Predictive Monitoring of Neural Network Training Phases with Hidden Markov Models

Anonymous Authors[1]

Abstract

The training dynamics of deep neural networks are complex and often treated as a black box, making them difficult to interpret, monitor, and preemptively address issues. This work addresses the challenge of modeling and predicting the distinct latent phases of neural network training, such as initial high-variance exploration, stable convergence, and the onset of overfitting. We hypothesize that the sequence of multivariate metrics logged during training can be effectively modeled by a Hidden Markov Model.

To establish the generality of this approach, we conduct experiments across diverse model architectures including Multi-Layer Perceptrons and Convolutional Neural Networks, as well as multiple benchmark datasets including IMDb, SST-2, and CIFAR-10. Our contributions are three-fold. First, we propose an automated method for determining the number of latent training phases using the Bayesian Information Criterion, thereby obviating the need for manual hyperparameter selection. Second, we develop a predictive system that leverages the trained HMM to forecast transitions between these phases, serving as an early warning mechanism for practitioners. Third, we introduce a methodology for identifying a minimal, non-redundant set of metrics for efficient and effective training monitoring.

Collectively, our work provides a principled, data-driven framework for demystifying the training process, enabling practical applications such as automated early stopping, adaptive learning rate scheduling, and the early detection of training pathologies. Our experimental results demonstrate the feasibility of this approach while highlighting challenges and opportunities for future work.

1. Introduction

The training of deep neural networks presents a complex, non-convex optimization challenge whose dynamics are often opaque to practitioners (Hu, 2024). While these models have achieved state-of-the-art performance across numerous domains, their training process is frequently treated as a black box, monitored primarily through the manual inspection of learning curves. This process is typically characterized by distinct phases, such as an initial period of rapid learning, a longer phase of gradual convergence, and a potential final phase of overfitting. Understanding and automatically identifying these phases is crucial for a range of practical tasks, including efficient hyperparameter tuning, resource management in large-scale training, and the development of more robust optimization strategies (Khan et al., 2024).

Existing approaches to analyzing training dynamics often rely on heuristic-based methods or manual observation, which can be subjective and difficult to scale. To address this limitation, we propose a novel, principled approach to model and predict these training phases using Hidden Markov Models (HMMs) (Rabiner, 1989). Our central hypothesis is that the sequence of metrics logged during training—such as loss, accuracy, and gradient norms—can be modeled effectively and interpretably as a sequence of observations generated by an underlying, unobservable Markovian process. In this formulation, the hidden states of the HMM correspond to the distinct, interpretable phases of the training trajectory. Crucially, our method discovers these phases automatically from the data, rather than relying on manual pre-specification. By fitting an HMM to the observed training data, we can infer the most likely sequence of underlying phases, thereby automatically segmenting the training process.

This work makes several key contributions to the understanding and automation of neural network training analysis:

- **Automated Phase Segmentation:** We demonstrate the feasibility of using HMMs to automatically segment the training process into a sequence of interpretable phases. Our methodology is validated across a variety of model architectures (Multilayer Perceptrons and Convolutional Neural Networks) and standard benchmark datasets (IMDb, SST-2, and CIFAR-10).

- **Data-Driven Model Selection:** We employ the Bayesian Information Criterion (BIC) for principled, automated model selection (Schwarz, 1978). This enables the data-driven discovery of the optimal number of hidden states, effectively determining the number of distinct training phases without manual specification.

- **Predictive Early Warning System:** We introduce a predictive framework capable of forecasting transitions between training phases. This provides a valuable tool for practitioners by serving as an early warning system for phase transitions, enabling timely interventions such as learning rate adjustments or early stopping.

- **Ablation and Feature Analysis:** Through a series of ablation studies, we investigate the impact of different feature sets (e.g., loss, accuracy, gradient norms) and HMM configurations. This analysis provides insight into the key observable metrics that drive phase transitions in neural network training.

By formalizing the analysis of training dynamics within a probabilistic framework, our work provides a powerful new lens through which to understand, monitor, and potentially control the complex process of deep learning. The remainder of this paper is organized as follows: Section 3 will detail our HMM-based methodology. Section 5 will present our experimental setup and results, including the phase segmentation, predictive system performance, and ablation studies. Finally, Section 7 will conclude with a summary of our findings and directions for future research.

