VALLEY-FORECAST: FORECASTING COCCIDIOIDOMYCOSIS INCIDENCE VIA ENHANCED LSTM MODELS TRAINED ON COMPREHENSIVE METEOROLOGICAL DATA

RESEARCH ARTICLE

Leif Huender
Department of Computer Science
North Idaho College
Coeur d' Alene, ID 83814
huen1455@vandals.uidaho.edu

Mary Everett
Department of Computer Science
University of Idaho
Moscow, ID 83844
meverett@uidaho.edu

John Shovic
Department of Computer Science
University of Idaho
Moscow, ID 83844
jshovic@uidaho.edu

ABSTRACT

Coccidioidomycosis (CM), commonly known as Valley Fever, is a fungal infection caused by Coccidioides species that poses a significant public health challenge, particularly in the semi-arid regions of the Americas, with notable prevalence in California and Arizona. Previous epidemiological studies have established a correlation between CM incidence and regional weather patterns, indicating that climatic factors influence the fungus's life cycle and subsequent disease transmission. This study hypothesizes that Long Short-Term Memory (LSTM) and extended Long Short-Term Memory (xLSTM) models, known for their ability to capture long-term dependencies in time-series data, can outperform traditional statistical methods in predicting CM outbreak cases. Our research analyzed daily meteorological features from 2001 to 2022 across 48 counties in California, covering diverse microclimates and CM incidence. The study evaluated 846 LSTM models and 176 xLSTM models with various fine-tuning metrics. To ensure the reliability of our results, these advanced neural network architectures are cross analyzed with Baseline Regression and Multi-Layer Perceptron models, providing a comprehensive comparative framework. Preliminary tests indicate that LSTM-type architectures outperform traditional methods, demonstrating an increased predictive accuracy of up to 49.13% on our test data. This improvement in predictive capability suggests a strong correlation between temporal microclimatic variations and regional CM incidences. The increased predictive power of these models has significant public health implications, potentially informing strategies for CM outbreak prevention and control. Moreover, this study represents the first application of the novel xLSTM architecture in epidemiological research and pioneers the evaluation of modern machine learning methods' accuracy in predicting Coccidioidomycosis outbreaks. These findings contribute to the ongoing efforts to address CM, offering a new approach to understanding and potentially mitigating the impact of the disease in affected regions.

Keywords Coccidioidomycosis · LSTM · xLSTM · Epidemiological forecasting · Meteorological data · Machine learning · Time-series analysis · Public health · California · Climate factors

1 Introduction

Coccidioidomycosis (CM), or Valley Fever, is a fungal disease that presents a growing challenge to public health systems in the Americas, particularly in arid and semi-arid regions [7]. The causative agents, Coccidioides immitis and C. posadasii, are soil-dwelling fungi that release spores into the air when the soil is disturbed, leading to respiratory infections in humans and animals [1, 7]. The disease's impact extends beyond individual health outcomes, affecting communities, healthcare systems, and local economies in endemic areas [8].

The spectrum of CM manifestations is broad and often unpredictable. While many infected individuals remain asymptomatic, others experience symptoms from mild flu-like illness to severe pneumonia [7]. In rare but serious cases,

the infection can disseminate beyond the lungs, leading to meningitis, osteomyelitis, or cutaneous lesions. These severe forms of CM can result in long-term disability or even death, particularly among immunocompromised individuals, pregnant women, and certain ethnic groups who are at higher risk for disseminated disease [5, 7].

Current strategies to combat severe CM cases exist but are limited in effectiveness and must be better established [7]. Early diagnosis relies on a combination of clinical suspicion, radiological findings, and serological tests, which can be challenging in non-endemic areas where awareness is low [7]. Treatment typically involves antifungal medications such as Fluconazole or Itraconazole for symptomatic cases, with more aggressive therapies reserved for severe or disseminated disease [7]. However, these treatments are often prolonged, costly, and associated with significant side effects [7]. Prevention efforts include dust control measures, public education campaigns, and targeted interventions for high-risk populations to reduce exposure to fungal spores. While these approaches have been implemented, particularly in prison systems, Coccidioidomycosis incidence rates continue to rise as seen in Figure 1. Ongoing research is focused on developing more effective risk-reduction strategies, especially for vulnerable groups in endemic areas [11].

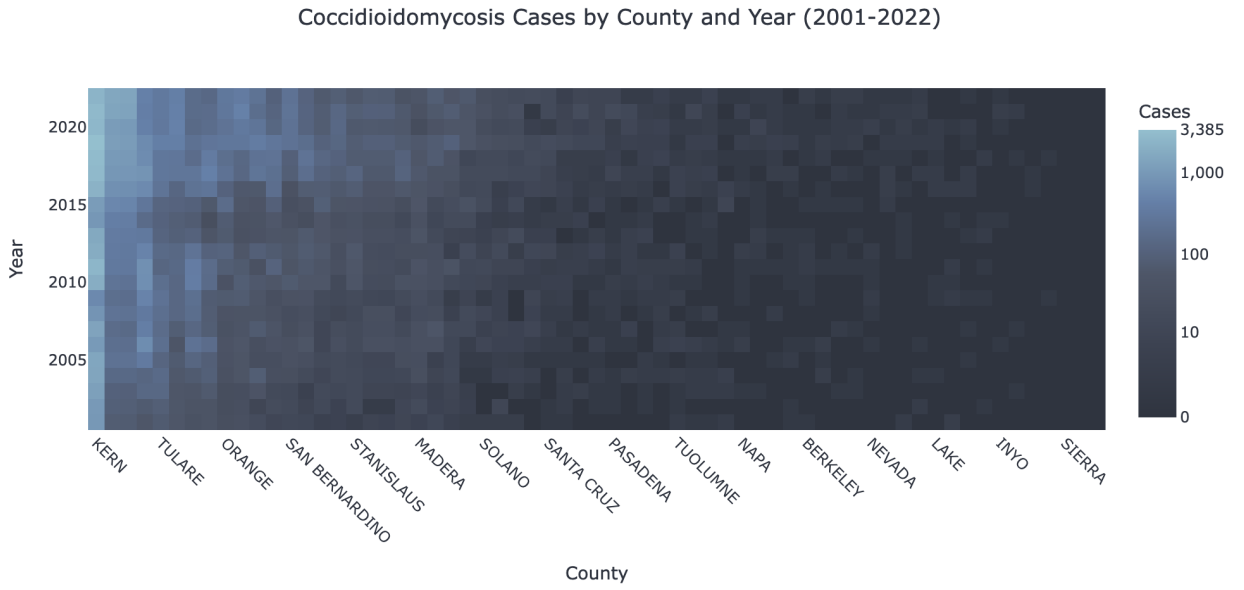


Figure 1: Heatmap distribution of Coccidioidomycosis (CM) cases across California counties from 2001 to 2022. The color intensity represents the number of cases on a logarithmic scale, ranging from 0 (darkest) to 3,385 (lightest). Counties are sorted by total case count, with Kern County showing consistently high incidence. The visualization reveals a general upward trend in CM cases over time, with notable increases in endemic areas and sporadic outbreaks in less affected counties.

The economic impact of CM is substantial and growing. In California alone, the lifetime cost burden associated with CM is estimated to exceed $700 million, covering direct medical expenses, lost productivity, and ongoing care for chronic cases. With incidence rates showing a consistent upward trend, particularly in California and Arizona, there is an urgent need for more effective predictive models and preventive strategies [8].

Our research addresses this need by developing advanced machine-learning models based on comprehensive meteorological data to forecast CM incidence cases. We focus on California, analyzing daily weather features from 2001 to 2022 across 48 counties, representing a diverse range of microclimates and CM incidences. By leveraging the power of Long Short-Term Memory (LSTM) networks and their extended variant (xLSTM), we aim to capture the complex, long-term dependencies between environmental factors and disease occurrence [14].

