applied sciences

MDPI

Article

A Hybrid Deep Learning Model Based on Local and Global Features for Amazon Product Reviews: An Optimal ALBERT-Cascade CNN Approach

Israa Mustafa Abbas [1], İsmail Atacak [2,*], Sinan Toklu [2], Necaattin Barışçı [2] and İbrahim Alper Doğru [2]

[1] Graduate School of Natural and Applied Sciences, Gazi University, Central Campus, 06500 Ankara, Turkey; 23833304013@os.gazi.edu.tr
[2] Department of Computer Engineering, Faculty of Technology, Gazi University, 06560 Ankara, Turkey; sinantoklu@gmail.com (S.T.); nbarisci@hotmail.com (N.B.); iadogru@gmail.com (İ.A.D.)
* Correspondence: iatacak@hotmail.com; Tel.: +90-537-726-68-91

Abstract

Natural Language Processing (NLP) is a valuable technology and business topic as it helps turn data into useful information with the spread of digital information. Nevertheless, there are some difficulties in its use, including the language's complexity and the data quality. To address these challenges, in this study, the researchers first performed a series of ablation experiments on 14 models derived from various variations in Deep Learning (DL) methods, including A Lite BERT (ALBERT) together with Convolutional Neural Networks (CNNs), Long Short-Term Memory (LSTM), Bidirectional LSTM (BiLSTM), Max Pooling layer, and attention mechanism. Subsequently, they proposed an ALBERT-cascaded CNN hybrid model as an effective method to overcome the related challenges by evaluating the performance results obtained from these models. In the proposed model, a transformer architecture with parallel processing capability for both word and subword tokenization is used in addition to creating contextualized word embeddings. Local and global feature extraction was also performed using two 1-D CNN blocks before classification to improve the model performance. The model was optimized using an advanced hyperparameter optimization tool called OPTUNA. The findings of the experiment conducted with the proposed model were obtained based on Amazon Fashion 2023 data under 5-fold cross-validation conditions. The experimental results demonstrate that the proposed hybrid model exhibits good performance with average scores of 0.9308 (accuracy), 0.9296 (F1 score), 0.9412 (precision), 0.9182 (recall), and 0.9797 (AUC) in the validation dataset, and scores of 0.9313, 0.9305, 0.9414, 0.9199, and 0.9800 in the test dataset. In addition, comparisons of the model with models in studies using similar datasets support the experimental results and reveal that it can be used as a competitive approach for solving the problems encountered in the NLP field.

Keywords: deep learning; ALBERT-Cascade CNN hybrid model; OPTUNA framework; natural language processing; sentiment analysis

Check for updates

Academic Editors: Tymoteusz I. Miller and Yoonsik Choe

Received: 27 November 2025
Revised: 16 December 2025
Accepted: 18 December 2025
Published: 19 December 2025

Copyright: © 2025 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license.

1. Introduction

Natural Language Processing (NLP) is a domain within computer science and Artificial Intelligence (AI) that allows computers to comprehend, analyze, and respond to human language. Its goal is to connect human communication with computer functionality. To capture human emotions and opinions expressed in written texts, NLP is employed,

studying these feelings to assist businesses in understanding customer perspectives and improving their services. Nonetheless, online reviews play an important role in e-commerce and are equivalent to trying products in stores. Seventy-five percent of consumers check online reviews when researching local businesses [1]. Especially on platforms like Amazon, Reviews play a crucial role in seller rankings and conversions, with 45% of European shoppers and 65% of Turkish consumers finding product reviews helpful [2]. Additionally, 97% of business owners consider online reputation management crucial [3].

Transformer models enhance computational efficiency through parallel processing. Traditionally, language models can only read the input text sequentially from left to right or right to left, struggling to capture contextual relationships across text inputs [4]. To address this limitation, transformer learning has been introduced with the capability to read sequences bidirectionally, known as bidirectionality, from both sides. Utilizing this feature, BERT is pre-trained on two interrelated NLP tasks: Masked Language Modeling (MLM) and Next-Sentence Prediction (NSP) [4,5]. In recent years, there has been a notable increase in the application of Deep Learning (DL) techniques to address various NLP challenges. Early DL models struggled to capture contextual relationships within text inputs [4]. To overcome this issue, hybrid models have emerged that aim to leverage multiple models simultaneously, potentially enhancing the extraction of features from sequences by integrating two or more models.

This study proposes the ALBERT-Cascade CNN hybrid model as an innovative approach for analyzing customer sentiments regarding products through binary classification. The effectiveness of the method was evaluated using the Amazon Fashion 2023 dataset. To ensure high performance, essential features were extracted from ALBERT's last four hidden layers using 1-D CNN blocks and MaxPooling in the output stages. The CNN stage is essential for identifying fundamental information through multiple parallel blocks with varying kernel sizes and filters to extract local and global features. This integration is particularly useful for managing uncertainty, which is a significant challenge in NLP. The proposed method's hyperparameter optimization was conducted using OPTUNA Bayesian Optimization (TPE—Tree-structured Parzen Estimator) with a 5-fold cross-validation. Considering the experimental results of the model, it stands out as a competitive NLP model that demonstrates successful performance among existing approaches. This study contributes to the literature in five primary ways:

- This approach, designed with ALBERT and dual-layer CNN blocks, combines both local and global information representation. By learning micro and macro-level relationships at the word and sentence levels, it becomes a highly effective method for NLP applications.
- The proposed approach introduces a computationally efficient model for NLP applications by leveraging ALBERT's parameter sharing (reducing model size) and CNNs' parallel processing capability (accelerating training).
- The main difficulty in sentiment analysis lies in managing long-term dependencies and various lexicons in the text [6]. This makes feature extraction and accurate classification of such texts significantly challenging in natural language processing [6,7]. Leveraging the unique strengths of ALBERT and CNN, the model effectively addresses long and short-term token lengths in the dataset, a key issue faced in real-world sentiment analysis.
- This study utilized advanced hyperparameter tuning using the next-generation OPTUNA framework to determine the optimal configuration for the proposed model. The empirical results show that the ALBERT-Cascade CNN architecture significantly improves performance compared to contemporary methods. In addition, the bias of the proposed model was reduced, and the effectiveness of the established architec-

ture was rigorously measured through cross-validation and ablation studies. These assessments validate the robustness and generalizability of the proposed approach by setting a new benchmark for sentiment analysis, particularly in scenarios with limited data availability.

- Proposed hierarchically designed cascade 1-D CNN architecture and ALBERT transformer prevent the increase in the number of parameters and effectively deal with memory limitation by deploying ALBERT transformer model that assists to compress the text representation (embedding factorization), to guarantee that resource efficiency from input to final classification.

This paper is organized into five sections. Section 1 explores the topic's relevance, identifies key issues, and discusses methodological approaches grounded in the existing literature while emphasizing how our proposed model contributes to this field. Section 2 reviews relevant literature based on BERT and DL foundations. Section 3 describes the dataset used in the study, the pre-processing steps, the proposed model, and the evaluation criteria. Section 4 analyzes experimental findings alongside their implications. Finally, Section 5 offers an overall assessment of the experimental results, acknowledges the limitations of the proposed model, and suggests future research directions.

2. Related Works

This section explores the application of models formed by various combinations of BERT and DL techniques to sentiment analysis tasks using diverse data sources. These studies predominantly highlight hybrid models that merge BERT's contextual word embedding with DL's capability to identify local patterns within textual data. In these frameworks, BERT is combined with DL approaches, such as CNN, LSTM, and BiGRU, to assess sentiments across various datasets, particularly those from social media interactions, product evaluations, and film reviews. The versatility of this integration has been confirmed across several fields, demonstrating its robustness and adaptability in processing distinct textual data for sentiment classification purposes.

In studies by Jain et al. [8], Mutinda, Mwangi, and Okeyo [9], and Zhang [10], researchers employed well-known hybrid approaches to introduce BERT- and CNN-based models as effective hybrid models for sentiment analysis across various fields' text reviews. In these models, BERT selects and vectorizes specific words from the input text segments, while CNN is a DL classifier for feature mapping and sentiment classification outputs. Jain et al. [8] suggested the use of a BERT-based dilated convolutional neural network (BERT-DCNN) as a robust sentiment analysis framework for text processing. This structure represents a hybrid system that integrates the BERT architecture with a triplet-dilated CNN architecture and a global pooling layer. The researchers validated the effectiveness of the proposed model using texts from comments made by airline passengers on Twitter, achieving notable results, including 86.3% accuracy, 81% macro F1 score, 87% precision, and 86% recall. Mutinda, Mwangi, and Okeyo [9] introduce a Lexicon-Enhanced Bert Embedding (LeBERT) Model with CNN to overcome the limitations of word-based and ML methodologies. This study leverages three datasets—Amazon product reviews, IMDb movie reviews, and Yelp restaurant reviews —to derive experimental results for their proposed models. The experiments yielded F-measure scores of 88.73%, 86.65%, and 82.52% for the Yelp, IMDb, and Amazon datasets, respectively, demonstrating that their model outperformed existing models in terms of performance. Zhang [10] investigated a BERT-CNN hybrid model designed explicitly for movie reviews. Experiments conducted using the IMDb dataset revealed superior performance in handling negative reviews.

