Classification of Paleographic Artifacts at Scale: Mitigating Confounds and Distribution Shift in Cuneiform Tablet Dating

Danlu Chen[1], **Jiahe Tian**[2], **Yufei Weng**[1], **Taylor Berg-Kirkpatrick**[1], **Jacobo Myerston**[1]
UC San Diego[1], Fudan University[2]
danlu@ucsd.edu

Abstract

Cuneiform is the oldest writing system used for more than 3,000 years in ancient Mesopotamia. Cuneiform is written on clay tablets, which are hard to date because they often lack explicit references to time periods and their paleographic traits are not always reliable as a dating criterion. In this paper, we systematically analyse cuneiform dating problems using machine learning. We build baseline models for both visual and textual features and identify two major issues: confounds and distribution shift. We apply adversarial regularization and deep domain adaptation to mitigate these issues. On tablets from the same museum collections represented in the training set, we achieve accuracies as high as 84.42%. However, when test tablets are taken from held-out collections, models generalize more poorly. This is only partially mitigated by robust learning techniques, highlighting important challenges for future work.

1 Introduction

Computational paleography (Vidal-Gorène and Decours-Perez, 2021; Srivatsan et al., 2021) is a growing interdisciplinary field that uses computational algorithms to decipher and analyse ancient writing systems. We investigate using machine learning to automate large-scale dating of cuneiform[1], the oldest writing system from around 3,500 BCE. Similar to general chronicle attribution tasks in paleography, cuneiform dating involves classifying cuneiform tablets into specific time periods rather than precise years. For example, Figure 1 shows a tablet comes from Ur III. Different from other historical languages, such as ancient Greek (Assael et al., 2022) or ancient Arabic (Adam et al., 2018), cuneiform tablets are more challenging to convert into a machine readable format because the writing system continually evolved over the 3,000 years it was in use.

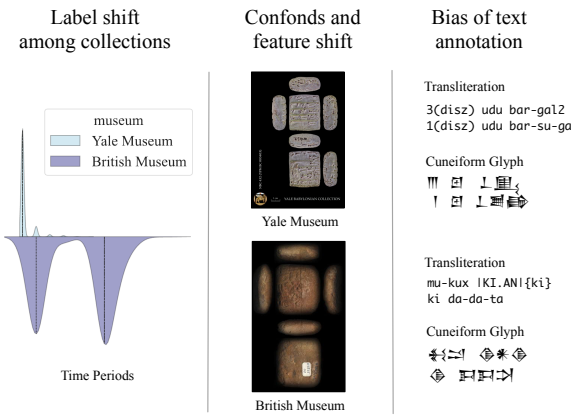

Figure 1: An overview of the cuneiform dating task. Tablets from different collection (museum or private collector) usually in different time period distribution and there is confound (undeier features to machine learning models) from different cameras. The transliteration is usually exhibit bias towards specific time periods.

For many writing systems, historians and paleographers have been able to identify distinguishing features in textual content and writing style that allow for inferences about date of origin for individual artifacts. For some writing systems, these processes have even been automated with machine learning to some extent. For example, Assael et al. (2022) showed encouraging results using neural networks trained on ancient Greek text to restore and date digitized ancient Greek artifacts.

Can we train similar textual models for cuneiform dating using accompanying manual transcriptions or transliterations? We conduct experiments with a series of light-weight recurrent models that show this is indeed possible. However, relying on manual transcriptions for the purpose of dating is somewhat circular: for Cuneiform, transcription and transliteration is as time-intensive as manually dating tablets. Further, transliterations themselves might exhibit bias—for example, an expert's approach to transliterating a tablet may al-

[1]Code is available at https://github.com/taineleau/CuneiML/tree/main/ml4al_2024_dating.

Proceedings of the 1st Workshop on Machine Learning for Ancient Languages (ML4AL 2024), pages 30–41

ready be influenced by preconceived notions about its time period—allowing models to overfit to the tendencies of individual transliterators.

Thus, we also study whether *visual representations* of Cuneiform tablets can be used effectively for automatic dating. Visual representations skirt the issues of manually-intensive transcription and confounds due to transliteration style. Further, visual representations may even allow models to automatically extract information about the visual style of writing, which paleographers have found useful for manual dating. In past work, Bogacz and Mara (2020) has shown that relatively accurate dating of cuneiform tablets using 3D scans is possible. However, currently it is not feasible to produce 3D scans of over 100,000 remaining tablets, which are dispersed among museums and private collections around the world.

Therefore, instead we explore the use 2D photographs from CDLI (CDLI contributors, 2024) to address the dating problem—a task that as far as we are aware has not been previously studied. Our experiments using convolutional neural models trained on 2D images demonstrate a new problem however: the different imaging setups used by different collections presents a confound that leads to poor generalization (shown in Figure 3). We find that the gap between performance on tablets from collections that were attested in training data versus those that were not is extremely large. Thus, we also evaluate to what extent robust learning methods that attempt to address out-of-distribution (OOD) generalization can mitigate this issue. We find that while these methods do help, they do not increase generalization to the point where accurate dating of tablets from unseen collections can be performed reliably. Thus, our empirical study highlights this important challenge as an area for future research. We summarize our primary contributions below:

1. We identify and analyze several challenging issues in cuneiform dating related to confounds, distribution shift, and domain generalization. These challenges are likely also present in the classification of other ancient artifacts with text.

