STATISTICS, OPTIMIZATION AND INFORMATION COMPUTING
Stat., Optim. Inf. Comput., Vol. **13**, June 2025, pp 2601–2615.
Published online in International Academic Press (www.IAPress.org)

Forecasting Scientific Impact: A Model for Predicting Citation Counts

Bao T. Nguyen*, Thinh T. Nguyen

Institute of Intelligent and Interactive Technology, College of Technology & Design,
University of Economics Ho Chi Minh City, Vietnam

Abstract Forecasting the citation counts of scientific papers is a challenging task, particularly when utilizing textual data such as author names, paper titles, abstracts, and affiliations. This task diverges from conventional regression problems involving numerical or categorical inputs, as it demands the processing of complex, high-dimensional text features. Traditional regression techniques, including Linear Regression, Polynomial Regression, and Decision Tree Regression, often fail to encapsulate the semantic intricacies of textual data and are susceptible to overfitting due to the expansive feature space. In the context of Vietnam, where research output is rapidly growing yet underexplored in predictive modeling, these limitations are especially pronounced. To tackle these issues, we leverage advanced Natural Language Processing (NLP) techniques, employing Long Short-Term Memory (LSTM) and Gated Recurrent Unit (GRU) networks. These deep learning models are adept at handling sequential data, capturing long-range dependencies, and preserving contextual nuances, rendering them well-suited for text-based citation prediction. We conducted experiments using a dataset of academic papers authored by Vietnamese researchers across diverse disciplines, sourced from publications featuring Vietnamese author contributions. The dataset includes features such as author names, titles, abstracts, and affiliations, reflecting the unique characteristics of Vietnam's research landscape. We compared the performance of LSTM and GRU models against traditional machine learning approaches, evaluating prediction accuracy with metrics like Mean Absolute Error (MAE) and Root Mean Squared Error (RMSE). The results reveal that LSTM and GRU models substantially outperform their traditional counterparts. The LSTM model achieved an RMSE of 8.54 and an MAE of 8.1, while the GRU model yielded an RMSE of 8.32 and an MAE of 7.83, demonstrating robust predictive capabilities. In contrast, traditional models such as Decision Tree Regression and Linear Regression exhibited higher error rates, with RMSEs exceeding 12.0. These findings underscore the efficacy of deep learning in forecasting citation counts from textual data, particularly for Vietnamese research outputs, and highlight the potential of LSTM and GRU models to uncover intricate patterns driving scientific impact in emerging research ecosystems.

Keywords Citation Counts, LSTM, GRU, Deep Learning, Machine Learning

DOI: 10.19139/soic-2310-5070-2524

1. Introduction

Citations are widely regarded as a fundamental metric for evaluating the impact and relevance of scientific research, serving as a proxy for a paper's influence within its field [1]. The ability to forecast citation counts offers significant value to diverse stakeholders, including researchers, funding agencies, and academic institutions. Such predictions provide insights into a paper's potential reach and can inform critical decision-making processes, such as grant allocations, faculty hiring, tenure evaluations, and strategies to enhance research visibility and collaboration [2]. However, accurately predicting citation counts remains a formidable challenge due to the multifaceted nature of academic publishing, the dynamic evolution of citation patterns, and the complexity of text-based data [3] [4].

*Correspondence to: Bao T. Nguyen (Email: baont@ueh.edu.vn).
Institute of Intelligent and Interactive Technology, College of Technology & Design,
University of Economics Ho Chi Minh City, Vietnam

ISSN 2310-5070 (online) ISSN 2311-004X (print)

Traditional citation prediction models have predominantly utilized structured data, such as the number of authors, journal impact factors, and historical citation trends [5]. These approaches, often grounded in regression techniques like Linear Regression, Polynomial Regression, or Decision Trees, have achieved moderate success when applied to numerical metadata. However, their performance diminishes when tasked with processing high-dimensional, text-rich features such as paper titles, abstracts, and affiliations [6]. These models struggle to capture the latent semantic relationships within textual data and are prone to overfitting, limiting their generalizability across diverse datasets [7].

Recent advancements in Natural Language Processing (NLP) and deep learning [8] have introduced promising solutions to these limitations. Techniques such as word embeddings [9], Recurrent Neural Networks (RNNs), and their advanced variants—Long Short-Term Memory (LSTM) units [4] and Gated Recurrent Units (GRUs) [10]—excel at modeling sequential data and preserving contextual information over long text sequences [**?**] These methods have proven effective in various NLP tasks, including text classification, machine translation, and sentiment analysis, owing to their capacity to learn complex temporal patterns [11]. For citation prediction, where the semantic content of a paper's title or abstract may signal its future impact, LSTM and GRU models offer a robust framework to bridge textual features with citation outcomes [12].

Several studies have demonstrated the potential of deep learning in this domain. For instance, Cohan et al. [13] employed a hierarchical attention network to leverage the full text of research papers, achieving improved citation prediction accuracy. Similarly, Dong et al. [12] utilized deep recurrent models to forecast citation counts based on metadata and abstracts, highlighting the efficacy of sequential modeling. These findings underscore the advantage of incorporating contextual and structural information from text, a capability that traditional machine learning approaches often lack [14].

In this study, we propose a novel model to predict the future citation counts of scientific papers by harnessing text-based features, including author names, paper titles, abstracts, and affiliations. We investigate the effectiveness of LSTM and GRU networks in capturing the intricate relationships between textual data and citation trends. Our hypothesis posits that these deep learning models, by encoding semantic and contextual nuances, will outperform conventional regression-based methods. To test this, we compare their performance against traditional techniques using a comprehensive dataset spanning multiple disciplines.

The remainder of this paper is structured as follows. Section 2 reviews prior work on citation prediction and text-based regression models. Section 3 outlines our methodology, including data collection, preprocessing, and model architecture. Section 4 presents the experimental results, and interprets their implications. Finally, Section 5 summarizes key findings and suggests avenues for future research.

2. Related works

Citation prediction has emerged as a pivotal research area in scientometrics, driven by the need to quantify and forecast the impact of scientific publications. Early approaches predominantly relied on structured features, such as author attributes, publication venues, and historical citation counts. For example, Bornmann and Daniel [1] investigated the role of author productivity, collaboration networks, and publication history in determining citation counts, demonstrating their strong predictive power. Similarly, Tahamtan et al. [15] conducted a systematic review of citation drivers, identifying high-impact journals, international collaborations, and keyword selection as significant factors influencing citation rates. Additional studies, such as those by Wang et al. [16], further underscored the importance of venue prestige and author reputation in early citation prediction models.

Traditional statistical methods, including Linear Regression, Polynomial Regression, and Decision Tree Regression, have been widely adopted to model citation counts based on these structured features. Yan and Ding [17] employed Decision Tree Regression with bibliometric indicators, revealing that integrating multiple features enhances prediction accuracy. Nevertheless, these methods struggle with text-based features, such as titles and abstracts, due to their limited capacity to capture semantic nuances and contextual relationships. Overfitting is a frequent issue when handling high-dimensional textual data, as noted by Li et al. [18], constraining the effectiveness of traditional regression techniques in forecasting long-term citation trends.

