two tasks in the same domain could be affected by temporal misalignment very differently.

Chapter 4 shows that incorporating temporal characteristics of a community in modeling can be useful to study persuasion. We aimed to estimate user persuasive skill for members of an online debate website, *debate.org*. To do so, we cast the problem as a forecasting problem where we predict the outcome of a debate based only on the previous debates and not the content of the current one at hand. We build on the Elo method by augmenting it with a linguistic profile, features based on a user's past debates. Our results indicate that a model combining our linguistic profile with Elo outperforms Elo by itself and even approaches an oracle model. Finally, we track the feature use of the best and worst users and find that the top users over time. We find that the best users improve while the worst users stagnate or get worse.

We have established in this thesis the utility of modeling natural language interactions to uncover social phenomena and knowledge of social interactions to motivate new NLP tasks. Throughout the thesis, we have emphasized the importance of recognizing nuances in the community, tasks, and domains studied, especially with respect to time. Each task studied in this work had unique characteristics that we took advantage of. In light of the strength of recent NLP models empowered by pretrained language models, we encourage practitioners to consider their final applications carefully and pay attention to unique characteristics in their data.

Bibliography

Tim Althoff, Cristian Danescu-Niculescu-Mizil, and Dan Jurafsky. 2014. How to ask for a favor: A case study on the success of altruistic requests. In *Proceedings of ICWSM*.

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*.

Pranav Anand, Joseph King, Jordan Boyd-Graber, Earl Wagner, Craig Martell, Doug Oard, and Philip Resnik. 2011. Believe me? we can do this! annotating persuasive acts in blog text. In *Proceedings of the Workshops at AAAI*.

Tal August, Lauren Kim, Katharina Reinecke, and Noah A. Smith. 2020. Writing strategies for science communication: Data and computational analysis. In *EMNLP*.

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

Fan Bai, Alan Ritter, and Wei Xu. 2021. Pre-train or annotate? domain adaptation with a constrained budget. In *EMNLP*, pages 5002–5015.

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

Chandra Bhagavatula, Sergey Feldman, Russell Power, and Waleed Ammar. 2018. Content-based citation recommendation. In *NAACL-HLT*.

Lutz Bornmann and Rüdiger Mutz. 2015. Growth rates of modern science: A bibliometric analysis based on the number of publications and cited references. *JASIST*.

Andrew Eliot Borthwick. 1999. *A maximum entropy approach to named entity recognition*. New York University.

Ralph Allan Bradley and Milton E. Terry. 1952. Rank analysis of incomplete block designs: I. the method of paired comparisons. *Biometrika*, 39(3/4):324–345.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020a. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901. Curran Associates, Inc.

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

Isabel Cachola, Kyle Lo, Arman Cohan, and Daniel S. Weld. 2020. Tldr: Extreme summarization of scientific documents. In *EMNLP*.

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*.

Tanmoy Chakraborty, Sandipan Sikdar, Niloy Ganguly, and Animesh Mukherjee. 2014. Citation interactions among computer science fields: a quantitative route to the rise and fall of scientific research. *Social Network Analysis and Mining*.

Shuo Chen and Thorsten Joachims. 2016. Predicting matchups and preferences in context. In *Proceedings of SIGKDD*.

Alexandra Chronopoulou, Matthew E. Peters, and Jesse Dodge. 2021. Efficient hierarchical domain adaptation for pretrained language models. *ArXiv*, abs/2112.08786.

Arman Cohan, Waleed Ammar, Madeleine van Zuylen, and Field Cady. 2019. Structural scaffolds for citation intent classification in scientific publications. In *NAACL-HLT*.

Arman Cohan, Franck Dernoncourt, Doo Soon Kim, Trung Bui, Seokhwan Kim, Walter Chang, and Nazli Goharian. 2018. A discourse-aware attention model for abstractive summarization of long documents. In *NAACL-HLT*.

Arman Cohan and Nazli Goharian. 2015. Scientific article summarization using citation-context and article's discourse structure. In *EMNLP*.

Arman Cohan and Nazli Goharian. 2017. Contextualizing citations for scientific summarization using word embeddings and domain knowledge. In *SIGIR*.

