universität wien

MASTERARBEIT | MASTER'S THESIS

Titel | Title

Temporal Adaptation Techniques in Diachronic Language Modelling

verfasst von | submitted by

Ksenia Dvorkina

angestrebter akademischer Grad | in partial fulfilment of the requirements for the degree of

Master of Science (MSc)

Wien | Vienna, 2025

Studienkennzahl lt. Studienblatt | Degree programme code as it appears on the student record sheet:	UA 066 645
Studienrichtung lt. Studienblatt | Degree programme as it appears on the student record sheet:	Masterstudium Data Science
Betreut von | Supervisor:	Univ.-Prof. Dr. -Ing. Benjamin Roth B.Sc. M.Sc.

Acknowledgements

I would like to express my sincere gratitude Benjamin Roth and Terra Blevins for their supervision, guidance and support throughout the development of this thesis.

I also acknowledge the use of AI tools, specifically GPT-3.5 and DeepSeek V3, which were employed for editing support, such as grammar correction, rephrasing, and stylistic improvements.

Abstract

This thesis explores strategies for temporally adapting generative language models to capture historical variation in English. While most modern language models treat language as static, this work develops a diachronic version of the Open Language Model (OLMo) by fine-tuning it on historical English texts. Two main strategies are explored: full fine-tuning with various forms of temporal conditioning, and a Mixture of Experts (MoE) approach, where each expert is fine-tuned on a different historical period and combined using a learned gating mechanism.

Experiments show that temporal conditioning during full fine-tuning had limited impact on perplexity. In contrast, the MoE approach led to clearer improvements in perplexity and enabled a structural encoding of time. Beyond language modelling, we show that the MoE's gating outputs can be used for text dating, demonstrating that model is able to predict the publication year of historical texts with high accuracy.

Kurzfassung

Diese Masterarbeit untersucht Strategien zur zeitlichen Anpassung generativer Sprachmodelle, um historische Sprachvariationen im Englischen abzubilden. Während die meisten modernen Sprachmodelle Sprache als statisch betrachten, wird in dieser Arbeit eine diachrone Version des Open Language Model (OLMo) entwickelt, indem es auf historischen englischen Texten feinabgestimmt wird. Es werden zwei Hauptstrategien verfolgt: die vollständige Feinabstimmung mit verschiedenen Formen zeitlicher Konditionierung sowie ein Mixture-of-Experts-(MoE)-Ansatz, bei dem jeweils ein Experte auf einen unterschiedlichen historischen Zeitraum feinabgestimmt wird und deren Ergebnisse durch einen gelernten Steuerungsmechanismus kombiniert werden.

Die Experimente zeigen, dass die zeitliche Konditionierung während der vollständigen Feinabstimmung nur begrenzte Auswirkungen auf die Perplexität hat. Im Gegensatz dazu führt der MoE-Ansatz zu deutlichen Verbesserungen der Perplexität und ermöglicht eine strukturelle Kodierung der Zeit. Über die Sprachmodellierung hinaus wird gezeigt, dass die Steuerungsausgaben des MoE zur Datierung von Texten genutzt werden können, wobei das Modell in der Lage ist, das Veröffentlichungsjahr historischer Texte mit hoher Genauigkeit vorherzusagen.

Contents

1. Introduction

Understanding how language evolves over time is essential for studying linguistic change, improving historical NLP tools, and building temporally-aware AI systems. Yet, most modern language models treat language as static, lacking awareness of historical variation.

This thesis addresses that gap by developing and evaluating a generative diachronic language model for English. Specifically, we fine-tune a generative, general-purpose English language model (OLMo) on historical English texts. This model adaptation is done via diachronic modelling: using specific model parameters to encode time-specific representations. Building on this, we introduce a Mixture of Experts (MoE) architecture to improve temporal adaptation, which is then applied to the task of text dating.

The evolution of language over time has always been a topic of lasting interest within linguistics and natural language processing (NLP). While significant advancements in language modelling have been achieved through pre-trained models such as GPT [18], a causal language model, and BERT [19], a masked language model, these models generally operate without temporal context, capturing language as it exists at a fixed point in time. Autoregressive language models, such as GPT, generate text sequentially by forecasting the next token based on preceding tokens. In contrast, masked language models like BERT are trained to reconstruct missing tokens using the context from both preceding and following tokens. However, this lack of temporal awareness makes it challenging to analyse linguistic changes across historical periods or account for phenomena such as semantic drift and syntactic evolution. Diachronic language models aim to address this gap by embedding temporal information into language representations, allowing for the study of language as a dynamic and evolving system. Unlike traditional models, which generalise across time periods without considering temporal nuances, diachronic models capture how linguistic features, such as semantics and syntax, change over specific historical intervals.

Recent advancements in diachronic language models include HistBERT [4], a pre-trained model designed for lexical semantic analysis over time, which has demonstrated its ability to identify temporal shifts in word meanings and date texts based on linguistic features. Similarly, approaches like the Temporal Referencing framework [8] and the Dynamic Word Embeddings model [2] have shown promise in capturing semantic evolution over time. However, these models primarily focus on classification or retrieval tasks and lack the ability

to generate text, which is a common feature of modern large language models (LLM). In contrast, diachronic generative language models could extend this capability by incorporating temporal dynamics into their architectures. By enabling the generation of text that captures semantic and linguistic changes over time, they support tasks that require historically accurate or temporally contextualised outputs.

Building on these advancements, generative language models, such as the Open Language Model (OLMo) [15], have proven beneficial by providing a foundation for a broad range of linguistic tasks. These models, initially trained on general corpora, can be fine-tuned for specific tasks, which reduces computational complexity and accelerates model adaptation in comparison to training from scratch. However, these models still retain the limitation of masked language models in that they are primarily trained on modern text without temporal awareness.

This research focuses on developing a generative diachronic language model, which enables us to (a) investigate the ability of general-purpose language models to learn and represent text stratified by time and (b) apply this model to the linguistic analysis of historical texts. This thesis examines how OLMo captures the evolution of language across different time periods, using perplexity analysis (PPL) as a key evaluation metric. It also considers a range of approaches to temporal adaptation, including the use of time indicators such as special tokens, year labels, or lexical prompts. Building on these experiments, (c) a Mixture of Experts architecture is introduced: rather than conditioning a single model on time, this approach fine-tunes separate instances of OLMo on distinct historical intervals and combines their outputs through a soft gating mechanism. The final stage of this research (d) assesses the potential of the MoE in text dating, evaluating its ability to accurately infer the temporal origins of texts.

The remainder of this thesis is structured as follows. Chapter 2 introduces background and related work. Chapter 3 describes the dataset, preprocessing steps, and data splits. Chapter 4 introduces the baseline fine-tuning and evaluates different temporal adaptation techniques. Chapter 5 presents the Mixture of Experts model, and Chapter 6 discusses MoE results in the context of text dating. Chapter 7 scales up the best performing MoE model to a model of a bigger size.

Unlike the typical thesis structure, results are presented directly within each of the main experimental chapters (Chapters 4 to 7) and are later summarised in Chapter 8. This decision reflects the dependency between chapters, as results from earlier experiments often inform the methodology of subsequent ones. Also, while full fine-tuning serves as an initial exploration of temporal embedding techniques and motivates the transition to the MoE approach, the two lines of investigation are largely analysed separately, as they provide different levels of insight. Additionally, the MoE results are foundational for year prediction and must be understood in advance.

2. Background and Related Work

The study of diachronic linguistics through computational methods has led to diverse approaches aimed at capturing language evolution. While early efforts relied on corpora analysis and static models, modern techniques increasingly employ neural architectures, enabling richer representations and insights into historical linguistic phenomena. This section critically examines key works that inform and contrast with the methodological choices of this thesis.

2.1. Diachronic NLP

Diachronic linguistics, the study of how languages evolve over time, provides important insights into phenomena such as semantic drift, syntactic change and lexical innovation. Traditional research in this area has relied on corpus-based methods that examine chronological patterns in word usage, grammatical structures and stylistic variation. For example, frequency analyses of corpora such as the Helsinki Corpus [13] and the Corpus of Historical American English [34] have revealed long-term trends in lexical and syntactic developments. Techniques including concordance analysis, collocation studies and manual annotation have been central to these studies [33]. However, traditional approaches face challenges such as data sparsity, orthographic variation and genre imbalance.

Development of machine learning (ML) techniques has enabled more sophisticated methods for semantic change detection. Embedding-based methods such as diachronic word embeddings [2] track shifts in word meaning across discrete time slices. Probabilistic models like SCAN [28] and genre-aware models such as GASC [27] address the complexities of polysemy and contextual variation. More recently, DiSC [29] and its successor EDiSC [30]

introduce clustering and sampling techniques that enable fine-grained semantic change detection even in low-resource settings.

The advent of contextualised embeddings has transformed diachronic Natural Language Processing (NLP). For instance, HistBERT [4] leverages transformer-based architectures for diachronic lexical analysis, demonstrating superior performance in capturing nuanced semantic shifts. Similarly, Ishihara et al. [31] employed word2vec embeddings to analyse eleven years of Japanese and English news articles, highlighting how embedding-based methods can reveal linguistic trends tied to social and historical events. Extending these approaches, recent studies emphasise the potential of large language models (LLMs) in diachronic tasks. Periti et al. [12] compared LLMs such as GPT and BERT to static embeddings for detecting semantic change, revealing their superior ability to capture nuanced, context-dependent shifts.

Foundational research also highlights challenges inherent to modelling diachronic language change. Jaidka et al. [7] examined the phenomenon of diachronic degradation, the decline in embedding quality when models generalise across temporal boundaries, revealing limitations of static embeddings in temporal contexts. To address these limitations, Dubossarsky et al. [8] proposed temporal referencing, a technique that incorporates explicit temporal context to enhance semantic shift detection. Building on these insights, Rijhwani & Preotiuc-Pietro [21] introduced temporally informed analysis for named entity recognition (NER), demonstrating how temporal awareness improves entity disambiguation and detection in historical texts, thus opening new avenues for diachronic NLP applications.

Innovative models have expanded diachronic NLP into diverse applications, such as MacBERTh [5], that integrates historical texts spanning five centuries, enabling robust analyses of long-term linguistic evolution. In contrast, TimeLMs [6] address shorter-term language change in social media, leveraging diachronic Twitter data and employing continual learning strategies to address temporal trends and concept drift.

While semantic shift detection has long dominated diachronic NLP, a growing body of work has emerged around temporal text classification and generation, which are the primary concerns of this thesis. In particular, text dating, the task of predicting when a document was written, has been explored as a means to learn temporally sensitive representations of language. Han et al. [36] introduced TALM, a time-aware language model that learns

temporal word representations by transferring general-domain language models to time-specific ones, achieving state-of-the-art performance in historical text dating. In the area of temporal language models for downstream reasoning tasks, Peng et al. [1] introduced ChapTER, a temporal reasoning framework for knowledge graphs that leverages prefix-tuning and contrastive learning. By incorporating historically grounded textual context into a pseudo-Siamese encoder architecture, their approach effectively balances temporal and semantic signals, achieving strong results with minimal additional parameters.

Building on these foundations, Li and Flanigan [16] introduced the task of future language modelling, formalizing the probabilistic prediction of future texts based on temporal document histories, showing temporally conditioned models outperform non-temporal baselines. Complementing this, Dhingra et al. [11] proposed time-aware language models as temporal knowledge bases that jointly model text with timestamps to improve memorization and calibration of temporal facts and enable efficient model updates without full retraining. Additionally, Ushio and Camacho-Collados [10] systematically analysed temporal generalization in social media language models, revealing performance degradation on entity-focused tasks due to temporal shifts and showing that continuous pre-training does not always enhance adaptability.

While prior work in diachronic NLP has primarily focused on tasks such as semantic change detection, named entity recognition and text dating, this thesis extends the scope towards generative diachronic modelling. Unlike studies that treat time as a prediction target or as a latent contextual factor, the present work explicitly incorporates temporal structure into the model architecture through historical expert specialisation. This shift towards temporally conditioned language generation highlights the need for modular representations that preserve temporal distinctions, motivating the integration of Mixture-of-Experts design and full fine-tuning strategies discussed in the following sections.

2.2. Mixture of Experts

Another significant advancement in language modelling is the Mixture-of-Experts (MoE) paradigm, which offers a scalable and computationally efficient framework by dynamically activating only a subset of specialised modules during inference. Early work by Shazeer et al. [39] introduced the sparsely gated MoE layer, enabling models to grow in parameter count without a proportional increase in compute. Building on this foundation, recent architectures

such as OLMoE (Open Mixture-of-Experts Language Models) [38] and GLaM (Generalist Language Model) [42] have demonstrated the benefits of modularity, achieving competitive results with reduced inference cost by routing inputs through only a fraction of available experts.

Recent studies have explored using MoE architectures to model temporal and semantic variation. Liu et al. [40] propose Mixture of Temporal Experts (MoTE), which treats time as a domain variable in multilingual classification, showing that temporal modularity can improve generalisation under diachronic data drift. Similarly, Chen et al. [41] use MoE in temporal word sense disambiguation, integrating syntactic and semantic cues to better capture sense shifts over time.

Beyond sparse activation, domain-specialised and softly gated MoEs have emerged as a promising direction. Gururangan et al. [3] introduce DEMix layers to disentangle domain specific knowledge via modular expert layers, supporting zero shot generalisation and modular fine-tuning. Seo et al. [32] explore frozen experts in a shared transformer, enabling token level expert routing, while Wang et al. [22] leverage metadata supervised gating to select experts trained on distinct speaker identities. Armi et al. [25] propose soft gating mechanisms that enable global, sequence-level expert weighting via trainable gating networks.

While recent work has demonstrated the potential of MoE architectures for modelling structured variation, such as across speakers, domains, or time, approaches that focus specifically on temporal specialisation remain limited. In particular, the combination of globally applied, soft gating mechanisms with supervision from historical metadata is not yet widely explored. This suggests the potential of temporally modular MoE architectures that incorporate historical information into the gating process, with the aim of improving temporal conditioning for language generation and enabling more accurate temporal inference from textual inputs.

This thesis builds on these insights by developing a temporally specialised MoE architecture for generative diachronic language modelling, in which each expert is fine-tuned on a distinct historical period and combined via a learned gating mechanism.

2.3. Fine-tuning

Adapting large language models to downstream tasks through fine-tuning has become a fundamental practice in modern Natural Language Processing research. Standard fine-tuning approaches involve updating all model parameters on task specific data. In the context of diachronic language modelling, full fine-tuning ensures that the model internalises both general linguistic knowledge and the temporal nuances critical to capturing historical language dynamics.

However, this approach is resource-intensive, leading to increased interest in parameter-efficient fine-tuning (PEFT) methods [24], which adapt models using fewer trainable parameters, such as sequential fine-tuning [26] , LoRA [9], adapters [14] and BitFit [23]. While recent work has explored combining PEFT with MoE architectures to reduce costs, such as MoELoRA [35] and MixLoRA [37], have become popular due to their reduced computational cost, these methods often freeze large portions of the model and adapt only limited components.

However, this thesis employs full fine-tuning exclusively, as the objective is to faithfully capture the distinct linguistic characteristics of specific historical periods. This design choice prioritises historical fidelity in the generated outputs over training efficiency and aligns with the overall goal of developing a generative diachronic language model that can reflect the evolving nature of language over time.

3. Data

This chapter introduces the dataset used in this thesis, beginning with an overview and motivation for its selection. We then describe the data cleaning steps and the strategy for splitting the corpus into training, validation and test subsets. Finally, we present a quantitative characterisation of the cleaned and split data, including genre and time period distributions.

3.1. Overview

To model historical variation in English through the task of text dating, a dataset is required that spans a substantial time period, offers linguistic diversity and is openly available for research use. The Corpus of Late Modern English Texts (CLMET) [20] meets these criteria, making it a suitable choice for this research. Spanning just over two centuries, from year

1710 to 1920, this corpus offers a diverse collection of texts by British and Irish authors, making it well-suited for temporal language modelling. Corpus also enables in-depth linguistic analysis by supplying each text with rich metadata. In this research, we will only be using ID {unique text identificator}, author, title, printedDate {year the text was printed}, words {number of words in the text}, genre and the text itself.

The CLMET is distributed under an Attribution–Non-Commercial–Share-Alike 4.0 International licence, ensuring that it is openly accessible for research purposes. This openness supports both the reproducibility of results and the scalability of experiments. The dataset is also actively maintained and continuously improved by its creators, with all earlier versions preserved.

Created by Uppsala University, CLMET includes 333 texts totalling over 34 million tokens. To maintain diversity, no author is represented by more than three texts, and the dataset's genres are systematically classified. The most presented genre is Narrative fiction (49.06%), followed by Narrative non-fiction (14.80%), Treatise (14.27%) and Drama (4.23%). A portion of the texts (17.64%) is classified as Other. While the overall genre distribution is uneven across the entire corpus, the distribution of genres is held fixed in each temporal split. The original corpus also included the genre "Letters", but we excluded it, as contents of these texts often span over a century, making them unsuitable for our task of text dating.

To illustrate linguistic and stylistic variation across time and genre, we present two texts. An early example, Treatise "An Inquiry into the Nature and Causes of the Wealth of Nations" (1766) by Adam Smith, opens with the following passage:

> "*The annual labour of every nation is the fund which originally supplies it with all the necessaries and conveniencies of life which it annually consumes, and which consist always either in the immediate produce of that labour, or in what is purchased with that produce from other nations.*"

The dense, formal prose reflects the syntactic complexity characteristic of 18th-century treatise writing. In contrast, a later text, Narrative Fiction by Rudyard Kipling's "The Jungle Book" (1894), exemplifies a narrative style more familiar to the modern reader:

> "*Mother Wolf lay with her big gray nose dropped across her four tumbling, squealing cubs, and the moon shone into the mouth of the cave where they all lived.*

``Augrh! ``
said Father Wolf.
``It is time to hunt again. ``
He was going to spring down hill when a little shadow with a bushy tail crossed the threshold and whined:
``Good luck go with you, O Chief of the Wolves. And good luck and strong white teeth go with noble children that they may never forget the hungry in this world. ``"

These excerpts highlight the range of linguistic styles present in the CLMET, supporting its suitability for language modelling.

3.2. Data Cleaning and Data Splits

To capture and analyse such variation over time, we adapt the dataset's predefined temporal segmentation. We divide corpus into three consecutive 70-year periods: 1710–1780, 1780–1850 and 1850–1920. The later periods contain slightly more words: approximately 10.5 million tokens for 1710–1780 (30.5%), 11.3 million for 1780–1850 (32.8%) and 12.6 million for 1850–1920 (36,6%). Texts in the later period also tend to be shorter in length, resulting in a bigger number of texts and authors included.

During the data cleaning, we aimed to remove non-linguistic noise and standardise formatting leveraging Python libraries such as *pandas*, *langdetect* and *re* for regular expression-based text cleaning. Example of the text cleaning is shown in the Figure 1. You can find a sample of symbols present in texts on the left, and a text after it was passed through the cleaning function on the right.

We systematically eliminated editorial artefacts and paratextual elements such as numerical and alphanumerical references enclosed in brackets (e.g., [123], [pg 154], [11a], [Greek: ...]), typographic markers (e.g., ß123, * * * * *), structural tags (e.g., <page type="Page 50"/>) and marginal annotations without context like "[illustration]" or "[image]". Redundant or decorative punctuation patterns (e.g., sequences of question marks) were normalised, and missing-word indicators were standardised to a unified placeholder ([...]). In addition to structured annotations, we also removed sequences of non-standard or corrupted characters, such as random symbols or mis encoded glyphs, which likely resulted from OCR errors or

incompatible character encoding. One text was comprised almost entirely of missing or unreadable symbols, so we made a decision of excluding it.

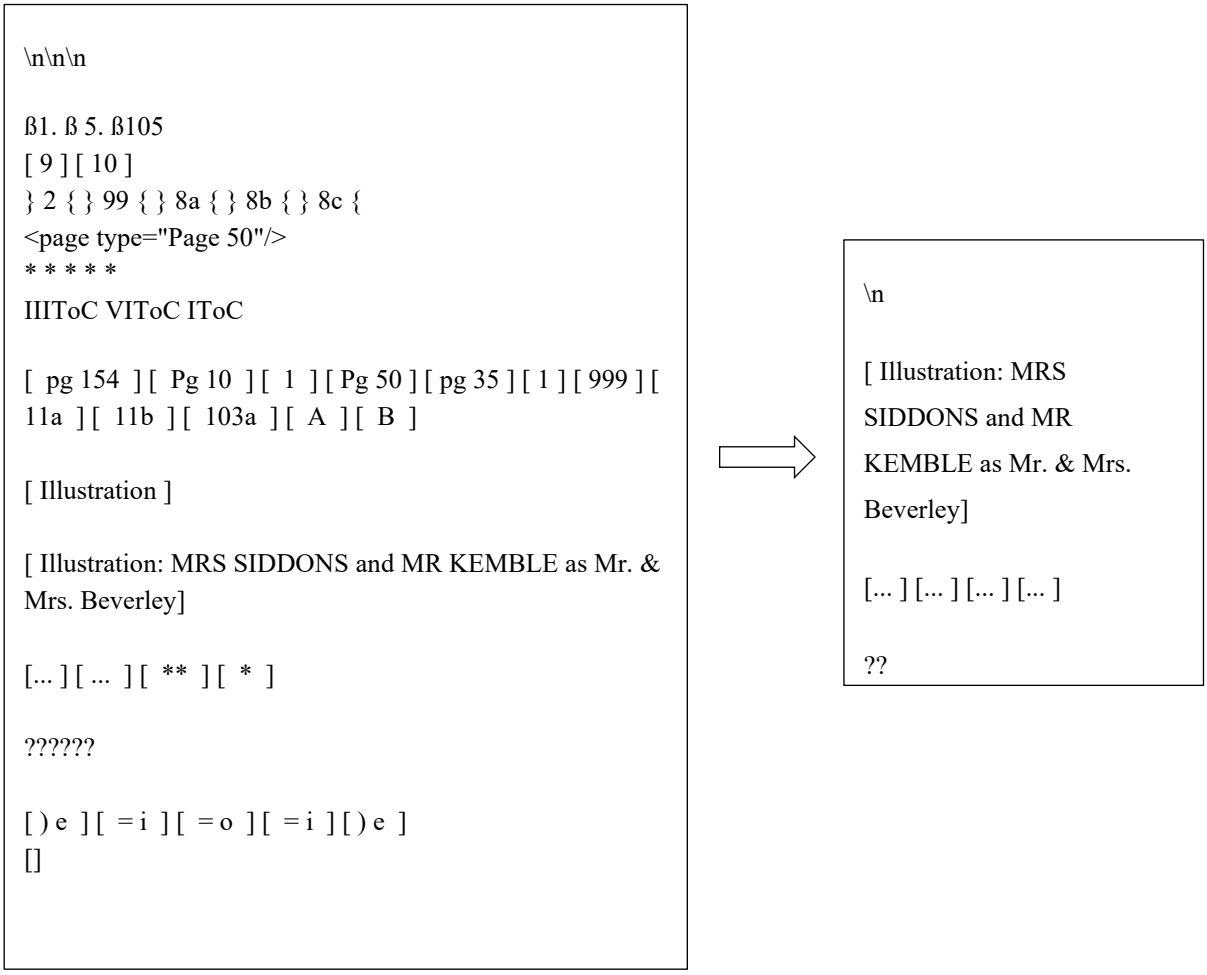

Figure 1. Example of the text cleaning function. Input text is on the left, the output text is on the right.