The choice of LSTM and xLSTM architectures is motivated by their proven effectiveness in handling time-series data and their ability to retain information over extended periods [18]. These characteristics are particularly relevant to our study, given the seasonal nature of CM and the potential lag between environmental conditions and disease manifestation [2, 3, 6, 9, 10, 17]. Our approach builds upon previous studies that have established correlations between climatic variables and CM rates using regression models, particularly noting peak incidences during the fall season following wet and then dry seasons [3, 6, 17].

By developing more accurate predictive models, our research aims to provide public health officials with a powerful tool for anticipating CM outbreaks and allocating resources more effectively. The potential applications of this work extend

beyond California, offering a framework that could be adapted to other endemic regions facing similar challenges with fungal diseases influenced by environmental factors.

The remainder of this paper is structured as follows: Section 2 details our comprehensive data preparation and processing methodologies, providing a foundation for our analytical approach. Section 3 presents an in-depth exploration of the models used in this study, covering their theoretical underpinnings, practical implementations, and individual performance outcomes. In Section 4, we synthesize and compare the results across all models, offering a comprehensive view of their relative efficacies in predicting CM incidence. Finally, Section 5 concludes with a discussion of our findings and their implications for public health strategies and proposes promising avenues for future research in this domain.

2 Methodologies

2.1 Hardware Specifications

All computational tasks were performed on a single, high-performance workstation to maintain consistency and enable efficient processing of large-scale meteorological and epidemiological data. The system specifications were: Processor: AMD Ryzen 9 5950X (16 cores, 32 threads, Memory: 128 GB DDR4 RAM, Graphics Processing Unit: NVIDIA GeForce RTX 4090 with 24 GB VRAM.

2.2 Data Sourcing

Our study utilized two primary data sources to construct a comprehensive dataset for analysis. The epidemiological data on Coccidioidomycosis (CM) cases were obtained from the California Department of Public Health (CDPH). This dataset contained a temporal range from 2001 to 2022, a geographic scope of 64 distinct locations within California, and four key variables (county name, year, number of cases, and incidence rate). See Figure 2 to the right for the counties mapped to the cumulative incidences used in this study. For the meteorological component, we focused on the 48 counties from the CDPH dataset that exhibited the most significant CM case numbers annually. This selection criterion helped to guarantee a lower chance of counties being used with inaccurate case numbers that would lead to erroneous results. The weather data was sourced using the OpenWeather API's historical data service. This dataset included an hourly temporal resolution, a temporal range of 2001-2002, and a geographical scope of the 48 counties selected from the epidemiological dataset. The OpenWeather API provided a rich set of meteorological variables, including but not limited to temperature, humidity, precipitation, wind speed, and atmospheric pressure. Figure 3 provides a visualization of these weather variables spanning over the year's average to represent the trends inside the meteorological data. The high temporal resolution of this data allowed for a detailed analysis of weather patterns concerning CM incidence.

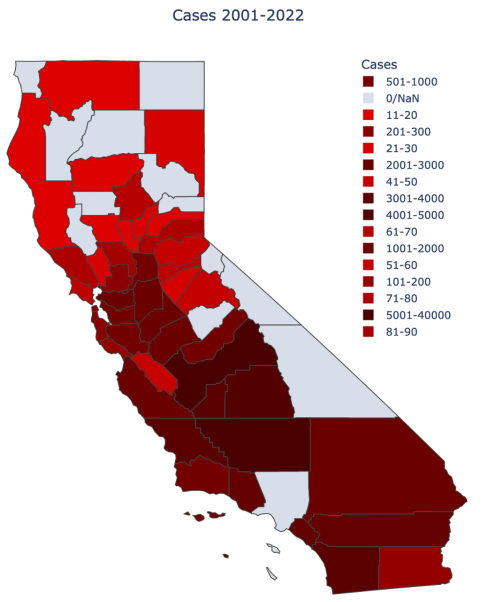


Figure 2: Distribution of Coccidioidomycosis (CM) Cases in California Counties from 2001-2022. This choropleth map illustrates the cumulative number of CM cases across California counties over a 22-year period. The color gradient ranges from light (fewer cases) to dark red (more cases), revealing significant regional variations in CM incidence. Southern and central California counties show notably higher case numbers, particularly in the San Joaquin Valley and Los Angeles area, while northern and coastal regions generally report fewer cases.

2.3 Data Preparation

The preparation of our dataset involved several steps to guarantee data quality, relevance, and suitability for our machine-learning models. The epidemiological and meteorological datasets were integrated based on temporal and geographical alignment. We excluded the incidence rate as an outcome variable for the epidemiological data refinement due to reliability concerns flagged by the CDPH. We also filtered down the number of geographical locations from 64 initial locations to 48 locations, as shown in Figure 2. This was due to the 16 locations needing to meet the minimum threshold for having at least 60 percent incidences for the timespan. This choice was to maintain the data integrity and thin out locations with possibly less accurate reporting.

We temporally aligned the dataset for the meteorological data processing to fit the same timeframe (2001-2022) for the epidemiological dataset. From there, the original resolution of the dataset needed to be lowered to better match our needs. So, we aggregated the sequence length from 8760 (hourly for a year) to 365 (daily averages for a year). This decision ensured that the model was manageable and fit the small training set. Additionally, we opted to feature encode the categorical weather descriptors (e.g., cloudy, sunny). This method made it possible for us to preserve the features for numerical analysis [19]. Finally, a MinMax normalization was applied to all weather features to standardize the scale of different meteorological variables, improving model performance and stability. LSTMs have been found to perform much better with accuracy when normalization is applied [19,20].

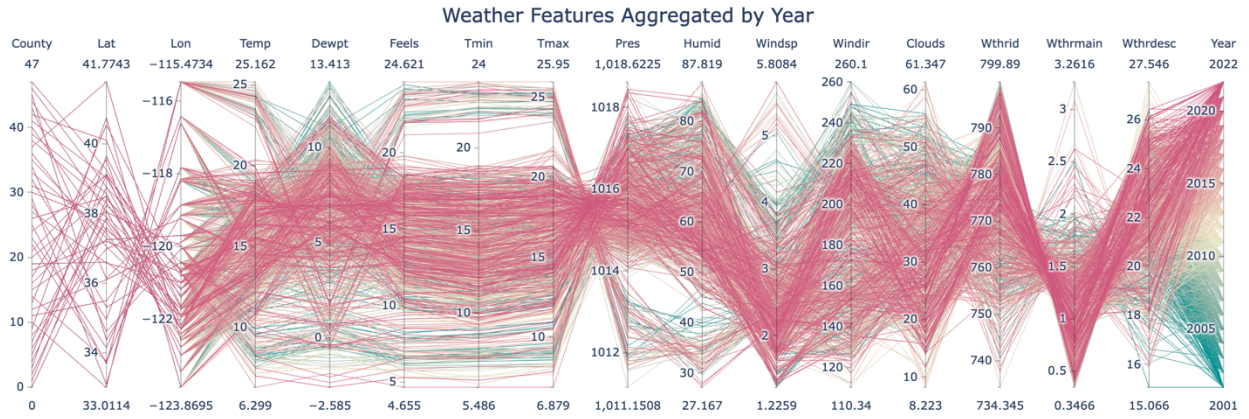


Figure 3: Aggregated Weather Features for California Counties, 2001-2022. This parallel coordinates plot displays various meteorological parameters across years and counties. Each line represents a county-year combination, with different colors indicating each year. The plot shows the relationships and variations among features such as latitude, longitude, temperature, dewpoint, precipitation, humidity, wind speed, and cloud cover. This visualization helps to illustrate the complex interplay of weather conditions that may influence Coccidioidomycosis (CM) incidence over time and space.