Cristian Danescu-Niculescu-Mizil, Lillian Lee, Bo Pang, and Jon Kleinberg. 2012. Echoes of power: Language effects and power differences in social interaction. In *Proceedings of WWW*.

Cristian Danescu-Niculescu-Mizil, Robert West, Dan Jurafsky, Jure Leskovec, and Christopher Potts. 2013. No country for old members: user lifecycle and linguistic change in online communities. *WWW*.

Hal Daumé III. 2007. Frustratingly easy domain adaptation. In *ACL*, pages 256–263.

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

Olivier Delalleau, Emile Contal, Eric Thibodeau-Laufer, Raul Chandias Ferrari, Yoshua Bengio, and Frank Zhang. 2012. Beyond skill rating: Advanced matchmaking in ghost recon online. *IEEE Transactions on Computational Intelligence and AI in Games*, 4(3):167–177.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. In *NAACL-HLT*.

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 large webtext corpora: A case study on the colossal clean crawled corpus. In *EMNLP*.

Esin Durmus and Claire Cardie. 2018. Exploring the role of prior beliefs for argument persuasion. In *Proceedings of NAACL-HLT*.

Esin Durmus and Claire Cardie. 2019. Modeling the factors of user success in online debate. In *Proceedings of WWW*.

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

Arpad E. Elo. 1978. *The Rating of Chessplayers, Past and Present*. Arco Pub.

Denis Emelin, Ronan Le Bras, Jena D. Hwang, Maxwell Forbes, and Yejin Choi. 2021. Moral stories: Situated reasoning about norms, intents, actions, and their consequences. *ArXiv*, abs/2012.15738.

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

Angela Fan, Mike Lewis, and Yann Dauphin. 2018. Hierarchical neural story generation. In *ACL*.

Hila Gonen, Ganesh Jawahar, Djamé Seddah, and Yoav Goldberg. 2020. Simple, interpretable and stable method for detecting words with usage change across corpora. In *ACL*.

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. 2020a. Don't stop pretraining: Adapt language models to domains and tasks. In *ACL*.

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

Ivan Habernal and Iryna Gurevych. 2016. What makes a convincing argument? Empirical analysis and detecting attributes of convincingness in web argumentation. In *Proceedings of EMNLP*.

Ivan Habernal and Iryna Gurevych. 2017. Argumentation mining in user-generated web discourse. *Computational Linguistics*, 43(1):125–179.

William L. Hamilton, Jure Leskovec, and Dan Jurafsky. 2016. Diachronic word embeddings reveal statistical laws of semantic change. In *ACL*.

Xiaochuang Han and Jacob Eisenstein. 2019. Unsupervised domain adaptation of contextualized embeddings for sequence labeling. In *EMNLP*.

David A Hanauer, Yang Liu, Qiaozhu Mei, Frank J Manion, Ulysses J Balis, and Kai Zheng. 2012. Hedging their mets: the use of uncertainty terms in clinical documents and its potential implications when sharing the documents with patients. In *Proceedings of AMIA*.

Ralf Herbrich, Tom Minka, and Thore Graepel. 2007. TrueskillTM: a Bayesian skill rating system. In *Proceedings of NIPS*.

Jack Hessel and Lillian Lee. 2019. Something's brewing! early prediction of controversy-causing posts from discussion features. In *NAACL*.

Christopher Hidey, Elena Musi, Alyssa Hwang, Smaranda Muresan, and Kathy McKeown. 2017. Analyzing the semantic types of claims and premises in an online persuasive forum. In *Proceedings of the Workshop on Argument Mining*.

Ari Holtzman, Jan Buys, M. Forbes, and Yejin Choi. 2020. The curious case of neural text degeneration. In *ICLR*.

Yufan Huang, Yanzhe Zhang, Jiaao Chen, Xuezhi Wang, and Diyi Yang. 2021. Continual learning for text classification with information disentanglement based regularization. In *ACL*.

Ken Hyland. 1996. Writing without conviction? Hedging in science research articles. *Applied Linguistics*, 17(4):433–454.

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

Kokil Jaidka, Muthu Kumar Chandrasekaran, Devanshu Jain, and Min-Yen Kan. 2017. The cl-scisumm shared task 2017: Results and key insights. In *BIRNDL@SIGIR*.