After data cleaning, dataset retained over 33 million tokens across 295 texts. Then the corpus was divided into training (~80%), validation/ development (~10%) and test (~10%) subsets. One of the main challenges in creating this split was ensuring that the division was based on the number of words rather than the number of texts, while keeping each text intact within a single subset. This was complicated by the wide variation in text length, ranging from 3,211 to 1,438,789 words. We also aimed to preserve the overall proportion of genres across the splits. To achieve this, we first distributed texts with known genre labels and then assigned texts with Undefined genre at the final stage of the split. The exact word counts in the final division are 26,693,626 (79,75%) for training, 3,107,948 (9,29%) for validation and 3,669,547 (10,96%) for test.

3.3. Data Distribution(s)

The genre distribution across data subsets is presented in Figure 2. Each horizontal bar represents a data subset, with segments showing the percentage contribution of each genre by word count. Narrative fiction dominates both the training (55%) and validation (61.4%) subsets, while the test subset shows greater genre balance, with Undefined texts constituting 40.9% alongside substantial proportions of Treatise (15.1%) and Narrative fiction (25.2%). Notably, Drama maintains consistent although minimal representation across all subsets (4.8-6.9%), while Undefined texts are entirely absent from the validation subset, as a consequence of our splitting methodology which prioritised longer texts and known genres during initial allocation.

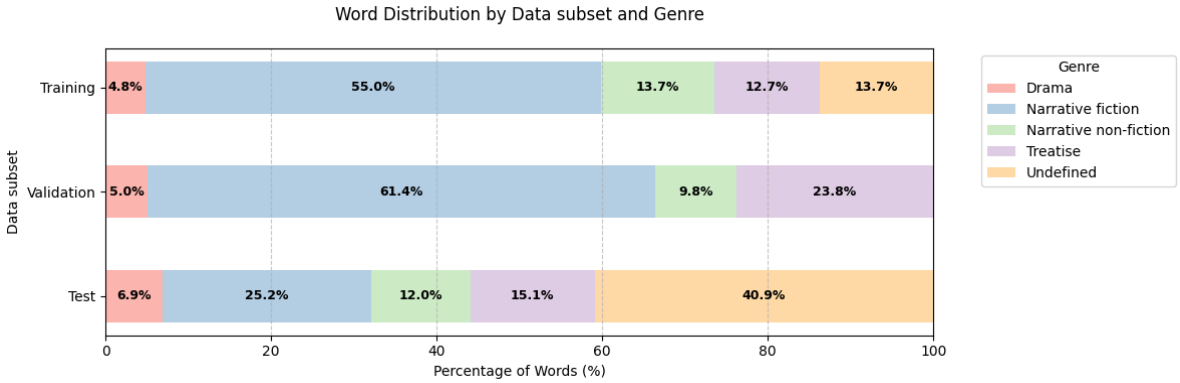

Figure 2. Distribution of genres in CLMET data across train, validation and test splits.

Next, we visualised distributing of words across data splits, genres and time ranges (Figure 3). The data distribution is presented as three bubble subplots, one for each data subset (training, validation and test). In every subplot, each column of the matrix corresponds to one of five genres (Drama, Narrative fiction, Narrative non-fiction, Treatise and Undefined), while each row represents one of the three time ranges (1710–1780, 1780–1850, 1850-1910). Each bubble encodes the percentage of words that a given genre accounts for within the specific time period and subset, using colour, size and label. Rows within each subplot sum to approximately 100%, representing the total word count for each data subset and period. The absence of a bubble indicates that the corresponding genre is not present in that time period and subset.

The Training set appears most balanced across time periods, with Narrative non-fiction consistently representing over 50% (52.6%–56.8%) and other genres generally below 22%. In contrast, the Validation and Test subsets exhibit greater imbalances, likely due to their

smaller sizes and the challenges inherent in their construction. Notably, Narrative non-fiction is absent from both subsets in the earliest period (1710–1780) and also missing from the Test set in the latest period (1850–1920). Furthermore, the Undefined genre is entirely absent from the Validation subset, and Drama is not represented in the Validation subset during the middle period (1780–1850). Clear imbalances also appear in the distribution of Narrative fiction: within the Validation set, it varies from 40% (1710–1780) to 81.7% (1780–1850), then down to 60.7% (1850–1920); within the Test set, it declines from 56.6% (1710–1780) to 21.5% (1780–1850) and further to just 7.6% (1850–1920). These observed disparities in genre representation across subsets and time periods highlight the need for data balance related experiments.

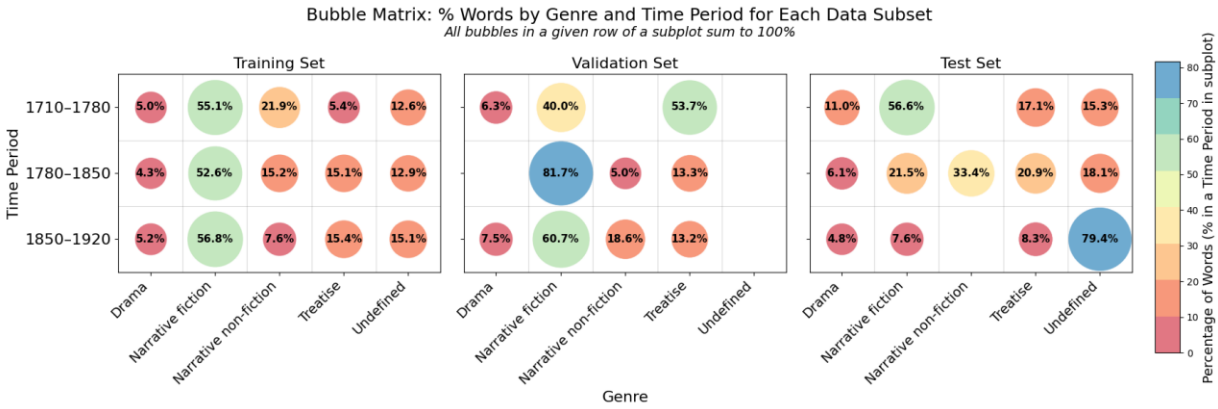

Figure 3. Percentage distribution of words by genre within each time period and data subset. One time range in one subplot is taken as 100% words.

This chapter has outlined the structure, preparation, and distribution of the CLMET corpus used in this thesis. By segmenting the data across defined historical periods and carefully controlling for genre imbalances, we have created a robust foundation for training and evaluating diachronic generative language models. The diversity and historical range of the dataset enable meaningful temporal adaptation and support the downstream task of text dating, ensuring that the experimental results in subsequent chapters are both interpretable and historically grounded.

4. Full Fine-tuning

This chapter focuses on full fine-tuning of the OLMo model on historical English texts. It begins with the methodology, including model architecture, evaluation metrics, and implementation setup. The chapter then moves onto the experimental setup, examining how the number of training epochs and the genre-related data distribution affect performance. Finally, it introduces and evaluated temporal adaptation techniques that were also later used in MoE setup. Results related to the number of epochs and genre-related data distribution are presented within the respective subchapters, while the analysis of temporal embeddings is concentrated in the final subchapter of full fine-tuning.

In summary, after establishing a stable training baseline, the model was evaluated using several forms of temporal conditioning. Across all tested configurations, the impact of temporal embeddings on model performance was minimal, and even incorrect temporal information caused only negligible degradation. These results suggest that the model did not learn to meaningfully condition on time prompts.

4.1. Model and Training Setup

4.1.1. Model Architecture

This thesis employs a generative model, Open Language Model (OLMo) [15], as the base language model for fine-tuning. OLMo is a decoder-only transformer pretrained autoregressively on English text; it follows the standard transformer architecture with causal self-attention, where each token prediction is conditioned on preceding tokens via masked attention.

As the pretraining corpus consists primarily of modern English, model's learned representations are placed out of distribution with respect to historical language. Therefore, the model’s capacity to capture diachronic lexical and syntactic variation without further fine-tuning might be limited, motivating us to employ the full fine-tuning to adapt model through the diachronic language modelling.

Full model architecture is outlined by Groeneveld et al [15]. We take the published, fully pre-trained model and employ it for fine-tuning. During fine-tuning, as in pre-training, model minimizes cross-entropy loss over the input sequence:

$$\mathcal{L} = -\frac{1}{\mathrm{N}} \sum_{\mathrm{i}=1}^{\mathrm{N}} \log p(x_i \mid x_{<i}),$$

where $x_{<i}$is the target token, $x_{<i}$ denotes all prior tokens in the sequence and N is the sequence length. This loss is computed for every token position (except the prompt) during autoregressive training.

This model serves as the foundation for both full fine-tuning and the expert models used in the Mixture of Experts framework.

4.1.2. Evaluation Metric

Model performance is evaluated with perplexity (PPL) as the primary evaluation metric. PPL is the exponentiated cross-entropy loss, providing an intuitive measure of the model's uncertainty in next-token prediction:

$$\mathrm{PPL} = \exp(\mathcal{L}) = \exp\left(-\frac{1}{N}\sum_{i=1}^{N} \log p\,(x_i \mid x_{<i})\right)$$

Lower PPL indicates better predictive performance, with values closer to 1 (the theoretical minimum for a perfect model) corresponding to highly confident and accurate predictions over the vocabulary.

In this thesis, perplexity serves as the primary evaluation metric for comparing fine-tuned models across different setups.

4.1.3. Implementation Details

Training and evaluation were conducted using the Hugging Face Trainer class, with mixed-precision training (bfloat16) and a cosine learning rate scheduler. Inference at test time was performed via a custom evaluation loop using *torch.utils.data.DataLoader*. Data preprocessing included tokenisation, dynamic chunking, metadata alignment and prompt construction with time-specific tokens or texts. A custom data collator was used to support variable-length inputs and efficient batch construction. All preprocessing, logging and

evaluation utilities were implemented as modular components supporting both training and evaluation workflows. Full implementation details and code are provided on GitHub[1].

All experiments in this thesis were conducted using the Galadriel node, provided by the Data Mining group at the University of Vienna. This server is equipped with four NVIDIA H100 Tensor Core GPUs, 192 logical CPU cores, and approximately 2 TB of RAM. The system runs on an x86_64 architecture and uses the Miniforge distribution for environment management. Initial experiments were executed directly on the standalone Galadriel node. Subsequently, most training and evaluation jobs were submitted to a SLURM-managed cluster using sbatch, following Galadriel's integration into the shared infrastructure maintained jointly with the VDA and SEC groups. A dedicated Python virtual environment was configured on Galadriel and consistently used for all experiments.

4.2. Experimental Setup

This section outlines the main experimental variables investigated in the full fine-tuning of the OLMo 1B model, namely the number of training epochs, the effect of genre-related data distribution, and various temporal adaptation strategies. We systematically evaluate how these factors influence model performance, using perplexity on validation and test sets as the primary metric.

First, we identify the optimal number of training epochs needed to balance under- and overfitting. Next, we examine whether variations in data ordering and genre composition affect the model's learning stability and genre-specific performance. Finally, we introduce different temporal conditioning techniques and test their effectiveness in enabling the model to condition on temporal information.

Subsequent subsections describe the experimental design and report key findings for each of these variables.

[1] Repository: https://github.com/kseniadvorkina/OLMo_thesis

4.2.1. Effect of Training Epochs

Before evaluating different strategies for temporal adaptation, it was necessary to establish a stable fine-tuning baseline. One important aspect of this setup is determining the appropriate number of training epochs. While longer training might improve performance on the training set, it can also lead to overfitting, particularly in smaller or more homogeneous datasets. At the same time, undertraining may result in underfitting, where the model fails to fully capture the structure of the data. Thus, identifying an optimal training duration is essential for balancing generalisation and performance, and for ensuring that subsequent comparisons between adaptation strategies are meaningful.

During model development, we adopted an 80–10–10 split for training, validation and testing. To investigate effect of training epochs (current section) and data distribution (the following section), we fine-tuned OLMo 1B model using special tokens for temporal adaptation. Specifically, we introduced three new tokens representing historical year ranges: [1710–1780], [1780–1850] and [1850–1920]. These tokens were added to the tokenizer's vocabulary and to the model's embedding layer. During the tokenisation process, the corresponding token was prepended to each chunk of data to condition the model on the temporal context.

Our first experimental series focused on determining the optimal number of epochs for full fine-tuning of the model. We conducted comprehensive trials across 1, 2, 3, 4, 5 and 7 epochs while maintaining a consistent effective batch size of 256. To accommodate varying epoch counts, we adjusted three key parameters inversely: learning rates (decreased for longer training), warmup ratios (increased) and weight decay values (increased). The complete configuration details for these training experiments are documented in Supplementary Table S1. For testing consistency across all models, we maintained fixed evaluation parameters with a batch size of 1, as testing was performed without parallelism on a single GPU.

Results are presented in Table 1 and visualised in Figure 4. The results clearly indicate that 3 epochs of fine-tuning yields the optimal balance between model performance and generalisation. As shown in Table 1, validation perplexity reaches its lowest point (14.75) at 3 epochs, while training perplexity continues to decrease beyond this point (from 3.95 at 3 epochs to 3.86 at 5 epochs). This divergence between validation and test metrics suggests that while the model continues to learn the training data, its ability to generalise peaks at 3 epochs

before overfitting occurs. Therefore, we have chosen 3 epochs as a standard of fine-tuning for the base model.

Number of train epochs	1 epoch	2 epochs	3 epochs	4 epochs	5 epochs	7 epochs
Train Perplexity	20,4999	3,9934	3,9484	3,9046	3,8598	3,8959
Val perplexity	19,2255	14,7574	14,7494	14,7621	14,8041	14,7930

Table 1. Training results for the experiments on the number of training epochs. OLMo 1B, fine-tuned using special tokens as temporal embeddings.

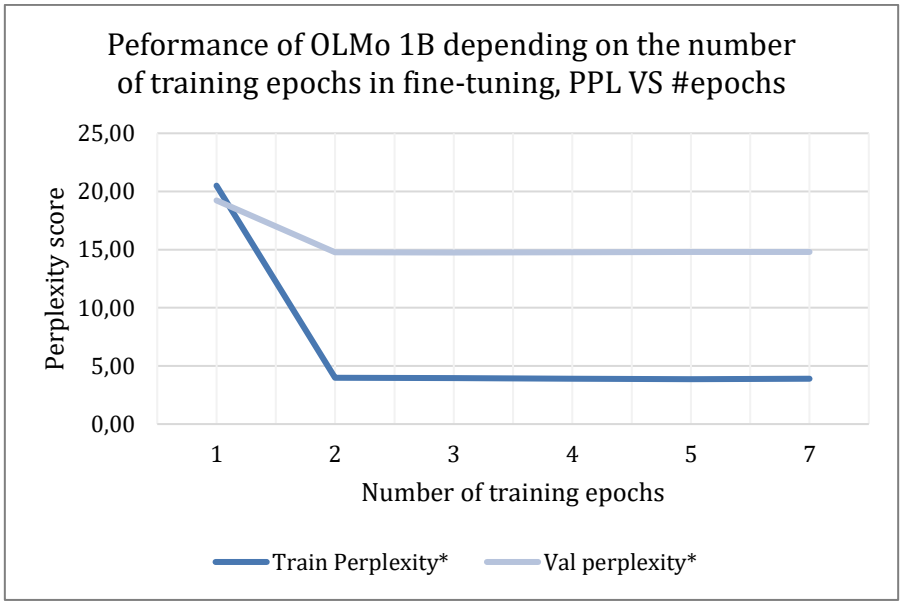

Figure 4. Training and validation perplexity after full fine-tuning of OLMo 1B with special tokens as temporal embeddings.

All the OLMo 1B models used in this research, both for full fine-tuning and MoE setup, will be adapted using 3 epochs as standard for the fine-tuning.

4.2.2. Impact of Genre-related Data Distribution

Another important aspect to check is whether the distribution of genres in the dataset influences model performance. Although the genre distribution is fixed across the training, validation, and test splits, it remains imbalanced overall.

To assess whether the model is learning consistently across different genres, we first tested the effect of data ordering. To this end, we fully fine-tuned five models using the outlined earlier setup of 3 epochs as training duration and special tokens for temporal adaptation. Each

model was adapted on a different shuffle of the same training data, specifically we used random seeds of 43, 123, 456, 789 and 101112 to shuffle the data. These values will be further used to distinguish the models in this subchapter.

Next, we calculated standard deviation of perplexity on the validation set across different models. As shown in Table 2, the model exhibits remarkable stability across different training data shuffles, with validation perplexity values clustering tightly around 14.75 (SD = 0.0037). This minimal variation suggests that the training process is robust to random seed variations in data ordering.

Random seed	**Perplexity on validation set**
42	14,758289
123	14,753099
456	14,756516
789	14,757515
101112	14,75151
Standard Deviation	0,003735357

Table 2. Perplexity on validation dataset across five models with different data shuffles during the training (fine-tuning).

To analyse the genre-related model performance, we evaluated each model separately on every genre, namely Drama, Narrative fiction, Narrative non-fiction, Treatise and Undefined (Table 3). Since no validation data for Undefined genre was available, we used test data. We then compared performance of each model across different genres, and calculated standard deviation, both for perplexity across different models, grouped by genres and for different genres, grouped by models.

The genre-specific analysis in Table 3 reveals significant performance stratification. Narrative non-fiction shows the most consistent performance across models (SD = 0.0023), while Treatise exhibits notably higher variability (SD = 0.0057). This pattern suggests that the model handles factual narrative content more reliably than the more structurally diverse Treatise genre. The Undefined category, evaluated on test data, demonstrates comparable stability (SD = 0.003) to other genres, providing no evidence of data leakage between splits.

Metric	**Perplexity on validation set**					
Model Seed	model 42	model 123	model 456	model 789	model 101112	**SD per genre**

Drama	10,375	10,376	10,373	10,37	10,367	**0,0035**
Narrative fiction	16,034	16,028	16,03	16,03	16,026	**0,003**
Narrative non-fiction	15,939	15,944	15,939	15,943	15,941	**0,0023**
Treatise	12,517	12,5089	12,52	12,523	12,513	**0,0057**
Undefined*	17,22	17,225	17,225	17,226	17,22	**0,003**
SD per model	**2,861**	**2,863**	**2,861**	**2,863**	**2,863**	–

*For validation on the Undefined genre, test data was used.

Table 3. Perplexity on validation subsets for different genres across five models with different data shuffles during the training (fine-tuning).

These findings are visualised in Figures 5 and 6. In both figures, the bar chart on the left shows model performance grouped either by model (Figure 5) or by genre (Figure 6). The box plots on the right show standard deviations corresponding to each group in the bar charts. Each data point in box plot on the right corresponds to SD within one group of bar charts on the left.

The between-model consistency in performance is particularly evident in Figure 5, where each model-group shows identical patterns of relative genre performance despite different training shuffles. The box plot confirms this stability, with all models showing similarly tight distributions of genre level SD values. Figure 6 further reinforces these findings, demonstrating that while absolute performance differs between genre-groups (bar chart), the within-genre consistency remains strong across all models (box plot).

The slightly higher variability observed for Treatise (SD = 0.0057) remains within a narrow range; even this maximum difference represents only a 0.05% variation in relative terms, suggesting it is not practically meaningful.