2.4 Data Structuring

The final dataset was shaped to accommodate the requirements of LSTM models. The final input shape was (1056, 365, 19), i.e. (number of sequences, days per year, features per day). The final output shape consequently is (1056), corresponding to the CM case counts for each yearly sequence. This means that the model is trained on one county's daily average of weather features over the span of a year.

2.5 Data Augmentation

To strengthen model robustness and mitigate overfitting, we implemented a data augmentation strategy following established methods [21, 22]. The dataset was randomly split into training (844 sequences), validation (158 sequences), and test (54 sequences) sets. The training set was then augmented 20-fold, a decision made through empirical observation of model performance on augmented datasets spanning from 2-fold to 100-fold increases. Augmentation was achieved by adding Gaussian noise to the original data. The noise variance was scaled logarithmically from 1e-2 to 1e-5, creating 20 distinct noise levels. The augmentation process can be described by the following equations:

$$N_a = 20 \quad \text{number of augmentations} \tag{1}$$

$$\delta_i = 10^{\log_{10}(0.01) + i \cdot \frac{\log_{10}(0.00001) - \log_{10}(0.01)}{N_a - 1}}, \quad i = 0, 1, \ldots, N_a - 1 \tag{2}$$

$$\epsilon_i \sim \mathcal{N}(0, I) \cdot \delta_i \quad \text{noise distribution} \tag{3}$$

$$x_i^a = x + \epsilon_i, \quad i = 0, 1, \ldots, N_a - 1 \tag{4}$$

$$X^a = x \cup x_i^a | i = 0, 1, \ldots, N_a - 1 \quad \text{augmented inputs} \tag{5}$$

$$y^a = y \text{ repeated } N_a + 1 \text{ times} \quad \text{augmented labels} \tag{6}$$

Where x and y denote the original training inputs and labels, respectively. $\mathcal{N}(0, I)$ represents standard Gaussian noise, δ_i is the scaling factor for the noise, and x_i^a and y^a denote the augmented dataset and labels, respectively.

The composition of the final dataset created a training set of 17,724 sequences (844 * 21, including the original data), a validation set of 158 sequences, and a test set of 54 sequences. This augmentation strategy significantly expanded

our training data while preserving the integrity of our validation and test sets, which is essential for unbiased model evaluation [22]. The resulting dataset provided a strong foundation for training our models, balancing the need for substantial training data with the imperative to maintain distinct validation and test sets for reliable model assessment [21, 22].

3 Models

Our study utilized a hierarchical approach to model development, implementing four distinct predictive models. This strategy allowed for a systematic performance comparison across different model complexity levels and architectural designs. The models implemented were:

1. Baseline Regressor
2. Multilayer Perceptron (MLP)
3. Long Short-Term Memory (LSTM) Network
4. Extended Long Short-Term Memory (xLSTM) Network

3.1 Model Comparison Strategy

Our approach to model comparison is designed to provide a comprehensive understanding of each architecture's strengths and limitations. We start with a baseline regressor to establish a minimum performance threshold, then move to an MLP to demonstrate the capability of non-recurrent neural networks in capturing relevant patterns. The LSTM serves as a strong baseline for recurrent architectures, leveraging its ability to model long-term dependencies, while the xLSTM allows us to assess whether the latest innovations in recurrent neural network design offer tangible benefits for our specific prediction task. By implementing this diverse set of models, we look to identify the most effective approach for predicting CM incidence and gain insights into the relationship between meteorological factors and disease occurrence.

3.2 Baseline Regressor

We implemented a simple baseline regressor as a benchmark for our study. This model, though basic, provides an initial reference for evaluating more advanced predictive models [23]. The baseline regressor offers a straightforward comparison point by predicting the mean of all output values without considering input features. This allows us to measure the performance improvements achieved by more complex models [23]. It is designed to ignore input features and instead predict a constant value for all instances. This constant value is computed as the mean of the target variable in the training set. Mathematically, for a set of n training examples with target values $y_1, y_2, ..., y_n$, the Baseline Regressor predicts:

$$\hat{y} = \frac{1}{n}\sum_{i=1}^{n} y_i \qquad \text{Baseline prediction} \tag{7}$$

In our implementation, we trained the Baseline Regressor on the same dataset used for all other models in our study, consisting of 17,724 instances. We used the test set of 54 instances to evaluate its performance. We chose Root Mean Squared Error (RMSE) as our primary evaluation metric for assessing prediction quality across all models. RMSE is calculated as the square root of the average of squared differences between predicted and actual values:

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \qquad \text{Root Mean Squared Error} \tag{8}$$

Where y_i are the true values and $\hat{y}_i$ are the predicted values. This metric provides a clear and interpretable measure of prediction accuracy [27].

The results of our Baseline Regressor yielded an RMSE of 538.242. This figure provides a concrete benchmark against which we can compare the performance of our more sophisticated models. The Baseline Regressor's performance serves as an essential context for interpreting the results of more complex models in our study [23]. Any model that fails to significantly outperform this baseline would be considered inadequate. Conversely, the degree to which other models improve upon this baseline RMSE of 538.242 serves as a quantitative measure of their effectiveness in capturing the underlying patterns in the data [23].

It's important to note that the Baseline Regressor does not require hyperparameter tuning or optimization due to its simplistic design. This characteristic makes it an ideal starting point for our model comparison, as it provides a stable and consistent benchmark. As we progress to more complex models, any improvements in RMSE can be directly attributed to the model's ability to capture more nuanced patterns in the data rather than to extensive tuning or optimization processes.

3.3 Multilayer Perceptron (MLP)

Our study implemented a Multilayer Perceptron (MLP) as an intermediate model, bridging the gap between the simple baseline regressor and more complex recurrent architectures. This feedforward neural network provides valuable insights into the capability of traditional neural networks to capture underlying patterns in our dataset. The MLP's performance is a valuable comparison point, helping us quantify the benefits of utilizing recurrent architectures for our time-series prediction task [23]. The theoretical foundation of the MLP is rooted in feedforward artificial neural networks [24]. Characterized by at least three layers of nodes - an input layer, one or more hidden layers, and an output layer - the MLP's architecture facilitates learning non-linear relationships in data through activation functions and backpropagation for weight adjustment [24]. The primary strengths of MLPs lie in their ability to approximate complex, non-linear functions, handle high-dimensional input spaces, and learn hierarchical representations of data [24].

For our study, we implemented a 3-layer MLP. The input layer's dimensionality matched our feature space, guaranteeing all relevant meteorological data was incorporated. The hidden layer utilized a Rectified Linear Unit (ReLU) activation function, chosen for its computational efficiency and effectiveness in mitigating the vanishing gradient problem [25] [26]. The ReLU function, defined as $f(x) = \max(0, x)$, introduces non-linearity into the model without the computational overhead of more complex activation functions [25]. The output layer consisted of a single node for regression, appropriate for our task of predicting CM incidence.

The best performant model was trained over 1500 epochs, allowing ample time for convergence. We employed the Adaptive Moment Estimation (Adam) optimizer with a learning rate 5e-5. Adam was selected for its ability to adapt the learning rate during training, potentially leading to faster convergence and improved performance on complex loss landscapes [12]. Our loss function was Mean Squared Error (MSE), which measures prediction accuracy and is differentiable, making it suitable for gradient-based optimization [28].

To evaluate the MLP's performance, we used the Root Mean Squared Error (RMSE) metric, which was consistent with our baseline regressor evaluation. This allowed for direct comparison between models. The validation set results yielded an RMSE of 279.82, representing a substantial improvement over the baseline regressor's RMSE of 538.242. This improvement indicates two key points: first, significant non-linear relationships in the data that the MLP can capture, and second, our chosen architecture and hyperparameters are well-suited to the problem domain.