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

Shagun Jhaver, Christian Boylston, Diyi Yang, and Amy Bruckman. 2021. Evaluating the effectiveness of deplatforming as a moderation strategy on twitter. *Proceedings of the ACM on Human-Computer Interaction*.

Jing Jiang and ChengXiang Zhai. 2007. Instance weighting for domain adaptation in nlp. In *ACL*.

Liwei Jiang, Jena D. Hwang, Chandrasekhar Bhagavatula, Ronan Le Bras, Maxwell Forbes, Jon Borchardt, Jenny Liang, Oren Etzioni, Maarten Sap, and Yejin Choi. 2021. Delphi: Towards machine ethics and norms. *ArXiv*, abs/2110.07574.

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. *arXiv preprint arXiv:2110.08534*.

Sukhwan Jung and Aviv Segev. 2013. Analyzing future communities in growing citation networks. *UnstructureNLP@CIKM*.

David Jurgens, Srijan Kumar, Raine Hoover, Dan McFarland, and Dan Jurafsky. 2018. Measuring the evolution of a scientific field through citation frames. *TACL*.

David Jurgens, James McCorriston, and Derek Ruths. 2017. An analysis of individuals' behavior change in online groups. In *SocInfo*.

J Peter Kincaid, Robert P. Fishburne Jr, Richard L. Rogers, and Brad S. Chissom. 1975. Derivation of new readability formulas (Automated Readability Index, Fog Count and Flesch Reading Ease Formula) for Navy enlisted personnel. *CNTECHTRA Research Branch Report 8-75*.

William Labov. 2011. *Principles of linguistic change, volume 3: Cognitive and cultural factors*, volume 36. John Wiley & Sons.

Angeliki Lazaridou, Adhiguna Kuncoro, Elena Gribovskaya, Devang Agrawal, Adam Liska, Tayfun Terzi, Mai Gimenez, Cyprien de Masson d'Autume, Tomáš Kočiský, Sebastian Ruder, Dani Yogatama, Kris Cao, Susannah Young, and Phil Blunsom. 2021. Mind the gap: Assessing temporal generalization in neural language models. In *NeurIPS*.

Jinhyuk Lee, Wonjin Yoon, Sungdong Kim, Donghyeon Kim, Sunkyu Kim, Chan Ho So, and Jaewoo Kang. 2019. Biobert: a pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics*.

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*.

Marco Lippi and Paolo Torroni. 2016. Argumentation mining: State of the art and emerging trends. *ACM Transactions on Internet Technology (TOIT)*, 16(2):10.

Kyle Lo, Lucy Lu Wang, Mark Neumann, Rodney Kinney, and Daniel S. Weld. 2020a. S2ORC: The Semantic Scholar Open Research Corpus. In *Proceedings of ACL*.

Kyle Lo, Lucy Lu Wang, Mark Neumann, Rodney Kinney, and Daniel S Weld. 2020b. 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*.

Li Lucy and David Bamman. 2021. Characterizing english variation across social media communities with bert. *TACL*.

Hans Peter Luhn. 1958. The automatic creation of literature abstracts. *IBM Journal of Research and Development*.

Stephanie M. Lukin, Pranav Anand, Marilyn Walker, and Steve Whittaker. 2017. Argument strength is in the eye of the beholder: Audience effects in persuasion. In *Proceedings of EACL*.

Kelvin Luu, Daniel Khashabi, Suchin Gururangan, Karishma Mandyam, and Noah A. Smith. 2022. Time waits for no one! analysis and challenges of temporal misalignment. *NAACL*.

Kelvin Luu, Chenhao Tan, and Noah A. Smith. 2019. Measuring online debaters' persuasive skill from text over time. *Transactions of the Association for Computational Linguistics*, 7:537–550.

Kelvin Luu, Xinyi Wu, Rik Koncel-Kedziorski, Kyle Lo, Isabel Cachola, and Noah A. Smith. 2021. Explaining relationships between scientific documents. In *ACL*, Online. Association for Computational Linguistics.

Julian McAuley and Jure Leskovec. 2013. From amateurs to connoisseurs: modeling the evolution of user expertise through online reviews. *WWW*.