2.1. NLP-Based Citation Prediction

With the growing availability of textual data from scientific publications, Natural Language Processing (NLP) techniques have gained prominence in citation prediction. NLP enables models to extract semantic insights from unstructured text, such as abstracts and titles, surpassing the capabilities of structured bibliometric indicators alone. Kanakia et al. [19] pioneered this shift by incorporating text features from abstracts into traditional machine learning algorithms, achieving modest performance gains. However, their approach relied heavily on manual feature engineering and failed to fully leverage the sequential and contextual properties of text.

Recent advances in deep learning have revolutionized NLP-based citation prediction by capturing complex, non-linear relationships within textual data. Word embeddings, such as Word2Vec [9] and GloVe [20], transform words into dense vector representations, facilitating the detection of semantic similarities. Building on these embeddings, Dong et al. [12] combined recurrent neural networks (RNNs) with text from abstracts and author affiliations, significantly outperforming traditional regression models. Similarly, Xu et al. [21] explored transformer-based architectures to model textual content, highlighting their potential to improve citation forecasting by capturing global dependencies in scientific texts.

2.2. LSTM and GRU Models in Citation Prediction

Among deep learning architectures, Long Short-Term Memory (LSTM) and Gated Recurrent Unit (GRU) networks have excelled in text-based regression tasks [22], including citation prediction. Designed to model long-range dependencies in sequential data, these architectures are well-suited for understanding context in scientific texts [4, 10]. For instance, LSTM and GRU models can retain critical information from earlier parts of an abstract—such as key topics or innovative terms—that may influence future citations. Cohan et al. [13] utilized a hierarchical attention network (HAN) to predict citation intent and counts by analyzing full-text articles, showing that attention mechanisms enhance context-aware predictions. Likewise, Dong et al. [12] employed GRU-based models to capture temporal citation dynamics, improving long-term trend forecasts.

Despite these advancements, deep learning models often demand substantial computational resources and large datasets to mitigate overfitting, as emphasized by Zhang et al. [22, 23] [24]. Moreover, integrating structured and unstructured data remains an open question. Hybrid approaches, such as that proposed by Chen et al. [25], combine biometric features (e.g., journal impact factor) with text embeddings via multi-layered neural networks, achieving state-of-the-art performance. Similarly, Abrishami and Aliakbary [26] introduced a hybrid framework blending network-based metrics with textual features, further advancing prediction accuracy.

In summary, while traditional models based on structured data fall short in capturing the multifaceted nature of scientific impact, NLP and deep learning techniques—particularly LSTM and GRU models—offer substantial improvements by harnessing textual semantics. Nonetheless, issues like dataset size, computational cost, and optimal feature integration remain unresolved. This study extends prior efforts by investigating LSTM and GRU networks for citation prediction, positing that their ability to model contextual text features can outperform conventional regression methods.

3. Approach

This section outlines the process of building a citation prediction model using text features. The raw textual data is initially preprocessed using Natural Language Processing (NLP) techniques. We then predict citation counts with Long Short-Term Memory (LSTM) networks and compare the results by experimenting with Gated Recurrent Unit (GRU) networks. In addition, traditional machine learning methods such as Polynomial Regression, Boosted Decision Trees, and Random Forests are incorporated to enhance performance and offer a broader basis for comparison.

3.1. Text Preprocessing with NLP Techniques

The initial stage of our approach involves preparing raw textual data—comprising paper titles, abstracts, and author information—for analysis by transforming it into a structured, numerical format suitable for machine learning models. This preprocessing is critical to enhance data quality and ensure compatibility with LSTM and GRU architectures [27]. We employ the following Natural Language Processing (NLP) techniques to achieve this:

1. **Stop Word Removal:** Common words with minimal predictive value, such as "and," "the," and "is," are filtered out. These stop words, as noted by Manning et al. [28], contribute little to semantic understanding in text analysis, reducing noise and dimensionality in the dataset.
2. **Number Removal:** While our initial approach excluded all numerical values—including publication years and page numbers—from the text to reduce irrelevant variance [29], we revised the preprocessing step to retain contextually meaningful numbers such as publication years and numeric components in model names (e.g., "BERT-2020"), as these may carry predictive signals for citation impact.
3. **Case Normalization:** All text is converted to lowercase to eliminate case-sensitive discrepancies (e.g., "Algorithm" vs. "algorithm"). This step ensures uniformity across the corpus, preventing the model from treating identical words as distinct entities [27].
4. **Special Character Removal:** Punctuation marks and special characters (e.g., $, !, @, %, ^, &, *, (,)) are stripped from the text. These symbols, while useful in syntactic parsing, lack semantic significance for citation prediction and may introduce noise [28].
5. **Lemmatization:** Words are reduced to their base or dictionary forms through lemmatization (e.g., "running" to "run"). This process, as described by Bird et al. [29], preserves meaning while reducing vocabulary size, enhancing model efficiency and generalization.
6. **Tokenization:** The cleaned text is segmented into individual tokens (words or subword units) via tokenization. This step transforms sentences into discrete units, enabling subsequent numerical encoding and analysis by machine learning models [27].
7. **One-Hot Encoding (Categorical Features):** Categorical variables, such as author affiliations or publication venues, are converted into binary vectors using one-hot encoding. This technique, widely adopted in NLP pipelines [11], allows categorical data to be seamlessly integrated into numerical models. We fine-tuned our Word2Vec embeddings on a domain-specific corpus consisting of $50,000$ scientific abstracts to better capture the semantics of academic language.
8. **Text-to-Numerical Conversion:** Tokenized words are mapped to dense numerical representations using word embeddings. We leverage pretrained embeddings like Word2Vec [9] or GloVe [20], which capture semantic relationships, though custom embeddings tailored to the citation dataset could also be trained for domain-specific precision [30].
9. **Sequence Padding:** To accommodate variable-length text inputs, sequences are padded or truncated to a fixed length. Shorter sequences are appended with zeros, while longer ones are cropped, ensuring a uniform input shape for LSTM and GRU processing [11]. This step is essential for batch training and model stability.

This meticulous preprocessing pipeline produces a clean, consistent, and numerically tractable dataset, optimized for input into LSTM and GRU models. By aligning the text data with the requirements of deep learning frameworks, we enhance the models' ability to extract meaningful patterns for citation prediction [30].

3.2. Citation Prediction Using LSTM

Following preprocessing, the cleaned and tokenized text data—encompassing titles, abstracts, and associated metadata—is input into a Long Short-Term Memory (LSTM) network to forecast future citation counts of scientific papers. Introduced by Hochreiter and Schmidhuber [4], LSTMs are a specialized form of recurrent neural networks (RNNs) engineered to model long-term dependencies in sequential data. Unlike traditional RNNs, which are hindered by the vanishing gradient problem, LSTMs employ a sophisticated gating mechanism to retain and propagate information across extended sequences [31].