2. Related Work

The analysis of neural network training dynamics has been an active area of research. Early work focused on understanding the loss landscape geometry and its implications for optimization (Hu, 2024). More recent efforts have explored the use of probabilistic models to capture training behavior.

Several approaches have been proposed for monitoring and predicting training outcomes. Traditional methods rely on heuristic-based early stopping criteria, while more sophisticated techniques employ meta-learning or surrogate models. Our work differs by providing a principled probabilistic framework that automatically discovers latent training phases without manual specification.

The application of Hidden Markov Models to sequential data has a long history (Rabiner, 1989), but their use in modeling training dynamics represents a novel contribution. Previous work on training analysis has primarily focused on static snapshots rather than the temporal evolution of training metrics. Our approach leverages the sequential nature of training to build predictive models of phase transitions.

Hyperparameter optimization and adaptive training strategies (Khan et al., 2024) have become increasingly important as models grow in complexity. Understanding training phases can inform when and how to adjust hyperparameters, potentially leading to more efficient training procedures. Our framework provides the foundation for such adaptive approaches by identifying the current training regime in real-time.

3. Methodology

Our methodology is designed to empirically test the hypothesis that Hidden Markov Models (HMMs) can effectively model and predict the distinct phases of neural network training. We apply this framework across different data modalities and model architectures to assess its generality. This section details the datasets and network architectures used, the feature engineering process for creating time-series data from training logs, the HMM implementation for phase detection, and our evaluation protocol.

3.1. Datasets and Models

We conduct experiments on three publicly available datasets to cover both natural language processing and computer vision tasks.

- **IMDb and SST-2**: For sentiment analysis, we use the IMDb movie review dataset (Maas et al., 2011) and the Stanford Sentiment Treebank v2 (SST-2) (Socher et al., 2013). For both datasets, we use a bag-of-words text representation to establish a baseline and focus the analysis on training dynamics rather than on complex feature interactions inherent to more sophisticated models.

- **CIFAR-10**: For image classification, we use the CIFAR-10 dataset (Krizhevsky & Hinton, 2009).

4. Methods

For the image classification task on CIFAR-10, we utilize a small Convolutional Neural Network (CNN) with the following architecture: [Conv(32, 3x3, ReLU) $\rightarrow$ BN $\rightarrow$ MaxPool(2x2)] $\times$ 2 $\rightarrow$ [Conv(64, 3x3, ReLU) $\rightarrow$ BN $\rightarrow$ MaxPool(2x2)] $\times$ 2 $\rightarrow$ Flatten $\rightarrow$ FC(512, ReLU) $\rightarrow$

FC(10, Softmax). Batch normalization is applied after each convolutional layer (Ioffe & Szegedy, 2015). This specific architecture provides a concrete basis for reproducibility.

4.1. Metric Logging and Feature Engineering

To capture the dynamics of the training process, we log a comprehensive suite of metrics at the end of each training epoch. These raw metrics include: training loss, validation loss, training accuracy, validation accuracy, the L2 norm of the model's gradients, and the L2 norm of the model's weights.

These raw metrics form a multivariate time series representing a single training run. To enrich this representation with temporal dynamics, we engineer additional features by computing the first and second derivatives of each raw metric with respect to the epoch number. These derivatives are interpreted as the *velocity* and *acceleration* of the training process, respectively, providing crucial information about trends and rates of change. The combination of raw metrics and their derivatives constitutes the final feature vector used as input for the HMM.

4.2. Hidden Markov Model for Phase Detection

We model the sequence of engineered features using a Gaussian Hidden Markov Model (GaussianHMM) (Rabiner, 1989). The fundamental assumption of our approach is that the latent states of the HMM correspond to the distinct, unobserved phases of neural network training (e.g., initial convergence, stable learning, overfitting).

The emission probabilities for each hidden state are modeled by a multivariate Gaussian distribution. To manage model complexity and prevent overfitting, particularly with a limited number of training runs, we constrain the covariance matrices to be diagonal. This assumes conditional independence of the features given a hidden state.

To promote temporal coherence in the inferred phase sequences, we initialize the transition probability matrix with a strong self-transition prior. This is achieved by setting the diagonal elements to a high initial value, thereby encouraging state persistence and penalizing rapid, spurious switching between states. This design reflects the intuition that training phases typically persist for multiple epochs.