Sean M. McNee, Istvan Albert, Dan Cosley, Prateep Gopalkrishnan, Shyong K. Lam, Al Mamunur Rashid, Joseph A. Konstan, and John Riedl. 2002. On the recommending of citations for research papers. In *CSCW*.

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*.

Joshua E. Menke and Tony R. Martinez. 2008. A Bradley–Terry artificial neural network model for individual ratings in group competitions. *Neural Computing and Applications*, 17(2):175–186.

Burt L. Monroe, Michael P. Colaresi, and Kevin M. Quinn. 2008. Fightin' Words: Lexical feature selection and evaluation for identifying the content of political conflict. *Political Analysis*, 16(4):372–403.

Preslav I Nakov, Ariel S Schwartz, and Marti Hearst. 2004. Citances: Citation sentences for semantic analysis of bioscience text. In *SIGIR*.

Ramesh Nallapati, Amr Ahmed, Eric P. Xing, and William W. Cohen. 2008. Joint latent topic models for text and citations. In *KDD*.

Franz Josef Och. 2003. Minimum error rate training in statistical machine translation. In *ACL*.

Nathan Ong, Diane Litman, and Alexandra Brusilovsky. 2014. Ontology-based argument mining and automatic essay scoring. In *Proceedings of the Workshop on Argumentation Mining*.

Chris D. Paice. 1980. The automatic generation of literature abstracts: An approach based on the identification of self-indicating phrases. In *SIGIR*.

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

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting on association for computational linguistics*, pages 311–318. Association for Computational Linguistics.

Joonsuk Park and Claire Cardie. 2014. Identifying appropriate support for propositions in online user comments. In *Proceedings of the Workshop on Argumentation Mining*.

Isaac Persing and Vincent Ng. 2015. Modeling argument strength in student essays. In *Proceedings of ACL*.

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

Jonathan Pilault, Raymond Li, Sandeep Subramanian, and Christopher Pal. 2020. On extractive and abstractive neural document summarization with transformer language models. In *EMNLP*.

Peter Potash and Anna Rumshisky. 2017. Towards debate automation: a recurrent model for predicting debate winners. In *Proceedings of EMNLP*.

Vinodkumar Prabhakaran, Ashima Arora, and Owen Rambow. 2014. Staying on topic: An indicator of power in political debates. In *Proceedings of EMNLP*.

Vinodkumar Prabhakaran and Owen Rambow. 2013. Written dialog and social power: Manifestations of different types of power in dialog behavior. In *Proceedings of IJCNLP*.

Vinodkumar Prabhakaran and Owen Rambow. 2014. Predicting power relations between participants in written dialog from a single thread. In *Proceedings of ACL*.

Vahed Qazvinian and Dragomir R. Radev. 2008. Scientific paper summarization using citation summary networks. In *Coling 2008*. Coling 2008 Organizing Committee.

Minghui Qiu, Yanchuan Sim, Noah A. Smith, and Jing Jiang. 2015. Modeling user arguments, interactions, and attributes for stance prediction in online debate forums. In *SDM*.

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

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019b. Language models are unsupervised multitask learners. *OpenAI Blog*, 1(8).

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 Pierrehumbert. 2021. Temporal adaptation of bert and performance on downstream document classification: Insights from social media. In *Findings of EMNLP*, pages 2400–2412.

Maja R. Rudolph and David M. Blei. 2018. Dynamic embeddings for language evolution. *WWW*.

Burr Settles, Masato Hagiwara, and Geoffrey T. LaFlair. 2020. Machine learning–driven language assessment. *TACL*.

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*.

Hidetoshi Shimodaira. 2000. Improving predictive inference under covariate shift by weighting the log-likelihood function. *Journal of Statistical Planning and Inference*.

Nate Silver. 2015. How our NFL predictions work. *https://fivethirtyeight.com/methodology/how-our-nfl-predictions-work/*. Accessed: 2019-02-30.

Swapna Somasundaran and Janyce Wiebe. 2009. Recognizing stances in online debates. In *Proceedings of ACL-IJCNLP*.

Yi Song, Michael Heilman, Beata Beigman Klebanov, and Paul Deane. 2014. Applying argumentation schemes for essay scoring. In *Proceedings of the Workshop on Argumentation Mining*.

Sandeep Soni, Kristina Lerman, and Jacob Eisenstein. 2021. Follow the leader: Documents on the leading edge of semantic change get more citations. *JASIST*.