LSTM Structure: [31, 32] Each LSTM unit comprises three key gates that regulate information flow:

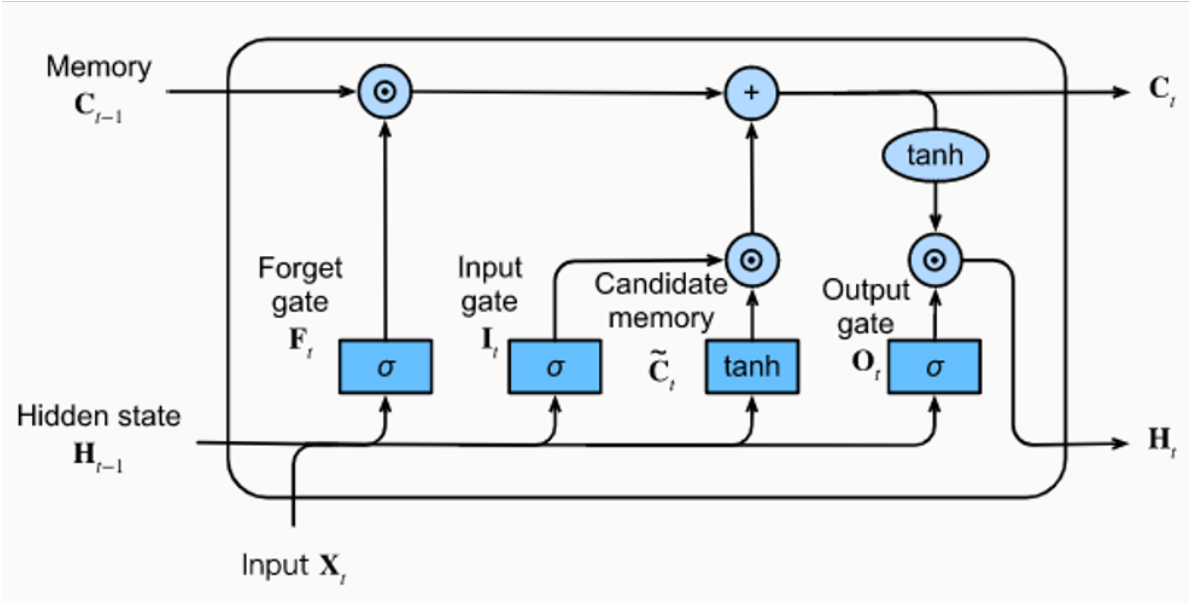

Figure 1. The LSTM (Long Short-Term Memory) model featuring three gates: Forget Gate, Input Gate, and Output Gate, as proposed by Hochreiter and Schmidhuber [4].

1. **Forget Gate:** This gate assesses the relevance of prior time step information, deciding what to discard or retain based on the current input and hidden state.
2. **Input Gate:** This gate determines which new information from the input should be stored in the cell state, updating the memory with pertinent features.
3. **Output Gate:** This gate filters the cell state to generate the output for the current time step, balancing memory and immediate context.

The cell state acts as a persistent memory conduit, modulated by these gates to preserve critical information across the sequence. This architecture enables LSTMs to effectively capture long-range dependencies, a capability extensively validated in sequence modeling tasks [11].

LSTMs offer distinct advantages for text-based prediction. Their ability to retain information over long periods makes them adept at processing sequential data like scientific abstracts, where early thematic elements may influence later citation outcomes [33]. By mitigating the vanishing gradient issue, LSTMs ensure stable learning over extended sequences, outperforming standard RNNs in tasks requiring deep memory retention [4]. However, their computational complexity—stemming from multiple gates and parameters—renders them resource-intensive and slower to train compared to simpler architectures like GRUs [34]. This trade-off between performance and efficiency is a key consideration in their deployment. We select LSTMs for citation prediction due to their proficiency in modeling the sequential and thematic structure of textual data, such as the progression of ideas within a research abstract. Citation patterns often hinge on a paper's conceptual coherence and foundational keywords, which LSTMs can adeptly capture across the entire sequence [12]. Their capacity to preserve early contextual cues—e.g., seminal phrases or topics introduced at the outset—aligns with the hypothesis that such elements shape a paper's long-term citation trajectory [13]. While computationally demanding, LSTMs provide a robust framework for uncovering the intricate relationships driving scientific impact in text-heavy datasets.

3.3. Citation Prediction Using GRU

To assess the comparative efficacy of Long Short-Term Memory (LSTM) networks, we implement a Gated Recurrent Unit (GRU) model, a streamlined variant of LSTMs introduced by Cho et al. [10]. GRUs are designed

to retain similar predictive power to LSTMs while minimizing computational overhead, making them an attractive alternative for sequential data tasks such as citation prediction [34].

GRU Structure: The GRU architecture relies on two primary gates, which govern the flow and retention of information through the network:

1. **Reset Gate:** This gate determines the extent to which prior information is discarded, allowing the model to focus on relevant features from recent inputs [10].
2. **Update Gate:** This gate regulates the integration of new input data into the current memory state, balancing past and present information to optimize predictions [35].