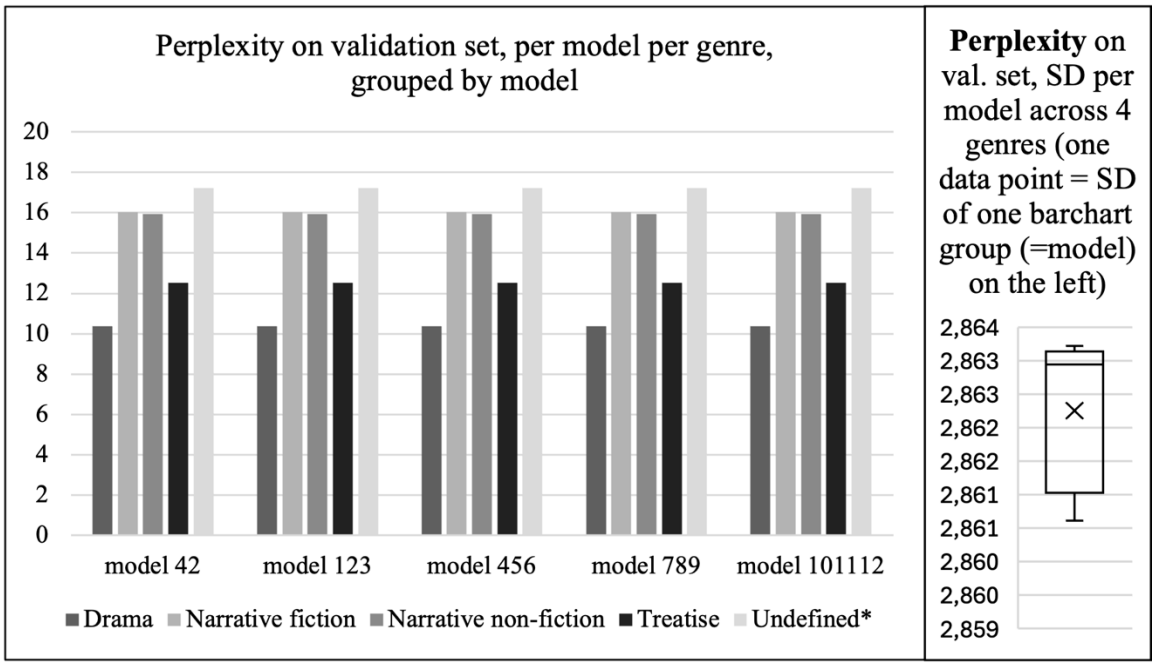

Figure 5. Model performance variability across genres. Left: Each bar represents perplexity per genre per model, grouped by model. Right: Box plot of standard deviations (one point per model's SD across PPL for different genres).

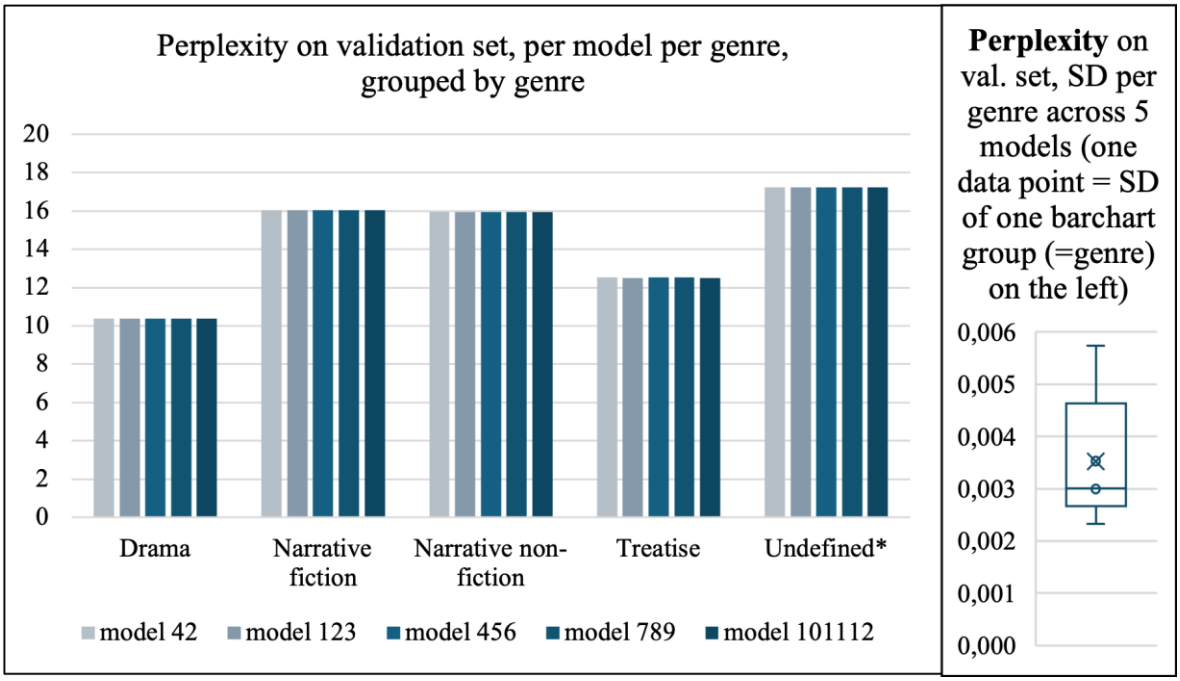

Figure 6. Model performance variability across genres. Left: Each bar represents perplexity per genre per model, grouped by genre. Right: Box plot of standard deviations (one point per genre's SD across PPL for different models).

Taken together, these results indicate that while genre does affect absolute performance levels, the model handles each genre consistently regardless of training data ordering. This suggests that the fine-tuning process is not disproportionately influenced by genre imbalance or data ordering, thereby reinforcing the suitability of the CLMET dataset and the current data split for the purposes of this research.

4.2.3. Temporal Adaptation Strategies

An essential aspect of diachronic language modelling is enabling the model to account for temporal variation in language use. To this end, different temporal conditioning techniques can be used to provide explicit time information during training and inference. Techniques used in this research vary in how they encode temporal context, ranging from discrete tokens to natural language prompts, and differ in how effectively they guide the model toward temporally coherent predictions. This section introduces the temporal conditioning strategies used, while the following results section evaluates their impact on model behaviour and investigates whether the model learns to associate linguistic patterns with specific time periods.

We begin by examining five conditioning approaches (Table 4) applied to the 1B-parameter OLMo model fully fine-tuned on the CLMET corpus. To illustrate the various temporal embedding strategies, we use a sample text CLMET3_1_1_67. This sample, beginning with the phrase "Preface Of The Author." and dated to the year 1776, serves as a running example to show how each temporal embedding is incorporated during the encoding preparation phase. During data preparation, the tokenised temporal prompt is prepended to each chunk of tokenised text, with an EOS (End Of Sentence) token added at the end. The length of each chunk is adjusted so that the combined sequence fits within the model's context window.

Model	Model Description	Embedding	Example Input (for CLMET3_1_1_67 text)
A	OLMo 1B (no fine-tuning)	–	–
B	Fine-tuned OLMo 1B	No temporal conditioning	*Preface Of The Author...*
C	Fine-tuned OLMo 1B + year ranges as special tokens	Special year range tokens like [1710–1780], [1780–1850] and [1850–1920] prepended	[1710–1780] *Preface Of The Author...*

D	Fine-tuned OLMo 1B + year range in natural language prompts	Year range in natural language	This is English text written between 1710 and 1780.\n\n*Preface Of The Author...*
E	Fine-tuned OLMo 1B + exact year in natural language prompts	Exact year in natural language	This text was written in the year 1776.\n\n*Preface Of The Author...*
F	Fine-tuned OLMo 1B + exact year (word) in natural language	Exact year as a single word in natural language	1776.\n\n*Preface Of The Author...*

Table 4. Temporal adaptation strategies for initial experiments.

In the initial stage, we fully fine-tuned the OLMo 1B model (with one billion parameters) on the entire CLMET corpus without incorporating any temporal embeddings (Model B) and compared its performance on the test set against the original, unfine-tuned OLMo 1B model (Model A). The second fine-tuned model (Model C) introduced temporal embeddings in the form of discrete special tokens, namely [1710–1780], [1780–1850] and [1850–1920], each added to the tokenizer's vocabulary and to the model's embedding layer and prepended to each chunk of data during the tokenization.

In the model D, the time periods were incorporated as natural language prompts: for our example, the sentence " This is English text written between 1710 and 1780." was placed at the beginning of each chunk of this text. The model E encoded temporal information at a more granular level by prepending the exact year of publication in a sentence, such as " This text was written in the year 1776". Lastly, model F incorporated exact year in minimal natural language prompt at the beginning of each text chunk.

To assess the effectiveness of the temporal conditioning strategies, we evaluated each model using deliberately incorrect temporal embeddings. For models trained with time ranges (Models C and D), inputs were modified by replacing the correct time period with a random incorrect one. For models that included explicit publication years in the prompt (Models E and F), these were randomly substituted with random years from other time periods. Model B was not tested with incorrect embeddings, as it does not use any explicit temporal conditioning.

4.3. Results: Full Fine-tuning with Temporal Conditioning

The full fine-tuning experiments investigate the extent to which different temporal embedding strategies can enhance the performance of a diachronic language model. We fine-tuned OLMo 1B using the hyperparameters listed in Supplementary Table S1 and temporal adaptation strategies B–F as described in Table 4. For each configuration, we computed perplexity and compared it against perplexity of the base model.

We also calculated perplexity at test, using false temporal embeddings for Models C-F to determine whether there was a substantial difference between using correct and incorrect embeddings, meaning if the model learned the intended temporal language representations. Similarly, for the test with deliberately false embeddings, perplexity was calculated.

The results are summarised in Table 5. Overall, the differences in perplexity across models B–F were minimal. The best test performance was achieved by Model B (no temporal embeddings), with a test perplexity of 17.5189, which is approximately 0.02 lower than other fine-tuned models and nearly 2 points lower than the unfine-tuned baseline (PPL = 19.4077). Validation performance was highly similar across models, with Model C exhibiting a marginally higher perplexity (PPL = 14.7494 comparing to ~14.72). Training perplexities were nearly identical, with Models B and C performing slightly better.

When incorrect temporal information was provided at test time, all tested models showed only negligible changes in perplexity. This suggests that the fine-tuned models did not learn to strongly condition on the provided temporal embeddings. The largest degradation was observed in Model C (PPL increase of ~0.1), though this is still minimal. These findings indicate that, in this setup, temporal embeddings had little impact on model performance and were not robustly internalised during training.

Model	**A**	**B**	**C**	**D**	**E**	**F**
Model Description	OLMo 1B (no fine-tuning)	FTed OLMo 1B	FTed OLMo 1B + year ranges as special tokens	FTed OLMo 1B + year ranges in natural language prompts	FTed OLMo 1B + exact year in natural language prompts	FTed OLMo 1B + exact year (word) in natural language
Train Perplexity	–	3.9491	3.9484	3.9516	3.9501	3.9528

Val perplexity	–	14.7179	14.7494	14.7178	14.719	14.7169
Test perplexity	19.4077	17.5189	17.5327	17.5409	17.5371	17.5488
Fake year range/ year perplexity (test)	–	–	17.6343	17.5429	17.5433	17.5674

Table 5. Initial experiments. OLMo 1B performance on the test set of CLMET with different temporal embedding strategies.

These findings motivated the development of more structurally constrained approaches, such as Mixture of Experts, where temporal specialisation is enforced through architectural modularity rather than auxiliary embeddings.

5. Mixture of Experts

Building on the results from Chapter 4, which found that temporal embeddings had minimal effect when applied uniformly across the model, this chapter investigates whether architectural modularity can offer a more effective form of temporal adaptation. Specifically, we implement a Mixture of Experts framework in which separate experts are fine-tuned on distinct temporal segments of the CLMET corpus. A soft gating mechanism is then used to dynamically weight expert outputs during inference and its results are compared to hard gating (argmax) and oracle (rule-based gate).

The chapter begins by introducing the model architecture and training procedure, followed by a description of the temporal experts and the design of the gating mechanism. The final section presents both qualitative and quantitative results, showing that while temporal gating does not yield major improvements in perplexity, it captures clear time-sensitive patterns in the gating distributions. These findings suggest that temporal specialisation is better learned through architectural separation than through embeddings alone.

5.1. Motivation and Initial Experiments

Our investigation of the Mixture of Experts architecture began with three independently fine-tuned OLMo 1B models. For this adaptation, we split training, validation and test datasets into temporal subsets (1710–1780, 1780–1850 and 1850–1920), fine-tuned one model for each time period without temporal encodings (Model B configuration), and then validated each model on all three temporal subsets.

This cross-period validation revealed significant performance stratification, as evidenced in Table 6. Each expert model achieved optimal performance on its designated time period, with the 1710–1780 model showing relatively stable performance across subsequent period (PPL +0.04 compared to its native period), while models from later periods exhibited more pronounced specialisation (approx. PPL +2 compared to its native period)

Year (data)\ Year(Model)	Model 1710–1780	Model 1780–1850	Model 1850–1920
Data 1710–1780	17.2181	18.2928	18.8545
Data 1780–1850	17.2553	16.3651	18.1274
Data 1850–1920	20.3878	18.8338	16.7124

Table 6. OLMo 1B, 3x experts (Model B) trained on different temporal splits, validation perplexity.

These expert models consistently outperformed the joint models with temporal embeddings (see Section 4.3), highlighting their stronger temporal specialisation. Subsequent experiments using explicit temporal conditioning (Models C and E; see Supplementary Tables S2 and S3) produced similarly strong specialisation effects.

These findings motivated the use of a Mixture of Experts approach, aiming to combine the strengths of independently specialised models within a unified architecture. In this context, architectural specialisation offers a complementary strategy to explicit conditioning, providing a robust foundation for modelling diachronic variation.

5.2. Architecture

Our Mixture of Experts architecture builds on prior work in domain-specialised and softly gated models, particularly Seo et al. [32] and Wang et al. [22]. Similar to Wang et al., we leverage expert specialisation guided by supervision from metadata; in their case, speaker identity labels used to supervise the selection of frozen wav2vec 2.0 experts for fake audio detection, and in ours, historical period labels used to guide temporal expert selection. From Armi et al. [25], we adopt a soft gating mechanism that computes a weighted sum of expert outputs, with gating probabilities derived from mean-pooled input representations. Unlike Seo et al.'s MoFE architecture [32], which performs sparse, token level routing to activate a subset of FFN blocks within a shared transformer backbone, our model activates all experts for every input and employs global, sequence-level gating.

Our contribution is to adapt these principles to the temporal domain by fine-tuning experts on distinct historical periods and supervising the gating network with historical metadata, thereby enabling dynamic expert selection based on inferred temporal characteristics of the input. We employ a frozen-expert, learned-gating, temporally specialised MoE trained with supervised gating using historical labels and using soft selection (weighted sum of logits) during inference.

5.2.1. Standard MoE Components

The architecture consists of two core modules: the frozen temporal experts and the learnable gating network. Below, we describe each in more detail.

We adapt a soft gating architecture similar to Armi et al. [25], in which the final prediction is computed as a weighted sum of expert outputs. Following prior work, we use mean-pooled token embeddings as input to the gating module, and a shallow MLP to produce expert selection logits.

The MoE architecture comprises three expert language models, each specialised on a distinct historical period. Let each expert be denoted (E_k) for ($k \in \{1, 2, 3\}$), where each (E_k) is an instance of the OLMo 1B model with parameters frozen during training of the gating mechanism.

Given an input sequence $(x = (x_1, x_2, \dots, x_T))$, the gating network computes a pooled embedding $\bar{\mathbf{h}} = \frac{1}{T}\sum_{t=1}^{T} \mathbf{h_t}$, where $\mathbf{h_t}$ is the embedding of token (x_t). This pooled embedding is passed through a two-layer feed-forward network with ReLU (Rectified Linear Unit) activation function, yielding logits $(z \in R^K)$, where each component (z_k) corresponds to expert (E_k):

$$\boldsymbol{z} = \boldsymbol{W_2} \cdot \mathrm{ReLU}\left(\boldsymbol{W_1}\bar{\boldsymbol{h}} + \boldsymbol{b_1}\right) + \boldsymbol{b_2},$$

Where $(\bar{\boldsymbol{h}} \in R^d)$ is the mean-pooled embedding, $(\boldsymbol{W_1} \in R^{512 \times d})$, $(\boldsymbol{b_1} \in R^{512})$, $(\boldsymbol{W_2} \in R^{K \times 512})$, $(\boldsymbol{b_2} \in R^K)$ and $(K = 3)$ is the number of experts.

Applying softmax function yields produces the gating probabilities

$$p_k = \frac{\exp(z_k)}{\sum_{j=1}^{K} \exp(z_j)}$$

During inference, the overall language model prediction is computed as the weighted sum of the expert logits:

$$\text{logits} = \sum_{k=1}^{K} p_k \cdot E_k(x),$$

where $(E_k(x) \in R^{T \times V})$ are the output logits of expert (E_k) over the vocabulary of size (V).

5.2.2. Temporal Specialisation in MoE

Unlike prior MoE work, which typically focuses on domain-specific specialisation, our contribution lies in adapting the MoE framework to the temporal domain. Specifically, each expert model is fine-tuned on texts from a distinct historical period: E_1 on texts from 1710–1780, E_2 on texts from 1780–1850 and E_3 on texts from 1850–1920. These expert models remain frozen during MoE training to preserve their pre-trained linguistic representations and temporal specialisations.

Training focuses exclusively on the gating network, which is responsible for dynamically weighting the contribution of each expert based on the input sequence. The gating network receives mean-pooled token embeddings derived from the shared input embedding layer of the experts and produces a categorical distribution over the experts via a softmax activation.

The gating network is supervised with ground-truth temporal labels $(y \in \{0,1,2\})$, indicating the correct expert corresponding to the input's historical period. The optimisation objective minimises the cross-entropy loss:

$$\mathcal{L}_{\text{gt}} = -\sum_{k=1}^{K} y_k \log p_k,$$

where (y_k) is the one-hot encoded true label.

Training employed the AdamW optimiser ($\eta = 10^{-5}$, $\beta_1 = 0.9$, $\beta_2 = 0.999$) over three epochs, with performance monitored through both gating accuracy and perplexity metrics.

For evaluation, language modelling performance is assessed by computing the cross-entropy loss and corresponding perplexity using the combined logits. Three inference variants were

evaluated: continuous soft gating, hard gating via $\arg\max_k p_k$ selection and an oracle using ground-truth temporal labels for test data.

5.3. MoE Results

This section analyses the performance of the Mixture of Experts architecture on the test text set. It begins by comparing three MoE variants employing different temporal adaptation strategies and gating mechanisms, evaluating their performance in terms of perplexity. Following this, the behaviour of the soft gating mechanism is examined through an analysis of gating weight distributions. Finally, the section presents a detailed performance analysis of the best-performing MoE configuration.

Our experiments with MoE architectures demonstrate three central insights. First, temporally specialised experts outperform non-specialised models when used in a MoE setup, particularly when conditioned on exact years via natural language prompts. Second, soft gating consistently outperforms hard (argmax) gating and often approaches the performance of oracle routing, making it a practical default choice. Third, while the middle temporal range (1780–1850) shows the most diffusion in expert selection, batch-level analyses confirm that the MoE can capture gradual transitions across time, especially when using natural language year conditioning.

5.3.1. All MoE

Given the limited temporal signal learned via embedding strategies, we explored a Mixture of Experts architecture to enforce stronger temporal specialisation. This decision was further motivated by promising cross-validation results obtained from temporally specialised expert models trained on the split CLMET corpus, as discussed in Section 5.1.

To evaluate whether combining temporally specialised models could improve language modelling performance, we constructed a full Mixture of Experts setup using fine-tuned models from previous experiments. For the expert sets, we selected three fine-tuned models that exhibited the best test-set perplexity in the initial experiments (see Table 5): Model B (which used no temporal embeddings), Model C (which incorporated year ranges as special tokens) and Model E (which embedded the exact year using natural language prompts). Each configuration was then fine-tuned separately on each of the three diachronic intervals (1710–1780, 1780–1850, 1850–1920), resulting in three sets of experts.

For each expert set, we tested three gating mechanisms as outlined in previous section. The first was a soft gating strategy, which assigns dynamic weights to each expert and averages their output accordingly. The second was a hard argmax gate, selecting the expert with the highest predicted weight for each input. The third was a rule-based oracle gate, which routes input texts to the correct expert based on ground-truth temporal labels and thus serves as a theoretical upper bound on MoE performance.

The results (Table 7) reveal several notable patterns. Across all gating mechanisms, the MoE model employing experts fine-tuned with natural language year prompts (Model E) consistently achieved the lowest perplexity, with a minimum of 16.9038 under oracle routing. This finding suggests that, for MoE, conditioning expert models on exact years via textual prompts yields stronger temporal differentiation than using no temporal information or relying on categorical tokens.

While oracle gating unsurprisingly achieved the best perplexity for most configurations, an interesting deviation is observed with the expert set based on Model B. Here, soft gating slightly outperforms the oracle gate. This result implies that, in the absence of explicit temporal embeddings, the probabilistic mixture of predictions from all three experts provides a more reliable estimate than deterministic routing to a single expert. It is possible that the soft gate in this case leverages complementary knowledge spread across the temporal spans, stabilising next token predictions.

On the other hand, the argmax gating mechanism consistently performed worst across all expert sets. The performance gap between argmax and soft gating, albeit numerically small (mostly within 0.03–0.05 perplexity points), remains consistent and supports the use of soft gating as a default strategy for balancing interpretability and stability.

Gate type	**Temporal embedding (expert' models' type)**		
	None (B)	**Special tokens (C)**	**Exact year in natural language (E)**
Soft gating	16.9687	16.993	16.9624
Argmax at Inference	17.0064	17.0723	16.9950
Rule-based (Oracle)	17.0032	16.9809	16.9038

Table 7. MoE (B, C, E), test PPL depending on the gate type.

Taken together, these results demonstrate that the MoE architecture improves upon single fine-tuned models when temporal specialisation is well-structured. The best-performing model, MoE (E) with soft gating, achieves a test perplexity of 16.9624, outperforming the best full fine-tuned model (Model B, test PPL of 17.5189) by over half a perplexity point. While this numerical improvement may seem modest, it reflects a more structurally grounded approach to temporal modelling, where temporal specialisation is enforced at the architectural level through expert-specific training on distinct historical periods.

To complement the perplexity-based evaluation, we analysed the outputs of the Mixture of Experts configurations by examining the distribution of expert predictions across the test set. Using the normalised gating weights as probabilistic indicators, we identified the top expert per prediction step and compared it against the ground truth time period labels (used only for analysis). At this stage of analysis, we averaged weights per texts, to account for the varying number of batches and re-normalised them.