2. We study a range of modeling approaches including simple methods like Naive Bayes, as well as neural methods for both images and text features. We demonstrate strong performance when using data splits that reduce distribution shift and OOD effects, but poor performance across museum collections.

3. We applied multiple robust learning techniques to mitigate distribution shift and the effect of confounds. While our results demonstrate improvements from these techniques, overall OOD generalization performance is still prohibitive for broader use.

In the following sections, we first formulate the problem and then describe the data collection splits we created to address our core research questions.

2 Problem Formulation

Technically, the dating task can be formulated as either a classification or a regression problem. However, after careful examination, we concluded that treating inferred dates as continuous variables (using regression) does not make sense in this domain because the annotation standard used for manual dating (the source of supervision for learning and evaluation) includes date categories with overlapping time intervals (see Figure 5). Instead, we represent each time period as a categorical class ID and treat dating as a multi-class classification problem.

Next, we layout the core research questions we attempt to answer in this empirical study. To address each, we will carefully design data splits that contain three separate test sets, each measuring a specific aspect of OOD generalization, along with a train and validation set.

RQ1: What models, configurations, and features—either visual or textual—are most effective for automatically dating cuneiform tablets?

RQ2: How much of a problem do OOD effects pose for generalization in this domain? For example, do models overfit to specific features present in individual museum collections? How well do models generalize to tablets from previously unseen museum collections?

RQ3: How well do existing robust learning techniques address the issue of distribution shift and OOD generalization in the context of cuneiform tablet dating?

In later sections, we will specify the datasets we use, which specific input representations we compare, and which modeling approaches

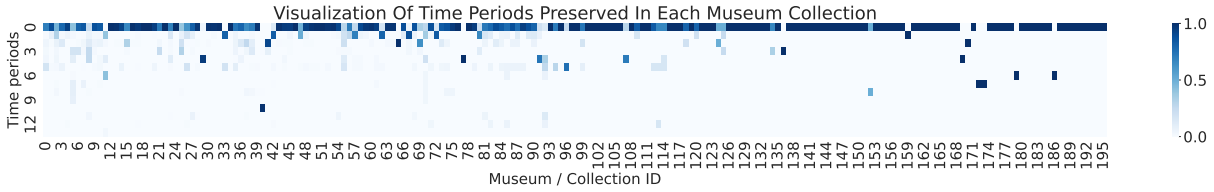

Figure 2: The normalized count density (by collection) of tablets from different time periods across museum collections. Darker colors indicate higher densities, highlighting that tablets from certain collections often belong to the same time period. This supports the hypothesis of distribution shifts between training and testing datasets. For a high-resolution version, see Appendix Figure 8.

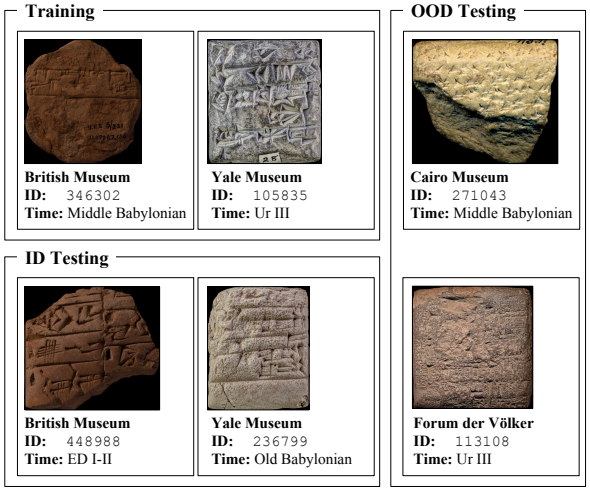

Figure 3: An overview one of the dating tasks using major-face cutouts of photographs to predict time periods. We held out several museums for the out-of-distribution (OOD) setting (e.g., the Cairo Museum), while the ID Testing set contains tablets from the same museums as the training set.

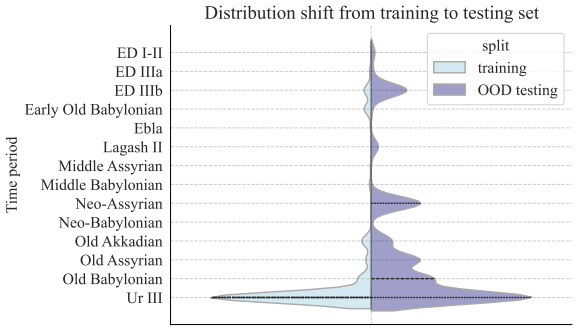

Figure 4: Visualization of *label shift* for the `collection shift` train/test split setting.

we evaluate. We will also carefully design test splits to answer specific questions about OOD generalization. Next, we describe and define some of the potential OOD effects in this domain and distribution shifts we seek to analyze.