The other significant combinations involve hybrid architectures that integrate BERT with CNN-LSTM/BiLSTM/GRU/BiGRU-based models. Deng et al. [11] introduced a

hybrid model in which BERT-embedded data structures were combined with CNN-LSTM-attention layers to extract features, which were then weighted and classified into sentiment categories using dense and output layers. This study evaluated the performance of this model against Text-CNN, LSTM, BiLSTM-ATT, Attention-Based Convolutional Neural Network (ABCNN), and Hierarchical Attention Network (HAN) models within the sentiment analysis framework. The experimental results indicate that the proposed BERT-CNN-LSTM-ATT model outperforms existing models with an accuracy rate of 90.4%, a Macro F1 score of 88.2%, and an F1 Score of 89.5%. Kaur and Kaur [12] discussed a novel model that integrates the BiLSTM and BERT frameworks with CNN layers (BERT-BiCNN) to address the challenges in identifying and classifying software requirements during system design. In this configuration, BERT acts as a word embedding layer, whereas BiLSTM captures contextual information. Meanwhile, CNN plays a role in diminishing the dimensionality of the feature space. The experimental results derived from the PROMISE dataset show that the proposed BERT-BiCNN model outperforms other methods in both binary and multiclass classification tasks. Silitonga et al. [13] presented comparative studies on sentiment analysis using BERT-CNN, Trans-BiLSTM, and RoBERTa models. Utilizing the Amazon Fine Food Reviews dataset, this study showed that Trans-BiLSTM could interpret human emotions expressed in textual data more effectively than BERT-CNN and RoBERTa, achieving an average F1 score of 85.37%. Gupta, Prakasam, and Velmurugan [14] introduced a novel approach that integrates BERT embeddings with BiLSTM-BiGRU and a 1-D CNN as a robust model for binary classification tasks. In this framework, both the BERT embedding and BiLSTM-BiGRU fulfill their designated roles. A self-attention layer was incorporated to improve the contextual understanding. For sensitivity classification within the dataset, a 1-D CNN is employed. The experimental results revealed that the proposed combination of BERT embedding—BiLSTM-BiGRU—self-attention—1-D CNN attained an impressive accuracy rate of 93.89% along with an AUC score of 98.28%, surpassing the current models in binary sentiment classification. Xiao and Luo [15] presented a sentiment analysis approach based on a transformer-BERT hybrid model as the final combination. This model enhances its capability to represent textual features by combining word vectors extracted via BERT with topic vectors. The integrated vectors were processed using the BiGRU technique to learn contextual attributes before being input into the transformer architecture. The experimental results demonstrate that the proposed framework achieves impressive metrics among the current models, including an accuracy of 93.77%, precision of 93.89%, recall of 93.69%, and an F1 score of 93.86%.

As can be seen from the studies mentioned, recent research in text review has been concerned with hybrid models that combine DL architecture with pretrained language models. Hybrid models combining BERT embedded with DL models like CNN, LSTM, BiLSTM, GRU, and attention layers have been effectively used in text reviews. However, the contributions of model components to performance have not been systematically studied in most studies, leaving a unique gap in the research in this field. Conversely, although various layer intersections have been successfully experimented with in hybrid designs, like LeBERT-CNN and BERT embedding—BiLSTM—BiGRU—self-attention—1-D CNN, research papers that have explored such models in considerable detail within the context of ablation analysis have been relatively few. In this context, this study performed comprehensive ablation analyses on hybrid models in which ALBERT-based embeddings were progressively integrated with CNN, LSTM, BiLSTM architectures, attention, and max-pooling layers. Based on experimental studies, the ALBERT-Cascade CNN architecture, which achieved successful performance results, was proposed for Amazon Fashion text review classification.

3. Materials and Methods

This section outlines the procedures involved in the study, including the dataset utilized, data preprocessing, the proposed model architecture, the hyperparameter optimization strategy, and the model evaluation. Detailed explanations of these processes are provided below under specific subheadings.

3.1. Dataset and Preprocessing

This study employed the Amazon Fashion 2023 dataset, a comprehensive collection of Amazon Reviews compiled by McAuley Lab in 2023 [16]. The Amazon Fashion 2023 dataset encompasses a wealth of valuable attributes, such as user reviews (ratings, text comments, and helpfulness ratings), product metadata (descriptions, prices, raw images), and co-purchase links (graphs illustrating how users interact with various items). For experimental purposes, a small function was created using the Pandas version 2.2.2' Python library to select 100,000 random samples from the main Amazon Fashion 2023 dataset containing 2.5 million data points. After removing irrelevant and duplicate entries, the final dataset (see the ReadMe.docx file in the Supplementary Materials section) comprised 49,957 positive and 49,992 negative reviews.

This dataset includes product ratings, user reviews in text format, helpfulness votes, and various other features of interest. The rating scale ranges from 1 to 5, where scores of 1–2 indicate negative reviews and 3–5 reflect positive reviews. This labeling follows the built-in classification criteria. For this study, only superficial data cleaning was performed, as deep cleaning often reduces the focus on the critical details. Recent studies have also shown that deep data cleaning is ineffective in DL applications [17,18]. In this context, HTML tags and numbers were removed from the datasets.

3.2. Proposed Model Architecture

The ALBERT-Cascade CNN method, proposed for analyzing Amazon product reviews, is an improved hybrid NLP model that integrates ALBERT's strengths with sequential CNN blocks and effective hyperparameter optimization. The main structure of the model combines ALBERT as a pre-trained language model with CNN's local and global feature extraction capabilities. Several distinct aspects differentiate the proposed model from similar hybrid approaches that combine BERT and CNN developed in this field. First, ALBERT's parameter-sharing enables a more dimensionally reduced architecture than BERT-based models. Second, the model's ability to extract both local and global features through cascade-connected CNN blocks allows it to better capture the multilayered nature of language based on word patterns and meaning blocks. Third, OPTUNA optimization ensures effective NLP performance.

In this study, instead of parallel architectures with different window sizes, a sequential 1-D CNN structure was preferred for text classification due to the need for Hierarchical Semantic Feature Extraction and Efficient Context Expansion. In Natural Language Processing, a 1-D CNN layer typically focuses on a specific word group. However, adding a second 1-D CNN layer sequentially allows the model to see a broader context. Therefore, the first layer will capture local features, while the second layer will focus on global features. This allows for deeper processing with fewer parameters compared to a parallel structure. Furthermore, sequential blocks increase the depth of the developed model, offering more nonlinear activation possibilities for the data used compared to a parallel architecture. This improves the ability to distinguish subtle differences in meaning in complex operations such as sentiment analysis. In addition, parallel architectures generally increase the size of the model, resulting in a higher number of parameters. In the proposed study, this sequential structure allows for the acquisition of deeper semantic features without inflating

the model size. This allows a larger portion of the input to be seen and the detection area to be effectively increased without the use of multiple parameters that arise in multi-branch parallel architectures.

As shown in Figure 1, this model contains a structure with six basic layers: an ALBERT Encoder, a 1-D CNN Block1, a 1-D CNN Block2, a Concatenate layer, a Fully Connected layer, and an output. In this structure, the raw text is first tokenized, with each token assigned a numerical ID for processing by the language model. The ALBERT layer then processes the digitized input text and extracts context-sensitive embedding vectors for each token. ALBERT was chosen over BERT in this layer due to its smaller size and superior ability to capture deep contextual information. The final output dimensions of ALBERT were shaped as [64, 128, and 768] based on the batch size, embedding dimension, and sequence length. An embedding dimension of 128 decreases the model size, allowing the CNN block to operate rapidly. However, a sequence length of 768 ensures absolute semantic learning by maintaining long contexts.

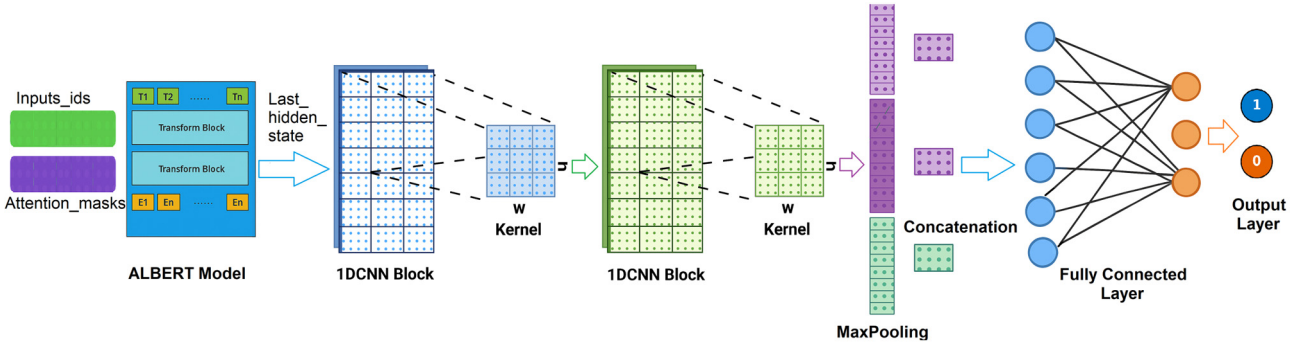

Figure 1. Architecture of the proposed ALBERT-Cascade CNN model.