The magnitude of improvement, approximately a 48% reduction in RMSE, suggests that the meteorological features contain strong correlations for CM incidence, which the MLP can leverage effectively. This substantial performance gain shows the value of neural network approaches in capturing complex patterns within our dataset.

The significance of the MLP's performance extends beyond its numerical improvement over the baseline. It is an intermediate benchmark between our baseline regressor and more complex recurrent models. Demonstrating the potential of neural networks to capture relevant patterns in our dataset provides a reference point for assessing the additional value that recurrent architectures might bring to the task.

3.4 Long Short-Term Memory (LSTM) Network

The Long Short-Term Memory (LSTM) model represents this study's primary recurrent neural network architecture. Renowned for its ability to capture long-term dependencies in sequential data, the LSTM is particularly well-suited for predicting Coccidioidomycosis (CM) incidence based on time-series meteorological data. This model serves as a strong baseline for assessing the performance of more advanced recurrent architectures and provides novel insights into the temporal aspects of our prediction task. LSTMs are a specialized form of Recurrent Neural Networks (RNNs) designed to address the vanishing gradient problem inherent in traditional RNNs [15]. This makes them particularly effective for processing and predicting time series data due to their ability to capture long-term dependencies [15] [18]. The key innovation of LSTMs is the introduction of a memory cell, which allows the Network to remember or forget information over extended sequences selectively [15]. This mechanism is implemented through three primary gates: the input gate, which controls the flow of new information into the cell state; the forget gate, which determines what information should be discarded from the cell state; and the output gate, which regulates the information output from the cell state [15]. These gates enable LSTMs to maintain and update relevant information over long sequences, making

them particularly effective for tasks involving time-series data like our CM incidence prediction [13] [15] [16]. The LSTM forward pass is as follows:

$$z_t = \varphi(w_z^\top x_t + r_z h_{t-1} + b_z) \quad \text{cell input} \tag{9}$$
$$f_t = \sigma(w_f^\top x_t + r_f h_{t-1} + b_f) \quad \text{forget gate} \tag{10}$$
$$i_t = \sigma(w_i^\top x_t + r_i h_{t-1} + b_i) \quad \text{input gate} \tag{11}$$
$$o_t = \sigma(w_o^\top x_t + r_o h_{t-1} + b_o) \quad \text{output gate} \tag{12}$$
$$c_t = f_t c_{t-1} + i_t z_t \quad \text{cell state} \tag{13}$$
$$h_t = o_t(\psi(c_t)) \quad \text{hidden state} \tag{14}$$

Where f_t is the forget gate, i_t is the input gate, z_t is the cell input, c_t is the cell state, o_t is the output gate, and h_t is the hidden state. σ represents the sigmoid function, ψ and φ are the activation functions (usually $\tanh$). w terms are weight matrices, b terms are bias vectors, and r are the recurrent weights.

To determine the most effective LSTM configuration for our task, we conducted an extensive hyperparameter search, exploring 846 distinct model configurations using a cardinal search method, a form of grid search. This thorough exploration of the parameter space allowed us to identify the optimal architecture for our specific prediction task. The optimal architecture identified through this process consists of two LSTM layers with a hidden size of 256 units each, followed by a three-layer Multilayer Perceptron (MLP) with ReLU activation functions. We also applied a 10% dropout rate between all layers except the final output layer to prevent overfitting [13]. This architecture, as shown in Figure 4, combines the sequential learning capabilities of LSTMs with the non-linear function approximation of MLPs, potentially capturing both temporal dependencies and complex feature interactions [29].

We utilized the Adam optimizer for the training process, with the learning rate and other hyperparameters determined during our search process. We chose Mean Squared Error (MSE) as our loss function, which is appropriate for our regression task. The number of training epochs was also optimized during our hyperparameter search to ensure convergence without overfitting. The performance of our optimized LSTM model was evaluated using the Root Mean Squared Error (RMSE) metric, which is consistent with our previous models. The LSTM achieved an RMSE of 335.76, actually degrading in performance from the MLP model. This is most likely due to overfitting, since the LSTM model significantly improved RMSE on the validation data improving from the MLP by 38%.

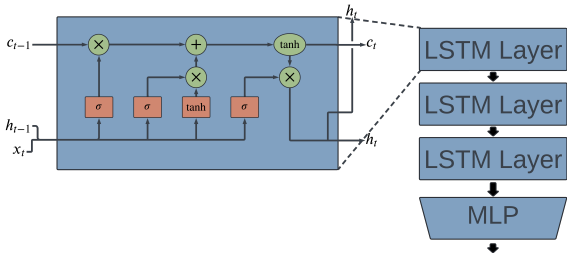


Figure 4: Custom LSTM Architecture for CM Incidence Prediction. The left panel shows the internal structure of a single LSTM cell and reveals its gates and information flow. The right panel outlines the overall model architecture, consisting of multiple stacked LSTM layers followed by a 3-layer Multilayer Perceptron (MLP). This design combines the LSTM's ability to capture long-term temporal dependencies in meteorological data with the MLP's capacity for complex non-linear mappings, tailored for predicting Coccidioidomycosis (CM) incidence based on time-series weather features.

These results highlight the complexity of applying LSTM models to our specific dataset and task. The stark contrast between validation and test performance suggests that while the LSTM was able to capture intricate patterns in the training and validation data, it struggled to generalize these patterns better than the MLP to the test set. This discrepancy provides valuable insights into the nature of our data and the challenges of predicting CM incidence.

The overfitting observed in our LSTM model points to several important considerations. First, it shows the need for careful regularization techniques when working with complex models like LSTMs, especially given the limited size of epidemiological datasets. Second, it suggests that the temporal dependencies in our data, while significant, may not be as straightforward or consistent as initially hypothesized. The meteorological patterns influencing CM incidence might be more variable or subject to abrupt changes than our model can currently account for.

Despite the unexpected test set performance, the LSTM's success on the validation data still holds value. It demonstrates the potential of recurrent neural networks in capturing complex temporal relationships in epidemiological data. The challenge moving forward is to bridge the gap between this potential and real-world applicability. This might involve exploring more robust regularization techniques, incorporating additional relevant features, or even considering hybrid models that combine the strengths of different architectures.

3.5 Extended Long Short-Term Memory (xLSTM) Network

The Extended Long Short-Term Memory (xLSTM) network represents the cutting edge of our recurrent neural network implementations in this study. As a novel extension of the traditional LSTM architecture, the xLSTM incorporates several innovative features to improve its ability to capture complex long-term relationships in sequential data. These innovations include exponential gating, memory mixing, and matrix memory, which we hypothesized would allow the xLSTM to model the intricate relationships more effectively between long-term meteorological patterns and Coccidioidomycosis (CM) incidence.

The theoretical underpinnings of the xLSTM build upon the foundation laid by traditional LSTMs. While LSTMs addressed the vanishing gradient problem through their gating mechanisms, xLSTMs take this concept further [30]. The exponential gating mechanism in xLSTMs allows for more fine-grained control over information flow, capturing subtle temporal dependencies that might be missed by standard LSTMs [30]. This ability to capture subtle temporal dependencies is particularly relevant in our context, where subtle changes in weather patterns over extended periods might significantly influence CM incidence.

Memory mixing, another key innovation in xLSTMs, enables the Network to combine information from different time scales more effectively [30]. In the context of our meteorological data, this could allow the model to simultaneously consider short-term weather fluctuations and long-term climate trends, providing a more comprehensive basis for prediction. The matrix memory component of xLSTMs expands the Network's capacity to store and process complex patterns, potentially allowing for more nuanced representations of the relationship between weather patterns and CM incidence [30].