Generally speaking, *distribution shift* occurs whenever the underlying distribution that generated the training data diverges from the distribution that will generate future test instances. Distribution shift poses a substantial challenge for learning systems: patterns that hold true on the training data may not generalize to the test set, leading to poor generalization performance. In the domain of cuneiform data the are two important types of distribution shift.

First, cuneiform datasets tend to exhibit substantial *label shift* due to how tablets are distributed across museum collections. We depict the distribution of tablet dates in museum collections in the CuneiML dataset (Chen et al., 2023) (which we use in experiments) in Table 2. Most museums contain tablets from a small range of time periods. Thus, if train and test setups for validating computational approaches are selected based on i.i.d. sampling from this dataset, the test performance may not accurately reflect expected performance on tablets from new, *unseen* museum collections. In Figure 4 we visualize actual label distribution shift in a i.i.d. train/test split.

Second, the input representations from individual museum collections may have properties that make the collection itself identifiable. For instance, as shown in Figures 1 and 3, the scanning methodologies used by separate museums leave artifacts like different amounts of color saturation and blurring. Similarly, it is possible that different transliteration styles may also be identifiable. Because individual collections are biased towards specific date ranges, the confounds mentioned above may cause *covariate shift*—a type of distribution shift where the distribution on the input variables and the relationship between input and output vary between train and test. For example, a model may learn to identify the collection based on properties of the scanning hardware in order to determine date. This may work on training data, but will not generalize

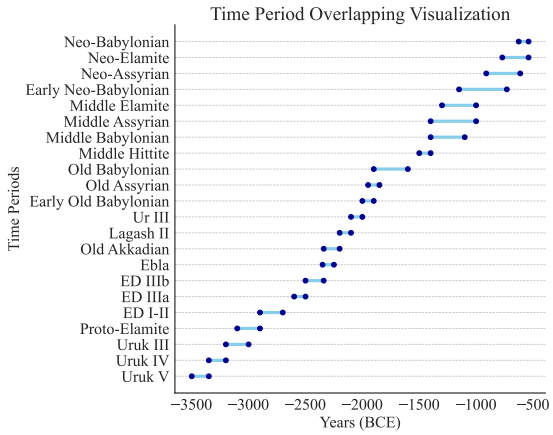

Figure 5: Time period overlapping visualization. The x-axis is years for BCE. Two time period classes can be parallel in time, for example, Middle Babylonian is almost completely overlaps in time with Middle Assyrian.

to new collections. Thus, one of our primary goals is to measure the effects of label and covariate shift for cuneiform dating and to evaluate to what extent robust learning methods may address these issues.

3 Data

We obtain 38,937 tablet images with transliterations from CDLI (CDLI contributors, 2024), using prepossessing from CuneiML (Chen et al., 2023). An example is shown in Figure 6. Besides transliteration and 2D images, we use several other attributes from the metadata entries, including **provenience**, **collection**, and **genre**, which we use in later experiments for both simple baselines and as additional supervision to mitigate distribution shift.

Figure 6: Left: Cuneiform Tablet images with six face photographs. Right: Example of transliteration in ATF format and the tokenization in cuneiform glyph. We use a special token <S> to separate the word in cuneiform.

Split	%	Count	Note
all	100%	38,937	-
train	80%	30,626	-
test 1	5%	2,065	OOD , $p(y)$ shift
valid	5%	2,116	OOD , $p(y)$ shift
test 2	5%	2,065	ID , $p(y)$ shift
test 3	5%	2,065	ID

Table 1: Dataset split statistics. OOD stands for Out-of-distribution compared to training set, and ID stands for in distribution compared to training set.

3.1 Data split

Inspired by Koh et al. (2021), we identify two kinds of distribution shift and would like to create splits that disentangle the issues and better answer the research questions. As we can see in Figure 2, most museum collections only own tablets from one or two time periods and most time periods are collected by a specific museum. To better study the distribution shift across collections, we split the data with regard to the collection id, i.e. tablets from the same collection only present in one split. This split we call OOD test split (test 1). We use $p(y)$ shift to denote a split where the label distribution $p(y)$ is significantly different from that of the training set. We describe briefly how we split the data (Table 1) below.

1. **Step 1: Getting an OOD and $p(y)$ shift set S_1 from the full data.** We sampled about 10% from the full dataset using the following rules: (i) We sample by collections, meaning tablets from an entire collection are either included or excluded. (ii) For a given time period, we do not select collections that constitute more than 30% of the data for that time period, ensuring that we do not remove most of the tablets for certain time periods from the training set. We named the remaining 90% of full data S_2. Figure 4 shows the shift of $p(Y)$.

2. **Step 2: Getting valid and test 1 set.** we evenly split S_1 We obtained from step 1 and we now have **valid** and **test 1** set.