First, we plotted distribution of average expert weights per MoE model per time period. For this, we created two sets of bar charts: grouped by MoE models (Figure 7, on the left) for high level comparison between the different time periods within each model and grouped by time periods (Figure 7, on the right) to compare how performance of different models varies in each time period. One vertically stacked bar in these visualisations represents one MoE model in one time period. Weights were first averaged per texts, then re-normalised within each text, and then averaged again for this visualisation.

In the earliest period, 1710–1780, all models favour the corresponding expert, with average weights of 0.628 in MoE (B), 0.601 in MoE (C) and 0.609 in MoE (E) (Supplementary Table S5). The differences are relatively small, though MoE (B) places slightly more emphasis on the correct expert. For the middle period, 1780–1850, the distribution becomes more balanced. Expert 1780–1850 receives the highest average weight in all models – 0.442 in MoE (B), 0.426 in MoE (C) and 0.444 in MoE (E), but both neighbouring experts also receive considerable weight. Finally, in the most recent period, 1850–1920, corresponding expert is dominant across all models, with average weights of 0.688 in MoE (B), 0.731 in MoE (C) and 0.704 in MoE (E). Here, MoE (C) shows the clearest preference for the correct expert.

We can see that while generally distributions are similar between the models, MoE (B) with no temporal embedding uses the earliest period the most, while the MoE (C) with special

tokens relies the most on the expert corresponding to the latest time range 1850–1920. MoE (E), which uses explicit natural language prompts, shows stronger expert separation in the early and middle periods. This illustrates how different forms of temporal input modulate the model's decision-making across time.

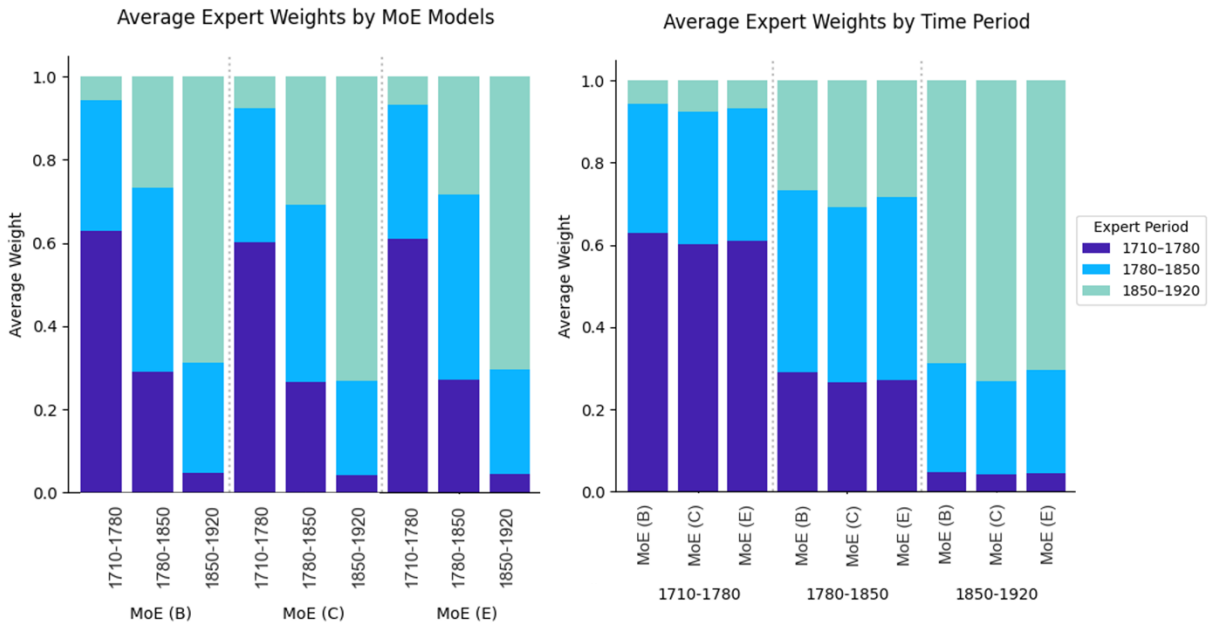

Figure 7. MoE (B, C, E), performance on test data. Average expert weights per unique text, grouped by the MoE (on the left) and true time ranges (on the right).

To explore more fine-grained patterns, Figure 8 plots the expert weights, still averaged per text, but now aligned with each text's exact publication year. Each subplot corresponds to one expert, and each line represents one MoE configuration. The test dataset comprises 34 texts spanning 210 years, meaning that not every year is represented. This results in sharp transitions between points, connected by linear interpolation.

The experts for the earliest and latest periods generally follow expected temporal trends. The 1710–1780 expert starts with higher weights and steadily declines across time, whereas the 1850–1920 expert starts with low weights and gradually increases. The trajectory of the 1780–1850 expert is less consistent, with peaks occurring both at the beginning and in the middle of the time range.

Several local patterns are also evident in the plots. Around the year 1735, the 1710–1780 and 1780–1850 experts show mirrored behaviour, forming an M-like shape in the former and a W-like shape in the latter. In 1840, there is a clear drop in the middle expert's weight and a

corresponding peak in the earliest expert. Towards the end of the timeline, the two experts corresponding to the younger time periods diverge in opposite directions.

Across the three configurations, the middle expert shows the most variation. MoE (C) consistently assigns it lower weights compared to the other models, while for the 1850–1920 expert, MoE (C) assigns the highest weights on average.

Figure 8. MoE (B, C, E), performance on test data. Average expert weights per unique text against the exact year the text was printed.

5.3.2. MoE (E)

To obtain more granular insights into the behaviour of the gating mechanism, we shift our focus from aggregated text-level metrics to those computed at the batch level. While text-level averaging provides a higher-level view of model behaviour, it can obscure finer

variations within individual texts, especially given that each test text is divided into multiple batches during preprocessing to fit the model's context window. In our case, 34 test texts result in 2044 batches. By examining the gating weights at the batch level, we can assess the consistency of temporal patterns across finer-grained input segments. This section focuses on results obtained by best-performing Mixture-of-Experts model, MoE (E), on the test set.

We first visualise the boxplots showing the distribution of expert weights for each true time period (Figure 9). Each group of boxplots corresponds to one time period, with each box corresponding to the weights of one expert in that period.

In all three periods, the expert corresponding to the true time range receives the highest median weight. However, expert activation is not exclusive, and considerable variation exists within each group. Notably, in the earliest period (1710–1780), while Expert 1710–1780 has the highest median and a relatively concentrated interquartile range (IQR), expert 1850–1920 displays numerous outliers. A similar pattern is visible in the latest period (1850–1920), where expert 1710–1780 exhibits frequent out-of-range values, despite being temporally distant.

Interestingly, the middle period (1780–1850) shows the most diffuse pattern: while expert 1780–1850 holds the highest median weight, the widest IQR is observed for two other experts, suggesting frequent overlap and uncertainty in expert selection during this transitional phase. This aligns with earlier observations of increased ambiguity in the middle period.

To complement the distribution analysis, we also report the average gating weights computed at the batch level (Supplementary Table S4). While the general patterns remain consistent, the average weights per expert are slightly more polarised compared to the text-level aggregation: the expert corresponding to the true period is used more frequently in all cases. Notably, the expert for the middle period (1780–1850) is frequently used across all time ranges, receiving nearly 29% of the weight in the earliest period and 38% in the latest. This pattern reinforces earlier observations of temporal overlap and diffusion between time-specific experts.

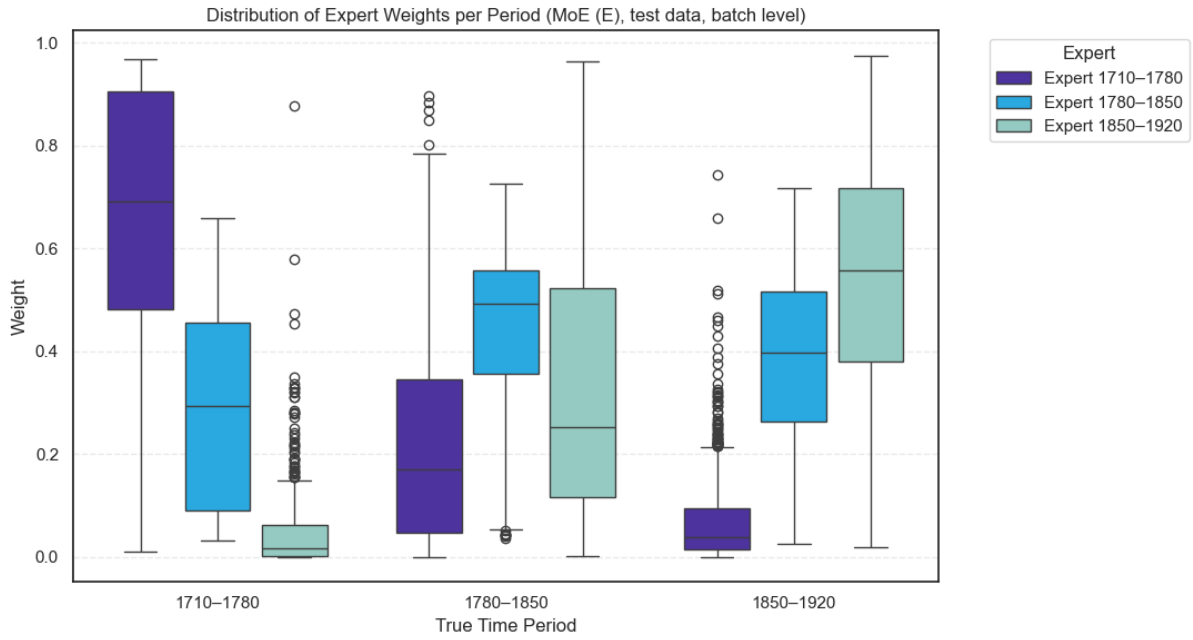

Figure 9. MoE (E), performance on test data. Distribution of expert weights per batch, grouped by the true time ranges.

Next, we assessed the confusion matrices based on the dominant expert predictions, using both aggregation strategies: predictions averaged per text (Figure 10, on the left) and predictions over all batches (Figure 10, on the right). In these matrices, rows correspond to the true time periods, and columns indicate the predicted time period based on the expert with the highest gating weight. Thus, correct predictions lie on the main diagonal, while off-diagonal entries represent misclassifications.

The confusion matrix aggregated per text (on the left) shows relatively clean classification performance, with only a few misclassified texts. All six texts from the most recent period (1850–1920) are classified correctly, while the middle period (1780–1850) exhibits mild confusion, with two misclassifications toward earlier and later periods respectively. The earliest period (1710–1780) shows moderate confusion with seven texts misclassified as belonging to the subsequent time period.

In contrast, the batch-level confusion matrix (on the right) reveals more nuanced dynamics. While the majority of batches are correctly classified within their true period, significant overlap is visible between adjacent periods, especially in the middle range. For instance, 211 batches from the 1780–1850 period are assigned to the most recent expert, and 287 batches from the latest period (1850–1920) are assigned to the middle expert. This further supports

earlier findings that the expert for 1780–1850 generalises broadly across the temporal spectrum and is frequently selected even outside its designated range.

The comparison between the two matrices suggests that temporal boundaries are more sharply defined at the document level, while batch-level predictions exhibit greater fluidity, likely due to localised stylistic or lexical variation within texts.

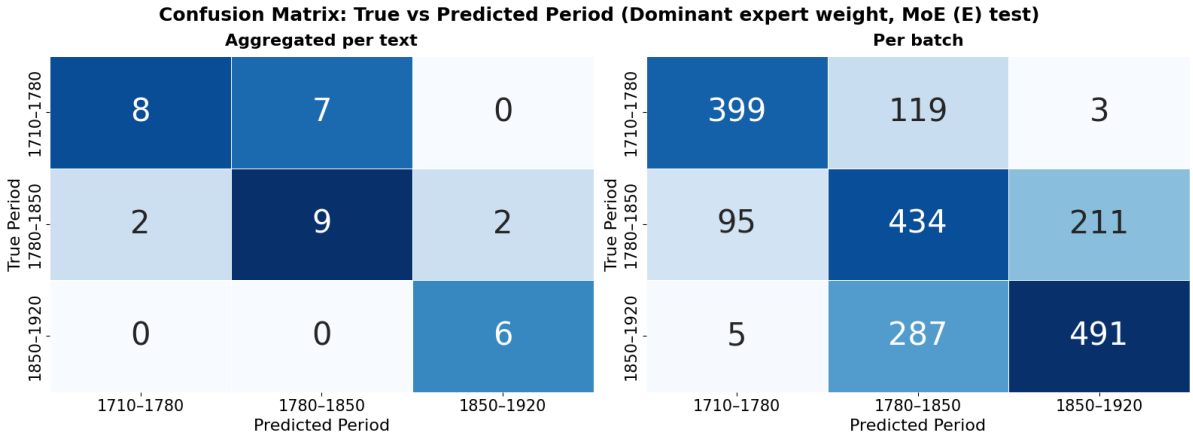

Figure 10. MoE (E), confusion matrix: true time period versus time period that corresponds to the experts with the highest weight at the gate (dominant expert); on the left: weights averaged per text and re-normalised; on the right: per batch reported weights.

In summary, the MoE approach improves temporal modelling by integrating fine-tuned experts for distinct historical intervals. The best configuration – MoE (E) with soft gating – achieves the lowest perplexity and demonstrates robust alignment between expert activation and the true publication year. Importantly, results from both text- and batch-level analyses highlight the usefulness of natural language prompts for encoding temporal context and reveal transitional ambiguity in the middle historical range. These findings support the use of MoE as an effective and interpretable strategy for diachronic language modelling.

6. Year Prediction

The idea behind this chapter is to repurpose the Mixture-of-Experts architecture from a previous chapter for a novel task: predicting the publication year of historical texts. While the model has a generative nature, and was never trained to output years directly, we leverage the soft gating weights as probabilistic signals of temporal proximity. This allows us to decode a hidden sense of time from the model, turning expert activations into interpretable historical predictions. The approach is not only resource-efficient, but also accurate: with no explicit

supervision, our MoE models achieve a mean absolute error of just over 24 years on a 210-year timespan, representing approximately a 66% improvement compared to the expected error of a random predictor. This demonstrates that temporal patterns in language are learnable and recoverable and opens the door to using large language models for historical inference and diachronic stylistic analysis.

Our approach focuses on text-level year prediction, using the gating weights as probabilistic indicators of a text's historical origin, and interpretating them with a linear combination of temporal intervals and gating weights. We further adjust prediction strategy by tuning the confidence threshold used in linear interpolation based on gating weights. This confidence threshold determines when the prediction should rely on the dominant expert alone and when on the two top experts. Lastly, we analyse how splitting text into batches during pre-processing influences the year prediction and complement this analysis with computation of both absolute and signed errors.

6.1. Methodology

To apply the Mixture of Experts framework beyond expert weighting, we leverage the gating network's soft weights to estimate the year of origin for each input text. Since each expert is specialised to a distinct historical period, the gating weights serve as probabilistic indicators of temporal proximity. The gating network outputs a vector of non-negative weights $\mathbf{w} = [w_0, w_1, w_2]$), representing the relative importance of each expert for the given input. These weights are normalised via softmax activation which are normalised via softmax activation to produce the gating probabilities $\boldsymbol{p} = [p_0, p_1, p_2]$ corresponding to the experts (E_0, E_1, E_2), trained on the intervals 1710–1780, 1780–1850 and 1850–1920, respectively.

We interpret these probabilities as reflecting the model's belief over temporal proximity and develop a heuristic for deriving a continuous year prediction from them. The year prediction process proceeds as follows:

(1) We first identify the two experts with the highest gating probabilities (p_a, p_b) and their corresponding intervals $([l_a, r_a])$ and $([l_b, r_b])$.

(2) We then check if the selected intervals are adjacent i.e., $(\max(l_a, l_b) = \min(r_a, r_b))$,

(3) and calculate the confidence score as ($c = p_a + p_b$)

(4) We then proceed with the year prediction. We consider three cases for the year calculation:

(case X) If the selected intervals are adjacent and confidence score ($c = p_a + p_b \geq \tau$) is over or equal to the confidence threshold τ, we consider the prediction to be locally reliable and proceed with a weighted interpolation within the union of the two adjacent intervals:

$$\hat{y} = \begin{cases} r_a + 70 * \dfrac{p_b - p_a}{p_a + p_b}, if\ l_a < l_b \\ r_b + 70 * \dfrac{p_a - p_b}{p_a + p_b}, otherwise \end{cases},$$

where 70 is the length of each interval.

(case Y) If the confidence threshold is met, but the two top intervals are not adjacent, we predict the year by shifting the middle of the interval towards the most dominant weight, as in our case for the intervals to not be adjacent, they will have to be 1710–1780 and 1850–1920.

$$\hat{y} = 1815 + 105 * \frac{p_2 - p_0}{p_0 + p_2},$$

Where 105 = 70 + 35 ensures that the prediction can reach either end of the time range, should the dominant interval have a probability close to 1.

(case Z) Otherwise, we predict based on the expert with the highest weight p_{max} and shift within its interval according to the relative weights of the other two experts:

$$\hat{y} = \begin{cases} 1815 + 35 * (p_2 - p_0), if\ p_1 = p_{max} \\ 1920 - 70 * (p_0 + p_1), if\ p_2 = p_{max} \\ 1710 + 70 * (p_1 + p_2), if\ p_0 = p_{max} \end{cases}$$

Here, 1815 is the midpoint of the middle interval, and 35 is half its length, ensuring the prediction remains within the interval. Note that probabilities

within each formula are not re-normalised, to preserve the relative dominance of the most confident expert.

To tune the confidence threshold τ, we used the validation set predictions of the MoE model. Threshold values ranging from 0.70 to 1.00 (in increments of 0.01) were evaluated. For each threshold setting, we recorded the number of predictions falling into Cases X, Y and Z, and computed the corresponding Mean Absolute Error (MAE) in predicted years, by comparing predicted year, rounded to integer, to the actual printed year. Prior to year estimation, the expert probabilities were first averaged across all batches representing the same input text. These averaged scores were then re-normalised to ensure they summed to one, after which the prediction heuristic was applied as described above.

6.2. Confidence Threshold Tuning

The next step in our analysis was to use the weights assigned by the gating mechanism to predict the year of publication. This analysis focuses on predicting the year in which a text was first printed. As before, the gate weights were averaged per text and re-normalised. We focus here on the results of the best-performing mixture-of-experts model, MoE (E), which uses the exact year in the natural language embedding. The findings are presented alongside results from MoE (B) and MoE (C) for comparison.

We began by tuning the confidence threshold used for prediction. This threshold determines how many predictions are made using a weighted combination of the two top experts (case X) and how many rely on the dominant expert (case Z). The validation dataset consisted of CLMET text IDs, text-level average perplexity, within-text perplexity standard deviation (between batches), the three gate weights and the ground truth year of publication. We applied a function to calculate, for each confidence threshold value between 0.70 and 1.00 (in steps of 0.01), the number of cases falling into categories X, Y and Z, the overall Mean Absolute Error and the MAE within each case group.

Results are presented in Table 8 for MoE (E) and in Supplementary Tables S6 and S7 for MoE (B) and MoE (C), respectively. Case Y was not observed in any of the settings, meaning that there was no instance where the sum of the two highest gate weights met the confidence threshold while the corresponding temporal intervals were non-adjacent. Meaning, there was never a situation where [1710–1780] and [1850–1920] were the top two intervals whose combined weight surpassed the threshold.

For MoE (E), the lowest overall MAE (18.54) was achieved in the threshold range of 0.70–0.74, where all predictions were made using a weighted combination of the top two experts (case X). As the threshold increased, more predictions shifted into case Z. While the MAE within case Z initially decreased (e.g. 31.42 at threshold 0.99), the overall MAE increased due to the reduction in high-performing X-type predictions. This reflects a trade-off between confident but less flexible predictions (case Z) and more accurate but blended ones (case X).

Confidence threshold	Mean Absolute Error	#X cases	#Y cases	#Z cases	X cases MAE	Z cases MAE
0.7	18.54	28	0	0	18.54	–
0.71	18.54	28	0	0	18.54	–
0.72	18.54	28	0	0	18.54	–
0.73	18.54	28	0	0	18.54	–
0.74	18.54	28	0	0	18.54	–
0.75	19.32	27	0	1	17.81	60
0.76	19.32	27	0	1	17.81	60
0.77	19.32	27	0	1	17.81	60
0.78	19.68	26	0	2	17.04	54
0.79	19.18	24	0	4	16.58	34.75
0.8	19.18	24	0	4	16.58	34.75
0.81	19.18	24	0	4	16.58	34.75
0.82	19.18	24	0	4	16.58	34.75
0.83	19.18	24	0	4	16.58	34.75
0.84	19.96	23	0	5	15.57	40.2
0.85	19.96	23	0	5	15.57	40.2
0.86	19.96	23	0	5	15.57	40.2
0.87	19.96	23	0	5	15.57	40.2
0.88	20.93	22	0	6	16.27	38
0.89	21.11	21	0	7	16.67	34.43
0.9	21.11	21	0	7	16.67	34.43
0.91	21.11	21	0	7	16.67	34.43
0.92	21.86	20	0	8	16.07	34.75
0.93	21.89	18	0	10	13.89	36.3
0.94	22.86	17	0	11	13.65	37.09
0.95	22.86	17	0	11	13.65	37.09
0.96	22.86	17	0	11	13.65	37.09
0.97	22.86	17	0	11	13.65	37.09
0.98	23.86	14	0	14	13.36	34.36
0.99	25.86	9	0	19	14.11	31.42
1	27.39	0	0	28		27.39

Table 8. Mean absolute error, number of cases (X, Y, Z) and mean absolute error per case for the confidence threshold based on the gate weights of MoE (E) on the validation set. Column with absolute error for cases Y is dropped, as no such cases were present.