The 1-D CNN Block1 captures meaningful patterns within word groups of specific lengths by running multiple filters with varying kernel sizes on ALBERT embedding vectors. This process is called local feature extraction because it targets a specific region. One-dimensional. 1-D CNN Block2, equipped with MaxPooling, performs another convolution on Block1's output and applies max pooling to obtain the most dominant activation for each filter. Consequently, information regarding whether the filters are active or not comes to the foreground rather than their spatial positions. This process is called the global representation phase because this information produces signals that summarize the entire text. The Concatenate layer gathers multi-scale features from the Block2 outputs into a single-scale vector. This vector of size [64, 192] represents a rich set of features for each pattern. The Fully Connected layer applies a linear transformation to the features combined by the concatenate layer, producing class scores. In the output layer, these scores are processed through the softmax function, normalizing them to a 0–1 range where their sum equals 1. The normalized values characterize the class probabilities, with the highest probability indicating the model's prediction. The layers of the proposed model are described in the following subsections.

3.2.1. ALBERT

A Lite BERT (ALBERT) is an efficient BERT model that utilizes two parameter reduction techniques, namely, Factorized Embedding Parameterization and "Cross-Layer Parameter Sharing", to reduce parameters and increase scalability. Both techniques significantly reduce the number of BERT parameters and improve parameter efficiency without major performance loss. An ALBERT configuration similar to BERT-large has 18 times fewer parameters and can learn approximately 1.7 times faster than BERT. The parameter reduction technique also functions as a form of regularization that balances learning and

aids generalization. As a result of these design decisions, it is possible to scale to larger ALBERT configurations with fewer parameters than BERT-large [19].

ALBERT reduces the model size without significantly compromising performance. Its factorized embedding parameterization is a key innovation that addresses scalability issues with large embedding matrices. Language models typically use a large O(V × H) embedding matrix to map words to numerical representations. ALBERT significantly reduces memory usage and computational overhead by splitting a single large matrix into two smaller matrices.

$$E_1 \in \mathbb{R}^{v \times E} \tag{1}$$

Equation (1) means that the vocabulary is projected into a lower-dimensional space.

$$E_2 \in \mathbb{R}^{v \times H} \tag{2}$$

Equation (2) maps the low-dimensional representation to the hidden space. This reduces the parameter count to:

$$O(V \times E + E \times H) \text{where } E \ll H \tag{3}$$

ALBERT decreases the number of training parameters without compromising expressive power through shared weight matrices across transformer layers. Replacing BERT's Next Sentence Prediction (NSP) with Sentence Order Prediction (SOP) enhances inter-sentence coherence, aligning with the pre-training goal. A classification layer applied to [CLS] embeddings from both segments determines if adjacent segments are properly ordered or swapped. The SOP loss is computed via cross-entropy, while ALBERT maintains BERT's self-attention mechanism, which relies on the scaled dot-product attention formula.

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V \tag{4}$$

where query Q, key K, and value V matrices are derived from the input representation. ALBERT's total pre-training loss combines the Masked Language Model (MLM) loss, used for predicting randomly masked tokens, with the SOP loss to form the overall loss.

$$\mathcal{L} = \mathcal{L}_{\text{MLM}+}\mathcal{L}_{\text{SOP}} \tag{5}$$

These innovations, including parameter sharing and embedding factorization, significantly reduce the number of parameters, making ALBERT more computationally efficient while preserving its ability to perform large-scale language understanding tasks [19].

The Albert Tokenizer transforms raw text into a format that the ALBERT model can comprehend and process. Figure 2 shows the processing strategy of the ALBERT tokenizer. In particular, it breaks down the input text into smaller components known as tokens.

Depending on the tokenization settings, these tokens can be words, subwords, or characters. The main features of the Albert Tokenizer include subword tokenization, such as the WordPiece method utilized in models such as BERT and ALBERT. This approach permits more efficient processing of words not found in the lexicon by breaking them into smaller subword units. [CLS] tokens are added at the beginning of a sequence and are often used in classification tasks. The [SEP] token is a separator that differentiates between various sentences or segments. The tokenizer also creates an attention mask that informs the model which tokens represent actual words, and which are padding tokens, enabling the model to concentrate on the pertinent parts of the input. In this study, the ALBERT

model was utilized as the first layer in the proposed architecture and was trained using two primary inputs: input IDs and attention masks.

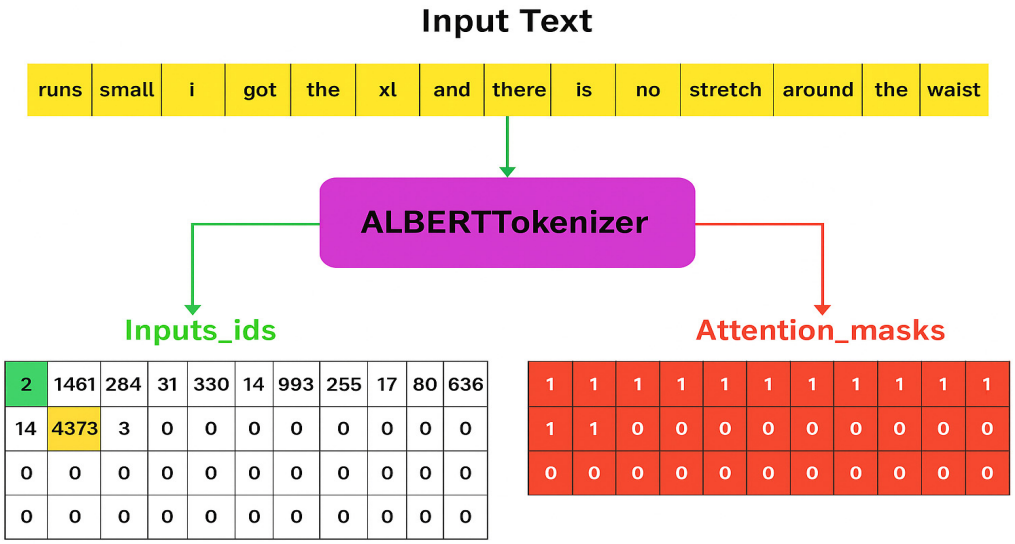

Figure 2. The processing strategy of the ALBERT tokenizer.

After completing the training process, data were extracted from the final hidden state of the ALBERT model, which holds contextualized representations of the input tokens. These representations were then forwarded as inputs to the 1-D CNN blocks. These blocks play a crucial role in retaining key features, thereby enhancing efficiency and performance.

3.2.2. One-Dimensional . (1-D) CNN Blocks

Recent advancements in deep learning have demonstrated a strong focus on CNN architecture to overcome challenges such as image, audio, and video processing, and object detection [20]. In the proposed model, where the cascaded 1-D CNN block was used, the preprocessed text input for the first CNN block was embedded with ALBERT. The last_hidden_state output of ALBERT was sent as input to the first 1-D CNN block for local feature extraction, and its output was applied as input to the second 1-D CNN block, followed by MaxPooling for global feature extraction.

The first block vectors are represented as a matrix $X \in \mathbb{R}^{T \times d}$, where T denotes the sequence length and d the word embedding dimension. The first 1-D CNN block takes the last_hidden_state as input, while the second block uses the output of the first block. For both the first and second blocks (where i = 1, 2), the convolution operation at position t is computed using the formulation in Equation (6).

$$z_t^{(i)} = \sum_{k=0}^{K^{(i)}-1} \sum_{j=0}^{d^{(i)}-1} x_{t+k,j}^{(i)} . w_{k.j}^{(i)} \quad (6)$$

where $K^{(i)}$ is the filter size; $d^{(i)}$ is the number of embedding dimensions for the first block; $w_{k.j}^{(i)}$ is the filter weight. The best model parameters are exhibited in Table 1. The out-feature maps $z^{(i)} \in \mathbb{R}^{T^{(i)}-K^{(i)}+1}$ is transferred directly to the next layer without applying any pooling layer or activation function.

Table 1. The hyperparameter value ranges, along with the best optimal values determined using the OPTUNA framework.

Parameter	Hypermeter Value Ranges	Best Parameter Values
Learning rate	1×10^{-5}, 1×10^{-3} altered to 1×10^{-7}, 1×10^{-2}	0.000031
Weight decay	1×10^{-6}, 1×10^{-2}	0.000005
Dropout rate	0.1, 0.5	0.20
Batch size	16, 32, 64	16
Number of epochs	5, 20	5
Number of filters	64, 128, 192, 256, 320	128
Hidden dimension	50, 600	498
Filter sizes	(2, 3, 4), (3, 4, 5), (1, 2, 3), (1, 3, 5)	(1, 2, 3)

To reduce the dimensions and extract features across all 1-D CNN blocks, the second block architecture was modified, as shown in Figure 1. As a result, the first layer was subjected to the MaxPooling operation with a $\mathcal{P}$ pool size.

$$\mathcal{P}_{\mathrm{f}} = \max\left(z_{\mathrm{f},\wp}^{(3)}, z_{\mathrm{f},\wp}^{(3)} + 1, \ldots, z_{\mathrm{f},\wp}^{(3)} + \wp - 1\right) \tag{7}$$

where $\mathcal{P}_f$ is the pooled value for the f-th feature map. This proposed architecture allows the model to capture local patterns from the first block and higher-level aggregated features from the second block. This layer is designed to identify and retain the optimal feature values, and furthermore, enhance the efficiency of the model.