3. **Step 3: Getting test 2.** We sampled 5% of the data from the subset S_2 against the label distribution of **test 1**. Therefore, test 2 has the same sub-population shift from the training set as **test 1**, but consists of in-domain (ID)

data instead of OOD data. We named the remaining 85% of the full data S_3.

4. **Step 4: Getting test 3 and train set.** We randomly sampled 5% of the data from S_3 to constitute **test 3**, and the remaining 80% is the final training set.

Therefore, we have three testing splits setup as shown in Table 1.

4 Methods

We describe the baseline models we used in experiments and also several training strategies, adversarial regularization and , to mitigate the distribution shift issues.

4.1 Baseline models

1. **Naive Bayes.** We use discrete categorical features, including genre, collection, provenance, and size, to predict the time period as a categorical prediction problem. Note that when there is only one feature, the performance indicates a correlation between the feature and the predicted class.

2. **Char-LSTMs.** We use a character-level two-layer bi-directional LSTMs to process cuneiform transliterations and sign tokens for dating ancient texts. The model has a hidden size of 128 and an embedding size of 256. We train for 200 epochs using the ADAMW optimizer with a learning rate of 5e-4 and a weight decay of 1e-3.

3. **ResNet.** Our study utilizes the ResNet (He et al., 2016) architecture, specifically ResNet-50 and ResNet-101. We apply these models to classify images of cuneiform inscriptions, leveraging their powerful feature extraction capabilities. The models are trained using a cross-entropy loss function, with adjustments made to the final layer to suit our specific class labels. The training regimen includes a batch size of 16, 30 epochs, ADAM with a learning rate of 3e-5, and no weight decay.

4.2 Baseline Objective

For all the neural models, we use cross entropy (CE) loss to train the models.

$$L = \text{CE}(y^{(t)}, p^{(t)})$$

4.3 Advanced Algorithms

To address the aforementioned issues, we explore several different robust training algorithms in this paper.

Adversarial Regularization. We use other attributes such as provenience and genre, to optimize a min-max objective. We attach a new branch of MLP to calculate the $p^{(adv)}$.

$L = \text{CE}(y^{(t)}, p^{(t)}) + \text{KLD}(y^{(const)}, p^{(adv)})$

where CE is cross entropy loss and KLD is the KL Divergence loss.

Correlation Alignment for Deep Domain Adaptation (CORAL). CORAL (Sun and Saenko, 2016) measures the divergence of means and covariance between batches of feature representations. The goal of CORAL is to match the feature distributions from different domains.

Invariant risk minimization (IRM). IRM (Arjovsky et al., 2019) penalizes feature distributions that result in different optimal linear classifiers across different domains. where where Φ is the entire invariant predictor, $w = 1.0$ is a fixed classifier, and the gradient norm penalty is the measure of the classifier at each environment.

5 Experiments and Results

5.1 Input Features

We have four different input features for training, describing as below.

1. **Raw image.** The raw images downloaded from CDLI. Each image usually contains photographs of six faces for each tablet.

2. **Major-face image.** The major-face cutout of the raw images, which are usually the front faces of the tablets.

3. **Raw transliteration.** We use the post-processed version from CuneiML, which removes formatting string such as line numbers, broken markers and etc.

4. **Cuneiform sign (glyph) token.** We tokenize cuneiform glyph at a character-level, with a vocabulary size of 764. See Figure 6 for an example. We keep the space between words and line break.

Features	Model	test 1 OOD $p(y)$ shift		test 2 ID $p(y)$ shift		test 3 ID	
		F_1	Acc.	F_1	Acc.	F_1	Acc.
-	random	2.92	7.80	2.91	7.12	2.96	6.30
-	majority	6.56	74.29	6.56	74.29	6.02	72.93
provenience	NBayes	**39.48**	83.63	51.09	79.95	61.15	89.20
genre	NBayes	15.31	72.88	19.77	75.11	22.72	80.63
provenience & genre	NBayes	37.94	83.49	**56.98**	83.24	**62.72**	91.91
museum (collection)	NBayes	6.56	74.29	13.92	75.16	21.78	77.85
transliteration	char-LSTM	16.14	10.72	**26.52**	10.87	**84.42**	95.73
sign token	char-LSTM	**16.59**	11.45	24.25	11.89	78.13	95.39
raw image	ResNet-50	28.46	82.03	64.33	93.51	78.73	94.26
+ *OOD mitigate*		29.42	83.24	47.46	92.13	48.63	88.17
major cutout	ResNet-50	34.82	87.36	**68.69**	94.74	**80.60**	95.19
+ *OOD mitigate*		**41.06**	88.37	49.55	91.62	54.78	88.03

Table 2: Main result table for cuneiform dating. Macro F_1 and Accuracy (Acc.) are reported. Macro F_1 denotes the average F_1 score calculated across all classes. Best F_1 scores for each subgroup are in **bold face** and the second best ones are underlined. Colored background highlight the best overall model for each setting.

The bounding boxes for major-face images and the Cuneiform sign (glyph) tokens are obtained from Chen et al. (2023)[2].