A similar pattern was observed for MoE (B) (Supplementary Table S6). The minimum MAE (18.43) was again reached when all predictions were classified as case X (thresholds 0.70–0.77). Once case Z predictions began to appear, they were associated with substantially higher errors (e.g. MAE of 59 at threshold 0.78), increasing the overall MAE. Notably, MoE (B) showed a slight delay in the emergence of case Z predictions compared to MoE (E), with case Z not appearing until threshold 0.78.

In MoE (C) (Supplementary Table S7), the lowest MAE (18.75) was reached slightly earlier at a threshold of 0.70, but the model transitioned to case Z even faster than MoE (B), with Z-type predictions appearing already at 0.73. The case Z errors were again large (e.g. MAE of 61) and the resulting overall MAE rose more quickly than in the other models.

Across all three models, no Y-type cases were observed, suggesting that the gating network consistently learned to assign high weights to temporally adjacent experts when uncertainty arose.

Overall, these results highlight that lower thresholds where predictions are made via a weighted combination of two adjacent experts tend to yield the best dating accuracy. As thresholds rise and the model commits to a single dominant expert, performance degrades, especially due to occasional large prediction errors. Among the three variants, MoE (E) performed best in terms of both minimum MAE and stability across thresholds, further validating the benefit of conditioning on the exact year in the natural language embedding.

Taken together, the best confidence threshold (CT) in each model corresponds to the lowest value at which all predictions fall into case X, i.e., where the top-2 experts are adjacent. Among these, we select 0.74 over 0.70, as it imposes a slightly stricter condition and provides greater confidence in the expert selection, with only a minimal increase in error. However, since all optimal results occurred under this minimax condition, where predictions remained entirely within case X, we decided to discard computation for the case Z (interpolation across non-adjacent experts). Case Y is retained for potential use when the top two dominant experts are non-adjacent. This strategy will be used for year prediction on the test set in the following chapters.

6.3. Year Prediction Results

This chapter examines the effectiveness of three Mixture of Experts (MoE) language models in predicting the year of historical English texts from 1710 to 1920. It presents a comprehensive quantitative evaluation of overall prediction accuracy, temporal variation in errors, and model behaviour across distinct time periods. The analysis further explores error distributions, potential temporal biases, and the impact of text length on prediction performance. Furthrmore, both positive and negative outlier texts are identified and analused qualitatively. Additionally, a detailed investigation of a specific MoE configuration highlights the relationship between model perplexity and prediction error at both the individual text and batch levels.

This chapter shows that explicit temporal embeddings have limited impact on overall year prediction accuracy, though different embedding strategies can influence performance during specific time periods. The middle historical period (1780–1850) is associated with the highest prediction errors, indicating particular difficulties in modelling texts from this period. Additionally, batch-level analysis reveals that text length does not directly affect prediction accuracy, but longer texts tend to produce greater variability in perplexity across batches. Prediction errors are fairly consistent across the timeline, with no systematic bias in their direction or magnitude. A key finding is the models' strong year prediction performance, achieving a Mean Absolute Error of approximately 24 years, representing an improvement of roughly 66% compared to a baseline random prediction.

6.3.1. All MoE

In this section, we analyse and compare the year prediction results across the three Mixture of Experts configurations.

Year predictions were generated using the updated prediction function, incorporating only cases X and Y, though no case Y instances were present in the test set across any of the models. This is consistent with patterns observed in the validation data. Gating weights were first aggregated per text prior to computing predictions.

The overall Mean Absolute Error in predicted publication year was 24.206 years for MoE (B), 23.912 years for MoE (C) and 24.029 years for MoE (E), suggesting comparable performance across the three configurations when averaged across the full time span. To put

this MAE results into perspective, if we were to predict a year randomly on the 1710–1920 time span, the Mean Absolute Error will be 70 years[2], making out result of 24 year an improvement of approximately 66%.

To understand the temporal sensitivity of the models, we visualised the MAE per historical period for each MoE (Figure 11). The resulting pattern shows a consistent trend across all three models: year predictions are least accurate for the middle period (1780–1850), while the earliest (1710–1780) and latest (1850–1920) periods exhibit smaller errors. This finding aligns with earlier observations (e.g. Figure 7), where the middle period also showed greater uncertainty in expert assignment.

The lowest MAE observed is 17.33 for MoE (C) during the 1850–1920 period, indicating that the use of special token embeddings may facilitate more temporally precise predictions for later texts. Conversely, all three models appear to struggle with the middle period, where MAE exceeds 31 years regardless of configuration.

These results suggest that although the presence or absence of explicit temporal encoding does not drastically affect overall prediction accuracy, the choice of temporal embedding strategy may have localised effects on performance. MoE (C), in particular, appears to benefit from its token-based temporal representation in later texts, while MoE (E), which had the exact year embedded in the prompt during training, does not necessarily outperform the others.

---

[2] Let $Y_{\text{true}}, Y_{\text{pred}} \sim \mathcal{U}(a, b)$ be two independent random variables drawn from a uniform distribution over the interval $[a, b] = [1710, 1920]$. Under the assumption of no prior information, random guess is equivalent to sampling predicted year Y_{pred} from the same distribution as true publication year Y_{true}. The expected absolute error of such guess is:

$$E\left[\left|Y_{\text{true}} - Y_{\text{pred}}\right|\right] = \frac{b-a}{3} = \frac{1920-1710}{3} = 70.$$

This serves as a baseline for evaluating the effectiveness of our model's year prediction.

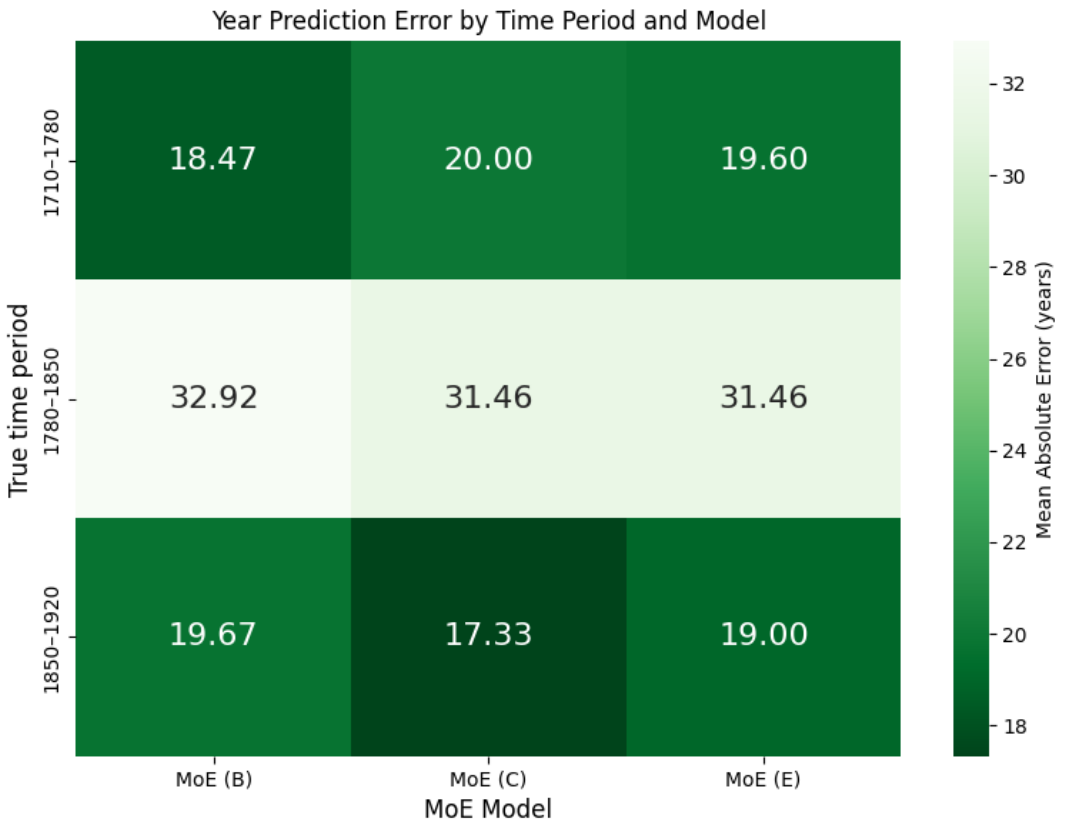

Figure 11. MoE (B, C, E), confusion matrix: MAE (years) per time period per MoE.

Next, we examined the absolute year prediction error of each model with respect to individual texts across the three historical periods (Figure 12). This visualisation presents the distribution of absolute errors per model and time range, using a swarmplot (scatterplot with points adjusted to be non-overlapping) layout that allows for precise comparisons between models within each temporal group.

Each point represents a single prediction and is coloured according to the model that generated it: MoE (B), MoE (C), or MoE (E). Model predictions are grouped by time interval along x-axis. Within each group, the positions of individual points are slightly adjusted along the horizontal axis to reduce overlap, although this jitter does not carry interpretive meaning. The y-axis shows the absolute error in years. A horizontal red line indicates the mean absolute error for each temporal group.

The chart reveals that MoE (B) and MoE (E) tend to follow similar patterns, often clustering near the MAE line, particularly in the periods 1710–1780 and 1850–1920. In contrast, MoE (C) shows more deviation, occasionally producing larger or smaller errors on the same texts.

In the middle period the predictions from all three models are more uniformly spread, with less distinct clustering and greater vertical dispersion, suggesting more variable performance and greater uncertainty in this temporal range. This aligns with the confusion matrix results and highlights a region of temporal ambiguity where textual features may be less distinctive or more transitional.

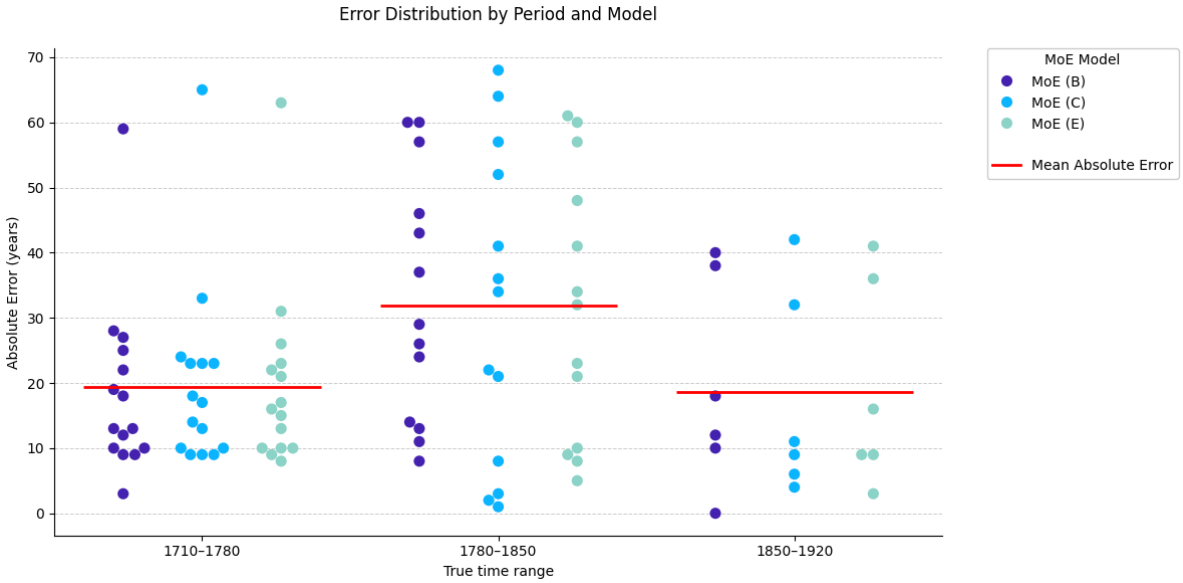

Figure 12. Absolute error of year prediction per text for MoE (B, C, E), grouped by time period.

To further examine the distribution of prediction errors across time, we visualised absolute prediction errors per text against their corresponding actual years in a scatterplot (Figure 13). Each point represents one text, coloured by the MoE model used and shaped by its true temporal label (one of three time periods). To highlight broader trends, we overlaid linear regression lines for each model, with dashed lines indicating the direction and strength of error change over time. Details of the year prediction for every text in the test dataset can be found in the Supplementary Table S8.

The figure demonstrates that while prediction errors fluctuate over the 210-year timespan, none of the models exhibit a strong increasing or decreasing trend. This suggests that all three MoE configurations achieve relatively stable predictive performance across the historical range.

MoE (E), which was provided with natural language prompts containing the target year, yields the flattest regression line, indicating highly uniform performance. MoE (B) shows slightly lower error in earlier periods, with a gradual increase towards the end of the timeline.

MoE (C) largely mirrors this trend. All three regression lines intercept around the year 1790, with MoE (C) and (B) switching the relative positions to (E).

Consistent with earlier observations, the middle period (1780–1850) is associated with the highest overall prediction errors. The most pronounced outlier appears in the earliest period: a single test text exhibits an absolute error of approximately 60 years across all three models. This text, Treatise “Human Nature and Other Sermons” by Joseph Butler (CLMET3_1_1_10), was printed in 1726 but consistently misclassified into the middle period by all MoEs, with predicted years ranging from 1785 to 1791. One possible explanation is the author’s abstract philosophical tone and discursive structure, which resemble the Enlightenment-era prose more common in the late 18th century.

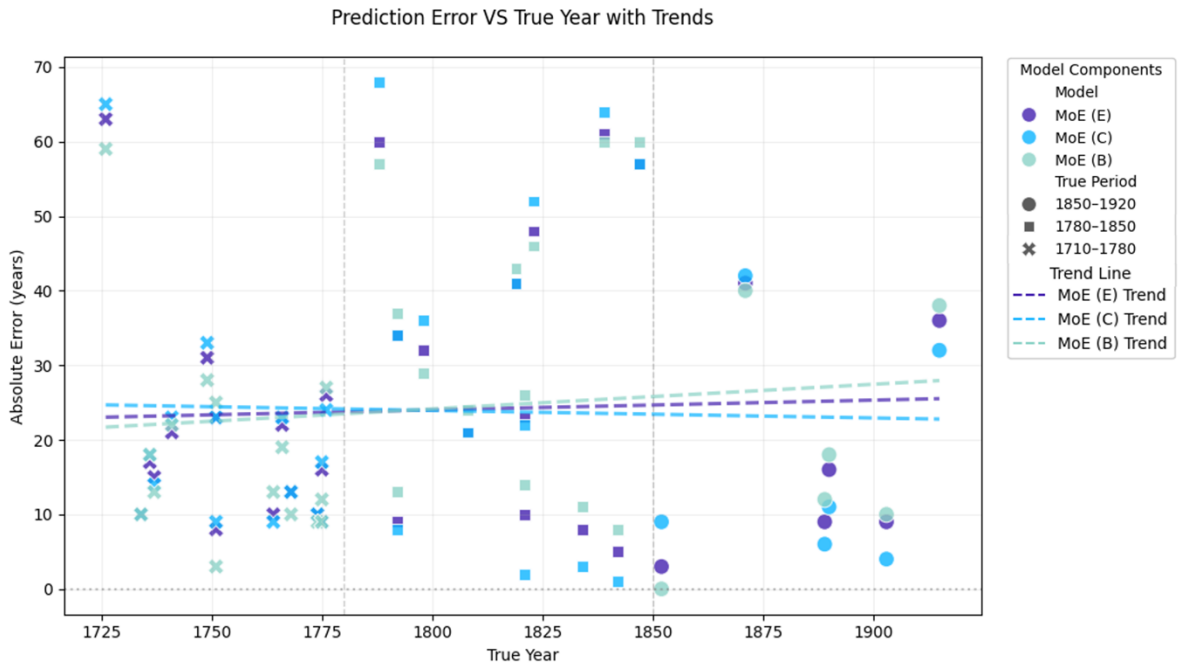

Figure 13. Year prediction of MoE (B, C, E) versus absolute error per text, with trend lines.

Three additional texts exhibiting high absolute errors are all situated within the middle period. These include Treatise “An Essay on the Principle of Population” by Thomas Malthus (CLMET3_1_2_116), misclassified approximately 57–68 years into the future; Narrative non-fiction “Reminiscences of Samuel Taylor Coleridge and Robert Southey” by Joseph Cottle (CLMET3_1_2_120), misclassified around 60 years into the past; and “The Voyage of the Beagle” by Charles Darwin (CLMET3_1_2_171), which was placed nearly 60 years too late by all models. The latter text, although unclassified at the top level, was assigned the

subgenre "travel/treatise", suggesting that thematic or stylistic overlaps with later scientific prose might have contributed to this misalignment.

To further examine the distribution and direction of prediction errors across all periods, we visualised the relationship between the predicted and true years of each test instance (Figure 14). Each point in the plot represents a single text, plotted by its actual publication year (x-axis) and the predicted year output by one of the three models (y-axis). Points are colour-coded according to the model and shaped according to the true time period, providing a comparative view of model behaviour both within and across periods.

A diagonal reference line marks the space of perfect prediction, where the predicted year equals the actual year. Points that fall close to this line indicate high temporal accuracy, whereas deviations from the line reflect varying degrees of misclassification. Dashed horizontal and vertical markers denote the period boundaries at the years 1780 and 1850, enabling clearer identification of cross-period misalignments.

The scatterplot reveals a clear directional bias in the models' errors. For texts from 1710–1780, most points lie below the y = x line, indicating underestimation of publication year. In contrast, texts from 1850–1920 are predominantly plotted above the line, indicating overestimation. Despite these systematic shifts, the bulk of predictions remain close to perfect alignment, with few extreme outliers.

Model agreement is high, as most individual points cluster closely across models. A notable exception occurs for Narrative fiction "The Last Days of Pompeii" by Edward George Bulwer-Lytton (CLMET3_1_2_162). Here, MoE (C) predicts almost the exact year (1831 predicted vs 1834 true), whereas MoE (B) and MoE (E) predict 1823 and 1826 respectively. This divergence suggests that the token-based gating in MoE (C) can, in certain cases, capture temporal cues more precisely than the alternative strategies.

The other two predictions hitting near-true year are Narrative non-fiction "The Bible in Spain" by George Borrow (CLMET3_1_2_164) and "Chambers's Edinburgh journal, no. 418-462" (CLMET3_1_3_332) with unknown author and unclassified genre. Both situated close to the 1850 year boundary, on the either side. George Borrow's text printed date (1842) was most closely predicted by MoE (C) (1843) and slightly underscored by MoE (E) and MoE (B) (1837 and 1834 respectively). As for the Chambers's Edinburgh journal, it was

predicted exactly by MoE (B) (1852) and very closely matched by MoE (E) (1855), followed by MoE (C) (1861). (Supplementary Table S8).

The case with Chambers's Edinburgh Journal is particularly interesting, as this is the largest text in the test dataset, spanning 1,116,620 words, more than four times longer than the next longest text. This text was split into 617 batches during pre-processing, which makes the precise accuracy of the year prediction, based on the averaged and re-normalised gate weights, especially noteworthy.

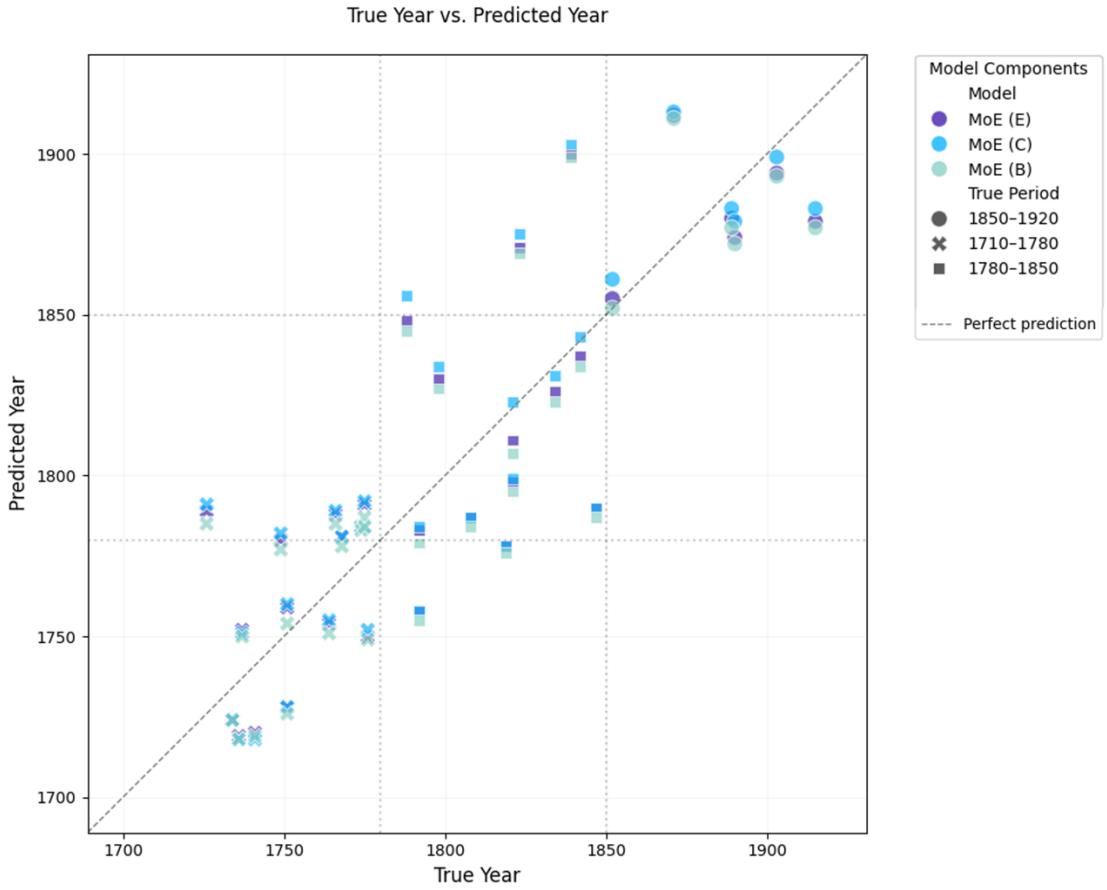

Figure 14. Predicted VS True year, MoE (B, C, E), with perfect year prediction line.