3.2.3. Fully Connected Layer

The Fully Connected (FC) layer is a key component of many neural network architectures and serves as the final stage, where all neurons in the previous layer are connected to those in the current layer. This layer combines the high-level features extracted by the previous layer to produce the final output. To reduce nonlinearity and improve the model's ability to learn complex patterns, a Rectified Linear Unit (ReLU) activation function, defined as $ReLU(x) = \max(0, x)$ was applied after the FC layer. This helps alleviate the vanishing gradient issue and speeds up convergence during training. Dropout is also employed alongside the FC layer before the output layer to prevent overfitting and improve generalization. The dropout operation randomly deactivates certain neurons during training, encouraging the network to rely on various pathways and reducing co-adaptation among neurons. The integration of the ReLU activation and dropout mechanisms in the FC layer contributes to building robust and efficient neural networks.

3.3. Hyperparameter Optimization

To optimize the hyperparameters of the proposed ALBERT-Cascade CNN model, we employed a robust framework, OPTUNA. During model training, it can automatically optimize hyperparameters, such as the learning rate, layer count, and batch size [21]. This feature distinguishes it from other optimizers used for hyperparameter optimization. In addition, several factors elevate OPTUNA above conventional optimizers, including its support for distributed computing, pruning features, and integration with various databases and search algorithms [22].

To perform hyperparameter optimization on our model, we utilized the OPTUNA version 4.5.0 Python library, which facilitates the automatic tuning of ML models [21]. This library is compatible with various ML frameworks, such as TensorFlow, Sklearn, and PyTorch. In this study, we opted for PyTorch in our experiments. We chose Adam as the optimizer for OPTUNA. The model optimization focused on several hyperparameters, including the learning rate, weight decay, dropout rate, batch size, filter number, hidden di-

mension, filter size, and epoch number. Twenty trials were conducted for these experiments, and the results are presented in Table 1 in Section 3.2.2.

3.4. Performance Assessment

Sentiment analysis involves various NLP techniques to identify emotional states within a text. These techniques primarily assess the emotional tone and classification of written content. Consequently, sentiment analysis is a challenging classification task. Performance measures based on confusion matrices were employed to evaluate the accuracy and effectiveness of the methods used to address such challenges. A confusion matrix effectively compares the predicted outcomes with the actual results, categorizing them into four key components: true positive (TP), false positive (FP), true negative (TN), and false negative (FN). This study leverages metrics such as accuracy, F1-score, precision, recall, and area under the curve (AUC) from confusion matrices when assessing the performance of the methods during experiments related to ablation studies and the proposed method. The measurement results for these metrics were derived from four core components within the confusion matrices [23]. The overall formulation is as follows:

$$\text{Accuracy} = \frac{TP + TN}{TP + FP + TN + FN} \tag{8}$$

$$\text{Recall} = \frac{TP}{TP + FN} \tag{9}$$

$$\text{Precision} = \frac{TP}{TP + FP} \tag{10}$$

$$\text{F1} - \text{score} = 2 \times \frac{Precision \cdot Recall}{Precision + Recall} \tag{11}$$

$$AUC = \int_0^1 TPR(fPR)d(FPR) \tag{12}$$

where TP is defined as the number of accurately identified samples within the positive category, FP as those inaccurately classified within the negative category, TN as accurately identified samples within the negative category, and FN as inaccurately classified samples within the positive category.

4. Experimental Results and Discussion

This section discusses the results concerning the development of the ALBERT-Cascade CNN hybrid model through ablation studies and evaluates the experimental findings related to its validation and testing. The entire workflow encompassing ablation studies and experiments for the proposed model was executed using Python version 3.12.12 and various libraries, including Pandas, TensorFlow, Scikit-learn, Keras, and Transformers, were leveraged throughout this process. Additional details regarding library configurations and source codes are available in the ReadMe.docx file within the Supplementary Materials at the provided link. Given the substantial computational demands, prolonged training durations, and optimization requirements of the models used in the study, Google Colab Pro, which offers graphics processing unit (GPU)/tensor processing unit (TPU) capabilities along with ample RAM, was selected as the coding platform. Experiments were conducted using the Amazon Fashion 2023 dataset under 5-fold cross-validation conditions. The OPTUNA optimization framework was utilized for hyperparameter tuning. For this optimization, twenty trials were conducted under 5-fold cross-validation conditions, with the parameter ranges detailed in Table 1. In these trials, the optimal parameters for the ALBERT-Cascade CNN method were a learning rate of 0.000031, weight decay of 0.000005, dropout rate of 0.2, and batch size of 16, delivering the best performance. Furthermore, the

number of filters was optimized to 128, the hidden dimensions were set to 498, and the filter sizes were configured as (1, 2, 3); default stride value was kept, which is 1, to prevent any skip and down sampling that assist convolution to work step by step, and it is not included in OPTUNA hyperparameter selection.

These configurations significantly enhance the accuracy and overall efficacy of the model. Table 1 presents the hyperparameter value ranges along with the best optimal values determined using the OPTUNA framework.

4.1. Validation and Test Studies

In order to showcase the effectiveness and robustness of the proposed model in terms of performance, a series of experiments was conducted on both the validation and test datasets under 5-fold cross-validation conditions. The results from these experiments were assessed according to various criteria, including accuracy, precision, recall, F1-score, and AUC on an individual-fold and average-fold basis. Table 2 shows the performance results on both individual and average folds under 5-fold cross-validation conditions for the validation dataset of the proposed model.

Table 2. The performance results on both individual and average folds under 5-fold cross-validation conditions for the validation dataset of the proposed model.

Metric	Fold-1	Fold-2	Fold-3	Fold-4	Fold-5	Average of 5-Fold
Accuracy	0.9302	0.9299	0.9328	0.9309	0.9301	0.9308
F1-score	0.9282	0.9288	0.9319	0.9300	0.9289	0.9296
Precision	0.9404	0.9392	0.9425	0.9422	0.9415	0.9412
Recall	0.9164	0.9186	0.9215	0.9181	0.9167	0.9182
AUC	0.9796	0.9795	0.9796	0.9803	0.9795	0.9797
Confusion Matrix						
TP	9030	9135	9190	9177	9136	45,668
TN	9564	9454	9456	9431	9456	47,361
FP	575	591	561	563	568	2855
FN	824	810	783	819	830	4066
FNR	0.0836	0.0814	0.0785	0.0819	0.0833	0.0818
FPR	0.0567	0.0588	0.0560	0.0563	0.0567	0.0569

Based on the results of individual folds, the proposed model achieved an accuracy ranging from 0.9299 to 0.9328, an F1-score from 0.9282 to 0.9319, a precision from 0.9392 to 0.9425, a recall from 0.9164 to 0.9215, and AUC scores from 0.9795 to 0.9803. The highest performance in the other metrics, except for the AUC metric, occurred in Fold 3. In this fold, the proposed model achieved an accuracy of 0.9328, an F1-score of 0.9319, a precision of 0.9425, and a recall of 0.9215. Regarding AUC performance, the highest score was obtained in Fold 4 (0.9803). The lowest performance occurred in different folds with varying metric scores. In this context, while a 0.9299 accuracy, 0.9392 precision, and 0.9795 AUC were realized in Fold 2 (the same AUC score was also obtained in Fold 5), a 0.9282 F1-score and 0.9164 recall score were obtained in Fold 1. From the average fold performances in the table, the proposed model demonstrated a performance proportional to its balanced individual fold performances on an average fold basis. In this respect, it demonstrated successful performance in Amazon reviews with an accuracy of 0.9308, F-score of 0.9296, precision of 0.9412, recall of 0.9182, and AUC of 0.9797 on the validation dataset. The proposed model's recall metric scores remain relatively lower on both individual and average fold bases when compared with other metric scores. A decreased score in this metric, representing the model's ability to capture positive samples, increases the false-negative rate (FNR). As can

be seen from the FNR and false positive rate (FPR) calculated on an individual fold and average fold basis, as presented in the confusion matrix parameters in Table 2, the FNR scores are higher than the FPR scores. These scores indicate that the model can capture negative samples more successfully than positive ones. The model achieved an FNR of 0.0818 and an FPR of 0.0569 on average, resulting in a negative sample detection success rate that was 2.49% higher than that of positive samples.

Table 3 presents the performance results on both individual and average folds under 5-fold cross-validation conditions for the test dataset of the proposed model. The individual and average fold performances on the test dataset, as shown in Table 3, were realized as scores very close to the performances in the validation dataset presented in Table 2.

Table 3. The performance results on both individual and average folds under 5-fold cross-validation conditions for the test dataset of the proposed model.