5.2 Metrics

As the label distribution $p(y)$ imbalance exists and there is a distribution shift, we primarily use the F_1 score and accuracy to evaluate our methods. Specifically, we use Macro F_1 and accuracy[3] as our major evaluation metrics.

Macro F_1 score computes the F_1 score independently for each class and then takes the average, thus treating all classes equally regardless of their frequency. This dual approach allows us to address both the overall accuracy and the individual class performance, ensuring a thorough evaluation in the face of skewed class distributions and shifts.

5.3 Results and Analysis

The main results for two split settings are shown in Table 2 and several key observation are summarized as follows.

1. **Random and Majority Baseline Models.** These models provide basic benchmarks with the majority model performing based on the most frequent class, note that the majority class contains more than 70% of the models, which accounts for the big discrepency between macro F_1 and accuracy. The low F1 scores, indicating poor performance across all classes evenly.

2. **Neural models perform the best across all settings.** Both visual and textual neural models work fairly good in ID setting (test 3), showing that both textual and visual features provide sufficient information to date tablets.

3. **Raw images contain confounded undesired features: collection**. When using a ResNet-50 model, features extracted from the raw images outperformed those obtained from front face cutouts on ID split (test 3). However, this performance was reversed on an OOD split (test 1). This reversal clearly indicates that raw images include collections as a confounding factor.

4. **Textual features are not effective for dating when label shift exists.** From test 3 to test 2, only the label distribution changes, while the data remains in-domain. However, textual models experience a dramatic drop in performance by 57.9%, revealing that textual features are not robust to label imbalance issues. In contrast, image models are not affected as significantly.

5. **Textual models are not robust to OOD shift; visual models are better but still have room for improvement.** Textual models exhibit nearly a 50% relative decrease in macro F_1 for the OOD setting (test 1) compared to visual models. With the application of OOD

[2] https://github.com/taineleau/CuneiML

[3] For single-label classification, Micro F_1 is equal to accuracy

mitigating algorithms (see section 6.3 for details), visual models improve from 34.82% to 41.06%, achieving the best F_1 score on test 1. This aligns with our earlier concerns that textual features do not capture any writing style of the tablet, making it difficult to determine the time period under OOD shift conditions.

6 Further Analysis

6.1 Zooming in on Textual Models

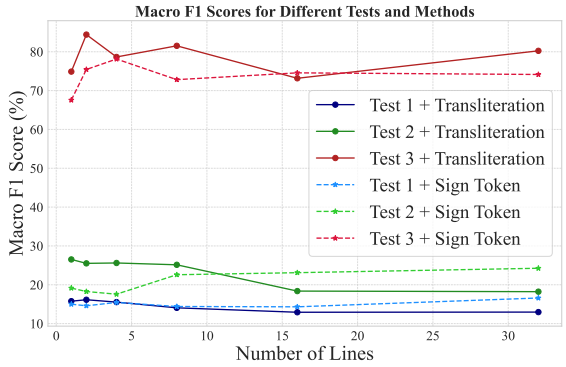

Figure 7: Analysis on best context length for textual features using char-LSTM on three test split.

As shown in Figure 7, we conducted extensive experiments on the number of lines per example fed into the models. As mentioned earlier, we use majority vote by default to ensemble predictions when we divide a full document. The performance of the glyph token features (sign token) increases as the number of lines in an example increases, while the transliteration features typically achieve the best performance with only one or two lines. This observation aligns with our understanding that transliteration already encodes some contextual knowledge, as signs are transliterated into Latin depending on the context. In contrast, for sign token features, the machine learning model requires more lines to discern the underlying information effectively.

6.2 Mitigating Label Imbalance Issues

Table 3 presents the results of label imbalance methods using char-LSTM on transliteration and glyph token features, with loss reweighing (LR) and upsampling (US). While both methods show varied effects on the performance metrics, loss reweighing generally improves F_1 scores and accuracy across the test sets, particularly for transliteration features, achieving a F_1 86.06% on test 3.

Features	test 1		test 2		test 3	
	F_1	Acc.	F_1	Acc.	F_1	Acc.
trans.	15.77	10.53	**26.52**	10.87	74.89	92.62
+ *LR*	**16.54**	11.45	24.98	11.89	**86.06**	85.33
+ *US*	12.63	9.89	25.32	10.63	81.75	94.61
glyph	15.00	8.82	19.14	9.52	67.56	92.08
+ *LR*	**17.63**	12.04	18.63	12.52	**74.08**	94.85
+ *US*	15.57	10.92	**19.17**	12.38	73.22	94.66

Table 3: Result for label Imbalance methods using char-LSTM on transliteraion and glyph token features. LR: loss reweighing, US: up sampling. The models trained with num_of_line=1.

6.3 Distribution shift and Confounds

Adversarial Regularization. Table 4 show results using adversarial regularization. Macro F_1 does not change as much as the accuracy. We also found that adversarial training requires very careful hyper-parameter tuning; otherwise, the model may completely underfit due to the noisy gradients provided by the adversarial branch.

