DOI: https://doi.org/10.36719/2789-6919/32/197-202

Ziya Mustafayev
Azərbaycan Dövlət Neft və Sənaye Universiteti
magistrant
z.mustafayev070@gmail.com

DATA MİNİNGDƏ SÜNİ NEYRON ŞƏBƏKƏLƏRİNDƏN İSTİFADƏNİN YENİLİKLƏRİ

Xülasə

Süni Neyron Şəbəkələri (Artificial Neural Networks) böyük verilənlər bazasında mürəkkəb nümunələri və əlaqələri öyrənmək qabiliyyətinə görə data miningdə geniş istifadə edilmişdir. Data mining-də ANN-lərin istifadəsində bir sıra yeniliklər olmuşdur. Əvvəlcə təsvirin tanınması üçün nəzərdə tutulmuş konvolutional neyron şəbəkələri (Convolutional Neural Networks) təbii dilin işlənməsi və zaman sıralarının təhlili kimi müxtəlif sahələrdə tətbiqlər tapmışdır. Onlar verilənlərdəki məkanları və müvəqqəti nümunələrin təyinində effektiv rol oynayır.

Dərin öyrənmə arxitekturaları data miningə tətbiq edilən süni neyron şəbəkələri (ANN) sahəsində əhəmiyyətli yeniliyə malik oldular. Bu irəliləyişlər mürəkkəb nümunələri idarə etmək və böyük məlumat dəstlərindən qiymətli fikirlər çıxarmaq üçün daha güclü, səmərəli və dəqiq modellərə gətirib çıxardı. CNN-lər image miningdən istifadə edirlər. Bu arxitekturalar vizual xüsusiyyətlərin ierarxik təsvirlərini avtomatik öyrənmək üçün xüsusi konvolyusiya qatlarından istifadə edir.

Yeniliklərə dərin öyrənmə arxitekturaları (məsələn, ResNet, DenseNet) və müvafiq görüntülərə fokuslanan diqqət mexanizmləri daxildir. Təkrarlanan Neyron Şəbəkələri (Recurrent Neural Networks) və Uzun- Qısamüddətli Yaddaş (Long Short Term Memory) şəbəkələri zaman sıralarının təhlili, təbii dil emalı (Natural Language Processing) və nitqin tanınması kimi tətbiqlər üçün ardıcıl məlumatların idarə edilməsində əsas rol oynamışdır. Yeniliklərə uzunmüddətli asılılıqların modelləşdirilməsini təkmilləşdirmək üçün iki istiqamətli RNN, diqqət mexanizmləri və keçidli təkrarlanan vahidlər (Gates Recurrent Units) daxildir.

Açar sözlər: *Süni Neyron Şəbəkələri, data mining, Təbii Dil Emalı, avtokodlayıcılar, Konvolutional Neyron Şəbəkələri, uzun-qısa müddətli yaddaş, şəkil mining*

Ziya Mustafayev
Azerbaijan State Oil and Industry University
master student
z.mustafayev070@gmail.com

Innovation in the use of artificial neural network in data mining

Abstract

Artificial Neural Networks (ANNs) have been widely used in data mining for their ability to learn complex patterns and relationships within large datasets. There have been several innovations in the use of ANNs in data mining. Originally designed for image recognition, convolutional neural networks (CNNs) have found applications in various domains such as natural language processing and time-series analysis. They are effective in capturing spatial and temporal patterns in data.

Deep learning architectures have witnessed significant innovation in the realm of artificial neural networks (ANNs) applied to data mining. These advancements have led to more powerful, efficient, and accurate models for handling complex patterns and extracting valuable insights from large datasets. CNNs use image mining tasks. These architectures leverage specialized convolutional layers to automatically learn hierarchical representations of visual features.

Innovations include deeper architectures (e.g., ResNet, DenseNet) and attention mechanisms that focus on relevant image regions. Recurrent Neural Networks (RNNs) and Long Short-Term Memory (LSTM) networks have been pivotal in handling sequential data for applications like time-series analysis, natural language processing (NLP), and speech recognition. Innovations include bidirectional RNNs, attention mechanisms, and gated recurrent units (GRUs) to improve the modeling of long-range dependencies.

Keywords: *Artificial Neural Networks, data mining, Natural Language Processing, autoencoders, Convolutional Neural Networks, long-short term memory, image mining*

Introduction

The Transformer architecture, introduced with models like BERT (Bidirectional Encoder Representations from Transformers) and GPT (Generative Pre-trained Transformer), revolutionized NLP and sequence mining. Transformers use self-attention mechanisms to capture contextual information effectively. Pre-training on large datasets has become a common practice, enabling models to transfer knowledge to downstream tasks. Graph Neural Networks have emerged as powerful tools for mining information from graph-structured data, such as social networks and molecular structures. Innovations include graph attention networks (GATs) and graph convolutional networks (GCNs), enabling the learning of node and graph-level representations (Rabunal, Dorado, 2006: 305).