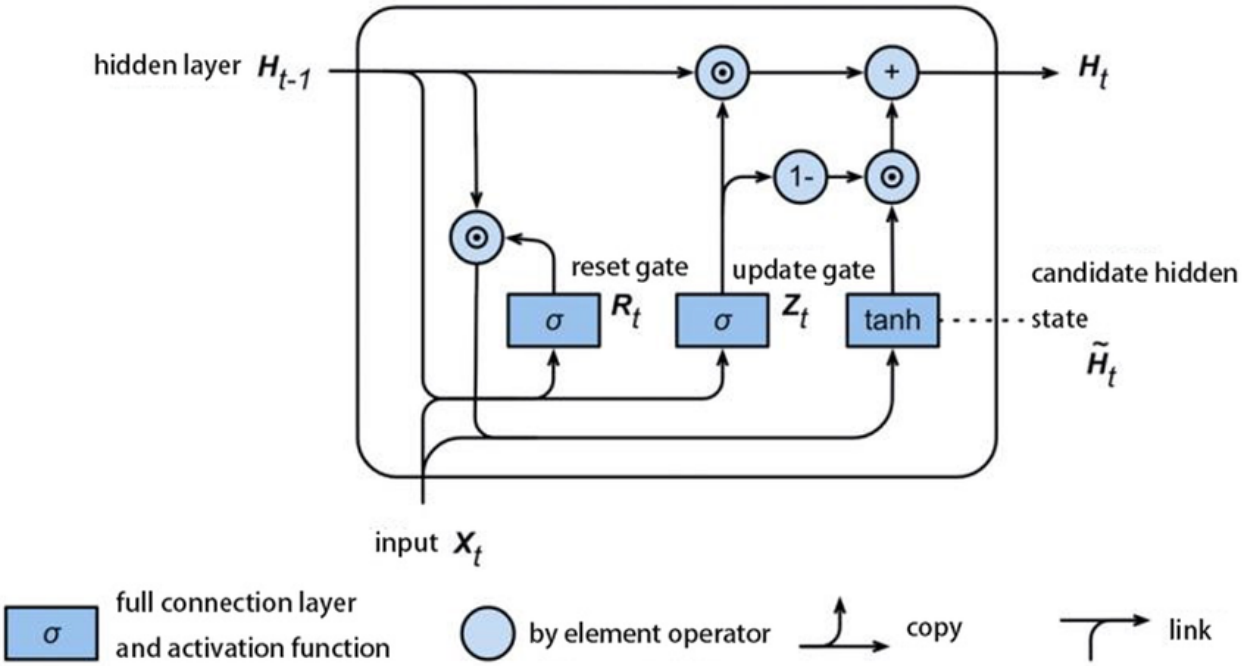

Figure 2. The GRU (Gated Recurrent Unit) model featuring two gates: Reset Gate and Update Gate, adapted from Cho et al. [10].

Unlike LSTMs, which maintain separate cell and hidden states, GRUs merge these into a single hidden state, simplifying the architecture and accelerating training times [34]. This design enables GRUs to efficiently propagate information from earlier time steps to the output, a feature particularly advantageous for tasks requiring rapid updates from sequential inputs, such as text-based citation analysis [32]. Studies by Chung et al. [34] demonstrate that GRUs achieve performance comparable to LSTMs with reduced computational complexity, owing to fewer parameters and gates.

GRUs offer several benefits over LSTMs. Their simpler structure—lacking a distinct cell state—results in faster training and lower resource demands, making them well-suited for resource-constrained environments [11]. Moreover, GRUs often exhibit superior generalization on smaller datasets due to their reduced risk of overfitting [35]. However, this simplicity comes at a cost: GRUs may struggle to capture long-term dependencies as effectively as LSTMs, a potential limitation for tasks requiring extensive memory retention over prolonged sequences [32].

We adopt GRUs as an efficient alternative to LSTMs for citation prediction, leveraging their ability to process textual sequences (e.g., abstracts, titles) swiftly while preserving critical dependencies. In scientometric applications, where datasets can be voluminous, GRUs provide a practical trade-off between computational efficiency and predictive accuracy [12]. Their streamlined architecture reduces training time, an essential

consideration for large-scale citation datasets, without significantly compromising performance compared to more complex models.

By implementing both LSTM and GRU networks, we seek to evaluate their effectiveness in forecasting citation counts, elucidating the trade-offs between accuracy and computational efficiency. These models are inherently suited to text-based prediction tasks, yet their architectural differences render them optimal for distinct scenarios—GRUs for speed and efficiency on moderately sized datasets, and LSTMs for deeper memory retention on complex, long-sequence data [36]. This dual approach enables a comprehensive analysis of their practical utility in citation forecasting.

3.4. Citation Prediction Using Traditional Machine Learning Techniques

To ensure a robust and equitable comparison with advanced methods, we investigate citation prediction using a suite of traditional machine learning algorithms. These include Linear Regression, Decision Tree Regression, and Neural Network Regression, each offering distinct strengths in modeling citation counts. Linear Regression provides a straightforward approach by assuming a linear relationship between input features and the target variable [37]. Decision Tree Regression captures non-linear patterns through hierarchical feature splitting [38], while Neural Network Regression leverages multi-layered architectures to model complex dependencies [39]. By evaluating these techniques, we aim to benchmark their performance against state-of-the-art deep learning models, providing insights into their applicability for citation forecasting.

Linear Regression: We employ Linear Regression to model the relationship between citation counts and a variety of features, including paper characteristics (e.g., author count, publication year) and bibliometric metadata (e.g., journal impact factor). This method assumes a linear dependency between the predictors and citation counts, offering a computationally efficient and interpretable baseline for prediction tasks [17]. Despite its simplicity, Linear Regression has been widely used in scientometric studies due to its ability to reveal key drivers of scientific impact [1].

Decision Tree Regression: Next, we apply Decision Tree Regression, which constructs a predictive model by recursively partitioning the feature space into subsets based on feature thresholds. This tree-based structure excels at capturing non-linear relationships and feature interactions, making it suitable for heterogeneous datasets common in citation analysis [38]. Previous work by Yan and Ding [17] demonstrated its efficacy in predicting citation counts using bibliometric indicators, highlighting its adaptability to diverse input features.

Neural Network Regression: Finally, we implement Neural Network Regression to address the limitations of linear models in capturing intricate, non-linear patterns. This approach utilizes a feedforward neural network with multiple hidden layers, where interconnected nodes learn to represent complex relationships between input features (e.g., text embeddings, citation history) and citation counts [39]. Goodfellow et al. [11] note that such architectures are particularly effective for high-dimensional data, offering a bridge between traditional machine learning and deep learning paradigms. We configure the network with varying layer depths and activation functions to optimize performance for citation prediction.

4. Experiments & Results

In this section, we present the experimental setup and evaluation of three distinct approaches for forecasting scientific impact, specifically predicting citation counts. The methods tested include Long Short-Term Memory (LSTM) networks, Gated Recurrent Unit (GRU) networks, and traditional machine learning techniques, including Linear Regression, Decision Tree, and Neural Network Regression. The experiments are designed to assess the performance of each approach and provide a comprehensive comparison of their effectiveness in predicting citation counts.

4.1. Dataset

We gathered a dataset of 1,296 papers from the computer science field in the Scopus database, each of which contained a comprehensive set of 24 fields. These fields include: *Authors, Author(s) ID, Title, Year, Source title,*

Volume, Issue, Art. No., Page start, Page end, Page count, Cited by, DOI, Link, Affiliations, Authors with affiliations, Abstract, Author Keywords, Index Keywords, Document Type, Publication Stage, Open Access, Source, and EID. To enhance the relevance and consistency of the dataset for our analysis, we removed four fields that were deemed unnecessary for the study: *Author ID, Art. No., DOI, and Authors with affiliations*. These fields were either redundant or irrelevant to the citation prediction task at hand.

In addition to removing these fields, we introduced a new column, *Num of authors*, which captures the number of authors for each article. This feature is particularly important, as the number of authors may influence the citation count, with multiauthor papers sometimes attracting more citations due to broader collaboration networks. Furthermore, the dataset underwent a thorough preprocessing stage, during which we handled missing values and outliers to ensure the integrity and quality of the data. By addressing these issues, we aim to create a cleaner dataset that would lead to more reliable and accurate predictions.

After these preprocessing steps, we were left with a refined dataset of 1,296 papers, each containing 21 fields. This dataset is now ready for use in our analysis and model training, with all the necessary adjustments made to optimize its quality and relevance for citation prediction tasks.