Input	adv. feat	Macro F_1	Acc.
raw	none	25.73	58.34
raw	collection	25.44	62.12
cutout	none	29.09	64.91
cutout	collection	30.39	68.64

Table 4: Adversarial study on image features. ResNet-50 is used for all experiments in this table. We run each experiments five time and report the mean F_1 scores. Note that the result is trained on a slightly different split than the main table.

OOD mitigation. Table 5 shows results using OOD methods. Among the OOD mitigating algorithms, CORAL consistently improves the performance across all test sets for both raw and cutout features. Notably, CORAL achieves the best F_1 scores of 29.42% and 40.46% on test 1 for raw and cutout features, respectively. The other algorithms, IRM and groupDRO, generally show a decline in performance, with groupDRO performing the worst, especially for the cutout features. Overall, the results indicate that while textual models struggle with domain shifts, visual models, particularly those enhanced with cutout features and CORAL, demonstrate a more robust performance, albeit with room for further improvement.

Features	test 1		test 2		test 3	
	F_1	Acc.	F_1	Acc.	F_1	Acc.
raw	28.46	82.03	**64.33**	93.51	**78.73**	94.26
+ *IRM*	26.97	85.28	47.28	90.91	48.63	88.17
+ *CORAL*	**29.42**	83.24	47.46	92.13	46.94	90.08
+ *groupDRO*	24.02	77.97	35.18	87.69	56.54	89.54
cutout	34.82	87.36	**68.69**	94.74	**80.60**	95.19
+ *IRM*	28.31	87.46	43.42	91.11	44.34	88.81
+ *CORAL*	**40.46**	89.39	52.51	93.05	48.61	90.92
+ *groupDRO*	27.94	80.77	50.47	86.82	60.50	86.99

Table 5: OOD setting results trained on images features using ResNet-50.

Num of Examples	ResNet-50		char-LSTM	
	F_1	Acc.	F_1	Acc.
Full	85.34	94.40	53.19	85.89
10,000	67.18	90.86	49.46	84.73
5,000	59.17	89.99	32.93	82.24
1,000	40.89	82.39	17.28	75.09
500	28.03	77.77	7.44	70.62
100	13.21	55.49	5.66	65.67

Table 6: Ablation study on different number of training data, on test 3 using ResNet-50 and BERT. Note that this table is running on a slightly different data split from the main table.

6.4 Cuneiform Dating at Scale

It is not possible to make an apple-to-apple comparison on 2D and 3D scans features because most of the HeiCuBeDa dataset (Bogacz and Mara, 2020) does not accompany with a 2D photo. The paper reported a weighted F_1 of 83% (which is roughly comparable to accuracy in our case). We conduct a set of experiments by varying the number of training examples, as shown in Table 6. Both models show a clear trend of improved performance with increased training data.

7 Related work

7.1 Automated classification for ancient languages

Sommerschield et al. (2023) provides a detailed overview of ancient languages processing using machine learning. Resler et al. (2021) classified artifact images using CNNs and nearest neighbors. Assael et al. (2022) train a BE to restore ancient Greek. There have been work on dating documents in various ancient languages, like Arabic, Korean and Chinese oracles bones among others (Sommerschield et al., 2023)

7.2 Cuneiform studies

There have been important efforts in cuneiform sign recognition, language identification (Bernier-Colborne et al., 2019), and machine translation for Akkadian have been explored (Gutherz et al., 2023). Bogacz and Mara (2020) use high resolution 3D scans to classify time periods, and more recently Yugay et al. (2024) have explored the dating of first millennium Assyrian and Babylonian documents, using stylistic criteria and CNN. As mentioned earlier, it is non-trivial to tokenize the transliteration. Gordin et al. (2020) uses HMM and neural models to automatically transliterate Unicode cuneiform signs. On the contrary, in our paper, we reverse this process by converting the transliteration back to Unicode cuneiform signs to reduce transliteration bias.

7.3 Distribution shift

Historical data always suffers from noise and therefore it is hard to have good generalization on held out data. Specially for cuneiform, the systematic distribution shift is the most salient one. The systematic distribution shift is a special cases in domain adaptation, and therefore can be mitigated by general domain adaptation methods (Koh et al., 2021)). Ahmed et al. (2020) analyses group invariant predictions, where dominant simpler correlations with the target variable. Zare and Nguyen (2022) studied similar scenario in medical diagnosis, which has a shift on several attributes such as sex, age and race. They use invariant risk minimization (IRM) (Arjovsky et al., 2019) to learn invariant features. Another branch of methods is adversarial regularization, which uses adversarial training (Gokhale et al., 2021) to improve the generalization ability. Li et al. (2018) uses Maximum Mean Discrepancy (MMD) to align loss in different class.

8 Conclusion

In this paper, we explore end-to-end cuneiform dating at scale using machine learning. We have identified three major challenges—label imbalance, distribution shift, and circular reasoning—that are prevalent in cuneiform dating. These issues and solutions explored in our paper are broadly applicable to the classification of other ancient artifacts as well. We hope our initial analysis will inspire the

community to further adopt machine learning for addressing problems in ancient language processing.