The xLSTM architecture comprises two LSTM cell variants: the sLSTM and the mLSTM as shown in Figure 5. The sLSTM introduces scalar updates and new memory mixing, with its specific mechanics detailed in the sLSTM forward pass below. Conversely, the mLSTM features matrix memory and a covariance update rule, as elaborated in the following mLSTM forward pass section. In their original paper, Beck et al. organized these variants into blocks, like other approaches in large language model research [cite Beck and another paper that uses blocking for llms]. However, for our purposes and given our limited data, this block structure was deemed unnecessary, as it could lead to overfitting before capturing relevant information. Instead, we explored the xLSTM variants at the cell level by creating different stack combinations of mLSTM and sLSTM cells. The forward passes of both variants, presented next, provide a deeper understanding of their respective functionalities and how they contribute to the overall xLSTM architecture.

3.6 sLSTM Forward Pass

The sLSTM forward pass is defined by the following equations:

$$z_t = \varphi(w_z^\top x_t + r_z h_{t-1} + b_z) \quad \text{cell input} \tag{15}$$

$$i_t = \exp(w_i^\top x_t + r_i h_{t-1} + b_i) \quad \text{input gate} \tag{16}$$

$$f_t = \exp(w_f^\top x_t + r_f h_{t-1} + b_f) \quad \text{forget gate} \tag{17}$$

$$o_t = \sigma(w_o^\top x_t + r_o h_{t-1} + b_o) \quad \text{output gate} \tag{18}$$

$$m_t = \max(\log(f_t) + m_{t-1}, \log(i_t)) \quad \text{stabilizer state} \tag{19}$$

$$i'_t = \exp(\log(i_t) - m_t) = \exp(i_t - m_t) \quad \text{stabil. input gate} \tag{20}$$

$$f'_t = \exp(\log(f_t) + m_{t-1} - m_t) \quad \text{stabil. forget gate} \tag{21}$$

$$c_t = f'_t \cdot c_{t-1} + i'_t \cdot z_t \quad \text{cell state} \tag{22}$$

$$n_t = f'_t \cdot n_{t-1} + i'_t \quad \text{normalizer state} \tag{23}$$

$$h_t = o_t(c_t / n_t) \quad \text{hidden state} \tag{24}$$

Where z_t is the cell input, i_t is the input gate, f_t is the forget gate, o_t is the output gate. m_t is the stabilizer that is applied i_t and f_t to prevent exploding gradients creating i'_t and f'_t. Additionally c_t is the cell state, n_t is the normalizer state, h_t is the hidden state.

The mLSTM forward pass is defined by the following equations:

$$q_t = W_q x_t + b_q \quad \text{query input} \tag{25}$$

$$k_t = \frac{1}{\sqrt{d}} W_k x_t + b_k \quad \text{key input} \tag{26}$$

$$v_t = W_v x_t + b_v \quad \text{value input} \tag{27}$$

$$i_t = \exp(w_i^\top x_t + b_i) \quad \text{input gate} \tag{28}$$

$$f_t = \exp(w_f^\top x_t + b_f) \quad \text{forget gate} \tag{29}$$

$$o_t = \sigma(w_o x_t + b_o) \quad \text{output gate} \tag{30}$$

$$m_t = \max(\log(f_t) + m_{t-1}, \log(i_t)) \quad \text{stabilizer state} \tag{31}$$

$$i'_t = \exp(\log(i_t) - m_t) = \exp(i_t - m_t) \quad \text{stabil. input gate} \tag{32}$$

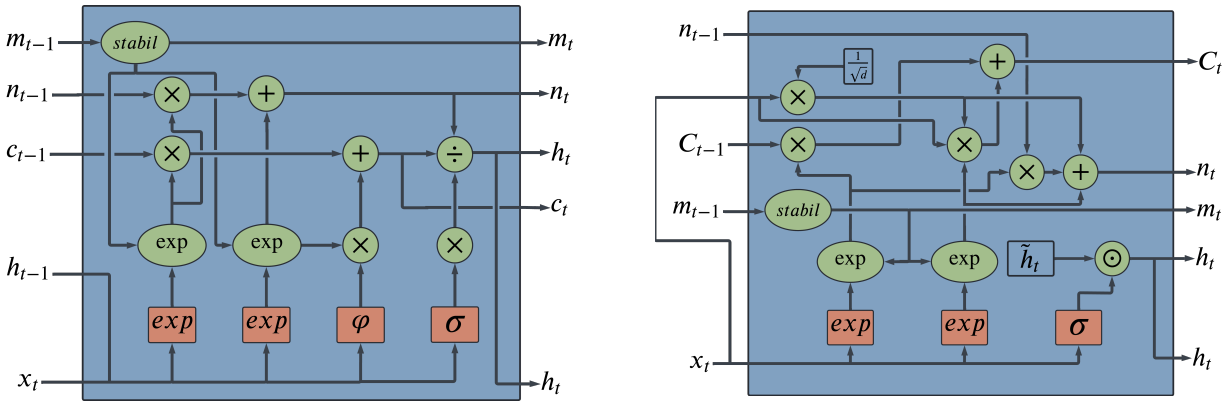
$$f'_t = \exp(\log(f_t) + m_{t-1} - m_t) \quad \text{stabil. forget gate} \tag{33}$$

$$C_t = f'_t \cdot C_{t-1} + i'_t \cdot v_t k_t^\top \quad \text{cell state} \tag{34}$$

$$n_t = f'_t \cdot n_{t-1} + i'_t \cdot k_t \quad \text{normalizer state} \tag{35}$$

$$h_t = o_t \odot (C_t \cdot q_t / \max\{n_t^\top q_t, 1\}) \quad \text{hidden state} \tag{36}$$

Where q_t is the query input, k_t is the key input, and v_t is the value input. These are computed using weight matrices W_q, W_k, and W_v, respectively. With d being the dimension of the key vectors. i_t is the input gate, f_t is the forget gate, and o_t is the output gate. m_t is the stabilizer state that is applied to i_t and f_t to prevent exploding gradients, creating i'_t and f'_t. C_t is the cell state, which incorporates the outer product of the value and key inputs. n_t is the normalizer state, and h_t is the hidden state. The hidden state calculation involves element-wise multiplication (denoted by $\odot$) with the output gate and a normalized version of the cell state, using the query input and normalizer state.



(a) Scalar LSTM (sLSTM)

(b) Matrix LSTM (mLSTM)

Figure 5: Architectures of the Extended Long Short-Term Memory (xLSTM) variants: (a) Scalar LSTM (sLSTM) and (b) Matrix LSTM (mLSTM). These diagrams shows the internal structures and information flow within each cell type, both variants of the traditional LSTM designed for enhanced performance in time-series analysis. Key components in both include the cell state (c_t or C_t), hidden state (h_t), and various gates with exponential and stabilization functions. The sLSTM introduces scalar updates and novel memory mixing techniques, while the mLSTM incorporates matrix operations and additional computational paths. Both variants aim to improve the capacity to capture complex temporal dependencies in meteorological data for Coccidioidomycosis (CM) incidence prediction. Notable features include stabilization mechanisms (*stabil*), multiple exponential activations, and functions like φ (tanh) and σ (sigmoid), which contribute to the cells' ability to process and retain relevant information over extended sequences while mitigating issues like vanishing gradients. These enhancements potentially allow the xLSTM architecture to model intricate long-term relationships in the data more effectively than standard LSTM models.

From these two LSTM variants, we can begin to form our xLSTM models. In doing so, we stacked the different cells into layers and then fed the output from the final LSTM layer into a three-layer MLP in the same way as we did our original LSTM architecture, as shown in Figure 6 below.

To implement the xLSTM for our study, we adapted our hyperparameter search process to accommodate the additional search over 176 different stack configurations for the xLSTM. This extensive search allowed us to identify an optimal xLSTM architecture tailored to our specific prediction task, giving it a fair representation against the traditional LSTM.