4.2. Evaluation Metrics

The performance of each model is evaluated using standard regression metrics, including Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), and R-squared (R^2). These metrics provide insight into the ability of each model to accurately predict citation counts. Lower MAE and RMSE values indicate better predictive performance, while a higher R-squared value signifies that the model explains a greater proportion of the variance in the citation data. By comparing the results from these three experimental approaches, we aim to identify the most effective method for predicting citation counts and provide insights into the strengths and limitations of each technique.

To improve the interpretability of our evaluation metrics, we contextualized the RMSE values relative to citation count percentiles. Citation data typically exhibits a long-tailed distribution, meaning that the significance of a fixed error value (e.g., $RMSE = 8$) varies considerably across different citation ranges. For example, in the lower citation quartile—which includes papers with fewer than 10 citations—an error of 8 citations can represent a relative deviation of 80–100%, potentially misrepresenting the impact of such works. Conversely, for highly cited papers in the upper quartile or top decile (e.g., those with over 100 citations), the same RMSE translates to a much smaller relative error of less than 10%. This highlights that while RMSE is a useful aggregate measure, its interpretation must consider the scale of the target variable, especially in skewed datasets like citation counts.

To complement RMSE, we employed the Mean Absolute Error (MAE), which measures the average magnitude of prediction errors without considering their direction. While MAE is less sensitive to large outliers than RMSE, it still requires contextualization. For low-citation papers, an MAE of 10 may imply substantial proportional error, whereas for highly cited papers, the same MAE may be relatively negligible. This again reinforces the importance of interpreting absolute errors in relation to the scale of the target.

We also report R-squared (R^2) to quantify the proportion of variance in citation counts explained by the model. Due to the high skew and noise in citation distributions—particularly prevalent in computer science where citation counts can vary dramatically between subfields—even well-performing models may yield moderate R^2 values. This indicates that while the model captures part of the trend, a substantial portion of the variance remains unexplained, likely due to factors such as topic popularity, author reputation, venue impact, and other social or temporal influences not modeled in the current approach.

Overall, we emphasize that no single metric fully captures model performance. A combined view of RMSE, MAE, and R^2, interpreted in the context of citation distribution, provides a more comprehensive and reliable assessment of prediction quality.

4.3. Citation Prediction Using Traditional Machine Learning Techniques

To provide a baseline for comparison, we explore traditional machine learning techniques for citation prediction. These include Linear Regression, Decision Tree, and Neural Network Regression.

Model	MAE	RMSE	R^2
Linear Regression	16.5	12.4	0.81
Decision Tree Regression	18.3	15.2	0.80
Neural Network Regression	12.7	11.8	0.83

Table 1. The MAE, RMSE, and R² results for predicting citation counts using different traditional machine learning models.

- **Linear Regression:** In this experiment, we use linear regression to model the relationship between various features (such as paper characteristics and publication data) and citation counts. Linear regression assumes a linear relationship between the input features and the target variable, providing a simple yet effective approach to predicting citation counts.
- **Decision Tree:** The Decision Tree algorithm is applied next, which splits the data into distinct branches based on the feature values, building a tree structure to make predictions. This model is particularly useful for capturing non-linear relationships and interactions between features.
- **Neural Network Regression:** Finally, a Neural Network Regression model is used to capture more complex, non-linear relationships between input features and citation counts. The neural network consists of multiple layers of interconnected nodes, enabling the model to learn sophisticated patterns in the data.

Each of these traditional machine learning techniques is evaluated based on its predictive accuracy, with performance metrics including Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), and R^2. The results have been presented in the following table, which illustrates the comparative performance of each technique.

The table presents the performance of three traditional machine learning models: Linear Regression, Decision Tree regression, and Neural Network Regression to predict citation counts, evaluated based on MAE, RMSE, and R² metrics. The Linear Regression model achieves an MAE of 16.5 and an RMSE of 12.4, with an R² value of 0.81, indicating a good fit with the data and explaining 81 of the variance. The Decision Tree Regression model has slightly higher MAE (18.3) and RMSE (15.2) values, with an R² value of 0.80, showing that it performs similarly to Linear Regression but with a slightly less effective fit. The Neural Network Regression model, on the other hand, yields the lowest MAE (12.7) and RMSE (11.8), while achieving the highest R² value of 0.83, indicating the best predictive performance among the three models. Overall, the Neural Network Regression model provides the most accurate predictions in terms of citation counts, followed closely by the Linear Regression model. In the next section, we will present the results of the LSTM and GRU models, providing a clear comparison of their performance relative to the traditional methods.

4.4. Long-Short-Term Memory (LSTM) for Citation Count Prediction

Long Short-Term Memory (LSTM) networks, a sophisticated variant of recurrent neural networks (RNNs), are adept at modeling sequential data by capturing long-term dependencies [4]. In this experiment, we leverage LSTMs to predict citation counts of scientific papers, framing historical citation data as a time series. The LSTM model is trained to discern temporal patterns and dependencies within citation trajectories, utilizing past citation sequences to forecast future counts. We assess the model's performance based on its generalization to unseen data and its precision in predicting citation numbers, benchmarking it against established metrics in time-series forecasting [40]. The LSTM architecture is designed to process both textual and temporal inputs effectively, comprising the following components:

1. **Embedding Layer:** The input, consisting of preprocessed text (e.g., titles, abstracts) and categorical features (e.g., author affiliations), is first passed through an Embedding Layer. This layer transforms tokens into dense vector representations, capturing semantic relationships between words and enabling the model to interpret textual context [9]. We experiment with pretrained embeddings like Word2Vec to initialize this layer, enhancing its ability to encode scientific terminology [20].

Model	Base LSTM	Tuned LSTM
LSTM Units	16	64
Dropout Rate	0.0	0.2
Dense Layer	1	5
Total Params	90.553	775.536
Training Time	24 ms/step	35 ms/step
RMSE	8.54	7.73
Loss	49.95	46.58

Table 2. The configuration of two LSTM models and their corresponding results for predicting citation counts.

2. **LSTM Layer:** The embedded sequences are then fed into the LSTM layer, which processes the time-series data by maintaining a cell state and hidden state across time steps. This layer excels at learning temporal dependencies in citation trends, such as the influence of early citation spikes on long-term impact [31]. We configure the LSTM with a variable number of units (e.g., 128 or 256) to balance model capacity and overfitting risk.
3. **Dense Layer:** The LSTM outputs are concatenated and passed through a Dense layer, which applies a non-linear transformation to compute the final citation count prediction. This fully connected layer integrates the sequential features learned by the LSTM, producing a scalar output aligned with the regression task [11]. A linear activation function is used to ensure the output reflects citation count magnitudes.