Acknowledgments

All the images and annotations are from CDLI (CDLI contributors, 2024), and our work would not have been possible without the numerous annotations and editorial work provided by their team and collaborators.

References

Kalthoum Adam, Asim Baig, Somaya Al-Maadeed, Ahmed Bouridane, and Sherine El-Menshawy. 2018. Kertas: dataset for automatic dating of ancient arabic manuscripts. *International Journal on Document Analysis and Recognition (IJDAR)*, 21:283–290.

Faruk Ahmed, Yoshua Bengio, Harm Van Seijen, and Aaron Courville. 2020. Systematic generalisation with group invariant predictions. In *International Conference on Learning Representations*.

Martin Arjovsky, Léon Bottou, Ishaan Gulrajani, and David Lopez-Paz. 2019. Invariant risk minimization. *arXiv preprint arXiv:1907.02893*.

Yannis Assael, Thea Sommerschield, Brendan Shillingford, Mahyar Bordbar, John Pavlopoulos, Marita Chatzipanagiotou, Ion Androutsopoulos, Jonathan Prag, and Nando de Freitas. 2022. Restoring and attributing ancient texts using deep neural networks. *Nature*, 603(7900):280–283. Number: 7900 Publisher: Nature Publishing Group.

Gabriel Bernier-Colborne, Cyril Goutte, and Serge Léger. 2019. Improving Cuneiform Language Identification with BERT. In *Proceedings of the Sixth Workshop on NLP for Similar Languages, Varieties and Dialects*, pages 17–25, Ann Arbor, Michigan. Association for Computational Linguistics.

Bartosz Bogacz and Hubert Mara. 2020. Period Classification of 3D Cuneiform Tablets with Geometric Neural Networks. In *2020 17th International Conference on Frontiers in Handwriting Recognition (ICFHR)*, pages 246–251.

CDLI contributors. 2024. Home. https://cdli.mpiwg-berlin.mpg.de/. [Online; accessed 2024-07-04].

Danlu Chen, Aditi Agarwal, Taylor Berg-Kirkpatrick, and Jacobo Myerston. 2023. Cuneiml: A cuneiform dataset for machine learning. *Journal of Open Humanities Data*.

Tejas Gokhale, Rushil Anirudh, Bhavya Kailkhura, Jayaraman J Thiagarajan, Chitta Baral, and Yezhou Yang. 2021. Attribute-guided adversarial training for robustness to natural perturbations. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 7574–7582.

Shai Gordin, Gai Gutherz, Ariel Elazary, Avital Romach, Enrique Jiménez, Jonathan Berant, and Yoram Cohen. 2020. Reading akkadian cuneiform using natural language processing. *PloS one*, 15(10):e0240511.

Gai Gutherz, Shai Gordin, Luis Sáenz, Omer Levy, and Jonathan Berant. 2023. Translating akkadian to english with neural machine translation. *PNAS nexus*, 2(5):pgad096.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778.

Pang Wei Koh, Shiori Sagawa, Henrik Marklund, Sang Michael Xie, Marvin Zhang, Akshay Balsubramani, Weihua Hu, Michihiro Yasunaga, Richard Lanas Phillips, Irena Gao, Tony Lee, Etienne David, Ian Stavness, Wei Guo, Berton A. Earnshaw, Imran S. Haque, Sara Beery, Jure Leskovec, Anshul Kundaje, Emma Pierson, Sergey Levine, Chelsea Finn, and Percy Liang. 2021. WILDS: A benchmark of in-the-wild distribution shifts. In *International Conference on Machine Learning (ICML)*.

Haoliang Li, Sinno Jialin Pan, Shiqi Wang, and Alex C Kot. 2018. Domain generalization with adversarial feature learning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5400–5409.

Abraham Resler, Reuven Yeshurun, Filipe Natalio, and Raja Giryes. 2021. A deep-learning model for predictive archaeology and archaeological community detection. *Humanities and Social Sciences Communications*, 8(1):295.

Thea Sommerschield, Yannis Assael, John Pavlopoulos, Vanessa Stefanak, Andrew Senior, Chris Dyer, John Bodel, Jonathan Prag, Ion Androutsopoulos, and Nando de Freitas. 2023. Machine learning for ancient languages: A survey. *Computational Linguistics*, pages 1–44.

Nikita Srivatsan, Jason Vega, Christina Skelton, and Taylor Berg-Kirkpatrick. 2021. Neural representation learning for scribal hands of linear b. In *Document Analysis and Recognition–ICDAR 2021 Workshops: Lausanne, Switzerland, September 5–10, 2021, Proceedings, Part II 16*, pages 325–338. Springer.

Baochen Sun and Kate Saenko. 2016. Deep coral: Correlation alignment for deep domain adaptation. In *European Conference on Computer Vision (ECCV)*.