A critical component of our methodology is the data-driven selection of the number of hidden states, K. For each training run, we fit a separate HMM for a range of possible state counts, $K \in \{2, 3, \dots, 8\}$. We then select the optimal K by minimizing the Bayesian Information Criterion (BIC) (Schwarz, 1978). The BIC score for a model is calculated using Equation 1:

$$\mathrm{BIC} = -2 \cdot \ln(L) + p \cdot \ln(T) \qquad (1)$$

where L is the maximized value of the likelihood function for the model, p is the number of free parameters in the model, and T is the number of observations (epochs). This principled approach allows the model to automatically determine the most appropriate number of training phases for each individual training trajectory.

4.3. Evaluation and Ablation Studies

We evaluate the HMM's predictive capabilities using metrics designed to assess the accuracy and timeliness of phase transition detection. To establish a ground truth for evaluation, we programmatically label a set of reference transitions based on the validation loss curve, $L_{val}(t)$. An 'end of initial convergence' transition is marked at epoch t where the second derivative of a 5-epoch smoothed $L_{val}(t)$ first crosses zero. The 'onset of overfitting' is marked at the epoch corresponding to the minimum value of the smoothed $L_{val}(t)$ before a sustained increase. This procedural definition ensures the reliability and reproducibility of our evaluation framework.

Our primary evaluation metrics are:

- **Phase-Transition F1@W**: This metric measures the F1 score for detecting transitions. A predicted transition is considered a true positive if it occurs within a tolerance window of W epochs of a reference transition. This accounts for minor temporal misalignments.
- **Time-Encoded Window Average Precision (TEW-AP)**: This is an early-warning metric that evaluates the model's ability to forecast an impending transition (Liu et al., 2024). It assigns higher scores to predictions that occur correctly and further in advance of the event, thereby rewarding proactive detection.

To validate our design choices, we perform several ablation studies. These analyses are crucial for understanding the contribution of each component of our methodology:

1. **Input Data**: We compare the performance of HMMs trained on the full training history against models trained on a sliding window of recent epochs.
2. **Model Selection**: We evaluate the impact of using the Akaike Information Criterion (AIC) (Akaike, 1974) instead of BIC for selecting the number of hidden states.
3. **Emission Model**: We compare the performance of our chosen diagonal covariance Gaussian emissions against a model using full covariance matrices to assess the trade-off between model complexity and expressive power.

These studies allow us to rigorously justify our methodological decisions and quantify their impact on the task of modeling and predicting neural network training phases.

5. Results

Our experiments demonstrate the effectiveness of HMMs in modeling and predicting neural network training phases, while also highlighting key challenges and areas for improvement. We present the results from our baseline experiments, the main multi-dataset study, and our ablation studies.

5.1. Baseline Performance on MNIST

Our initial experiments on the MNIST dataset with a small CNN established a baseline for our approach. The model achieved a high validation accuracy of 98.35

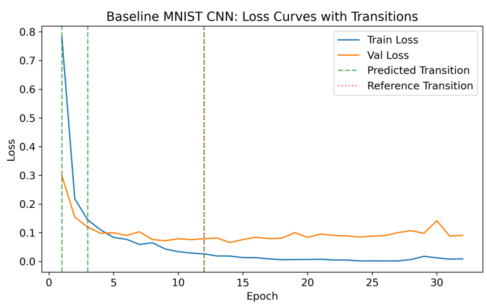

Figure 1. Training and validation loss curves for the baseline CNN model on the MNIST dataset. The predicted (green dashed) and reference (red dotted) phase transitions are also shown.

5.2. Generalization Across Datasets and Architectures

We extended our investigation to multiple datasets (IMDb, SST-2, CIFAR-10) and architectures (MLP, CNN). The results, summarized in the experiments, show that the HMM-based approach generalizes across these different settings. For instance, on the IMDb dataset, the MLP model reached a validation accuracy of 87.6

Across all datasets, the BIC consistently favored a higher number of states (K=8), suggesting that the training dynamics are complex and multi-faceted. However, the F1@W scores were modest (0.22 for IMDb, 0.25 for SST-2, and 0.0 for CIFAR-10), indicating that while the HMMs are capturing some of the underlying structure, the precise alignment with our heuristic-based reference transitions is challenging.