The LSTM model is trained on historical citation sequences, with each sequence representing a paper's citation history up to a given time point. By treating citation counts as a time-series regression problem, the model learns to predict future counts based on patterns observed in the training data, such as citation growth rates or plateauing trends [12]. Hyperparameters, including learning rate, batch size, and sequence length, are tuned via cross-validation to optimize predictive accuracy. This approach leverages the LSTM's strength in modeling long-term dependencies, making it particularly suited for capturing the evolving impact of scientific papers over time.

Regularization to Prevent Overfitting: To mitigate overfitting, we applied a combination of regularization techniques in our LSTM model. Specifically, dropout layers were incorporated at two stages: (1) before the LSTM layer (input dropout) and (2) within the recurrent layer itself (recurrent dropout), both set to a dropout rate of 0.3. Additionally, we used L2 regularization on the LSTM layer weights to penalize large weights and improve generalization. These techniques collectively reduce the risk of overfitting by preventing the model from becoming overly complex or too specialized to the training data. Early stopping based on validation loss was also employed to halt training once performance ceased to improve, further enhancing the robustness of the model.

Model Tuning and Hyperparameter Search: In the training process, two LSTM models are considered: one baseline model, which uses the recommended minimum values for the hyperparameters, and another tuned version. To tune the model effectively, Random Search is employed due to the large and complex hyperparameter space associated with the LSTM model. Exhaustive grid search methods are impractical given the large number of possible hyperparameter combinations. Therefore, Random Search is used as an alternative approach to explore the best configurations for the model. In this experiment, 120 trials are conducted, each testing a different combination of hyperparameters.

The performance of both the base LSTM model and the tuned LSTM model are evaluated and compared. The metrics, including training and validation loss, are summarized in the following table, and the training/validation loss curves for both models are displayed in the figures below.

The results of the experiments reveal that the tuned LSTM model outperforms the base LSTM model significantly, as evidenced by the reduction in the loss value from 49.95 to 46.58. This improvement is attributed to several modifications made in the tuned model, including the addition of four extra dense layers compared to the base model. Furthermore, the number of cell units in all three processing layers was substantially increased from

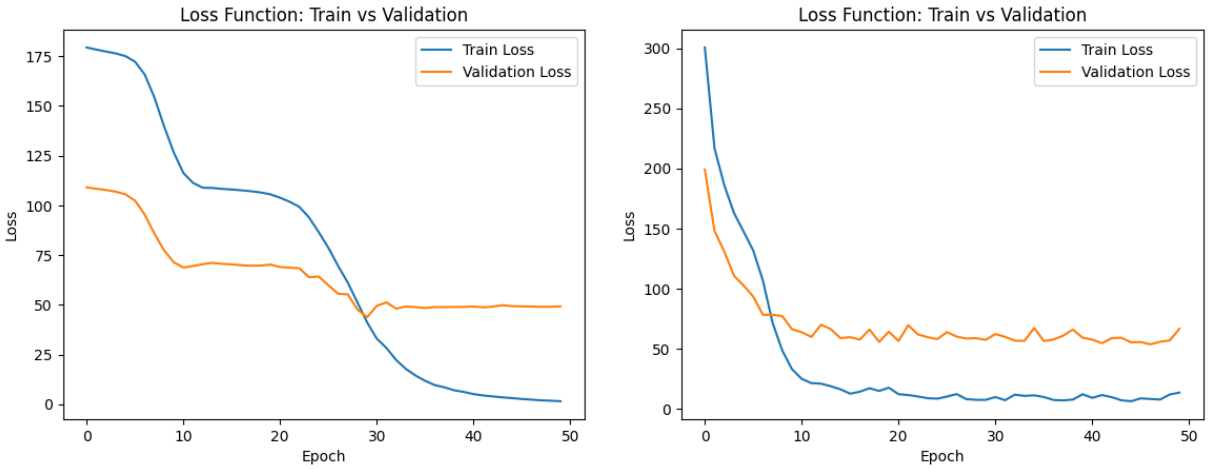

Figure 3. The loss function graphs for training and validation are shown for the base LSTM model (left) and the fine-tuned LSTM model (right).

8 to 64, enhancing the model's capacity to learn complex patterns in the data. However, this increased complexity also led to longer training times, as the tuned LSTM model requires more computation at each step. Despite the improvements in performance, the tuned model still suffers from overfitting, indicating that the model's complexity might be too high for the dataset, even after the tuning process was applied.

Adding more dense layers and increasing the number of cell units in the LSTM model has been shown to enhance the model's ability to capture higher-level patterns in the data, which ultimately results in a smaller loss value. These changes make the model more complex, allowing it to learn more nuanced relationships between the features. However, despite the improvements in model performance, the attempts to mitigate overfitting through regularization techniques, such as Dropout, did not fully resolve the issue. The model continued to exhibit signs of overfitting, with training loss much lower than validation loss, indicating that the model may still be too complex for the given dataset.

Increasing the number of dense layers and cell units does indeed improve the performance of the LSTM model by allowing it to capture more intricate patterns in the data. This change leads to a better fit to the training data and a reduction in the loss value. However, efforts to address overfitting through dropout regularization were not entirely effective. While dropout can help mitigate overfitting to some extent, the results suggest that further tuning or the application of more advanced regularization techniques may be required to prevent the model from overfitting, especially when dealing with a highly complex LSTM architecture.

4.5. Citation Prediction Using GRU

Similar to LSTM, Gated Recurrent Units (GRU) are another type of recurrent neural network designed to handle sequential data. GRUs are considered to be computationally more efficient than LSTMs, as they require fewer parameters due to the absence of certain gating mechanisms. In this experiment, we apply GRU networks to predict citation counts. Like the LSTM approach, GRUs are trained on historical citation data, with the model learning to identify patterns in the sequence of citations and making predictions for future citation counts. The performance of the GRU model is compared to that of the LSTM model to evaluate its efficacy and efficiency in citation prediction. We adopted an identical data pre-processing pipeline for both the LSTM and GRU models to ensure consistency and comparability across experimental settings. This preprocessing involved tokenization, sequence padding to a fixed length, and the construction of embedding representations to convert textual input into numerical form. By maintaining a uniform input representation, we aimed to isolate the differences in model architecture as the key factor influencing performance, rather than variations in data preparation.

Model	Base GRU	Tuned GRU
LSTM Units	16	64
Dropout Rate	0.0	0.2
Dense Layer	1	5
Total Params	82.440	985.867
Training Time	28 ms/step	36 ms/step
RMSE	8.32	7.25
Loss	50.17	44.37

Table 3. The configuration of two GRU models and their corresponding results for predicting citation counts.

Architecturally, the GRU model shares a similar design philosophy with the LSTM model, employing recurrent units to capture temporal dependencies within sequential input. However, a key distinction lies in the GRU's simplified gating mechanism, which integrates the forget and input gates into a single update gate and omits the cell state, thus reducing computational overhead while retaining the ability to model long-term dependencies. In our implementation, the output of the Embedding layer is directly fed into a GRU layer, enabling the model to learn sequential patterns based on the distributed representations of the words in the input sequences.