Metric	Fold-1	Fold-2	Fold-3	Fold-4	Fold-5	Average of 5-Fold
Accuracy	0.9300	0.9317	0.9311	0.9314	0.9325	0.9313
F1-score	0.9294	0.9312	0.9305	0.9308	0.9308	0.9305
Precision	0.9404	0.9430	0.9406	0.9420	0.9412	0.9414
Recall	0.9187	0.9197	0.9205	0.9199	0.9205	0.9199
AUC	0.9787	0.9797	0.9802	0.9810	0.9805	0.9800
Confusion Matrix						
TP	9214	9241	9211	9221	9067	45,954
TN	9377	9383	9402	9398	9573	47,133
FP	584	559	582	568	566	2859
FN	815	807	795	803	783	4003
FNR	0.0813	0.0803	0.0795	0.0801	0.0795	0.0801
FPR	0.0586	0.0562	0.0583	0.0570	0.0558	0.0572

When the model's results for the test dataset were evaluated on an individual fold basis, it was seen that the highest performance was achieved in different folds for different metrics. The best accuracy and recall performances were achieved in Fold 5 with scores of 0.9325 and 0.9205 (the same score was also obtained in Fold 3). For the F1-score and precision metrics, this performance was achieved in Fold 2, with scores of 0.9312 and 0.9430, respectively. The highest performance in terms of the AUC metric was achieved in Fold 4, with a score of 0.9810. The model had the lowest performance in all metrics for Fold 1. In this fold, the accuracy, F1-score, precision, recall, and AUC scores were 0.9300, 0.9294, 0.9404, 0.9187, and 0.9787, respectively. The average fold performance of the model was achieved with an accuracy of 0.9313, F1-score of 0.9305, precision of 0.9414, recall of 0.9199, and AUC of 0.9800. As in the validation dataset, the recall metric in the test dataset also lagged slightly behind the other metrics regarding the scores on both the individual and average folds. This decrease in the model's positive sample prediction can also be understood from the FNR and FPR performances calculated based on the confusion matrix parameters in the table. The FNR scores obtained in the range of 0.0795–0.0813 and the FPR scores obtained in the range of 0.0558–0.0586 indicate that the model's prediction of negative samples is more successful than its prediction of positive samples. A similar situation was also observed on an average fold basis, with an FNR of 0.0801 and an FPR of 0.0572. The model demonstrated a performance that was 2.288% higher in predicting negative samples than in predicting positive samples on an average fold basis.

Experimental results show that the proposed model captures negative examples better than positive examples. This is not due to the imbalance of the dataset. The dataset used in the study is largely balanced, containing 49,992 negative and 49,951 positive data points.

Therefore, the observed performance difference is not due to imbalance. This is due to linguistic discrimination. Negative comments generally contain more specific vocabulary and expressions of dissatisfaction. In contrast, positive comments sometimes contain more general expressions or implicit criticisms. Therefore, it is more difficult for the model to capture positive examples than negative ones. In conclusion, the difference in performance is due to the stronger expression of the negative class, not the imbalance of the proposed model dataset.

4.2. Analysis of Error Cases

In this study, to minimize information loss in the dataset and to better classify the entire dataset, comments with scores of 1–2 were labeled as 'Negative', and comments with scores of 3–4-5 were labeled as 'Positive'. This allowed us to separate definite dissatisfaction from other situations using this defined threshold value and achieve a higher accuracy rate. However, including neutral comments in the 'Positive' category may create label noise within the 'Positive' category. For qualitative analysis, the dataset was manually examined. The examination revealed typical linguistic difficulties that detract from the model's performance, some examples of which are shown in Table 4.

Table 4. The examples shown illustrate data that may be interpreted in a mixed light in the inheritance records used.

Sample Text From Dataset	Label	Model Prediction	Why Does the Model Produce Incorrect Results? (Analysis and Defense)
1. cute but run small sadly this dress be way too small for give to a friend who wear a medium it fit she perfectly	1	0	Mixed Sentiment: The model recognizes the words "sadly" and "too small" as negative, or 0. However, when the phrase "fit she perfectly" appears in the sentence, it changes to positive, or 1. It is difficult for the model to detect this contrast.
2. have to return it i like the idea and want to love the dress but it be too tight around the next to be comfortable in it have to return	1	0	Label Noise: The text contains clearly negative phrases like "return," "too tight," and "want to love… but." Therefore, the model predicts this as "0," but the dataset labels it as "1." This indicates a clear labeling error.
3. do not fit true to size the shirt itself be beautiful i order a x because i know they run small but when it come today it be definetly not true to size it fit like a xl i have such high hope	0	1	Misleading Positive Words: Although the label in the dataset is set to Negative (0), the model may likely register a Positive (1) due to the presence of very strong positive expressions in the text, such as "beautiful" and "high hope."

As shown in Table 4, comments containing 'Mixed Feelings' and 'Irony' particularly challenge the model's decision-making mechanism. In such cases, the model may focus on the dominant word, such as "good," leading to incorrect decisions. Furthermore, as shown in Table 4, it is understood that internal inconsistencies in the dataset affect the model's performance. In Example 2, Label Noise occurs because there is a contradiction between the Negative expressions in the text and the Positive expression in the label. This leads to incorrect learning and problems in the model's generalization ability. From another perspective, Examples 1 and 3 contain Complex Structures and Conflicting Emotions. Here, there are contrasting emotions in the text, such as "cute" but "too small." In situations where the model needs to provide a single output, it may not be able to grasp the clear meaning of the text due to such contradictory expressions. In addition, it may make incorrect predictions due to strong positive words. Therefore, due to the nature of the

data, it is difficult to reduce these texts to a single label in such cases. Therefore, the model architecture cannot be considered inadequate due to such datasets.

4.3. Literature Comparison

The current prevalence of e-commerce platforms has rendered product reviews essential. As one of the largest e-commerce platforms, Amazon contains a vast array of product reviews. Recently, many studies have focused on the automated generation of Amazon product reviews and the development of sentiment analysis models. These studies mainly center around straightforward DL models involving CNN, RNN, and their derivatives (LSTM, BiLSTM, and GRU), as well as transformers such as BERT and LeBERT, and hybrid models that combine transformer architectures with simple DL structures. Table 5 illustrates a comparison between the proposed model and those studies listed were selected for their use of similar Hybrid Architectures on Amazon Review datasets. Since training sub-categories (e.g., Electronics vs. Fashion) may vary, these results are presented to demonstrate the proposed model's competitive positioning among comparable architectural approaches rather than as a direct benchmark under identical data distributions.

Table 5. Comparison of the proposed model with those in similar studies using the different Amazon dataset and hybrid models.

Ref. No	Models	Dataset Length	Class Type	Dataset	Results Accuracy(%)
[24]	BERT-TF-IDF	100,000	Binary	Amazon product reviews	88.00
[25]	BERT base	75,000	Multi classification	Amazon fine food reviews	79.82
[26]	BERT-CNN	-	Binary classification	Amazon product reviews	92.10
[27]	LR, BI-LSTM BERT base	400,000	Triple	Amazon consumer reviews	86.10 87.10 89.00
[28]	CNN, RNN, LSTM, GRU	Data1:20,742 Data2:66,666 Data3:49,870	Binary	Amazon product reviews	85.00 71.00 70.00
[9]	LeBERT-CNN	70,000	Binary	Amazon product reviews	82.40
Proposed model	(average of 5-fold)	99,949	Binary	Amazon fashion reviews (2023)	93.13

Priya Kamath et al. [24] tested the effectiveness of these approaches on various ML models by applying various combinations of feature engineering methods, such as Word2Vec, bag of words (BoW), N-grams, term frequency-inverse document frequency (TF-IDF), and BERT to Amazon product review data. They achieved the highest performance for the Word2Vec model utilizing a skip-gram architecture with a window size set at 10, along with a linear Support Vector Classifier (SVC), resulting in an impressive accuracy rate of 87.00%. Furthermore, they developed a hybrid model by integrating TF-IDF with BERT, which yielded an accuracy rate of 88%, demonstrating promising outcomes for analyzing Amazon product reviews. Zhao and Sun [25] developed an NLP model based on the basic BERT algorithm that obtained a general estimation score through online text reviews. After applying several pre-processes, including removing punctuation marks and special characters, and cleaning processes on Amazon fine food reviews, they experimented with the BERT model. The results of these experiments showed that the model could successfully perform text reviews with an accuracy of 79.82% for food businesses. In a study that yielded an accuracy rate close to that of the proposed model, Wang, Li, and Wang [26] proposed a new approach that combined a pre-trained BERT model with a CNN for text classification. The results obtained from experiments conducted on the AG News and Amazon product reviews datasets demonstrated that their model outperformed traditional methods in terms of accuracy and F1-score metrics. Using the new approach, they achieved an accuracy score of 92.10% on the Amazon product review dataset. Ali et al. [27] conducted a comparative

study employing various BoW and TF-IDF techniques across ML, DL, and transformer-based methods on Amazon customer reviews. Their experimental findings indicated that the BERT model outperformed existing ML and DL approaches, with an accuracy score of 89.00%. Ahmed [28] compared the RNN, LSTM, GRU, and CNN DL methods for classifying Amazon product reviews. In experiments utilizing three distinct Amazon datasets, he achieved the highest performance with an accuracy rate of 85.00% using an RNN on the first dataset. However, this method yielded accuracy rates of only 71.00% and 70.00% on the second and third datasets, respectively, reflecting a considerably lower performance than that of the other models presented in the table. Mutinda, Mwangi, and Okeyo [9] created a model named LeBERT that utilizes N-Gram, BERT word embeddings, and CNN to conduct sentiment analysis on social media reviews. They confirmed the efficacy of the LeBERT model by comparing it with the Word2Vec, GloVe, and BERT word embeddings. Achieving an accuracy rate of 82.40% on Amazon product reviews places this model as the lowest-performing model among binary classification studies, following the RNN scores obtained from two Amazon datasets.