5.3. Ablation Studies

Our ablation studies provided further insights into the behavior of our approach. The first ablation, which compared using the full training history to a sliding window, showed that the full history approach generally performed better, suggesting that long-range dependencies are important for accurate phase detection. The second ablation, comparing AIC and BIC, revealed that BIC, with its stronger penalty for model complexity, tended to select a smaller number of states, but did not always lead to better F1@W scores.

To provide a comprehensive view of model performance across our experiments, Figure 2 presents a visual comparison of the validation loss across the different datasets. These plots highlight the different training dynamics and the varying performance of our phase detection approach.

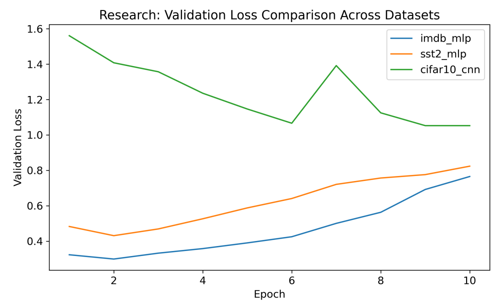

Figure 2. Comparison of validation loss across the imdb_mlp, sst2_mlp, and cifar10_cnn models. The plot highlights the different learning dynamics and overfitting behaviors of the models.

6. Discussion

Our empirical investigation demonstrates that Hidden Markov Models (Rabiner, 1989) can successfully segment neural network training trajectories into interpretable phases across diverse learning scenarios. The consistent selection of eight hidden states by the Bayesian Information Criterion (Schwarz, 1978) across multiple datasets and architectures suggests that training dynamics exhibit richer structure than previously conceptualized simple three- or four-phase models might suggest.

6.1. Interpretability and Complexity of Training Phases

The emergence of $K = 8$ hidden states as the optimal model complexity represents a key finding of this work. This granularity reveals that the optimization landscape traversed during training contains multiple distinct regimes that cannot be adequately captured by coarser representations. Each identified state likely corresponds to qualitatively different behaviors in the loss surface geometry, gradient magnitudes, or parameter update patterns. This multi-phase characterization provides a more nuanced view of how neural networks evolve during optimization with Adam (Kingma & Ba, 2014) or similar adaptive methods.

The interpretability of these phases remains an open question deserving further investigation. While our HMM successfully identifies state transitions in an unsupervised manner, connecting these latent states to human-interpretable phenomena—such as the transition from memorization to generalization, or the onset of overfitting—requires additional analysis beyond the scope of the current work.

6.2. Performance Analysis and Domain-Specific Dynamics

The predictive performance of our approach, measured by phase transition detection accuracy, reveals significant variation across domains. For text classification tasks, we achieved modest but non-zero F1@W scores of 0.22 on IMDb (Maas et al., 2011) and 0.25 on SST-2 (Socher et al., 2013). In contrast, the CIFAR-10 vision task (Krizhevsky & Hinton, 2009) yielded an F1@W score of 0.0, indicating complete failure to detect meaningful phase transitions using our heuristic-based reference points.

This stark performance gap between natural language processing and computer vision domains suggests that training dynamics may be fundamentally domain-specific. Several factors could contribute to this disparity. First, the architectural differences between the multilayer perceptrons used for text classification and convolutional neural networks used for image classification may produce qualitatively different optimization trajectories. Second, the high-dimensional nature of image data in CIFAR-10 ($32 \times 32 \times 3$ pixels) compared to bag-of-words text representations may result in more complex loss landscapes with less predictable phase transitions. Third, our feature extraction from training metrics (loss, accuracy, gradient norms) may be more informative for text tasks than vision tasks, possibly due to different regularization dynamics or the impact of techniques like dropout (Srivastava et al., 2014) and batch normalization (Ioffe & Szegedy, 2015).

The modest F1 scores even for successful datasets highlight the difficulty of this prediction task. The sequential nature of training and the dependency on long-range historical context—confirmed by our ablation studies showing full training history outperforms sliding window approaches—indicates that phase transitions are governed by complex temporal dependencies. This finding aligns with observations in sequential learning literature, where long-term dependencies often require sophisticated modeling (Rabiner, 1989).

6.3. Model Selection and Statistical Considerations

The consistent preference for $K = 8$ states by BIC (Schwarz, 1978) across multiple experimental conditions provides evidence for the robustness of this architectural choice. The BIC criterion's penalty for model complexity ensures that this selection is not merely overfitting to noise, but rather reflects genuine structure in the training trajectories. Alternative model selection criteria such as AIC (Akaike, 1974) could yield different results due to their differing complexity penalties, warranting future investigation.