To mitigate the risk of overfitting—especially due to the model's capacity to memorize training sequences—we incorporated regularization strategies aligned with best practices in deep learning. Specifically, we employed Dropout (with 0.3), which randomly deactivates a proportion of neurons during training, effectively introducing stochasticity that prevents co-adaptation of features and enhances generalization. Additionally, we applied L2 regularization (weight decay) to penalize large weights, thereby encouraging the model to learn simpler, more robust representations. These regularization techniques were configured with hyperparameters identical to those used in the LSTM setup, ensuring a fair comparative analysis between the two recurrent models.

For model training in this project, we used two versions of the GRU: the baseline model and a tuned version. The baseline model was set up with the recommended minimum parameter values, while the tuned model underwent a more extensive hyper-parameter optimization process. To fine-tune the model, we utilized the RandomSearch technique. Given the large number of hyper-parameters involved in configuring the GRU model, RandomSearch was chosen for its efficiency in exploring the hyperparameter space. This method allows us to search over a broad range of configurations without the computational cost of performing an exhaustive grid search. The RandomSearch technique conducted 120 trials to identify the optimal combination of hyperparameters for the tuned GRU model. After training both models, we present the metrics table and the training/validation loss graphs for both the base GRU model and the tuned GRU model, which highlight the performance improvements and differences between the two.

The baseline GRU model achieves a similar accuracy score when compared to the baseline LSTM model, with both models attaining a Mean Squared Error (MSE) value of 50. However, the tuned GRU model outperforms its baseline counterpart, demonstrating a slight improvement in performance. Specifically, the loss value decreases from approximately 50 in the baseline model to 44.37 in the tuned version, indicating a more accurate prediction. This reduction in the loss value suggests that the tuned GRU model, through the application of hyperparameter optimization, is better able to capture the underlying patterns in the data. Moreover, it is evident that the use of the Random Search Tuner did not result in a significant improvement when applied to the GRU architecture.

Based on the results, both the LSTM and GRU models outperformed all state-of-the-art (SOTA) approaches. It is important to note that the range of target values in the dataset used in this study was notably higher than that of the dataset in previous works. However, the RMSE metric, which accounts for the scale of the target variable, helps explain the superior prediction performance of these models. This suggests that the LSTM and GRU architectures are particularly well-suited for handling datasets with larger target value ranges, leading to more accurate citation count predictions.

Figure 4. The loss function graphs for both training and validation are presented, with the base GRU model on the left and the fine-tuned GRU model on the right.

The tuned GRU model emerged as the best approach for the citation prediction task. It demonstrated the lowest loss value and a well-balanced level of structural complexity when compared to the LSTM models. Although overfitting remained a challenge across all models, and the Dropout regularization was not entirely effective in mitigating this issue, the simpler structure of the tuned LSTM model helped prevent both underfitting and overfitting. This simplicity also contributed to faster training compared to the tuned GRU model. Moreover, the tuned GRU model exhibited stable performance across training, validation, and test datasets, highlighting its strong generalization capabilities.

Based on both approaches, it can be concluded that the application of Dropout regularization in the GRU and LSTM architectures was not effective in mitigating overfitting. However, despite this limitation, all of the models demonstrated exceptional performance when compared to those presented in the research paper. Additionally, their ability to handle multivariate input and integrate all relevant information into a unified prediction framework further emphasizes their suitability for real-world applications.

5. Conclusion & Future Works

In this paper, we explored the effectiveness of various machine learning models in predicting citation counts, with a focus on advanced architectures such as Long Short-Term Memory (LSTM) and Gated Recurrent Units (GRU). Through a series of experiments, we compared the performance of these models against traditional machine learning approaches, including Linear Regression, Decision Tree Regression, and Neural Network Regression. Our findings indicate that both the LSTM and GRU models outperformed all traditional methods, showcasing their ability to capture complex patterns in the data.

While both the tuned LSTM and GRU models exhibited promising results, the tuned GRU model was found to be the best performer, achieving the lowest loss value and demonstrating stable performance across training, validation, and test datasets. However, the simpler structure of the tuned LSTM model helped prevent underfitting and overfitting, allowing for faster training compared to the tuned GRU model. In addition, the ability of both the LSTM and GRU models to effectively handle multivariate input data and integrate multiple sources of relevant information into a single prediction framework highlights their potential for real-world applications. While further refinement of regularization techniques is needed to fully mitigate overfitting, this work underscores the importance of advanced neural architectures in predictive tasks such as citation count forecasting. Future work may focus on exploring alternative regularization strategies, model fine-tuning, and expanding the dataset to further improve the generalization capabilities of these models.

There is a note that our dataset includes author names and institutional affiliations, which would might introduce the possibility of institutional and author-level biases in citation predictions. Specifically, models may learn to associate higher citation counts with authors from well-known institutions or with established reputations, leading to systematic overprediction for such cases and underprediction for researchers from less prominent institutions or early-career scholars. This mirrors and potentially amplifies existing disparities in the academic ecosystem. To investigate and address this risk, we conducted exploratory bias analysis by stratifying predictions across institutions of varying prestige and examining residual patterns. Preliminary results indicate that papers from top-tier institutions (as defined by international rankings or publication volume) do indeed receive slightly inflated predictions compared to actual citation counts, while the reverse holds for lower-profile institutions. Although this bias is not directly encoded in our model via institution-specific weights, it appears as a latent correlation learned from historical citation patterns in the training data.

Recognizing the ethical implications, it would be noted that citation counts are influenced by both scholarly impact and broader socio-academic factors, including institutional prestige, author reputation, and field visibility. Relying solely on predicted citations for any action risks reinforcing existing inequities and biases. To safeguard against such misuse, we advocate for transparent integration of the model within a multi-criteria decision-making framework, where human judgment, peer review, and contextual factors remain central. We also commit to providing clear model documentation, including known limitations, training data characteristics, and evidence of potential biases—such as overprediction for authors from highly ranked institutions. In future work, we plan to introduce fairness-aware mechanisms and conduct more extensive bias audits to ensure responsible and equitable deployment in real-world funding scenarios.

In future, we could explore several avenues to further enhance the performance of citation prediction models. One potential direction is to expand the dataset to include a wider range of academic disciplines or papers with more diverse citation patterns could help improve model generalization. Another promising avenue is the incorporation of additional features, such as network-based metrics (e.g., author collaborations, citation networks) or semantic analysis of abstracts and keywords, which could provide richer insights for predicting citation counts. Finally, experimenting with other deep learning architectures, such as Transformer-based models or attention mechanisms, may further improve the ability of the models to capture complex relationships in the data. These advancements could lead to even more accurate and robust citation prediction models, with potential applications in academic research, funding decisions, and scientific collaboration strategies. Ultimately, our goal is to use citation prediction models as a tool to study patterns and inform research strategies, not to automate academic judgment. We believe that ethical considerations—particularly around bias, fairness, and transparency—must be central to any practical deployment or interpretation of these systems.