Compared with the proposed model, the findings in this table indicate that our approach achieves a higher accuracy score than all other sentiment analysis models that use similar datasets. Under 5-fold cross-validation, the proposed model demonstrated a performance that was 1.03% better than the pre-trained BERT-CNN model, producing the closest accuracy rate.

4.4. Ablation Studies

This subsection discusses and evaluates the results obtained from the experiments to determine the hybrid ALBERT-based model to be used as an effective NLP approach for product-review classification. Here, the experiments were conducted in two stages. In the first phase, the results obtained by running different BERT models under the same conditions were discussed, and the most suitable BERT model was selected. In the second phase, hybrid models consisting of a combination of different deep learning models with the selected ALBERT model were analyzed, and the most suitable hybrid NLP model for the Amazon dataset was determined.

Phase 1: The BERT, ALBERT, RoBERTa, LeBERT and DistilBERT transformer models were retrained under the same conditions on the Amazon Fashion dataset used in the study, and the comparative results of the relevant models are presented in Table 6. The experiments here were carried out by taking a sample (3000) from the dataset used.

Table 6. Comparative results of some transformer models under the same conditions.

Model Type	Params(M)	Model Size in RAM (MB)	Training Time (s/epoch(M))	Peak VRAM (MB)	Accuracy Score	Inference Time (Second)
ALBERT	21	81.89	18.939587	550.55	0.736667	2.31 s
BERT	119	455.56	17.069057	754.36	0.840000	2.06 s
RoBERT	140	536.00	17.489917	835.43	0.813333	1.96 s
LeBERT	199	455.56	15.688810	754.36	0.853333	1.89 s
DistilBERT	76	291.07	9.180782	589.87	0.866667	1.10 s

As shown in Table 6, the ALBERT model was chosen due to parameter efficiency and memory constraints. Because, as shown in Table 6, ALBERT uses only 21 million parameters. Therefore, compared to the BERT model, it uses approximately 82% fewer parameters. Also, looking at Table 6, the RAM size of the model is 81.89 MB for ALBERT. However, for BERT, this value is 455.56 MB. Compared to other BERT models, ALBERT is more successful in situations where there may be significant storage limitations. In addition, the VRAM usage of the ALBERT model is 550.55 MB. This shows that it is better than other BERT models for training on GPUs with smaller memory capacities. This makes the

model more repeatable than other BERT models. In addition, Table 6 shows that ALBERT has a higher inference time and lower accuracy compared to other BERT models. This is considered normal due to ALBERT's weight-sharing mechanism. However, the aim of this study is to achieve high accuracy rates with a lightweight architecture that minimizes storage and memory load. Therefore, ALBERT was evaluated as the most suitable option. In addition, the shortcomings of ALBERT are addressed thanks to the ALBERT + Cascade 1DCNN architecture proposed in this study. Thus, the lightweight advantage provided by ALBERT is preserved with the proposed method, and in addition, a high accuracy rate is obtained in terms of classification performance.

Phase 2: We aimed to determine a robust model for product reviews by analyzing which components contribute to the performance increase in hybrid models obtained by combining ALBERT's strong embedding capability with CNN, LSTM, BiLSTM, and their layers, along with attention mechanisms. The experiments were conducted in four steps. In the first step, the pre-processed data were divided into 80% training data, 10% validation data, and 10% test data. In the second step, the models were trained and optimized using the OPTUNA framework, which consisted of five experiments. In the third step, the optimized models were applied to the validation and test datasets, and the performance results were obtained based on accuracy, F1-score, precision, recall, and AUC metrics. In the final step, these results were evaluated, and a decision was made on the most suitable model for product reviews. Tables 7 and 8 show the performance results obtained by applying 14 hybrid models consisting of ALBERT with CNN, LSTM, BiLSTM methods, and combinations of pooling and attention layers to the validation and test datasets, respectively.

Table 7. The performance results obtained from applying 14 hybrid models to the validation dataset.

Models	Accuracy	F1-Score	Precision	Recall	AUC
ALBERT-base	0.9164	0.9180	0.8949	0.9424	0.9780
ALBERT-MaxPooling	0.9090	0.9031	0.9586	0.8536	0.9730
ALBERT-Cascade CNN	0.9318	0.9301	0.9468	0.9140	0.9804
ALBERT-LSTM	0.9299	0.9281	0.9457	0.9112	0.9778,
ALBERT-BILSTM	0.9276	0.9277	0.9205	0.9350	0.9795
ALBERT-CNN-LSTM	0.9303	0.9283	0.9493	0.9082	0.9786
ALBERT-CNN-BILSTM	0.9312	0.9306	0.9328	0.9283	0.9787
ALBERT-LSTM-CNN	0.9322	0.9304	0.9491	0.9124	0.9790
ALBERT-BLSTM-CNN	0.9315	0.9301	0.9433	0.9173	0.9786
ALBERT-MHAttention	0.9222	0.9218	0.9202	0.9235	0.9768
ALBERT-LSTM-MHAttention-CNN	0.9269	0.9257	0.9353	0.9162	0.9764
ALBERT-BILSTM-MHAttention-CNN	0.9297	0.9289	0.9339	0.9239	0.9783
ALBERT-CNN-LSTM-MHAttention	0.9287	0.9280	0.9311	0.9249	0.9792
ALBERT-CNN-BILSTM-MHAttention	0.9246	0.9240	0.9251	0.9229	0.9770

The ALBERT-base model demonstrated successful performance as the main component of the hybrid models, with accuracy scores of 0.9164 for the validation dataset and 0.9148 for the test dataset. The model's precision scores (0.8949 and 0.9000) on the validation and test datasets were significantly lower than recall scores (0.9424 and 0.9356), indicating that while it successfully detected positive samples, it occasionally produced false positives when detecting negative samples. In this context, it cannot be said that the model shows balanced performance between classes during the prediction process. Adding the MaxPooling layer to the ALBERT-base model decreased the performance scores for all metrics, except for precision for both datasets. In terms of the precision metric, there was an increase of over 6% for the validation dataset and over 5% for the test dataset, respectively. Conversely, in the recall metric, there was a decrease of nearly 9% for both datasets. In

other words, the imbalance in the class-level prediction performance compared to the ALBERT-base model further increased. On the other hand, integrating the MHAttention mechanism into the ALBERT-base model resulted in a slight performance improvement in terms of accuracy, F1-score, and precision metrics on the validation dataset, and accuracy and precision metrics on the test dataset. In fact, the most significant contribution of this mechanism to the base model lies in reducing the scoring difference between the precision and recall metrics. This contribution also reduced the class imbalance in the predictions among classes in the ALBERT-base model. The ALBERT-MHAttention model reduced the difference between the precision and recall metrics from 0.0475 to 0.0033 on the validation dataset and from 0.0356 to 0.0093 on the test dataset.

Table 8. The performance results obtained from applying 14 hybrid models to the test dataset.

Models	Accuracy	F1-Score	Precision	Recall	AUC
ALBERT-base	0.9148	0.9175	0.9000	0.9356	0.9762
ALBERT-MaxPooling	0.9031	0.8985	0.9561	0.8475	0.9709
ALBERT-Cascade CNN	0.9280	0.9272	0.9491	0.9064	0.9780
ALBERT-LSTM	0.9235	0.9228	0.9438	0.9026	0.9759
ALBERT-BILSTM	0.923	0.9243	0.9203	0.9283	0.9775
ALBERT-CNN-LSTM	0.9264	0.9256	0.9482	0.9040	0.9764
ALBERT-CNN-BILSTM	0.9225	0.9227	0.9315	0.9141	0.9764
ALBERT-LSTM-CNN	0.9275	0.9267	0.9494	0.9050	0.9780
ALBERT-BLSTM-CNN	0.9261	0.9256	0.9432	0.9087	0.9762
ALBERT-MHAttention	0.9163	0.9169	0.9216	0.9123	0.9750
ALBERT-LSTM-MHAttention-CNN	0.9208	0.9205	0.9357	0.9058	0.9741
ALBERT-BILSTM-MHAttention-CNN	0.9252	0.9251	0.9375	0.9131	0.9792
ALBERT-CNN-LSTM-MHAttention	0.9227	0.9232	0.9285	0.9180	0.9757
ALBERT-CNN-BILSTM-MHAttention	0.9202	0.9208	0.9255	0.9160	0.9740