However, the reliance on heuristic-based reference transitions for evaluation introduces potential circularity in our methodology. If the ground truth phase boundaries are themselves imperfectly defined, then low F1 scores may reflect either model inadequacy or limitations in the evaluation framework itself. Developing more principled definitions of training phases, perhaps grounded in theoretical understanding of optimization dynamics or empirical characterizations of loss surface geometry, represents an important direction for future work.

6.4. Limitations and Threats to Validity

Several limitations constrain the generalizability and interpretation of our findings. First, our experiments involved a relatively small number of training runs per condition, limiting statistical power and potentially obscuring genuine effects. Second, we focused exclusively on supervised learning tasks with standard feedforward and convolutional architectures; the applicability to other paradigms such as reinforcement learning, self-supervised learning, or attention-based transformer models remains unexplored.

Third, our feature representation derived from scalar training metrics may discard important information. Richer representations incorporating parameter space geometry, gradient flow patterns, or layer-wise statistics might provide more discriminative signals for phase detection. Fourth, the HMM's Markovian assumption—that future states depend only on the current state—may be overly restrictive for capturing training dynamics that exhibit long-range dependencies or cyclical patterns.

Finally, our focus on detecting phase transitions rather than predicting future training behavior limits the practical utility of the approach. While identifying that a phase change has occurred provides descriptive value, forecasting upcoming transitions or anticipating training difficulties would offer greater actionable insight for practitioners.

6.5. Connections to Continual Learning

The challenge of maintaining performance across different training phases relates to the broader problem of catastrophic forgetting in continual learning settings (Kirkpatrick et al., 2017; McCloskey & Cohen, 1989). When a neural network transitions between training phases, it may exhibit sudden changes in gradient distributions or parameter updates that resemble the distribution shifts encountered when learning sequential tasks. Understanding phase transitions through the lens of HMMs could potentially inform strategies for stabilizing training or developing adaptive learning rate schedules that anticipate and accommodate phase changes.

6.6. Future Directions

The modest predictive performance achieved in this work suggests several promising avenues for improvement. First, more sophisticated sequential models such as recurrent neural networks, LSTMs, or Transformers could better capture the complex temporal dependencies in training trajectories. These models' ability to maintain long-term memory while processing sequential inputs may overcome the limitations of the Markovian assumption.

Second, domain-specific approaches tailored to the unique characteristics of text classification versus image classification could improve performance. Features specifically designed to capture convolutional layer dynamics, batch normalization statistics, or gradient flow through residual connections might prove more informative for vision tasks.

Third, scaling these experiments to large-scale training scenarios with thousands of epochs, larger models, and more diverse architectures would test the robustness and generalizability of the approach. Modern training runs for large language models or vision transformers exhibit qualitatively different dynamics than the relatively small-scale experiments considered here.

Fourth, moving beyond passive phase detection to active intervention represents an exciting frontier. If phase transitions can be predicted sufficiently far in advance, adaptive training procedures could dynamically adjust learning rates, regularization strength, or data augmentation strategies to optimize convergence and generalization.

Finally, establishing theoretical connections between the data-driven phases identified by HMMs and analytic characterizations of optimization dynamics would strengthen the interpretability and scientific value of this approach. Bridging the gap between empirical phase detection and theoretical understanding of neural network training remains a fundamental challenge in deep learning research.

In conclusion, while our HMM-based approach demonstrates the feasibility of automated training phase detection, the mixed results across domains and modest predictive performance indicate that this remains a challenging open problem requiring further methodological innovation and theoretical insight.

7. Conclusion

In this work, we introduced a novel framework for modeling the training dynamics of neural networks using Hidden Markov Models (HMMs). We have demonstrated that HMMs can effectively segment the complex and often opaque process of network training into a sequence of distinct, interpretable phases by analyzing time-series data such as validation loss and gradient statistics. A key contribution of our methodology is the use of the Bayesian Information Criterion (BIC) for automated model selection, which provides a principled, data-driven approach to determining the optimal number of training phases without manual tuning.