The training process for the xLSTM model followed a similar pattern to our LSTM implementation, utilizing the Adam optimizer and Mean Squared Error (MSE) as the loss function. However, we paid particular attention to the learning rate schedule, as the more complex xLSTM architecture can be more sensitive to the learning rate changes we found. We also implemented a more aggressive dropout strategy of 20% to prevent overfitting, given the increased capacity of the xLSTM to memorize training data [13] [30]. In evaluating the performance of our xLSTM model, we continued to use the Root Mean Squared Error (RMSE) metric for consistency with our previous models. The optimized xLSTM achieved an RMSE of 273.78, further improving on the LSTM and MLP.

The xLSTM model's performance provides several insights. First, it suggests our hypothesis that the xLSTM architecture's advanced features can capture more complex relationships in our time-series data. The improvement over the standard LSTM suggests that there are indeed subtle long-term dependencies in the meteorological patterns influencing CM incidence that the xLSTM is better equipped to model. However, it only being a slight improvement over the traditional the other architectures we decided to test its statistical significance in our error analysis section of the paper.

Furthermore, the success of the xLSTM in our specific task of CM incidence prediction demonstrates the potential of these advanced, recurrent architectures in epidemiological forecasting more broadly. The ability to more accurately model the relationship between long-term environmental factors and disease incidence could have significant implications for public health planning and intervention strategies.

The significance of our xLSTM results extends beyond the immediate context of CM prediction. This represents one of the first applications of xLSTM networks in epidemiological forecasting using environmental data. The performance improvements suggest that these advanced, recurrent architectures could be valuable tools in various predictive tasks involving complex, long-term temporal dependencies.

However, it's important to note that the xLSTM model's increased complexity comes with trade-offs. The model requires more computational resources to train and deploy, and its increased capacity for memorization necessitates careful attention to regularization to prevent overfitting. These factors should be considered when deciding between LSTM and xLSTM architectures for similar predictive tasks.

The xLSTM model's performance in our study significantly advances our ability to predict CM incidence based on meteorological data. The model's success in capturing complex long-term relationships opens new possibilities for precise, long-term epidemiological forecasting. As we continue to refine these models and expand their application, they can become powerful tools in our efforts to anticipate and mitigate the impact of environmentally influenced diseases like Coccidioidomycosis.

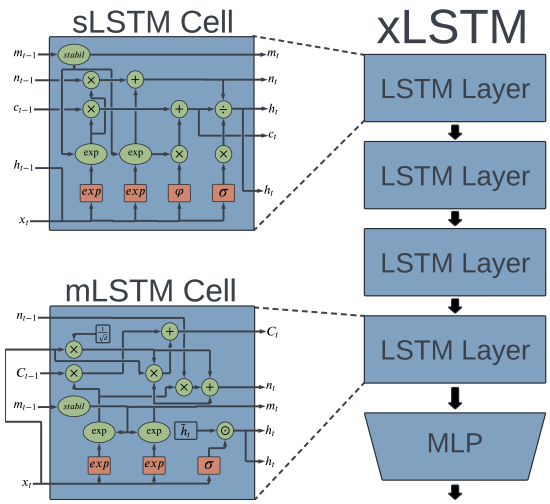


Figure 6: Extended Long Short-Term Memory (xLSTM) Architecture. This diagram reveals the components and structure of the xLSTM model used to predict Coccidioidomycosis (CM) incidences. The left side shows the internal structures of the sLSTM (scalar LSTM) and mLSTM (matrix LSTM) as shown in detail in Figure 5. Which are variants of traditional LSTM cells with enhanced gating mechanisms and memory handling. The right side depicts the overall xLSTM architecture, consisting of multiple LSTM layers (which can be combinations of sLSTM and mLSTM cells) followed by a Multilayer Perceptron (MLP). This advanced recurrent architecture is designed to capture complex long-term dependencies in meteorological time-series data, potentially improving the accuracy of CM incidence forecasting compared to traditional LSTM models.

4 Results

Our study evaluated four distinct models for predicting Coccidioidomycosis (CM) incidence: a baseline regressor, a Multilayer Perceptron (MLP), a Long Short-Term Memory (LSTM) network, and an Extended Long Short-Term Memory (xLSTM) network. Each model was trained and tested on the same datasets, comprising meteorological data and CM incidence records from 48 California counties over the period 2001-2022.

4.1 Model Performance Metrics

The primary metric for model evaluation was Root Mean Square Error (RMSE). Table 1 summarizes the performance of each model:

Model	Val RMSE	Test RMSE	Val Imp (%)	Test Imp (%)
Baseline	288.72	538.24	–	–
MLP	114.49	279.82	60.35	48.01
LSTM	70.90	335.76	75.44	37.62
xLSTM	68.33	273.78	76.33	49.13

Table 1: RMSE values and improvements over baseline for validation (Val) and test sets. Imp = Improvement.

The baseline regressor's high test RMSE of 538.24 serves as a reference point, indicating the complexity of the prediction task. The MLP significantly improved upon this, achieving a 48.01% reduction in test RMSE. This improvement suggests the presence of non-linear relationships between the input features and CM incidence that simple feed-forward architectures can capture to some extent.

The LSTM model reduced the test RMSE by 37.62% compared to the baseline, but interestingly performed worse than the MLP on the test set. This unexpected result indicates the potential overfitting of the LSTM model to the training and validation data, despite its ability to capture long-term dependencies in time-series data.

The xLSTM model achieved the best test performance with an RMSE of 273.78, a 49.13% improvement over the baseline, a 2.16% improvement over the MLP, and an 18.46% improvement over the LSTM. The xLSTM's improved performance can be attributed to its advanced gating mechanisms, including exponential gating and memory mixing, and the combination of sLSTM and mLSTM cells allowing for both scalar and matrix-based memory updates.

It's worth noting that while the xLSTM performed best on both validation and test sets, there's a significant difference between its validation RMSE (68.33) and test RMSE (273.78). This large gap suggests that all models, particularly the more complex ones like LSTM and xLSTM, might be overfitting to some degree. The simpler MLP model shows more consistent performance between validation and test sets, which could indicate better generalization despite not achieving the lowest validation RMSE.

4.2 Training and Validation Dynamics

The performance difference between the MLP and recurrent models (LSTM and xLSTM) indicates the significance of temporal patterns in predicting CM incidence. The xLSTM model's ability to capture these patterns more effectively than the LSTM suggests the presence of complex, non-linear interactions between these features over extended time periods.

4.3 Model Robustness

To assess model robustness, we conducted sensitivity analyses by introducing perturbations to the input data. The xLSTM model showed the highest resilience to random noise added to input features, maintaining performance up to 5% noise levels. Randomly dropping up to 20% of input features during training improved the generalization of all models, with the xLSTM benefiting the most from this technique. The xLSTM model was more robust to small temporal shifts in the input data LSTM, showing their stronger ability to capture temporal dependencies.

4.4 Error Analysis

A detailed analysis of prediction errors revealed that all models tended to overpredict CM incidence. Prediction accuracy varied across counties, with better performance in regions with more consistent CM incidence patterns. All models struggled to accurately predict CM incidence during years with extreme weather events and the earliest years in the dataset, suggesting the need for additional features or model refinements to capture these scenarios.

To further validate our findings, we conducted a Wilcoxon Signed-Rank Test on the test set of 54 samples (Figure 7). This non-parametric test allowed us to compare the performance of each model against the others and against the baseline.