Hybrid models consisting of dual and triple combinations of Cascade CNN, LSTM, and BiLSTM models with ALBERT demonstrated performance above the ALBERT-base model. The ALBERT-Cascade CNN hybrid model achieved the best performance among the dual hybrid models for all metrics, except for recall, on the validation and test datasets. Moreover, with its 0.9280 accuracy and 0.9292 F1-score on the test dataset, it outperformed even the triple hybrid models in these metrics. It also achieved the highest performance regarding the AUC metric on the validation dataset, with a score of 0.9804. The model demonstrated a score superiority of 1.54%, 1.21%, 5.19%, and 0.24% over the ALBERT-base model in terms of accuracy, F1 score, precision, and AUC metrics on the model validation dataset, while achieving a similar superiority on the test dataset with rates of 1.32%, 0.97%, 4.91%, and 0.18%, respectively, for these metrics. The ALBERT-LSTM hybrid model was the best-performing approach among the dual hybrid models on the validation dataset, following the ALBERT-Cascade CNN model. The relevant model achieved improvements of 1.35%, 1.01%, and 5.08% in terms of accuracy, F1 score, and sensitivity metrics, respectively, compared to the ALBERT-base model on this dataset. It also obtained a performance score similar to that of the validation dataset in terms of accuracy and precision metrics on the test dataset. On that test dataset, it provided a performance score increase of 0.87% in accuracy, 0.53% in F1 score, and 4.38% in precision, compared to the ALBERT-base model. Although the ALBERT-BiLSTM hybrid model lagged behind the ALBERT-LSTM hybrid model's accuracy scores on the validation and test datasets, the score differences remained relatively low. The ALBERT-BiLSTM model fell slightly behind the ALBERT-LSTM model in the F1 score metric on the validation dataset, with a minimal score difference (0.0004), while achieving a better result on the test dataset, with a slight score difference (0.0015).

The model outperformed ALBERT-LSTM in terms of recall and AUC metrics with scores of 0.9350 and 0.9795 on the validation dataset and 0.9283 and 0.9775 on the test dataset. It demonstrated a score superiority of 1.12%, 0.97%, 2.56%, and 0.15% on the validation dataset, and 0.82%, 0.68%, 2.03%, and 0.13% on the test dataset in terms of accuracy, F1-score, precision, and AUC metrics compared to the ALBERT-base model, respectively.

The performance of the triple hybrid models on the validation dataset followed a mixed trend. On this dataset, ALBERT-LSTM-CNN with scores of 0.9322 and 0.9790 in accuracy and AUC metrics, ALBERT-CNN-BiLSTM with scores of 0.9306 and 0.9283 in F1-score and recall metrics, and ALBERT-CNN-LSTM with a score of 0.9493 in precision metric were the models that achieved the best performance scores in terms of the relevant metrics among the triple hybrid models. In addition, ALBERT-LSTM-CNN demonstrated a performance quite close to that of the other triple hybrid models, falling behind with scores of 0.9304 and 0.9491 in F1-score and precision metrics, respectively. Therefore, when assessed generally, the best performance among the triple hybrid models on the validation dataset was achieved by the ALBERT-LSTM-CNN model. On the test dataset, the ALBERT-LSTM-CNN model produced the highest performance score among the triple hybrid models in all metrics except the recall metric. The model achieved this with an accuracy of 0.9275, F1-score of 0.9267, precision of 0.94494, and AUC score of 0.9780. The relevant model obtained higher performance scores than the ALBERT-base model, with improvements in accuracy, F1-score, precision, and recall metrics of 1.58%, 1.24%, 5.42%, and 0.10% on the validation dataset, and 1.27%, 0.92%, 4.94%, and 0.18% on the test dataset, respectively. Adding the MHAttention mechanism to the triple hybrid models resulted in a slight performance drop, in terms of accuracy, F1-score, and precision metrics, except for the ALBERT-CNN-LSTM model, which also showed a decline in the AUC metric.

The overall results of the ablation studies show that ALBERT can be successfully used as the base model for creating hybrid models. The dual hybrid models, created by adding CNN, LSTM, and BiLSTM layers to ALBERT, contributed to an accuracy rate in the range of 1.12–1.54% on the validation dataset and 0.82–1.32% on the test dataset, while this contribution in the triple hybrid models occurred with an accuracy rate in the range of 1.39–1.58% on the validation dataset and 0.77–1.27% on the test dataset. Among the dual models, the ALBERT-Cascade CNN hybrid model, with the best overall performance on both datasets, showed a higher accuracy rate on the validation dataset than the ALBERT-LSTM-CNN hybrid model, which particularly reflected the best test performance among the triple hybrid models (it also achieved performances close to this hybrid model in other performance metrics except recall). On the test dataset, it demonstrated better performance in all metrics except recall. Adding a MaxPooling layer to the ALBERT-base model caused a significant performance drop in all metrics except the precision metric. While adding the MHAttention layer to the ALBERT-base model significantly improved accuracy, F1-score, and precision metrics, especially on the validation dataset, it was observed that integrating this layer into the triple hybrid models had a minimal impact on performance and even slightly decreased the models' overall performance. Therefore, the ALBERT-Cascade CNN hybrid model stood out among all models due to its high performance and less complex structure compared to the models with similar performance.

5. Conclusions and Future Work

Today, as a result of the widespread use of online platforms, it has become an urgent need for users to quickly and accurately interpret the large volumes of text shared there. In the process carried out within the scope of sentiment analysis, traditional models prove inadequate at capturing the complex structures and shifts in sentiment present in texts. Therefore, most recent studies have turned to DL–based methods that provide effective

contextual richness beyond traditional models for this process. This study proposes a new DL–based hybrid model in which the contextual meaning, which is capable of overcoming the aforementioned shortcomings in sentiment analysis, is evaluated using local and global features. To determine the proposed model, ablation studies were conducted on hybrid models that combined ALBERT-based language representations with CNN, LSTM, BiLSTM layers, and the Multi-Head Attention mechanism. Fourteen hybrid models constructed in these experiments were applied to the Amazon Fashion 2023 dataset, which was split into 80% training, 10% validation, and 10% test. As a result of the ablation experiments, performance measurements were carried out by evaluating the results obtained on the validation and test datasets using accuracy, F1-score, precision, recall, and AUC metrics. Considering both the performance results and model complexities, it was determined that the ALBERT-Cascade CNN hybrid model is the most suitable model to address the requirements in sentiment analysis studies. OPTUNA, an effective optimization framework for parameter tuning, was used for all models.

The ALBERT block in the first layer of the proposed model produces contextualized word embeddings from the tokenized input text. The tokenized form of the input text and attention masks, used to distinguish between real and padding masks, were applied as additional inputs. This usage allows the model to focus on the relevant text effectively. The last_hidden_state of ALBERT was used to extract contextual representations, which were then transferred to the first 1-D CNN block to extract local features. Subsequently, these extracted features were applied to the second 1-D CNN block equipped with MaxPooling to obtain information summarizing the entire text (global features). This structure allows the proposed model to better capture multilayered language structures based on word patterns and semantic blocks.

The performance of the proposed model in the sentiment classification process was tested with experiments conducted on the Amazon Fashion 2023 dataset under 5-fold validation conditions. The experiments on the validation and test datasets showed that the model could successfully perform sentiment analysis. The experimental results show that the model achieved average performance scores of approximately 0.93 in accuracy and F1-score metrics, above 0.94 in precision metric, and approximately 0.98 in AUC metric on both datasets. However, model's performance in the recall metric fell behind that in the precision metric, with an average score difference of over 2%. Consequently, the FNR scores followed a course above the FPR scores. This indicates that the model is more successful in capturing negative samples than positive samples. In addition, the comparison results of the proposed model with those of studies using a similar dataset in the literature show that it has a higher accuracy than existing models. These results prove that the ALBERT-Cascade hybrid model can be successfully used as a sentiment analysis model for text reviews.

Although the proposed model has a less complex structure than the ALBERT-based triple hybrid models used in this study, it is still structurally complex. It incurs high computational costs and memory consumption due to the generation of high-dimensional feature vectors. When viewed in this sense, the model presents certain limitations in terms of its structural basis and real-time applicability. In small datasets, performance issues related to overfitting and generalization may arise due to problems emerging from the training process. The high computational cost leads to long training periods and real-time applicability issues with large datasets. Future studies could reduce model complexity by integrating lighter layers instead of those performing local and global feature extraction. By incorporating a layer from lightweight BERT models instead of the ALBERT layer, it may be possible to minimize the aforementioned limitations. Furthermore, in this study, three-score ratings were included in the positive category. This eliminates the possibility of semantic ambiguity when using the entire dataset, as three-score ratings often contain

mixed emotions. To avoid such ambiguities in future studies, neutral ratings could be removed from the dataset.

Supplementary Materials: The following supporting information can be downloaded at: https://www.mdpi.com/article/10.3390/app16010025/s1.

Author Contributions: Conceptualization, I.M.A., İ.A. and S.T.; methodology, I.M.A., İ.A., S.T., N.B. and İ.A.D.; software, I.M.A.; validation, İ.A., S.T., N.B. and İ.A.D.; formal analysis, S.T., N.B. and İ.A.D.; writing—original draft preparation, I.M.A. and İ.A.; writing—review and editing, I.M.A., İ.A., S.T., N.B. and İ.A.D.; visualization, I.M.A., İ.A. and S.T.; supervision, İ.A., S.T., N.B. and İ.A.D. All authors have read and agreed to the published version of the manuscript.