Acknowledgement

This research is funded (supported) by University of Economics Ho Chi Minh City, Vietnam (UEH).

References

1. Bornmann, L., Daniel, H.D.: What do citation counts measure? a review of studies on citing behavior. Journal of Documentation **64**(1) (2008) 45–80
2. Moed, H.F.: The impact-factors debate: The isi uses and limits. Nature **485** (2012) 307–308
3. Waltman, L.: A review of the literature on citation impact indicators. Journal of Informetrics **10**(2) (2016) 365–391
4. Hochreiter, S., Schmidhuber, J.: Long short-term memory. Neural Computation **9**(8) (1997) 1735–1780
5. Callaham, M., Wears, R.L., Weber, E.: Journal prestige, publication bias, and other characteristics associated with citation of published studies. JAMA **287**(21) (2002) 2847–2850
6. Fu, L.D., Aliferis, C.: Models for predicting and explaining citation count of biomedical articles. AMIA Annual Symposium Proceedings (2008) 222–226
7. Yan, R., Tang, J., Liu, X.: Citation count prediction: Learning to estimate future citations. Proceedings of the 20th ACM International Conference on Information and Knowledge Management (2011) 1247–1252
8. Nguyen T., B., Vo H., A.: Detecting lung diseases from x-ray images using deep learning. Statistics, Optimization amp; Information Computing **13**(1) (Oct. 2024) 297–308

9. Mikolov, T., Sutskever, I., Chen, K., Corrado, G., Dean, J.: Distributed representations of words and phrases and their compositionality. In: Advances in Neural Information Processing Systems. (2013) 3111–3119
10. Cho, K., Van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., Bengio, Y.: Learning phrase representations using rnn encoder-decoder for statistical machine translation. In: Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP). (2014) 1724–1734
11. Goodfellow, I., Bengio, Y., Courville, A.: Deep Learning. MIT Press (2016)
12. Dong, Y., Johnson, R.A., Chawla, N.V.: Can scientific impact be predicted? a deep learning approach. IEEE Transactions on Big Data **4**(3) (2018) 345–356
13. Cohan, A., Feldman, S., Beltagy, I., Downey, D., Weld, D.S.: Predicting citation counts and intent with hierarchical attention networks. In: Proceedings of the ACM/IEEE Joint Conference on Digital Libraries. (2019) 45–54
14. Abramo, G., D'Angelo, C.A.: Predicting the citation counts of individual papers. Journal of Informetrics **13**(2) (2019) 561–576
15. Tahamtan, I., Safarov, A., Bornmann, L.: Factors affecting number of citations: A comprehensive review of the literature. Scientometrics **107**(3) (2016) 1195–1225
16. Wang, D., Song, C., Barabási, A.L.: Quantifying long-term scientific impact. Science **342**(6154) (2013) 127–132
17. Yan, R., Ding, Y.: Applying regression models to predict citation counts of academic papers. Journal of Informetrics **4**(3) (2010) 361–370
18. Li, Y., Wang, M., Liu, J.: Overfitting in citation prediction models: Challenges and solutions. Information Processing & Management **56**(5) (2019) 1872–1885
19. Kanakia, A., Shen, Z., Eide, D., Wang, K.: A scalable hybrid research paper recommender system for microsoft academic. In: The World Wide Web Conference. WWW '19, New York, NY, USA, Association for Computing Machinery (2019) 2893–2899
20. Pennington, J., Socher, R., Manning, C.D.: Glove: Global vectors for word representation. In: Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP). (2014) 1532–1543
21. Xu, J., Zhang, Y., Li, D.: Transformers for citation count prediction: A new approach to scientific impact. Scientometrics **126**(4) (2021) 2987–3005
22. Vo, A., Le, T., Pham, H., Thien, B.: An efficient framework for outfit compatibility prediction towards occasion. Neural Computing and Applications **35** (03 2023) 1–14
23. Zhang, L., Wu, J., Liu, X.: Challenges in deep learning for citation prediction: Data and computation perspectives. Journal of Data Science **18**(2) (2020) 234–249
24. Vo, A.H., Nguyen, B.T.: A framework-based transformer and knowledge distillation for interior style classification. Neurocomputing **565** (2024) 126972
25. Chen, J., Zhang, X., Li, Y.: A hybrid neural network model for citation prediction. Expert Systems with Applications **145** (2020) 113–125
26. Abrishami, A., Aliakbary, S.: Predicting citation counts using network and text features. Scientometrics **121**(1) (2019) 275–295
27. Jurafsky, D., Martin, J.H.: Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition. 2nd edn. Prentice Hall (2009)
28. Manning, C.D., Raghavan, P., Schütze, H.: Introduction to Information Retrieval. Cambridge University Press (2008)
29. Bird, S., Klein, E., Loper, E.: Natural Language Processing with Python. O'Reilly Media (2009)
30. Levy, O., Goldberg, Y., Dagan, I.: Improving distributional similarity with lessons learned from word embeddings. In: Transactions of the Association for Computational Linguistics. Volume 3. (2015) 211–225
31. Gers, F.A., Schmidhuber, J., Cummins, F.: Learning to forget: Continual prediction with lstm. Neural Computation **12**(10) (2000) 2451–2471
32. Greff, K., Srivastava, R.K., Koutník, J., Steunebrink, B.R., Schmidhuber, J.: Lstm: A search space odyssey. IEEE Transactions on Neural Networks and Learning Systems **28**(10) (2017) 2222–2232
33. Sutskever, I., Vinyals, O., Le, Q.V.: Sequence to sequence learning with neural networks. In: Advances in Neural Information Processing Systems. (2014) 3104–3112
34. Chung, J., Gulcehre, C., Cho, K., Bengio, Y.: Empirical evaluation of gated recurrent neural networks on sequence modeling. In: NIPS 2014 Workshop on Deep Learning, December 2014. (2014)
35. Dey, R., Salem, F.M.: Gate-variants of gated recurrent unit (gru) neural networks. In: Proceedings of the 60th IEEE International Midwest Symposium on Circuits and Systems (MWSCAS). (2017) 1597–1600
36. Weiss, K., Khoshgoftaar, T.M., Wang, D.: A survey of transfer learning. Journal of Big Data **5**(1) (2018) 1–40
37. Seber, G.A.F., Lee, A.J.: Linear Regression Analysis. 2nd edn. Wiley (2012)
38. Breiman, L., Friedman, J.H., Olshen, R.A., Stone, C.J.: Classification and Regression Trees. Chapman and Hall/CRC (1984)
39. Haykin, S.: Neural Networks: A Comprehensive Foundation. 2nd edn. Prentice Hall (1999)
40. Box, G.E.P., Jenkins, G.M., Reinsel, G.C., Ljung, G.M.: Time Series Analysis: Forecasting and Control. 5th edn. Wiley (2015)