One final aspect we examined was the relationship between text length (the number of batches it was split into during pre-processing) and year prediction accuracy. To this end, texts were grouped into quartiles based on batch count, and the predicted versus true publication year was visualised separately for each quartile (Figure 15). Each subplot reflects a different range of text lengths, from the shortest (Q1) to the longest (Q4), spanning across the entire time range.

Seventy-five percent of batch sizes fall under the length 54 (Q1-Q3), although three texts far exceed 100: CLMET3_1_2_164 (143 batches), CLMET3_1_2_171 (134 batches) and CLMET3_1_3_332 (617 batches). Interestingly, texts 164 and 332 were just discussed as producing near-perfect year predictions in the previous paragraph. Furthermore, all top five texts in terms of largest batch sizes (101 to 617) are from a relatively narrow time period: 1834–1852.

Across all quartiles, the majority of predictions remain close to the diagonal line of perfect prediction, suggesting that text length does not systematically distort model output. However, some tendencies emerge: in the lower-middle length texts (Q2), prediction spread is more pronounced, and there is a visible number of deviations from the trend line. Conversely, upper-middle length texts (Q3) exhibit tighter clustering around the true year, particularly for MoE (C). Q4 appears to have the largest number of near-perfect predictions (points on the perfect prediction line), although this quartile also contains the greatest number of visible outliers.

Overall, it appears that text length does not have a direct effect on year prediction accuracy.

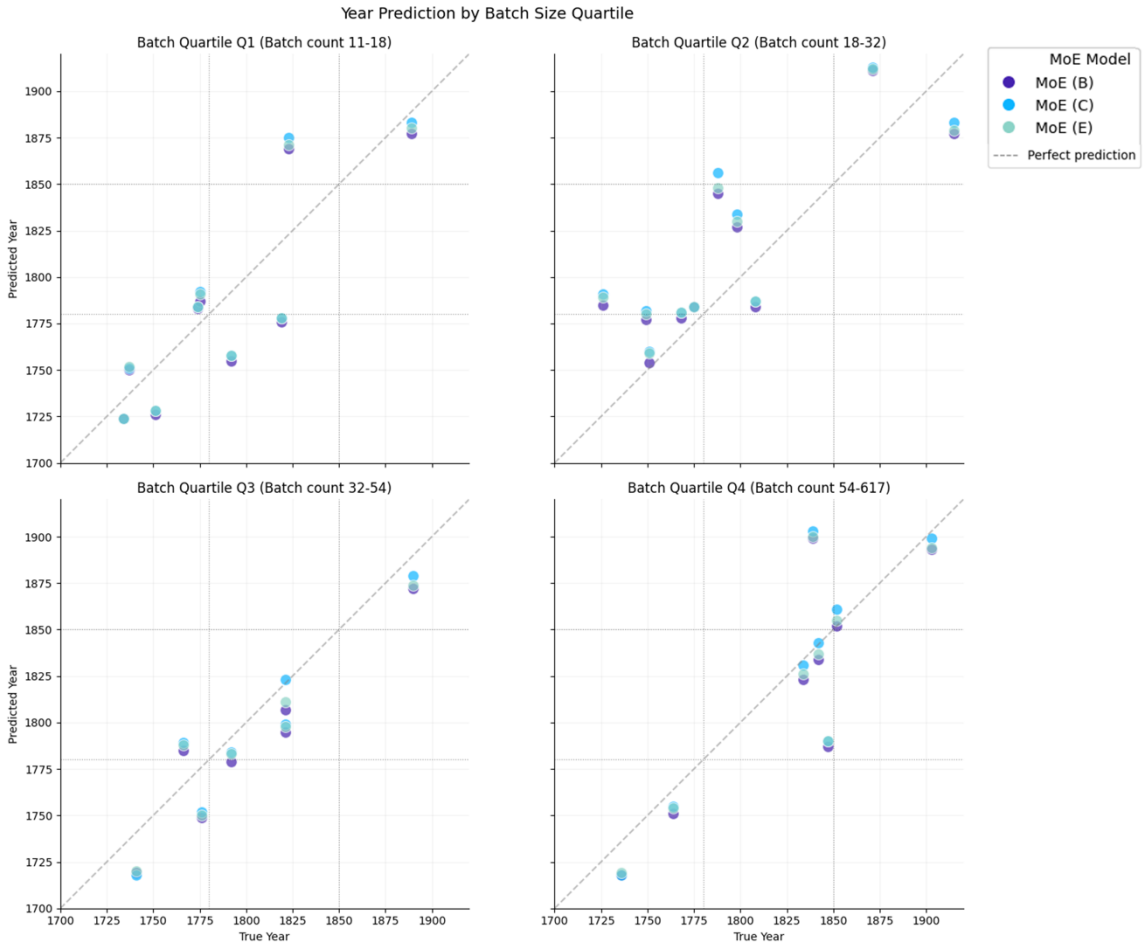

Figure 15. Predicted VS True year for four batch size quartiles, MoE (B, C, E), with perfect year prediction line.

Across all three Mixture of Experts models, prediction accuracy for dating historical English texts from 1710 to 1920 remains strong, with consistent performance improvements over a random baseline. While the presence or absence of explicit temporal embeddings does not drastically alter overall accuracy, different temporal encoding strategies appear to influence results in certain periods more than others. Notably, texts from the middle period (1780–1850) exhibit the highest prediction errors, highlighting particular difficulties in modelling this era. Analysis of error distributions reveals no systematic bias of absolute prediction errors both across the timeline and text length.

6.3.2. MoE (E)

To analyse the year prediction metrics further, we examine the results of MoE (E) individually, complementing them with the perplexity-based evaluation.

The first visualisation shows the distribution of perplexity values between batches in every text, indexed by their batch count and coloured by the true time period (Figure 16). Importantly, the horizontal axis does not reflect the magnitude of batch count but rather its order, as plotting 34 box plots proportionally spaced by actual batch count values would not be feasible. When multiple texts share the same batch count (e.g., batch count 11), they are displayed in a randomised order but under the same x-axis tick label.

The vertical axis in this and the two subsequent figures is capped at a maximum perplexity value of 50, omitting one extreme outlier with a perplexity of 186. This excluded value corresponds to the text CLMET3_1_1_64 (Treatise “On Conciliation with America” by Edmund Burke), which appears as an outlier associated with the box at batch count 18.

From the plot, we observe that the largest interquartile range (IQR) corresponds to text at batch count 38, followed closely by text at batch count 17. Most of the shortest boxes, indicating small IQR and therefore lower variation in perplexity, are located towards the bottom of the chart, suggesting lower perplexity values. However, there is no clear overall distinction between shorter and longer texts in terms of perplexity distribution.

Notably, a greater number of outliers (perplexity values outside the ±1.5×IQR range) are visible towards the right side of the plot, suggesting that bigger texts are associated with greater perplexity variation.

Finally, a trend emerges among the colour-coded time periods: purple boxes (representing 1850–1920) generally lie lower on the plot, indicating lower perplexity; red boxes (1710–1780) are mostly situated in the middle; and blue boxes (1780–1850) exhibit both higher perplexity values and broader IQRs.

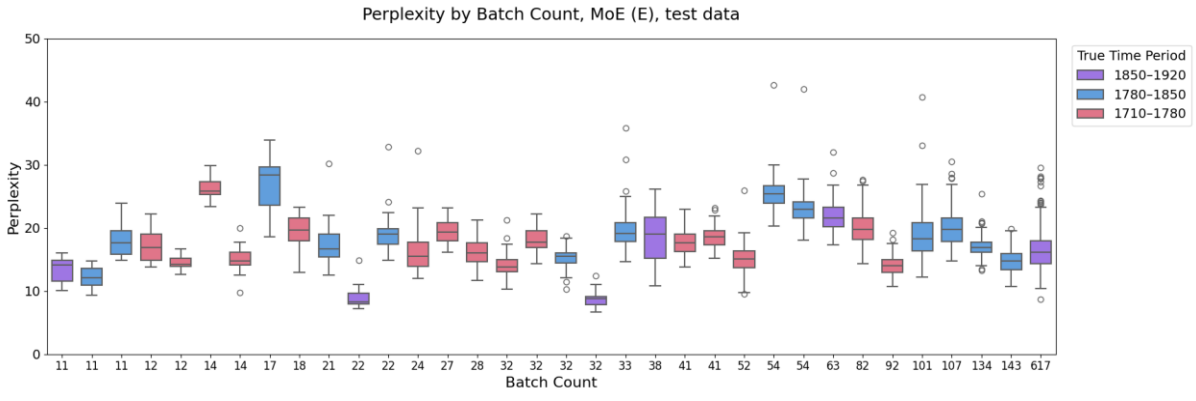

Figure 16. Boxplots of perplexity for every text in test set, by batch count. MoE (E).

To observe the relationship between absolute error and perplexity more closely, we created a scatter plot (Figure 17). Perplexity is averaged per text, and the year prediction is based on the averaged and re-normalised gate weights. Circle size indicates batch count quartiles (Q1–Q4), with the exact batch count ranges shown in the legend, resulting in four distinct circle sizes. The quartiles correspond to those used in Figure 15. Circles are coloured according to the true time period. A black dashed line represents the fitted regression line (trend line) across all data points.

Most of the circles appearing on the right-hand side of the plot (i.e., those with larger absolute error) are coloured blue, indicating they belong to the 1780–1850 time period. This indicates that texts from this middle time period tend to have higher year prediction error. There is also a concentrated group of circles of various sizes and colours clustered around an absolute error of approximately 10 ± 5 and a perplexity of around 20 ± 5, suggesting that these metric values are common.

Purple circles, representing the 1850–1920 period, again appear to be generally located lower on the plot, indicating lower perplexity values. These circles tend to lie beneath the trend line. The regression line itself is nearly horizontal, with a barely visible decline towards the higher end of absolute error.

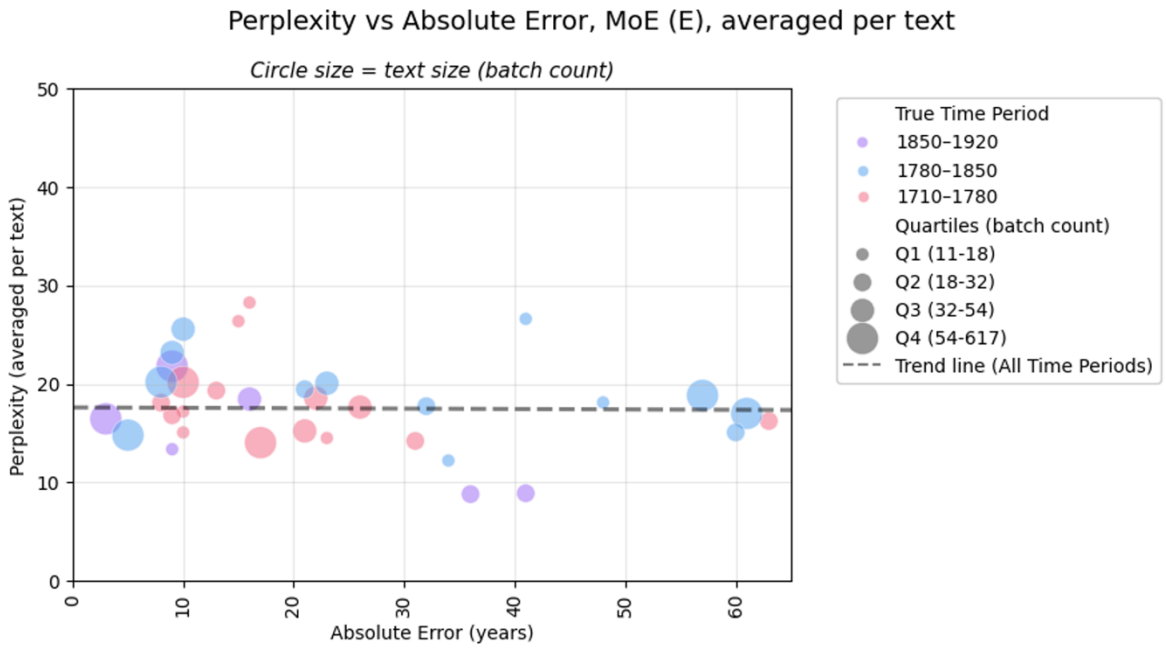

Figure 17. Scatter plot of perplexity VS absolute error, aggregated by text. MoE (E).

To mitigate the sparsity of data at the text level, we also generated a similar scatter plot at the batch level (Figure 18). In this second visualisation, we plotted absolute error against perplexity for every batch, again coloured by true time period. Here, the trend line is again roughly horizontal, centred around a perplexity value of 18, but a slightly more noticeable decline is visible on the right-hand side.

Pink circles, corresponding to the 1850–1920 time period, are once again concentrated in the lower region of the plot, indicating lower perplexity. Meanwhile, most of the outliers (data points deviating visibly from the trend line) belong to the 1780–1850 time period.

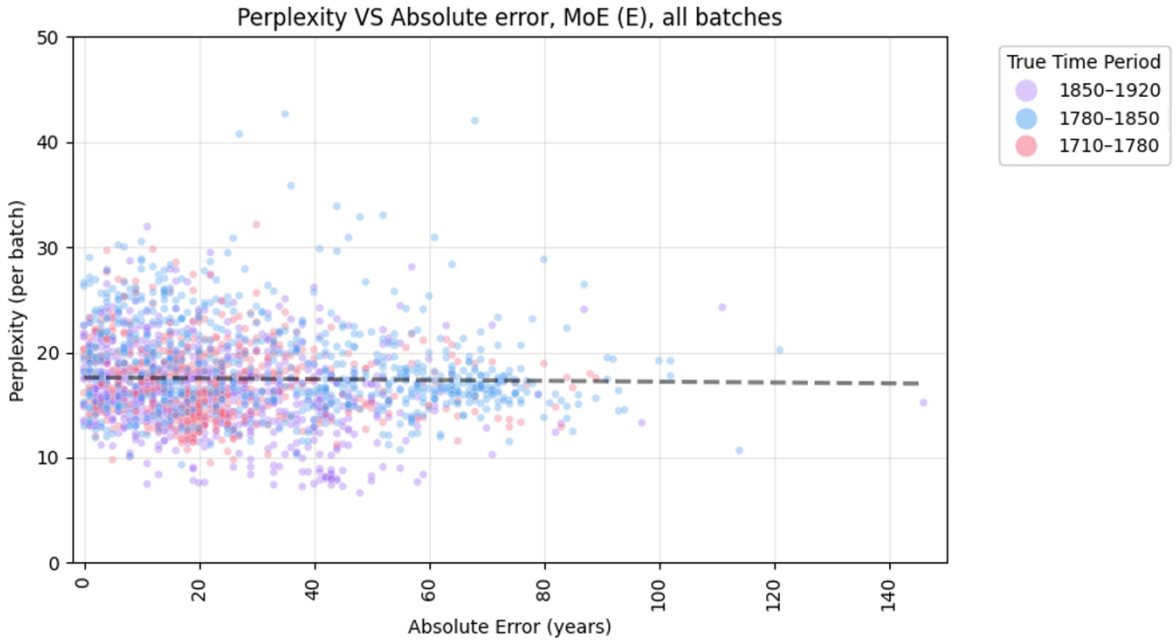

Figure 18. Scatter plot of perplexity VS absolute error for every batch in test set. MoE (E).

In all three of the preceding plots, the vertical axis has been capped to accommodate a single outlier with a perplexity value of 186. This value stems from a single batch of the Treatise "On Conciliation with America" by Edmund Burke (CLMET3_1_1_64). While this clear outlier inflates the text's average perplexity, the associated absolute error (16) is relatively small.

To investigate this further, we examine every batch of this text individually. The true printed year is 1775, placing it at the very end of the earliest period (1710–1780). Table 9 presents each batch in the order it was processed (Batch #). For each batch, the expert weights are displayed, with the dominant expert (highest weight) highlighted in bold font. The resulting

year prediction and the associated time period is shown. The predicted period is highlighted in red if it is two periods off from the true one, and in orange if it is one period off. Perplexity is also shown for every batch.

Instead of absolute error, we compute the signed error, the difference between the predicted year and the true year, in order to capture the direction of the misprediction. Predicted years are coloured green if the absolute value of the signed error is below 20; otherwise, they are left in default black.

From the table, we can see that while the very first batch yields an extremely high PPL (186) and a large signed error of +129, the rest of the batches have much lower perplexity values, ranging from approximately 13,4 to 23,3. Importantly, all batches but one predict the correct period, or are just one period off. There is considerable variation in the expert weights across batches. For instance, Batch 0 assigns 0.88 weight to the most recent expert (1850–1920), while Batch 15 places 0.63 weight on the earliest expert. Despite this variation, the predictions are often close to the true year of 1775.

In the final row of the table, we average the weights across all batches and re-normalise them. Based on these averaged weights, we calculate a final year prediction of 1791, with a signed error of -16. This corresponds to an absolute error of 16, well below the mean absolute error for both this period and the overall model results. The average perplexity across batches is 28.3, which remains elevated due to the extreme value from batch 0 but does not severely affect the accuracy of the final prediction.

Overall, this analysis demonstrates that although there might be significant variation in expert attention across batches, the final year prediction for this text is fairly accurate.

Batch #	**PPL**	**Weight 0** (expert 1710-1780)	**Weight 1** (expert 1780-1850)	**Weight 2** (expert 1850-1920)	**Predicted period** (dominant weight)	**Predicted year** (via formula)	**Signed error**
0	186	0,01	0,11	**0,88**	1850–1920	1904	129
1	23,3	**0,49**	0,44	0,08	1710–1780	1776	1
2	22,5	**0,46**	0,46	0,08	1710–1780	1780	5
3	21,8	**0,52**	0,41	0,07	1710–1780	1772	-3
4	21,7	0,24	**0,48**	0,28	1780–1850	1832	57
5	21,1	0,42	**0,47**	0,11	1780–1850	1784	9
6	20,7	0,29	**0,5**	0,21	1780–1850	1799	24
7	20,5	0,23	**0,49**	0,28	1780–1850	1831	56

8	19,7	0,27	**0,52**	0,22	1780–1850	1802	27
9	19,6	0,33	**0,52**	0,15	1780–1850	1796	21
10	19	**0,52**	0,42	0,06	1710–1780	1773	-2
11	18,9	0,22	**0,53**	0,25	1780–1850	1825	50
12	18,3	0,42	**0,46**	0,12	1780–1850	1784	9
13	17,8	0,39	**0,54**	0,08	1780–1850	1791	16
14	17,8	0,29	**0,55**	0,16	1780–1850	1802	27
15	14	**0,63**	0,34	0,03	1710–1780	1759	-16
16	13,4	0,16	**0,56**	0,28	1780–1850	1827	52
17	12,9	0,09	**0,46**	0,45	1780–1850	1849	74
AVG+renorm	28,3	0,33	**0,46**	0,21	1780–1850	1791	-16

Table 9. Metrics of batches for the text CLMET3_1_1_64, MoE (E), test data.

Lastly, we assess how perplexity varies with respect to the true (printed) year, while encoding the direction and magnitude of prediction error using colour (Figure 19). Each batch is represented as a point, with the x-axis showing the true year and the y-axis showing perplexity. The colour of each point corresponds to the signed error (predicted year minus true year), with a blue-to-yellow colour scale: blue indicating overshooting the true year, and yellow indicating undershooting. To aid visual orientation, the middle range (1780–1850) is faintly highlighted with a grey background.

In this plot, we observe a distinct cluster of yellow points near 1726, corresponding to the earliest text in the corpus, which also has the largest overshoot in prediction. Most batches, however, show negative signed error, meaning that predicted years are lower than the actual printed years. Despite this, the average signed error for the model overall is only -1.4 years, and the median is 8.5 years.

A very dark blue column of points is found near the lower-right corner of the plot, representing batches with very low perplexity but still considerable negative signed error. Text represented by these batches undershoot the true year by 64 years, once again reinforcing poor correlation between perplexity and year prediction accuracy.

No clear trend in signed error is visible across the timeline. This aligns with the average signed error per time range, which is -5.3 years (1710–1780), 0.5 years (1780–1850) and 4.3 years (1850–1920), showing no strong bias for any specific period.

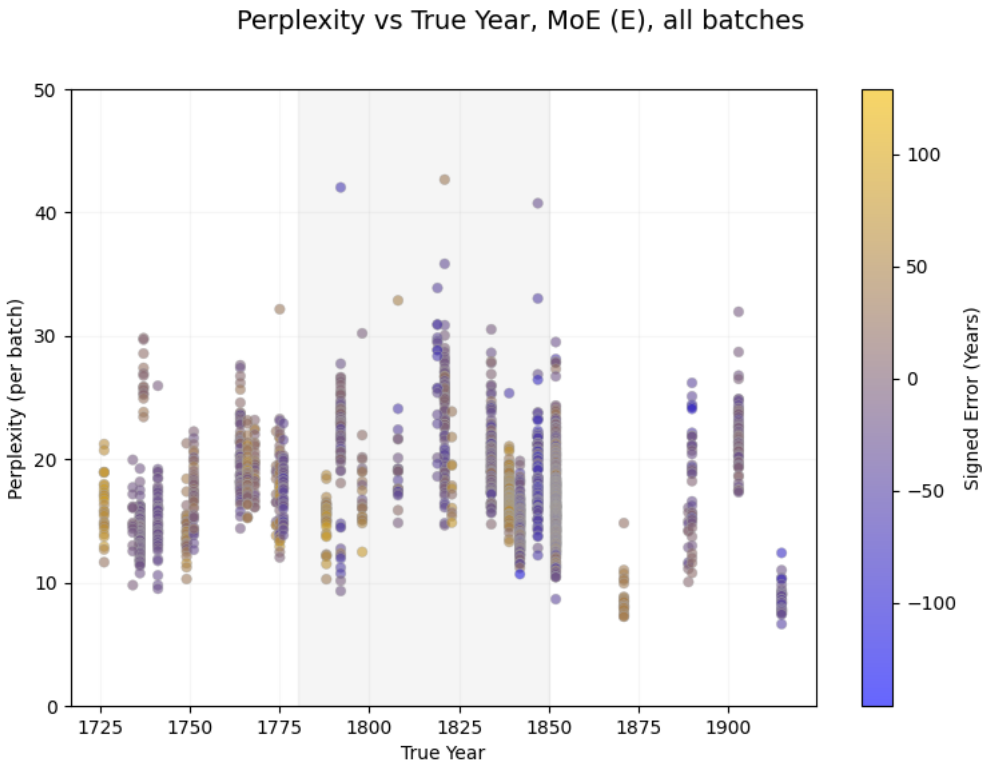

Figure 19. Scatter plot of perplexity VS true year for every batch in test set. MoE (E), coloured by signed error.