Funding: This study was not supported by any external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The Amazon Fashion 2023 dataset was used in this study. A raw dataset can be accessed using this source: https://amazon-reviews-2023.github.io/ (accessed on 17 February 2025).

Acknowledgments: The authors would like to thank Gazi University Academic Writing Application and Research Center for proofreading the article.

Conflicts of Interest: The authors declare no conflicts of interest.

Abbreviations

The following abbreviations are used in this manuscript:

NLP	Natural Language Processing
DL	Deep Learning
ALBERT	A Lite BERT
CNN	Convolutional Neural Network
RNN	Recurrent Neural Network
LSTM	Long Short-Term Memory
BiLSTM	Bidirectional Long Short-Term Memory
AI	Artificial Intelligence
MLM	Masked Language Modeling
TPE	Tree-structured Parzen Estimator
DCNN	Dilated Convolutional Neural Network
LeBERT	Lexicon-Enhanced Bert Embedding
GRU	Gated Recurrent Unit
BiGRU	Bi-directional Gateway Recurrent Unit
ABCNN	Attention-Based Convolutional Neural Network
HAN	Hierarchical Attention Network
NSP	Next Sentence Prediction
SOP	Sentence Order Prediction
MLM	Masked Language Model
FC	Fully Connected
ReLU	Rectified Linear Unit
TP	True Positive
FP	False Positive
TN	True Negative
FN	False Negative
AUC	Area Under Curve

GPU	Graphics Processing Unit
TPU	Tensor Processing Unit
FNR	False-Negative Rate
FPR	False Positive Rate
ML	Machine Learning
BoW	Bag of Words
TF-IDF	Term Frequency-Inverse Document Frequency
SVC	Support Vector Classifier

References

1. Backlinko Team. 15 Online Review Statistics, 2025. Available online: https://backlinko.com/online-review-stats (accessed on 11 July 2025).
2. Statista. Statista Research Department and Content Philosophy. Available online: https://www.statista.com/aboutus/our-research-commitment (accessed on 11 July 2025).
3. FinancesOnline. 62 Customer Reviews Statistics You Must Learn: 2024 Market Share Analysis & Data. Available online: https://financesonline.com/customer-reviews-statistics (accessed on 13 July 2025).
4. Wang, J.; Huang, J.X.; Tu, X.; Wang, J.; Huang, A.J.; Laskar, M.T.R.; Bhuiyan, A. Utilizing BERT for Information Retrieval: Survey, Applications, Resources, and Challenges. *ACM Comput. Surv.* **2024**, *56*, 33. [CrossRef]
5. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, Ł.; Polosukhin, I. Attention Is All You Need. *Adv. Neural Inf. Process. Syst.* **2017**, *30*.
6. Bao, T.; Ren, N.; Luo, R.; Wang, B.; Shen, G.; Guo, T. A BERT-Based Hybrid Short Text Classification Model Incorporating CNN and Attention-Based BiGRU. *J. Organ. End User Comput.* **2021**, *33*, 1–21. [CrossRef]
7. Tan, K.L.; Lee, C.P.; Anbananthen, K.S.M.; Lim, K.M. RoBERTa-LSTM: A Hybrid Model for Sentiment Analysis with Transformer and Recurrent Neural Network. *IEEE Access* **2022**, *10*, 21517–21525. [CrossRef]
8. Jain, P.K.; Quamer, W.; Saravanan, V.; Pamula, R. Employing BERT-DCNN with Sentic Knowledge Base for Social Media Sentiment Analysis. *J. Ambient Intell. Humaniz. Comput.* **2023**, *14*, 10417–10429. [CrossRef]
9. Mutinda, J.; Mwangi, W.; Okeyo, G. Sentiment Analysis of Text Reviews Using Lexicon-Enhanced BERT Embedding (LeBERT) Model with Convolutional Neural Network. *Appl. Sci.* **2023**, *13*, 1445. [CrossRef]
10. Zhang, B. A BERT-CNN Based Approach on Movie Review Sentiment Analysis. *SHS Web Conf.* **2023**, *163*, 04007. [CrossRef]
11. Deng, L.; Yin, T.; Li, Z.; Ge, Q. Analysis of the Effectiveness of CNN-LSTM Models Incorporating BERT and Attention Mechanisms in Sentiment Analysis of Data Reviews. In *Proceedings of the 2023 4th International Conference on Big Data and Informatization Education (ICBDIE 2023)*; Atlantis Press: Dordrecht, The Netherlands, 2023; pp. 821–829. [CrossRef]
12. Kaur, K.; Kaur, P. Improving BERT Model for Requirements Classification by Bidirectional LSTM-CNN Deep Model. *Comput. Electr. Eng.* **2023**, *108*, 108699. [CrossRef]
13. Silitonga, C.A.A.; Dermawan, M.D.; Adeta, F.; Nadia, N. Comparative Study of BERT-CNN, TRANS-BLSTM, and RoBERTa Models for Sentiment Analysis. In Proceedings of the 8th International Conference on Information Technology, Information Systems and Electrical Engineering (ICITISEE 2024), Yogyakarta, Indonesia, 29–30 August 2024; pp. 358–363. [CrossRef]
14. Gupta, B.; Prakasam, P.; Velmurugan, T. Integrated BERT Embeddings, BiLSTM-BiGRU and 1-D CNN Model for Binary Sentiment Classification Analysis of Movie Reviews. *Multimed. Tools Appl.* **2022**, *81*, 33067–33086. [CrossRef]
15. Xiao, H.; Luo, L. An Automatic Sentiment Analysis Method for Short Texts Based on Transformer-BERT Hybrid Model. *IEEE Access* **2024**, *12*, 93305–93317. [CrossRef]
16. Hou, Y.; Li, J.; He, Z.; Yan, A.; Chen, X.; McAuley, J. Bridging Language and Items for Retrieval and Recommendation. *arXiv* **2024**, arXiv:2403.03952. [CrossRef]
17. Parningotan Manik, L.; Kurniasih, A. On the Role of Text Preprocessing in BERT Embedding-Based DNNs for Classifying Informal Texts. *Int. J. Adv. Comput. Sci. Appl.* **2022**, *13*. [CrossRef]
18. Shukla, D.; Dwivedi, S.K. The Study of the Effect of Preprocessing Techniques for Emotion Detection on Amazon Product Review Dataset. *Soc. Netw. Anal. Min.* **2024**, *14*, 191. [CrossRef]
19. Lan, Z.; Chen, M.; Goodman, S.; Gimpel, K.; Sharma, P.; Soricut, R. ALBERT: A Lite BERT for Self-Supervised Learning of Language Representations. *arXiv* **2020**, arXiv:1909.11942. [CrossRef]
20. Khan, S.H.; Iqbal, R. A Comprehensive Survey on Architectural Advances in Deep CNNs: Challenges, Applications, and Emerging Research Directions. *arXiv* **2025**, arXiv:2503.16546. [CrossRef]
21. Akiba, T.; Sano, S.; Yanase, T.; Ohta, T.; Koyama, M. Optuna: A Next-Generation Hyperparameter Optimization Framework. In Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Anchorage, AK, USA, 4–8 August 2019; pp. 2623–2631. [CrossRef]

22. İstanbullu, C. Parametre Optimizasyonuna Pratik Bir Çözüm: Optuna. Miuul. Available online: https://miuul.com/blog/parametre-optimizasyonuna-pratik-bir-cozum-optuna (accessed on 26 July 2025).
23. Vaibhav, J. Performance Metrics: Confusion Matrix, Precision, Recall, and F1 Score. Towards Data Science. Available online: https://towardsdatascience.com/performance-metrics-confusion-matrix-precision-recall-and-f1-score-a8fe076a2262 (accessed on 16 August 2025).
24. Priya Kamath, B.; Geetha, M.; Acharya, U.D.; Singh, D.; Rao, A.; Rai, S.; Shetty, R. Comprehensive Analysis of Word Embedding Models and Design of Effective Feature Vector for Classification of Amazon Product Reviews. *IEEE Access* **2025**, *13*, 25239–25255. [CrossRef]
25. Zhao, X.; Sun, Y. Amazon Fine Food Reviews with BERT Model. *Procedia Comput. Sci.* **2022**, *208*, 401–406. [CrossRef]
26. Wang, C.; Li, Y.; Wang, Z. A Novel Approach for Text Classification by Combining Pre-Trained BERT Model with CNN Classifier. In Proceedings of the 6th IEEE International Conference on Information Systems and Computer Aided Education (ICISCAE 2023), Dalian, China, 23–25 September 2023; pp. 57–62. [CrossRef]
27. Ali, H.; Hashmi, E.; Yayilgan Yildirim, S.; Shaikh, S. Analyzing Amazon Products Sentiment: A Comparative Study of Machine and Deep Learning, and Transformer-Based Techniques. *Electronics* **2024**, *13*, 1305. [CrossRef]
28. Ahmed, I. Comparative Study of Sentiment Analysis on Amazon Product Reviews Using Recurrent Neural Network (RNN). *Int. J. Adv. Trends Comput. Sci. Eng.* **2022**, *11*, 141–146. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.