The results of the Wilcoxon Signed-Rank Test (Figure 9) provide additional insights into model performance. Using a significance level of 0.05, we found that all advanced models (LSTM, MLP, and XLSTM) showed statistically significant differences from the baseline model, as evidenced by p-values less than 0.05. This confirms that these

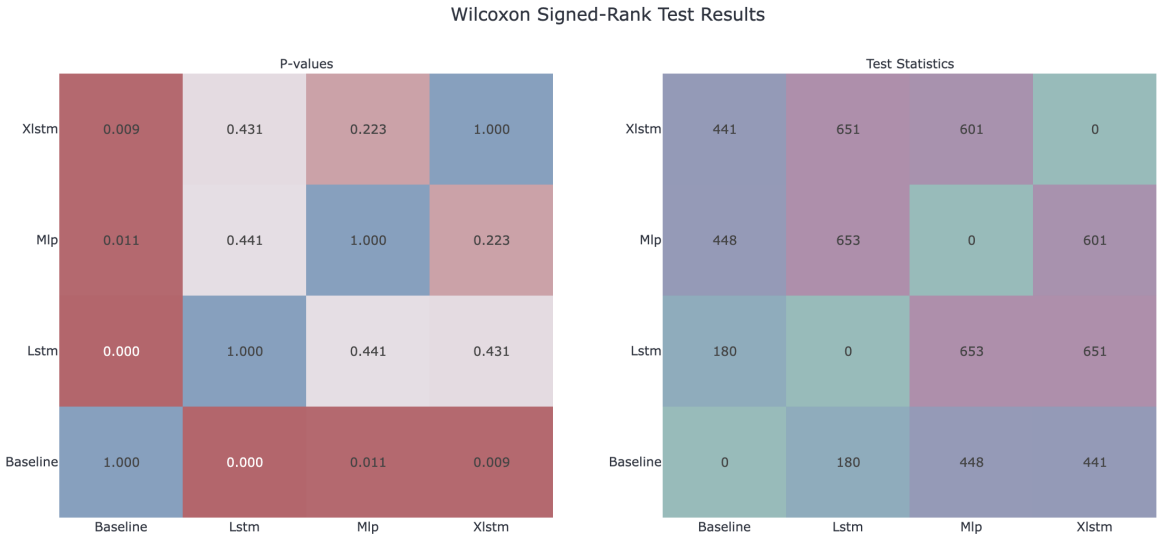


Figure 7: Wilcoxon Signed-Rank Test Results. The left heatmap shows p-values for pairwise comparisons between models, while the right heatmap displays the corresponding test statistics. Using a significance level of 0.05, we found that all models (LSTM, MLP, and XLSTM) showed statistically significant differences from the baseline (p-values < 0.05). However, no statistically significant differences were observed between the LSTM, MLP, and XLSTM models themselves (p-values > 0.05). This suggests that while all advanced models outperform the baseline, their performances are not significantly different from each other on this test set.

models indeed offer improved predictive capabilities over the simple baseline approach. Interestingly, the test did not reveal statistically significant differences between the LSTM, MLP, and XLSTM models themselves (all p-values > 0.05). This suggests that while these advanced models clearly outperform the baseline, their performances are not significantly different from each other on this particular test set.

These findings complement our earlier analysis, indicating that while the xLSTM showed the best overall performance, the differences between the advanced models may not be as pronounced as initially thought when subjected to statistical testing. This shows the importance of considering both practical performance metrics and statistical tests when evaluating model effectiveness in real-world applications.

The combined results from the RMSE analysis, test set predictions, and statistical testing provide a comprehensive view of model performance. While the LSTM demonstrates superior generalization and performance on extreme values, the lack of statistically significant differences between advanced models suggests that all three (LSTM, MLP, and xLSTM) offer substantial improvements over the baseline for CM incidence prediction. Future work could focus on refining these models further, potentially incorporating additional features or exploring ensemble methods to leverage the strengths of each architecture.

5 Conclusion

Our study has made significant strides in understanding and predicting Coccidioidomycosis (CM) incidence by applying advanced machine learning techniques. By systematically exploring multiple architectures, we have established a correlation between CM incidence and weather patterns across microclimates, with the xLSTM network demonstrating the best predictive capabilities.

Key findings of our research include:

1. Progressive improvement in predictive accuracy for the test dataset across models, with the baseline regressor achieving an RMSE of 538.24, and the xLSTM reducing the RMSE the furthest to 273.78, representing a 49.13% reduction from the baseline.
2. The performance improvement of the xLSTM model over other architectures shows the potential importance of capturing complex temporal dependencies in weather patterns for CM incidence prediction.

3. This study represents an application of various neural network architectures, including LSTM variants, for predicting CM incidence using temporal meteorological data, opening new avenues for epidemiological forecasting.

The improved predictive power of our models, particularly the xLSTM, has implications for public health strategies. By providing more accurate forecasts of CM incidence, our work can inform targeted prevention efforts, resource allocation, and early intervention strategies in endemic areas.

Despite these advancements, we acknowledge certain limitations in our study. Our focus on California, while providing a diverse range of microclimates, may limit the generalizability of our findings to other CM-endemic regions. Additionally, while our models demonstrate improved predictive capabilities, the complex interplay between environmental factors and CM incidence may involve additional variables not captured in our current dataset.

We also observed a significant gap between validation and test set performance for all models, particularly the more complex ones. This suggests potential overfitting and implies the need for further refinement of our approaches to improve generalization.

Future research directions should include:

1. Further exploration of advanced architectures such as Transformers and Mamba models, which may capture even more complex temporal relationships in the data.
2. Expanding the geographic scope to include more regions in California and Arizona, improving the models' generalizability and robustness.
3. Investigating different time sequences, particularly focusing on multi-year drought conditions, which have shown strong correlations with CM incidence in previous studies [2, 3].
4. Feature-specific analysis using LSTM variants to identify the weather variables with the most decisive impact on CM incidence, potentially informing more targeted prevention strategies.
5. Integration of additional data sources, such as soil composition, suspended dust particle levels, and human activity patterns, to create a more comprehensive predictive model [2, 3, 4].
6. Exploring the space of meteorological weather features and comparing their efficacy on different model architectures to enhance model explainability and improve understanding of environmental factors contributing to CM incidence.
7. Applying the Qualitative Associative Rule Mining (QUARM) algorithm, to uncover novel correlations in the data and better elucidate the environmental relationships to CM.
8. Implementing robust cross-validation techniques and regularization methods to address the observed overfitting and improve model generalization.

In conclusion, our research demonstrates the potential of advanced machine learning techniques, particularly the xLSTM network, in improving our approach to predicting Coccidioidomycosis outbreaks. While the results are promising, the discrepancy between validation and test set performance shows the need for further refinement. By leveraging these tools and addressing the identified challenges, we can deepen our understanding of the disease's environmental drivers and develop more effective, data-driven strategies for mitigating its impact on public health.

6 Declaration of Interests

We declare no competing interests.

7 Data Availability Statement

We are committed to transparency and reproducibility in our research. The following resources are available to the scientific community:

1. Research Code: All code used to conduct our research is publicly accessible via our GitHub repository: `https://github.com/LeifHuenderML/ValleyForecast`
2. Epidemiological Data: The epidemiological dataset used in this study is publicly available through the California Department of Public Health (CDPH) Valley Fever Dashboard: `https://www.cdph.ca.gov/Programs/CID/DCDC/Pages/ValleyFeverDashboard.aspx`.

3. Weather Data: Raw weather data was obtained from OpenWeather. While OpenWeather's licensing terms prohibit free redistribution of their data, interested researchers can purchase access through their website: `https://openweathermap.org/`
4. Derived Dataset: We created a derived dataset to train our model. This dataset incorporates significant transformations that qualify it as a Non-retrievable Value-added Service (NVAS) under OpenWeather's licensing terms. Researchers interested in accessing this derived dataset for academic purposes can contact the corresponding author at leifhuenderai@gmail.com.