Autoencoders, a type of unsupervised learning architecture, have been applied to learn compact representations of data. Variational autoencoders (VAEs) introduce probabilistic modeling, enabling the generation of new data samples. Autoencoders are also utilized for anomaly detection by reconstructing normal patterns. Capsule networks, proposed as an alternative to traditional pooling layers, aim to capture hierarchical relationships among features more effectively. Capsules represent entities in an image and their spatial relationships, potentially improving generalization and interpretability. Attention mechanisms have become a ubiquitous component in various architectures. They enhance the model's ability to focus on relevant information while processing data, leading to improved performance in tasks such as machine translation, image captioning, and summarization. The concept of transfer learning, especially using pre-trained models on large datasets, has become a standard practice. Models like OpenAI's GPT and Google's BERT are pre-trained on massive corpora and then fine-tuned for specific tasks, achieving state-of-the-art results with less data.

Recurrent Neural Networks (RNNs) and Long Short-Term Memory (LSTM)

Particularly useful for sequential data, RNNs have been employed in tasks like time-series prediction, natural language processing, and speech recognition. Long Short-Term Memory (LSTM) and Gated Recurrent Unit (GRU) architectures have been introduced to address the vanishing gradient problem, enabling the network to learn long-term dependencies (Bakardjieva, 2019: 246).

Recurrent Neural Networks (RNNs) have undergone various innovations to enhance their capabilities in the context of data mining. RNNs are particularly well-suited for handling sequential data and time-series analysis due to their ability to capture dependencies over time. Long Short-Term Memory (LSTM) networks address the vanishing gradient problem in traditional RNNs, allowing them to capture long-range dependencies in sequences more effectively. LSTMs include memory cells and gates to control the flow of information, enabling the network to retain and update information over extended time periods.

Gated Recurrent Units (GRUs) are another variant of RNNs designed to address the vanishing gradient problem. GRUs are computationally less complex than LSTMs, making them more efficient for certain tasks. They include update and reset gates to control the information flow through the network. Bidirectional RNNs process input data in both forward and backward directions, allowing the network to capture dependencies from past and future contexts. This is particularly useful for tasks where context from both directions is important, such as natural language processing. Attention mechanisms enable the network to focus on specific parts of the input sequence when making predictions. This innovation enhances the model's ability to selectively attend to relevant information, improving performance in tasks like machine translation, summarization, and sequence-to-sequence learning. RNNs have been extended to handle sequence-to-sequence tasks, where the input and output are both sequences of varying lengths. This is widely used in machine translation, speech recognition, and other applications where the goal is to transform one sequence into another. RNNs, especially with attention mechanisms, have been pivotal in the development of Neural Machine Translation systems. These systems have significantly improved the quality of machine translation by learning contextually rich representations of source and target languages (Daniel, Barbara, 2020: 286).

Temporal Convolutional Networks (TCNs) are an alternative to traditional RNNs for sequence modeling. They use causal convolutions to capture temporal dependencies and have demonstrated success in tasks like time-series forecasting. Inspired by the success of residual connections in convolutional neural networks (CNNs), residual connections have been introduced in RNNs. These connections facilitate the training of deeper networks by mitigating the vanishing gradient problem. RNNs have been used for meta-learning tasks, where the model learns to quickly adapt to new tasks with limited examples. This innovation has potential applications in scenarios where data is scarce or where models need to adapt to changing environments.

Types of autoencoders and Generative Adversarial Networks (GANs)

These unsupervised learning models are used for feature learning and dimensionality reduction. Variational Autoencoders (VAEs) and Generative Adversarial Networks (GANs) are examples of advanced autoencoder architectures. Autoencoders, a type of artificial neural network, have seen several innovations in their applications within the field of data mining. Autoencoders are particularly useful for unsupervised learning and dimensionality reduction (Arens, Weigold. 2020: 683).

- Variational Autoencoders (VAEs) introduce a probabilistic framework to autoencoders, making them capable of generating new data samples. This innovation allows VAEs to learn not only the most compact representation of input data but also the underlying probability distribution. VAEs have found applications in generative tasks and anomaly detection.

- Denoising autoencoders are trained to reconstruct clean data from corrupted input. By adding noise to the input data during training, these autoencoders learn to capture essential features and discard noise. Denoising autoencoders are valuable for feature learning and handling noisy datasets.

- Sparse autoencoders introduce sparsity constraints in the learned representations, encouraging the model to activate only a subset of neurons for a given input. This innovation helps in feature selection and can lead to more interpretable representations.