Chahan Vidal-Gorène and Aliénor Decours-Perez. 2021. A computational approach of armenian paleography. In *Document Analysis and Recognition – ICDAR 2021 Workshops*, pages 295–305, Cham. Springer International Publishing.

Vasiliy Yugay, Kartik Paliwal, Yunus Cobanoglu, Luis Sáenz, Ekaterine Gogokhia, Shai Gordin, and Enrique Jiménez. 2024. Stylistic classification of

cuneiform signs using convolutional neural networks. *it - Information Technology*. Publisher: De Gruyter Oldenbourg.

Samira Zare and Hien Van Nguyen. 2022. Removal of confounders via invariant risk minimization for medical diagnosis. In *Medical Image Computing and Computer Assisted Intervention – MICCAI 2022*, pages 578–587, Cham. Springer Nature Switzerland.

A Appendix

A.1 Tokenization

There are $7,000$ glyphs across different time periods. We use Chen et al. (2023) tokenization of text.

1. **word boundary.** Empty space is manually inserted between word. We by default keep the space by inserting.

2. **Logogram.** A tilde sign before a sign indicate it is a logogram. By default we differentiate whether a sign is syllable or logogram.

3. **Intrusions.** `(...)` indicates unknown number of signs is missing.

4. **Modifier**. In ATF, at-sign precedes a sign or group. For example, `@c` means curved.

5. **Compound.** `|GA`$_2$ $\sim$ `a×EN|`, means: "the a-allograph of the sign GA_2 containing sign EN".

6. **Breakage.** Hash tag is used to mark breakage.

B Details

B.1 OOD Experiments Details

1. **Raw**

   (a) **IRM**. We train for 30 epochs with a learning rate of 3e-5, an IRM lambda 1, and seed 2.

   (b) **CORAL**. We train for 30 epochs with a penalty weight 10 and seed 0.

   (c) **groupDRO** We train for 30 epochs with a learning rate of 3e-5 and seed 1.

2. **Front**

   (a) **IRM**. We train for 30 epochs with a learning rate of 3e-5, an IRM lambda 1, and seed 2.

   (b) **CORAL**. We train for 30 epochs with a penalty weight 10 and seed 2.

   (c) **groupDRO** We train for 30 epochs with a learning rate of 3e-5 and seed 2.

B.2 Hyperparameters and ablation study

We provide further analysis and conduct a comprehensive ablation study in the following section,

exploring the effects of hyperparameters, input feature selection, and the number of training examples on our model's performance.

As shown in Table **??**, larger model or larger resolution of input can boost model performance.

C Visualization of tablets counts

A full resolution with number annotated heatmap for the time periods preserved in each museum collection is shown in Figure 8.

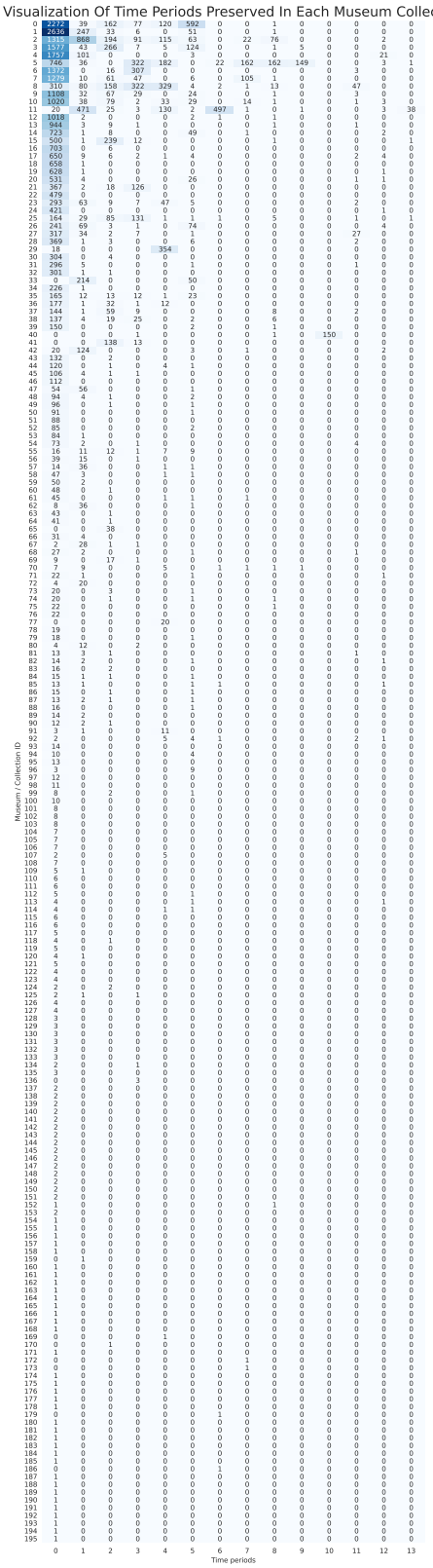

Figure 8: Visualization of the time periods preserved in each museum collection.