The MoE (E) model confirms many of the trends observed across all MoE configurations. Detailed investigation at the batch and individual text level reveals a nuanced relationship between model perplexity and prediction error: while perplexity variation tends to increase with longer texts, this does not directly translate into poorer prediction accuracy. Year prediction constitutes mean signed error of -1.4 years and a median signed error of 8.5 years, indicating minimal overall bias in its year predictions. However, most batches show a negative signed error, reflecting a tendency to predict publication years slightly earlier than their true publishing dates.

7. Scaling Up: Diachronic Mixture of Experts with OLMo 7B

To investigate whether diachronic adaptation benefits scale with model size, this chapter explores the application of the Mixture of Experts architecture using the larger OLMo 7B language model, using the best-performing configuration identified in previous chapter (MoE (E)). While previous experiments with OLMo 1B demonstrated clear improvements in temporal modelling through expert specialisation, it remains an open question whether similar gains can be achieved at larger scales. Scaling up not only tests the robustness of the

architecture but also provides insight into how increased model capacity interacts with historical domain shifts and temporal conditioning.

We trained three temporal experts based on OLMo 7B, using the same temporal embedding scheme (E) as before, namely, the exact year expressed in natural language. Each expert was trained on one of the previously defined temporal intervals: 1710–1780, 1780–1850 and 1850–1920. We then validated the performance of each model on both its corresponding time range and the other two ranges (Table 10).

For these experiments, we trained for five epochs, applied a slightly reduced learning rate of 1e-5 (as opposed to 2e-5 used for the 1B models), and used a per-device batch size of 1. To compensate for the small batch size, we increased the gradient accumulation steps to 64. Additionally, the warmup ratio was raised from 0.1 to 0.15, while the weight decay remained fixed at 0.005.

The results mirrored the pattern observed with the 1B models: each expert achieved the lowest perplexity on its corresponding temporal range. For instance, the 1710–1780 model reached a perplexity of 12.29 on texts from its own period, compared to 14.23 and 14.43 on texts from the middle and late periods, respectively. Similarly, the 1780–1850 expert achieved its best result (11.85) on its matching time span, while showing higher perplexities on earlier (13.61) and later (14.51) data. The 1850–1920 model also followed this trend, obtaining a perplexity of 12.15 on in-period validation data, compared to 15.86 and 15.06 on earlier time ranges.

These results reinforce the observation that temporal specialisation is effective, with each model learning features most relevant to its own period. Moreover, the sharper performance drops across temporal boundaries in the 7B models, as compared to 1B, suggests that the larger models may be more sensitive to historical domain shifts, but also more capable of leveraging in-period information effectively when available.

Year (data)\ Year(Model)	**Model 1710–1780**	**Model 1780–1850**	**Model 1850–1920**
Data 1710–1780	12.2948	14.2267	14.4251
Data 1780–1850	13.612	11.8451	14.5114
Data 1850–1920	15.8569	15.0615	12.1553

Table 10. OLMo **7B (E)**, PPL on the validation set across time ranges.

As the next step, we trained a gating network for the MoE 7B (E) model using the same architecture and training setup as for the 1B version, but with a smaller batch size (1 instead of 16) and a lower learning rate (1e-5 instead of 1e-4). We then evaluated its performance across the same three gating strategies as before: soft gating, argmax at inference, and oracle (i.e. rule-based) gating (Table 11).

Interestingly, the results diverge from the 1B findings. While the MoE 1B (E) performed best with oracle gating (16.9038) and worst with argmax (16.9950), the MoE 7B (E) showed a different pattern: the lowest perplexity for the 7B model was achieved with argmax gating (11.1006), followed by oracle (11.1307) and the highest perplexity with soft gating (11.4044). This reversal suggests that the larger 7B experts are more sharply specialised to their respective temporal ranges, allowing the gating network to assign expert weights more decisively. The improved separation is also supported by stronger validation performance difference across time ranges.

Gate type	**Test perplexity**	
	MoE 1B (E)	MoE 7B (E)
Soft gating	16.9624	11.4044
Argmax at Inference	16.9950	11.1006
Rule-based (Oracle)	16.9038	11.1307

Table 11. MoE 1B (E) and MoE 7B (E) performance on the test data depending on the gate type.

From a practical perspective, these findings imply that argmax gating may be preferable in settings where expert specialisation is pronounced, and the gate can reliably distinguish between temporal domains. In contrast, soft gating may still be useful for smaller models or in more ambiguous settings where a blended expert response helps reduce uncertainty.

7.1.Year Prediction Results MoE 7B (E)

Next, we evaluated year prediction performance both at the batch level and at the aggregated text level, using averaged and re-normalised expert weights. As with the 1B model, no type B cases (where two dominant experts are non-adjacent in time) were observed. In terms of Mean Absolute Error (MAE), the performance of MoE 7B (E) was comparable to that of MoE 1B (E), with scores of 24.147 and 24.029 respectively. To visualise the distribution of absolute errors across texts, we employed an overlaid step histogram with kernel density estimate (KDE) smoothing (Figure 20).

The x-axis indicates the absolute error in years, the y-axis reflects frequency counts. The step histogram provides a binned view of how often errors of particular magnitudes occur. Kernel density estimate lines are overlaid to convey the smoothed probability density, visualising the general trends and shifts in the distributions.

The plot illustrates the frequency and distribution of absolute errors (in years) made by each model. Step histograms provide discrete binning of error magnitudes, while KDE lines show the smoothed density estimates. Overall, the two curves are highly similar. The KDE for MoE 1B (E) is marginally higher than that of MoE 7B (E) until approximately 65 years of error, after which the 7B curve slightly overtakes, though this occurs within the final histogram bin. The step histograms reveal that the 7B model exhibits a sharper peak in the 10 to 18 year error range, while the 1B model displays a flatter, more uniform error distribution in the lower ranges.

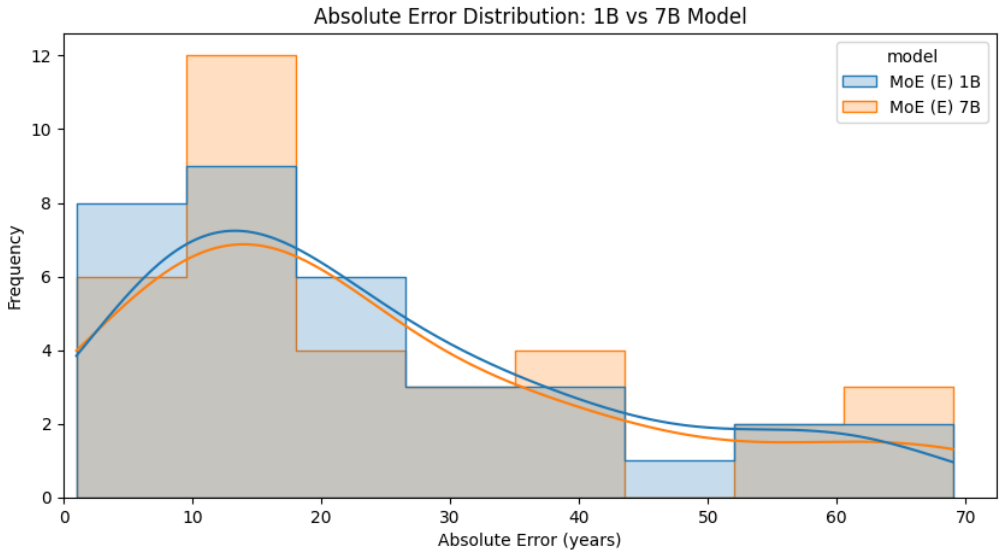

Figure 20. Absolute error distribution plot MoE (E) 1B VS 7B. Curved lines plot probability density.

Lastly, we visualised the difference in absolute error between MoE (E) 1B and MoE (E) 7B for each text individually (Figure 21). Each horizontal bar represents a text, with the direction and magnitude of the bar indicating which model achieved lower error and by how many years. Bars in blue correspond to cases where the 7B model outperformed the 1B model, and those in red mark the opposite.

Although there are noticeable differences in individual prediction errors, both models performed best on exactly 16 texts each, with two texts yielding identical absolute errors. The 1B model tended to yield smaller improvements in most cases but achieved a substantial error reduction of nearly 50 years for CLMET_1_1_64. This text included a batch with the single highest perplexity across all evaluations, yet the final year prediction based on averaged expert weights was relatively accurate in MoE (E) 1B. In contrast, for MoE (E) 7B, the same batch also produced an unusually high perplexity (115), and the gating mechanism disproportionately favoured the later temporal intervals (1780–1850 and 1850–1920). As a result, the predicted year (1840) was substantially displaced from the true year (1775), contributing to the larger error (see Supplementary Table S8).

The two texts where both models produced identical absolute errors, CLMET3_1_1_28 and CLMET3_1_1_58, are Narrative Fiction works from the end of the first temporal interval: “The Governess; or The Little Female Academy” by Sarah Fielding (1749) and “The Vicar of Wakefield” by Oliver Goldsmith (1766), with errors of 31 and 22 years respectively. Prediction performance on these texts was also highly consistent across all 1B expert models, suggesting that their temporal characteristics are particularly difficult to disambiguate.

Overall, both mixtures of experts demonstrated similar performance, with the MoE (E) 1B model achieving only a marginal advantage of 0.1 years in terms of average absolute error.

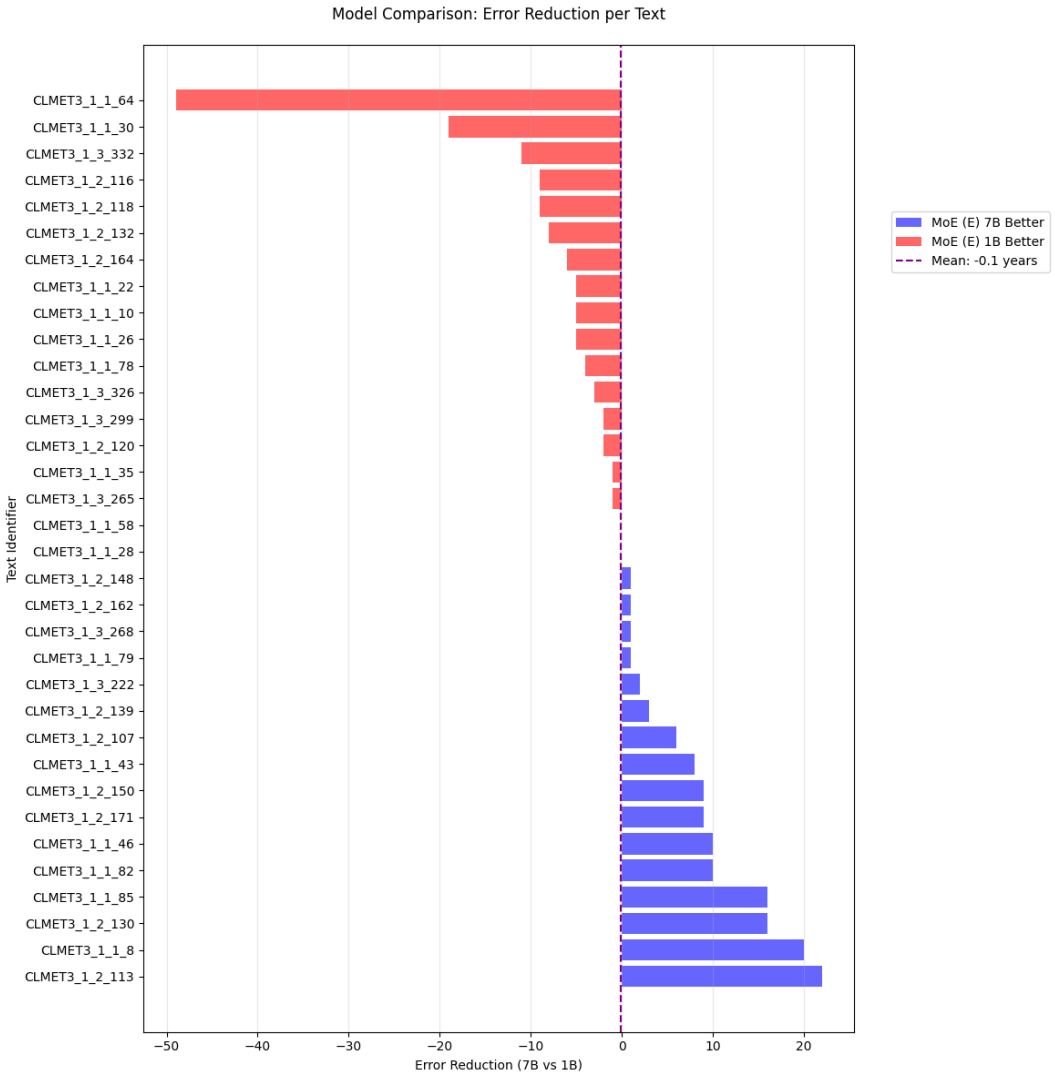

Figure 21. Error reduction plot comparing MoE (E) 1B and 7B models across texts. Positive values indicate lower error for MoE (E) 7B.

In summary, this chapter demonstrates that the Mixture of Experts architecture remains effective when scaled to larger model sizes. The OLMo 7B experts trained with natural-language year prompts (MoE (E)) exhibited clear temporal specialisation, achieving the lowest perplexity on their respective time intervals and showing sharper performance drops across boundaries than their 1B counterparts. Notably, the 7B model performed best with argmax gating, surpassing both soft and oracle strategies, indicating a stronger alignment between gating decisions and expert specialisation. Year prediction accuracy was comparable to that of the 1B variant, with both models achieving a Mean Absolute Error of approximately 24 years. While performance differences on individual texts were observed, overall prediction patterns and error distributions were highly consistent. These findings suggest that diachronic adaptation via temporal experts is robust across model scales.

8. Discussion

8.1. Summary of Findings

This thesis has examined several approaches to temporal adaptation in the context of fine-tuning a diachronic language model on historical English texts. Two principal strategies were explored: full fine-tuning of the model with various forms of temporal conditioning, and a modular Mixture of Experts architecture with soft gating.

The performance of all models on the test set is summarised in Table 12 and visualised in Figure 22. In this figure, a bar chart, each bar corresponds to the perplexity of one model on the test set, with the exception that MoE configurations are presented twice: with PPL corresponding to soft gating and to argmax. Bars are ordered descending with respect to perplexity and coloured according to the model type (base models, fully fine-tuned models, MoE with soft gating and MoE with argmax at inference).

	Base models (no fine-tuning)			**Fully fine-tuned 1B model**					**MoE: soft gating**				**MoE: argmax**			
									MoE 1B			MoE 7B	MoE 1B			MoE 7B
Model	OLMo 1B	OLMo 7B	OLMo 13B	OLMo 1B (B)	OLMo 1B (C)	OLMo 1B (D)	OLMo 1B (E)	OLMo 1B (F)	MoE 1B (B)	MoE 1B (C)	MoE 1B (E)	MoE 7B (E)	MoE 1B (B)	MoE 1B (C)	MoE 1B (E)	MoE 7B (E)
Test PPL	19.4077	13.8507	10.6748	17.5189	17.5327	17.5409	17.5371	17.5488	16.9687	16.993	16.9624	11.4044	17.0064	17.0723	16.9950	11.1006

Table 12. Performance comparison between original (base) models and all fine-tuned models.

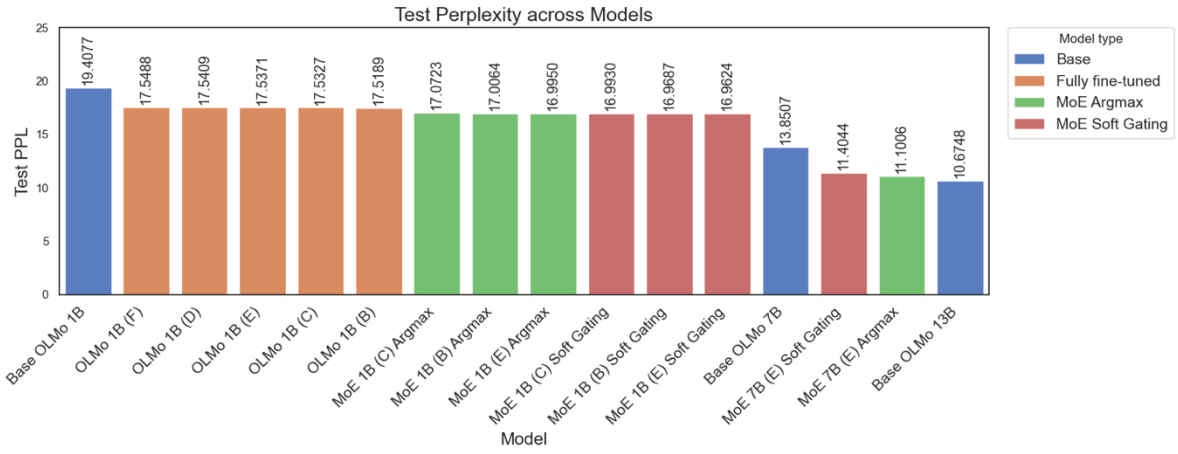

Figure 22. Bar chart: model performance on test set, original (base) models and all fine-tuned models.

Full fine-tuning did not yield substantial benefits from temporal embeddings. Across all experiments, only marginal differences in perplexity were observed between models with temporal input (OLMo 1B (C-F), PPL 17.53-17.55) and without it (OLMo 1B (B), PPL 17.52). In fact, the best-performing model (OLMo 1B (B)), which had no explicit temporal conditioning, slightly outperformed the alternatives, including those using natural language prompts and explicit time tokens (Figure 22). This suggests that the temporal representations used during fine-tuning were not reliably internalised by the model. Further evidence for this comes from the experiments where the model was tested with incorrect time inputs (Table 5). These had only minimal effects on perplexity, indicating that the model's predictions were largely independent of the embedded temporal information.

In contrast, the MoE architecture introduced in the section 6.2 led to clearer gains in temporal modelling. By fine-tuning three expert models on different temporal slices of the training data, and combining them using a soft gating mechanism, the MoE setup enforced a structural division based on time. The best-performing MoE 1B configuration (MoE 1B (E)) achieved a test perplexity of 16.96, representing a reduction of more than 0.5 points compared to the best single model (OLMo 1B (B)).

Comparative analysis of 1B and 7B parameter MoE models revealed the divergent gating behaviours. While the MoE 1B (E) configuration performed best with oracle gating and worst with argmax, the MoE 7B (E) exhibited the inverse pattern (Table 7). From the preliminary testing on the unfine-tuned models, we observed that the original OLMo 1B (base model) achieved a perplexity of 19.41 on the test set, while OLMo 7B demonstrated a considerable improvement with a perplexity of 13, and OLMo 13B improved this even further to 10.67.

Although fine-tuning the 1B model did not improve its performance sufficiently to match the 7B base model (PPL 16.96 VS 13.85), constructing a Mixture of Experts with the 7B expert models (MoE 7B (E), PPL 11.1) achieved performance only slightly worse than that of base 13B model (OLMo 13B, PPL 10.67). This demonstrates that by fine-tuning smaller models, and combining them in a MoE architecture, we can achieve performance comparable to the bigger models.

Lastly, analysis of the gating weights allowed us to attempt the task of predicting the year in which a text was written. All of the MoE 1B models performed relatively well on this task, with a mean absolute error of just over 24 years, constituting an improvement in year prediction of approximately 66% when compared to a random prediction. Interestingly, the 7B model performed slightly worse than the 1B model, with an average difference of 0.1 years, despite achieving a lower perplexity overall (more than 3 points lower). This suggests that while larger models may better model token level likelihoods, they do not necessarily yield improved performance in year prediction task within the current experimental setting.

8.2. General Implications

These findings speak to broader issues in the design of diachronic language models. The results of this study suggest that embedding temporal information through tokens or prompts may have limited effect unless the model architecture explicitly supports temporal specialisation. The failure of the full fine-tuning approach to make meaningful use of time information indicates that temporal embeddings alone are insufficient, particularly when added late in training.

By contrast, the success of the MoE model highlights the advantages of modular architectures for historical language modelling. Rather than relying on a single model to generalise across centuries, the MoE structure provides separate temporal components, each trained to model a specific range of linguistic variation. Argmax gating may be preferable in high-capacity models where expert boundaries are well-defined, as it enables confident expert selection without introducing the smoothing effects observed in soft gating. In contrast, smaller models may benefit from soft gating in scenarios with greater temporal ambiguity, as blending expert predictions can help reduce uncertainty.

8.3. Limitations

The conclusions above should be interpreted in light of several limitations.

First, evaluation relied primarily on perplexity, which offers a coarse and indirect measure of temporal understanding. A model with lower perplexity may predict the next token more accurately, but this does not necessarily imply that it has internalised historical linguistic change or diachronic structure. More targeted probing or classification tasks would be needed to establish this.