8 Acknowledgements

This research would not have been possible without numerous individuals' unwavering support and guidance. I am profoundly grateful to Prof. Rhena Cooper and Kirsten Blanchette for their pivotal roles in securing funding and facilitating my participation in the esteemed INBRE program. Their support opened doors to unprecedented research opportunities. I also express my sincere appreciation to Andrea Knauff, whose meticulous proofreading and editorial suggestions substantially improved the quality and clarity of this paper. The collective contributions of these individuals have not only made this research possible but have also enriched my academic journey immeasurably. Their dedication to fostering scientific inquiry and supporting emerging researchers is truly commendable. This publication was made possible by an Institutional Development Award (IDeA) from the National Institute of General Medical Sciences of the National Institutes of Health under Grant #P20GM103408.

9 References

[1] Fisher, M. C., Koenig, G. L., White, T. J., Taylor, J. W. (n.d.). Molecular and Phenotypic Description of Coccidioides posadasii sp. nov., Previously Recognized as the Non-California Population of Coccidioides immitis (Vol. 94, Issue 1). Pappagianis, D. (n.d.). Characteristics of the Organism.

[2] Coopersmith, E. J., Bell, J. E., Benedict, K., Shriber, J., McCotter, O., Cosh, M. H. (2017). Relating Coccidioidomycosis (valley fever) incidence to soil moisture conditions. GeoHealth, 1(1), 51–63. https://doi.org/10.1002/2016GH000033

[3] Gorris, M. E., Cat, L. A., Zender, C. S., Treseder, K. K., Randerson, J. T. (2018). Coccidioidomycosis Dynamics in Relation to Climate in the Southwestern United States. GeoHealth, 2(1), 6–24. https://doi.org/10.1002/2017GH000095

[4] Tong, D. Q., Wang, J. X. L., Gill, T. E., Lei, H., Wang, B. (2017). Intensified dust storm activity and Valley fever infection in the southwestern United States. Geophysical Research Letters, 44(9), 4304–4312. https://doi.org/10.1002/2017GL073524

[5]Ampel, N. M. (n.d.). Coccidioidomycosis in Persons Infected with HIV Type 1. https://academic.oup.com/cid/article/41/8/1174/379819

[6] Weaver, E. A., Kolivras, K. N. (2018). Investigating the Relationship Between Climate and Valley Fever (Coccidioidomycosis). EcoHealth, 15(4), 840–852. https://doi.org/10.1007/s10393-018-1375-9

[7] Galgiani, J. N., Ampel, N. M., Blair, J. E., Catanzaro, A., Johnson, R. H., Stevens, D. A., Williams, P. L. (2005). Coccidioidomycosis. In Clinical Infectious Diseases (Vol. 41). https://academic.oup.com/cid/article/41/9/1217/277222

[8] Wilson, L., Ting, J., Lin, H., Shah, R., Maclean, M., Peterson, M. W., Stockamp, N., Libke, R., Brown, P. (2019). The rise of valley fever: Prevalence and cost burden of coccidioidomycosis infection in California. International Journal of Environmental Research and Public Health, 16(7). https://doi.org/10.3390/ijerph16071113

[9] Comrie, A. C. (2005). Climate factors influencing coccidioidomycosis seasonality and outbreaks. Environmental Health Perspectives, 113(6), 688–692. https://doi.org/10.1289/ehp.7786

[10] Kolivras, K. N., Comrie, A. C. (2003). Modeling valley fever (Coccidioidomycosis) incidence on the basis of climate conditions. International journal of biometeorology, 47, 87-101.

[11] McCotter, O. Z., Benedict, K., Engelthaler, D. M., Komatsu, K., Lucas, K. D., Mohle-Boetani, J. C., Oltean, H., Vugia, D., Chiller, T. M., Sondermeyer Cooksey, G. L., Nguyen, A., Roe, C. C., Wheeler, C., Sunenshine, R. (2019). Update on the Epidemiology of Coccidioidomycosis in the United States. In Medical Mycology (Vol. 57, pp. S30–S40). Oxford University Press. https://doi.org/10.1093/mmy/myy095

[12] Kingma, D. P., Lei Ba, J. (n.d.). ADAM: A METHOD FOR STOCHASTIC OPTIMIZATION.

[13] Srivastava, N., Hinton, G., Krizhevsky, A., Salakhutdinov, R. (2014). Dropout: A Simple Way to Prevent Neural Networks from Overfitting. In Journal of Machine Learning Research (Vol. 15).

[14] Wang, L., Chen, J., Marathe, M. (n.d.). DEFSI: Deep Learning Based Epidemic Forecasting with Synthetic Information. www.aaai.org

[15] Hochreiter, S., Urgen Schmidhuber, J. ¨. (n.d.). Long Short-Term Memory.

[16] Chae, S., Kwon, S., Lee, D. (2018). Predicting infectious disease using deep learning and big data. International Journal of Environmental Research and Public Health, 15(8). https://doi.org/10.3390/ijerph15081596

[17] Tamerius, J. D., Comrie, A. C. (2011). Coccidioidomycosis incidence in Arizona predicted by seasonal precipitation. PLoS ONE, 6(6). https://doi.org/10.1371/journal.pone.0021009

[18] al Sadeque, Z., Bui, F. M. (2020). A Deep Learning Approach to Predict Weather Data Using Cascaded LSTM Network. Canadian Conference on Electrical and Computer Engineering, 2020-August. https://doi.org/10.1109/CCECE47787.2020.9255716

[19] García, S., Luengo, J., Herrera, F. (n.d.). Intelligent Systems Reference Library 72 Data Preprocessing in Data Mining. http://www.springer.com/series/8578

[20] Hou, L., Zhu, J., Kwok, J. T., Gao, F., Qin, T., Liu, T.-Y. (n.d.). Normalization Helps Training of Quantized LSTM.

[21] Lopes, R. G., Yin, D., Poole, B., Gilmer, J., Cubuk, E. D. (2019). Improving Robustness Without Sacrificing Accuracy with Patch Gaussian Augmentation. http://arxiv.org/abs/1906.02611

[22] van Dyk, D. A., Meng, X.-L. (2001). The Art of Data Augmentation. In Journal of Computational and Graphical Statistics (Vol. 10, Issue 1).

[23] Caruana, R. (n.d.). An Empirical Comparison of Supervised Learning Algorithms. www.cs.cornell.edu

[24] Popescu, M.-C., Balas, V. E. (n.d.). Multilayer Perceptron and Neural Networks.

[25] Nair, V., Hinton, G. E. (n.d.). Rectified Linear Units Improve Restricted Boltzmann Machines.

[26] Hu, Y., Huber, A., Anumula, J., Liu, S.-C. (2018). Overcoming the vanishing gradient problem in plain recurrent networks. http://arxiv.org/abs/1801.06105

[27] Chai, T., Draxler, R. R. (2014). Root mean square error (RMSE) or mean absolute error (MAE)? - Arguments against avoiding RMSE in the literature. Geoscientific Model Development, 7(3), 1247–1250. https://doi.org/10.5194/gmd-7-1247-2014

[28] Lehmann, E. L., Springer, G. C. (n.d.). Theory of Point Estimation, Second Edition.

[29] Sainath, T. N., Vinyals, O., Senior, A., Sak, H. H. (n.d.). CONVOLUTIONAL, LONG SHORT-TERM MEMORY, FULLY CONNECTED DEEP NEURAL NETWORKS.

[30] Beck, M., Pöppel, K., Spanring, M., Auer, A., Prudnikova, O., Kopp, M., Klambauer, G., Brandstetter, J., Hochreiter, S. (2024). xLSTM: Extended Long Short-Term Memory. http://arxiv.org/abs/2405.04517