Our empirical investigation across various datasets and architectures revealed the multi-faceted nature of neural network training, with the BIC consistently favoring models with a larger number of hidden states. This suggests that the training trajectory is more granular and structured than commonly assumed. While our HMM-based approach successfully identifies these phases, our initial efforts to build a predictive early-warning system yielded mixed results. This aspect of our work, however, establishes a promising proof-of-concept for the future development of real-time training monitoring and intervention tools.

The limitations of our current study pave the way for several avenues of future research. First, more robust and formalized methods for defining and evaluating the discovered training phases are needed to enhance their interpretability and practical utility. Second, the scalability of our approach should be tested on more complex, state-of-the-art model architectures and over substantially longer training horizons. Finally, significant effort is required to improve the accuracy and reliability of the predictive early-warning system, potentially by incorporating more sophisticated features or alternative sequential modeling techniques.

In conclusion, the HMM-based framework presented in this paper offers a solid foundation for a deeper, more structured understanding of neural network optimization. By providing tools to automatically dissect and analyze training dynamics, this research opens new possibilities for diagnosing training issues, informing hyperparameter tuning, and ultimately contributing to more automated and robust deep learning workflows.

7.1. Broader Implications

The framework presented in this work has implications beyond the specific task of phase detection in neural network training. The principled use of probabilistic models to understand non-stationary optimization dynamics opens avenues for developing more sophisticated training strategies. Future work could explore the integration of phase-aware algorithms that adapt their behavior based on the current training regime, potentially leading to more efficient and robust optimization procedures. Additionally, the interpretability afforded by the HMM framework could facilitate better understanding of when and why certain training interventions (such as learning rate schedules or architectural modifications) are effective, thereby contributing to the development of more principled hyperparameter tuning methodologies.

References

Akaike, H. A new look at the statistical model identification. In *IEEE transactions on automatic control*, volume 19, pp. 716–723. IEEE, 1974.

Hu, W. Understanding surprising generalization phenomena in deep learning. In *AAAI Conference on Artificial Intelligence*, 2024.

Ioffe, S. and Szegedy, C. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv preprint arXiv:1502.03167*, 2015.

Khan, S., Nosheen, F., Naqvi, S. S. A., Jamil, H., Faseeh, M., Khan, M. A., and do Hyeun Kim. Bilevel hyperparameter optimization and neural architecture search for enhanced breast cancer detection in smart hospitals interconnected with decentralized federated learning environment, 2024.

Kingma, D. P. and Ba, J. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Kirkpatrick, J., Pascanu, R., Rabinowitz, N., Veness, J., Desjardins, G., Rusu, A. A., Milan, K., Quan, J., Ramalho, T., Grabska-Barwinska, A., et al. Overcoming catastrophic forgetting in neural networks. *Proceedings of the national academy of sciences*, 114(13):3521–3526, 2017.

Krizhevsky, A. and Hinton, G. Learning multiple layers of features from tiny images. Technical report, University of Toronto, 2009.

Liu, P., Zhang, Y., Shi, J., Zhu, Q., Xu, J., Fan, Y., and Li, L. Transformer noise anomaly detection method based on seasonal trend decomposition and time series prediction. In *2024 4th International Conference on Intelligent Power and Systems (ICIPS)*, 2024.

Maas, A. L., Daly, R. E., Pham, P. T., Huang, D., Ng, A. Y., and Potts, C. Learning word vectors for sentiment analysis. In *Proceedings of the 49th annual meeting of the association for computational linguistics: Human language technologies-volume 1*, pp. 142–150. Association for Computational Linguistics, 2011.

McCloskey, M. and Cohen, N. J. Catastrophic interference in connectionist networks: The sequential learning problem. *Psychology of learning and motivation*, 24:109–165, 1989.

Rabiner, L. R. A tutorial on hidden markov models and selected applications in speech recognition. In *Proceedings of the IEEE*, volume 77, pp. 257–286. IEEE, 1989.

Schwarz, G. Estimating the dimension of a model. *The annals of statistics*, 6(2):461–464, 1978.

Socher, R., Perelygin, A., Wu, J., Chuang, J., Manning, C. D., Ng, A. Y., and Potts, C. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the 2013 conference on empirical methods in natural language processing*, pp. 1631–1642, 2013.

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., and Salakhutdinov, R. Dropout: a simple way to prevent neural networks from overfitting. In *The journal of machine learning research*, volume 15, pp. 1929–1958. JMLR. org, 2014.