Second, the test set was relatively small, comprising of only 34 texts in total. While the qualitative patterns observed in expert gating appear robust, they may be sensitive to noise or sampling artefacts. A larger and more balanced evaluation set would support stronger claims.

Third, experiments were limited to models of 1B and 7B parameters. The larger model (7B) was only used for a "scaling up" section and was trained in only one configuration (E). It is possible that larger models may be more sensitive to prompt-based conditioning or more effective at internalising time embeddings, though this remains to be tested.

Fourth, the dataset (CLMET) covers only British English and spans a relatively narrow range of genres and time periods. This limits the generalisability of the findings to other varieties or domains. The success of the MoE setup may depend on the temporal distinctiveness of the training periods in CLMET, which might not be present in other corpora.

Fifth, the year prediction method was tailored specifically to the three-period setup used in this study. It relies on a weighted average of the expert scores aligned with fixed time intervals, which would not generalise to settings with different or more granular temporal splits. Therefore, a different strategy of interpreting gating weights would be required for other temporal splits.

Sixth, while the Mixture of Experts architecture enables temporal specialisation, the models were not explicitly conditioned on genre, topic, or authorship, all of which are known to influence stylistic features and could confound temporal signals.

Seventh, the averaging of gate weights across batches and texts, while computationally efficient, may obscure meaningful intra-textual variation or amplify anomalous predictions, particularly in cases where individual batches yield inconsistent expert assignments.

Finally, all training and evaluation were carried out on a high-performance SLURM cluster. The computational demands of fine-tuning large language models, especially in an MoE framework, far exceed the capacity of a standard PC. As such, the feasibility of replicating or extending this work may depend on access to comparable hardware resources. Model scaling introduced its own constraints; while the 7B model offered sharper expert separation, its training and inference costs were substantially higher, raising questions about its practicality in resource-constrained settings or real-world deployments.

8.4. Future Directions

There are several directions in which this work could be extended.

One possible avenue is to pre-train a language model from scratch with explicit temporal embeddings, rather than introducing them only at the fine-tuning stage. This would provide a stronger test of whether temporally grounded representations can be learned throughout training and potentially improve the model's ability to generate historically appropriate text. If full pre-training is not feasible, it would be valuable to explore the effect of initialising fine-tuning from a different checkpoint. Recent work by Mitchell et al. [17] suggests that alternative initialisation can enhance downstream adaptation, particularly when domain specific features are involved.

Another direction concerns the use of parameter-efficient fine-tuning methods to mitigate the computational demands required. Techniques such as LoRA, adapters, or BitFit could help mitigate the computational demands of fine-tuning large models.

Lastly, the MoE framework could be applied to other diachronic corpora, especially those with greater regional, dialectal, or genre diversity. This would help assess the robustness of the gating mechanism and test whether similar patterns of temporal specialisation emerge in more heterogeneous data. A multilingual extension would offer further insight into how models capture language change across typologically and culturally distinct contexts.

References

[1] Peng, M., Liu, B., Xu, W., Jiang, Z., Zhu, J., & Peng, M. (2024). Deja vu: Contrastive Historical Modeling with Prefix-tuning for Temporal Knowledge Graph Reasoning. *arXiv preprint arXiv*: *2404.00051.*

[2] Hamilton, W. L., Leskovec, J., & Jurafsky, D. (2016). Diachronic Word Embeddings Reveal Statistical Laws of Semantic Change. *arXiv preprint arXiv:1605.09096*.

[3] Gururangan, S., Lewis, M., Holtzman. A., Smith N. A., Zettlemoyer, L. (2021). DEMix Layers: Disentangling Domains for Modular Language Modeling. *arXiv preprint arXiv*: *2108.05036.*

[4] Qiu, W., & Xu, X. (2022). HistBERT: A Pre-trained Language Model for Diachronic Lexical Semantic Analysis. *arXiv preprint arXiv:2202.03612.*

[5] Manjavacas, E., & Fonteyn, L. (2021). MacBERTh: Development and Evaluation of a Historically Pre-trained Language Model for English (1450-1950). In *Proceedings of the Workshop on Natural Language Processing for Digital Humanities*, (pp.23-36).

[6] Loureiro, D., Barbieri, F., Neves, L., Anke, L. E., & Camacho-Collados, J. (2022). TimeLMs: Diachronic language models from Twitter. *arXiv preprint arXiv:2202.03829*.

[7] Jaidka, K., Chhaya, N., & Ungar, L. (2018). Diachronic degradation of language models: Insights from social media. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)* (pp. 195-200).

[8] Dubossarsky, H., Hengchen, S., Tahmasebi, N. and Schlechtweg, D. (2019). Time-out: Temporal referencing for robust modeling of lexical semantic change. *arXiv preprint arXiv:1906.01688*.

[9] Hu, E. J., Shen, Y., Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., Wang, L., & Chen, W. (2022). Lora: Low-rank adaptation of large language models. *ICLR*, *1*(2), 3.

[10] Ushio, A., & Camacho-Collados, J. (2024). A Systematic Analysis on the Temporal Generalization of Language Models in Social Media. *arXiv preprint arXiv*:*2405.13017.*

[11] Dhingra, B., Cole, J. R., Eisenschlos, J. M., Gillick, D., Eisenstein, J., & Cohen, W. W. (2022). Time-aware language models as temporal knowledge bases. *Transactions of the Association for Computational Linguistics*, *10*, 257-273.

[12] Periti, F., Dubossarsky, H., & Tahmasebi, N. (2024). (Chat) GPT v BERT: Dawn of Justice for Semantic Change Detection. *arXiv preprint arXiv:2401.14040*.

[13] Rissanen, M., Kytö, M., & Palander-Collin, M. (Eds.). (1993). *Early English in the computer age: Explorations through the Helsinki Corpus* (No. 11). Walter de Gruyter.

[14] Houlsby, N., Giurgiu, A., Jastrzebski, S., Morrone, B., De Laroussilhe, Q., Gesmundo, A., A., Attariyan, M., & Gelly, S. (2019). Parameter-efficient transfer learning for NLP. In *International conference on machine learning* (pp. 2790-2799). PMLR.

[15] Groeneveld, D., Beltagy, I., Walsh, P., Bhagia, A., Kinney, R., Tafjord, O., Jha, A.H., Ivison, H., Magnusson, I., Wang, Y., & Arora, S. (2024). Olmo: Accelerating the science of language models. *arXiv preprint arXiv:2402.00838*.

[16] Li, C., & Flanigan, J. (2024). Future Language Modeling from Temporal Document History. *arXiv preprint arXiv*:*2404.10297.*

[17] Mitchell, E., Rafailov, R., Sharma, A., Finn, C., & Manning, C. D. (2023). An emulator for fine-tuning large language models using small language models. *arXiv preprint arXiv:2310.12962*.

[18] Radford, A., Narasimhan, K., Salimans, T., & Sutskever, I. (2018). Improving Language Understanding by Generative Pre-Training. *OpenAI.* Retrieved from *https://openai.com/research/language-unsupervised*

[19] Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2019). BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: human language technologies, volume 1 (long and short papers)* (pp. 4171-4186).

[20] Pettersson, E., & Megyesi, B. (2018). The histcorp collection of historical corpora and resources. In *DHN 2018 The Third Conference on Digital Humanities in the Nordic Countries, March 7-9, 2018, Helsinki, Finland* (pp. 306-320). University of Helsinki.

[21] Rijhwani, S., & Preoţiuc-Pietro, D. (2020). Temporally-Informed Analysis of Named Entity Recognition. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics* (pp. 7605-7617).

[22] Wang, Z., Fu, R., Wen, Z., Tao, J., Wang, X., Xie, Y., ... & Li, G. (2025). Mixture of experts fusion for fake audio detection using frozen wav2vec 2.0. In *ICASSP 2025-2025 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (pp. 1-5). IEEE.

[23] Zaken, E. B., Ravfogel, S., & Goldberg, Y. (2021). Bitfit: Simple parameter-efficient fine-tuning for transformer-based masked language-models. *arXiv preprint arXiv:2106.10199*.

[24] V. Lialin, V. Deshpande, A. Rumshisky et al. (2023). Scaling down to scale up: A guide to parameter-efficient fine-tuning. *arXiv preprint arXiv:2303.15647.*

[25] Armi, L., Abbasi, E., & Zarepour-Ahmadabadi, J. (2021). Mixture of ELM based experts with trainable gating network. *arXiv preprint arXiv:2105.11706*.

[26] Hu, H., Yu, S., Chen, P., & Ponti, E. M. (2024). Fine-tuning large language models with sequential instructions. *arXiv preprint arXiv:2403.07794*.

[27] Perrone, V., Palma, M., Hengchen, S., Vatri, A., Smith, J. Q., & McGillivray, B. (2019). GASC: Genre-aware semantic change for Ancient Greek. *arXiv preprint arXiv:1903.05587*.

[28] Frermann, L., & Lapata, M. (2016). A Bayesian model of diachronic meaning change. *Transactions of the Association for Computational Linguistics*, *4*, 31-45.

[29] Zafar, S., & Nicholls, G. K. (2022). Measuring diachronic sense change: New models and Monte Carlo methods for Bayesian inference. *Journal of the Royal Statistical Society Series C: Applied Statistics*, *71*(5), 1569-1604.

[30] Zafar, S., & Nicholls, G. K. (2024). An embedded diachronic sense change model with a case study from ancient Greek. *Computational Statistics & Data Analysis, 199*, 108011.

[31] Ishihara, S., Takahashi, H., & Shirai, H. (2023). Quantifying Diachronic Language Change via Word Embeddings: Analysis of Social Events using 11 Years News Articles in Japanese and English. In *9th International Conference on Computational Social Science*.

[32] Seo, J., Kim, J., & Shin, H. (2025). MoFE: Mixture of Frozen Experts Architecture. *arXiv preprint arXiv:2503.06491*.

[33] McEnery, T., & Hardie, A. (2012). *Corpus linguistics: Method, theory, and practice.* Cambridge University Press.

[34] Davies, M. (2010). *The corpus of historical American English: COHA*. BYE, Brigham Young University.

[35] T. Luo, J. Lei, F. Lei, W. Liu, S. He, J. Zhao, K. Liu (2024). MoELoRA: Contrastive Learning Guided Mixture of Experts on Parameter-Efficient Fine-tuning for Large Language Models. *arXiv preprint arXiv:2402.12851*.

[36] Ren, H., Wang, H., Zhao, Y., & Ren, Y. (2023). Time-Aware Language Modeling for Historical Text Dating. In *Findings of the Association for Computational Linguistics: EMNLP 2023* (pp. 13646-13656).

[37] Li, D., Ma, Y., Wang, N., Ye, Z., Cheng, Z., Tang, Y., ... & Tang, M. (2024). MixLoRA: Enhancing large language models fine-tuning with lora-based mixture of experts. *arXiv preprint arXiv:2404.15159*.

[38] Muennighoff, N., Soldaini, L., Groeneveld, D., Lo, K., Morrison, J., Min, S., ... & Hajishirzi, H. (2024). OLMoE: Open mixture-of-experts language models. *arXiv preprint arXiv:2409.02060*.

[39] Shazeer, N., Mirhoseini, A., Maziarz, K., Davis, A., Le, Q., Hinton, G., & Dean, J. (2017). Outrageously large neural networks: The sparsely-gated mixture-of-experts layer. *arXiv preprint arXiv:1701.06538*.

[40] Liu, W., Han, G., & Huang, X. (2025). Examining and adapting time for multilingual classification via mixture of temporal experts. *arXiv preprint arXiv:2502.08825*.

[41] Chen, Z., Wang, K., Cai, Z., Zheng, J., He, J., Gao, M., & Zhang, J. (2022). Using Deep Mixture-of-Experts to Detect Word Meaning Shift for TempoWiC. *arXiv preprint arXiv:2211.03466*.

[42] Du, N., Huang, Y., Dai, A. M., Tong, S., Lepikhin, D., Xu, Y., ... & Cui, C. (2022, June). GLaM: Efficient Scaling of Language Models with Mixture-of-Experts. In *International conference on machine learning* (pp. 5547-5569). PMLR.

Supplementary Materials

Model	OLMo 1B training parameters					
Number of train epochs	1 epoch	2 epochs	3 epochs	4 epochs	5 epochs	7 epochs
Per device train batch size	8					
Per device eval batch size	8					
Number of GPUs for job	1	2	2	2	2	2
Eval strategy	epoch					
Metric for best model	eval loss					
Max grad norm	0.5					
Learning rate	3e-5	2.5e-5	2e-5	2e-5	2e-5	1.5e-5
LR scheduler type	cosine					
Gradient accumulation steps	32*	16	16	16	16	16
Warmup ratio	0,05	0,075	0,1	0,125	0,15	0,2
Weight decay	0,005	0,005	0,005	0,0075	0,01	0,02
Results						
Train Perplexity	20,4999	3,9934	3,9484	3,9046	3,8598	3,8959
Val perplexity	19,2255	14,7574	14,7494	14,7621	14,8041	14,7930

Table S1. Training parameters for experiments on the number of epochs.

Year (data)\ Year(Model)	Model 1710–1780	Model 1780–1850	Model 1850–1920
Data 1710–1780	17.2203	18.3697	18.9139
Data 1780–1850	17.3081	16.372	18.1592
Data 1850–1920	20.5327	18.8875	16.7611

Table S2. OLMo 1B, 3x experts (Model C: special tokens embeddings) trained on different temporal splits, validation perplexity.

Year (data)\ Year(Model)	Model 1710–1780	Model 1780–1850	Model 1850–1920
Data 1710–1780	17.2049	18.271	18.8318
Data 1780–1850	17.2833	16.3544	18.1262
Data 1850–1920	20.5218	18.8071	16.7022

Table S3. OLMo 1B, 3x experts (Model E: exact year in a natural language embeddings) trained on different temporal splits, validation perplexity.

True Time Range	Expert 1710–1780	Expert 1780–1850	Expert 1850–1920
1710–1780	0.662	0.288	0.050
1780–1850	0.216	0.436	0.348
1850–1920	0.070	0.385	0.545

Table S4. MoE (E), performance on test data. Average expert weights per true time range. **All batches.**

True Time Period	MoE	Expert 1710–1780	Expert 1780–1850	Expert 1850–1920
1710–1780	B	0.628	0.315	0.057
	C	0.602	0.323	0.076
	E	0.609	0.324	0.067
1780–1850	B	0.290	0.442	0.268
	C	0.265	0.426	0.309
	E	0.271	0.444	0.285
1850–1920	B	0.048	0.264	0.688
	C	0.041	0.228	0.731
	E	0.043	0.253	0.704

Table S5. MoE (B, C, E), performance on test data. Average expert weights true time range. Averaged per text.

Confidence threshold	Mean Absolute Error	#X cases	#Y cases	#Z cases	X cases MAE	Y cases MAE
0.7	18.43	28	0	0	18.43	
0.71	18.43	28	0	0	18.43	
0.72	18.43	28	0	0	18.43	
0.73	18.43	28	0	0	18.43	
0.74	18.43	28	0	0	18.43	
0.75	18.43	28	0	0	18.43	
0.76	18.43	28	0	0	18.43	
0.77	18.43	28	0	0	18.43	
0.78	19.18	27	0	1	17.7	59
0.79	19.18	27	0	1	17.7	59
0.8	19.21	25	0	3	16.92	38.33
0.81	19.04	24	0	4	16.42	34.75
0.82	19.79	23	0	5	15.3	40.4
0.83	19.79	23	0	5	15.3	40.4
0.84	19.79	23	0	5	15.3	40.4
0.85	19.79	23	0	5	15.3	40.4
0.86	19.79	23	0	5	15.3	40.4
0.87	19.79	23	0	5	15.3	40.4
0.88	19.79	23	0	5	15.3	40.4
0.89	20.71	22	0	6	15.95	38.17
0.9	20.96	21	0	7	16.48	34.43
0.91	22	20	0	8	16.15	36.62
0.92	20.96	19	0	9	13.74	36.22
0.93	22.68	17	0	11	13.41	37
0.94	22.68	17	0	11	13.41	37
0.95	22.68	17	0	11	13.41	37

0.96	22.68	17	0	11	13.41	37
0.97	22.68	16	0	12	13.88	34.42
0.98	24.71	12	0	16	14.25	32.56
0.99	25.64	9	0	19	13.67	31.32
1	27.11	0	0	28		27.11

Table S6. Mean absolute error, number of cases (X, Y, Z) and mean absolute error per case for the confidence threshold based on the gate weights of MoE (B) on the validation set. Column with absolute error for cases B is dropped, as no such cases were present.

Confidence threshold	Mean Absolute Error	#X cases	#Y cases	#Z cases	X cases MAE	Y cases MAE
0.7	18.75	28	0	0	18.75	
0.71	18.75	28	0	0	18.75	
0.72	18.75	28	0	0	18.75	
0.73	19.54	27	0	1	18	61
0.74	19.54	27	0	1	18	61
0.75	19.61	26	0	2	18.5	34
0.76	19.14	25	0	3	16.76	39
0.77	19.14	25	0	3	16.76	39
0.78	19.14	25	0	3	16.76	39
0.79	19.43	24	0	4	16.92	34.5
0.8	19.43	24	0	4	16.92	34.5
0.81	19.43	24	0	4	16.92	34.5
0.82	19.43	24	0	4	16.92	34.5
0.83	19.43	24	0	4	16.92	34.5
0.84	19.43	24	0	4	16.92	34.5
0.85	20.36	23	0	5	16.17	39.6
0.86	21.5	21	0	7	17.33	34
0.87	21.5	21	0	7	17.33	34
0.88	21.5	21	0	7	17.33	34
0.89	21.5	21	0	7	17.33	34
0.9	21.5	21	0	7	17.33	34
0.91	22.29	20	0	8	17.5	34.25
0.92	22.29	20	0	8	17.5	34.25
0.93	23.18	19	0	9	16.79	36.67
0.94	23.14	17	0	11	14.18	37
0.95	23.14	17	0	11	14.18	37
0.96	23.14	17	0	11	14.18	37
0.97	23.18	16	0	12	14.44	34.83
0.98	24.11	14	0	14	14.93	33.29
0.99	26.57	9	0	19	15	32.05
1	28.04	0	0	28		28.04

Table S7. Mean absolute error, number of cases (X, Y, Z) and mean absolute error per case for the confidence threshold based on the gate weights of MoE (C) on the validation set. Column with absolute error for cases B is dropped, as no such cases were present.

CLMET text ID (#34 texts)	True Printed Year	Predicted Year			
		MoE (B)	MoE (C)	MoE 1B (E)	MoE 7B (E)
histcorp-english-clmet-CLMET3_1_1_10	1726	1785	1791	1789	1794
histcorp-english-clmet-CLMET3_1_1_22	1734	1724	1724	1724	1749
histcorp-english-clmet-CLMET3_1_1_85	1736	1718	1718	1719	1735
histcorp-english-clmet-CLMET3_1_1_26	1737	1750	1751	1752	1757
histcorp-english-clmet-CLMET3_1_1_8	1741	1719	1718	1720	1740
histcorp-english-clmet-CLMET3_1_1_28	1749	1777	1782	1780	1780
histcorp-english-clmet-CLMET3_1_1_82	1751	1726	1728	1728	1764
histcorp-english-clmet-CLMET3_1_1_30	1751	1754	1760	1759	1778
histcorp-english-clmet-CLMET3_1_1_79	1764	1751	1755	1754	1773
histcorp-english-clmet-CLMET3_1_1_58	1766	1785	1789	1788	1788
histcorp-english-clmet-CLMET3_1_1_35	1768	1778	1781	1781	1782
histcorp-english-clmet-CLMET3_1_1_43	1774	1783	1784	1784	1776
histcorp-english-clmet-CLMET3_1_1_64	1775	1787	1792	1791	1840
histcorp-english-clmet-CLMET3_1_1_78	1775	1784	1784	1784	1788
histcorp-english-clmet-CLMET3_1_1_46	1776	1749	1752	1750	1760
histcorp-english-clmet-CLMET3_1_2_116	1788	1845	1856	1848	1857
histcorp-english-clmet-CLMET3_1_2_113	1792	1755	1758	1758	1780
histcorp-english-clmet-CLMET3_1_2_107	1792	1779	1784	1783	1789
histcorp-english-clmet-CLMET3_1_2_130	1798	1827	1834	1830	1782
histcorp-english-clmet-CLMET3_1_2_132	1808	1784	1787	1787	1779
histcorp-english-clmet-CLMET3_1_2_148	1819	1776	1778	1778	1779
histcorp-english-clmet-CLMET3_1_2_118	1821	1807	1823	1811	1840
histcorp-english-clmet-CLMET3_1_2_139	1821	1795	1799	1798	1841
histcorp-english-clmet-CLMET3_1_2_150	1823	1869	1875	1871	1862
histcorp-english-clmet-CLMET3_1_2_162	1834	1823	1831	1826	1841
histcorp-english-clmet-CLMET3_1_2_171	1839	1899	1903	1900	1891
histcorp-english-clmet-CLMET3_1_2_164	1842	1834	1843	1837	1853
histcorp-english-clmet-CLMET3_1_2_120	1847	1787	1790	1790	1788
histcorp-english-clmet-CLMET3_1_3_332	1852	1852	1861	1855	1866
histcorp-english-clmet-CLMET3_1_3_222	1871	1911	1913	1912	1910
histcorp-english-clmet-CLMET3_1_3_326	1889	1877	1883	1880	1877
histcorp-english-clmet-CLMET3_1_3_268	1890	1872	1879	1874	1875
histcorp-english-clmet-CLMET3_1_3_299	1903	1893	1899	1894	1892
histcorp-english-clmet-CLMET3_1_3_265	1915	1877	1883	1879	1878

Table S8. True and predicted year for every MoE 1B (B, C, E) and MoE 7B (E) for every text in the test dataset, ordered ascending by the true printed year.