Dhanya Sridhar, James Foulds, Bert Huang, Lise Getoor, and Marilyn Walker. 2015. Joint models of disagreement and stance in online debate. In *Proceedings of ACL*.

Christian Stab and Iryna Gurevych. 2014. Identifying argumentative discourse structures in persuasive essays. In *Proceedings of EMNLP*.

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

Ian Stewart and Jacob Eisenstein. 2018b. 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. In *ACM/IEEE Joint Conference on Digital Libraries (JCDL)*, pages 170–179. IEEE.

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.

Chenhao Tan, Lillian Lee, and Bo Pang. 2014. The effect of wording on message propagation: Topic-and author-controlled. In *Proceedings of ACL*.

Chenhao Tan, Vlad Niculae, Cristian Danescu-Niculescu-Mizil, and Lillian Lee. 2016. Winning arguments: Interaction dynamics and persuasion strategies in good-faith online discussions. In *Proceedings of WWW*.

Chenhao Tan, Hao Peng, and Noah A. Smith. 2018. You are no Jack Kennedy: On media selection of highlights from presidential debates. In *Proceedings of WWW*.

Yla R. Tausczik and James W. Pennebaker. 2010. The psychological meaning of words: LIWC and computerized text analysis methods. *Journal of Language and Social Psychology*, 29(1):24–54.

Simone Teufel, Advaith Siddharthan, and Dan Tidhar. 2006. Automatic classification of citation function. In *EMNLP*.

Marco Valenzuela, Vu A. Ha, and Oren Etzioni. 2015. Identifying meaningful citations. In *AAAI Workshop: Scholarly Big Data*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *NeurIPS*.

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*.

Marilyn A. Walker, Pranav Anand, Robert Abbott, and Ricky Grant. 2012. Stance classification using dialogic properties of persuasion. In *Proceedings of NAACL-HLT*.

Alex Wang, Yada Pruksachatkun, Nikita Nangia, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R Bowman. 2019. Superglue: a stickier benchmark for general-purpose language understanding systems. In *NeurIPS*.

Lu Wang, Nick Beauchamp, Sarah Shugars, and Kechen Qin. 2017. Winning on the merits: The joint effects of content and style on debate outcomes. *Transactions of the Association for Computational Linguistics*, 5:219–232.

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*.

Xinyu Xing, Xiaosheng Fan, and Xiaojun Wan. 2020. Automatic generation of citation texts in scholarly papers: A pilot study. In *ACL*.

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

Yi Yang and Jacob Eisenstein. 2016. Part-of-speech tagging for historical english. In *NAACL*.

Michihiro Yasunaga, Jungo Kasai, Rui Zhang, Alexander Richard Fabbri, Irene Li, Dan Friedman, and Dragomir R. Radev. 2019. Scisummnet: A large annotated corpus and content-impact models for scientific paper summarization with citation networks. In *AAAI*.

Rowan Zellers, Ari Holtzman, Elizabeth Clark, Lianhui Qin, Ali Farhadi, and Yejin Choi. 2021. Turingadvice: A generative and dynamic evaluation of language use. In *NAACL*.

Rowan Zellers, Ari Holtzman, Hannah Rashkin, Yonatan Bisk, Ali Farhadi, Franziska Roesner, and Yejin Choi. 2019. Defending against neural fake news. In *NeurIPS*.

Fan Zhang, Homa B. Hashemi, Rebecca Hwa, and Diane Litman. 2017a. A corpus of annotated revisions for studying argumentative writing. In *Proceedings of ACL*.

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*.

Justine Zhang, William L. Hamilton, Cristian Danescu-Niculescu-Mizil, Dan Jurafsky, and Jure Leskovec. 2017b. Community identity and user engagement in a multi-community landscape. *ICWSM*.

Justine Zhang, Ravi Kumar, Sujith Ravi, and Cristian Danescu-Niculescu-Mizil. 2016. Conversational flow in Oxford-style debates. In *Proceedings of NAACL-HLT*.

Kun Zhang, Bernhard Schölkopf, Krikamol Muandet, and Zhikun Wang. 2013. Domain adaptation under target and conditional shift. In *ICML*.

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*.