- Contractive autoencoders are designed to learn representations that are robust to small input variations. The model is penalized for large changes in the learned representation with respect to small changes in the input. This innovation is beneficial for capturing essential features while ignoring irrelevant details.

- Adversarial Autoencoders (AAEs) combine the concepts of autoencoders with adversarial training. By introducing an adversarial network, AAEs can learn more meaningful and continuous latent representations. AAEs are especially effective for generating high-quality, realistic data samples. Autoencoders have been extended to semi-supervised learning scenarios where the model is trained on both labeled and unlabeled data. This innovation allows autoencoders to leverage limited labeled data for better representation learning. Inspired by capsule networks, capsule autoencoders aim to represent hierarchical structures in data more effectively.

- Capsule autoencoders can capture relationships among features in a way that traditional autoencoders might struggle with, offering improved expressiveness. Autoencoders have been used in joint learning settings, where they are combined with other neural network architectures to perform tasks such as clustering, classification, or regression simultaneously. This innovation allows for the extraction of meaningful features while accomplishing specific tasks. In the context of computer vision, spatial autoencoders focus on learning spatial hierarchies in data. These models can be particularly effective for tasks such as image segmentation and object localization by capturing spatial dependencies. Generative Adversarial Networks (GANs) and Autoencoders and autoencoders have been combined to create networks that benefit from both generative capabilities and efficient feature learning. This combination has led to the development of models like Adversarially Regularized Autoencoders (ARAEs) and improved capabilities in data generation tasks.

Explainable AI (XAI)

As neural networks become more complex, there is a growing emphasis on interpretability and explainability. Innovations in XAI aim to provide insights into how the neural network arrives at its decisions, making it more transparent and understandable for users. Explainable AI (XAI) focuses on developing models and techniques that provide clear, understandable, and interpretable explanations for the decisions made by artificial intelligence systems. In the context of artificial neural networks (ANNs) in data mining, innovations in XAI are crucial to enhancing transparency and building trust in AI systems. Methods have been developed to assess the importance of individual features in the decision-making process of neural networks. Techniques such as permutation importance and sensitivity analysis help identify the features that contribute the most to model predictions. Layer-wise Relevance Propagation (LRP) is a method that allocates the prediction's relevance back to the input features (Chaffey, Ellis, 2019: 417).

It decomposes the final prediction and assigns relevance scores to different neurons and features, offering insights into which parts of the input contributed the most to the output. Activation maximization involves altering input features to maximize the activation of specific neurons or target classes. This technique helps reveal what patterns or features in the input data lead to the neural network making certain decisions.

LIME (Local Interpretable Model-agnostic Explanations) is a model-agnostic approach that approximates the decision boundaries of complex models, including neural networks, using locally interpretable models. It generates perturbed samples around a data point and observes the changes in predictions, providing insights into the model's behavior (Brown, 2018: 413).

SHAP (SHapley Additive exPlanations) SHAP values, inspired by cooperative game theory, assign a value to each feature indicating its contribution to the prediction. SHAP values are applied to neural networks to provide a global view of feature importance and individual predictions' explanations. Attention mechanisms, initially developed for natural language processing tasks, have been adapted to enhance interpretability in neural networks. These mechanisms highlight specific parts of the input data that are crucial for the network's decision, offering insights into the decision-making process. Rule-based approaches involve extracting interpretable rules from neural networks to describe decision boundaries. These rules provide a more human-understandable representation of the model's behavior. Visualizing the activations of neurons in different layers of a neural network can help interpret how information is processed.

Heatmaps and saliency maps visually highlight the regions of an input that are most influential in determining the output. Using decision trees as surrogate models to approximate the behavior of neural networks has become a popular approach. Decision trees are inherently interpretable, providing a more straightforward representation of the underlying decision logic. Developing interactive interfaces that allow users to explore and interact with model predictions and explanations in real-time. This facilitates a more user-friendly and intuitive understanding of the AI system's decisions.

Optimization algorithms

New optimization algorithms have been developed to train neural networks more efficiently. Techniques such as Adam, RMSprop, and Nadam aim to address challenges like slow convergence and getting stuck in local minima. Optimization algorithms play a crucial role in training artificial neural networks (ANNs) in data mining. The goal is to find the optimal set of weights and biases that minimize the error or loss function (Chaffey, Ellis, 2019: 320).

Over the years, several innovations in optimization algorithms have been introduced to address challenges such as convergence speed, handling complex loss surfaces, and improving the efficiency of training. Stochastic Gradient Descent (SGD) variants forms the foundation of many optimization algorithms for neural networks. Innovations include: Introduces a moving average of past gradients to accelerate convergence. Gradient clipping limits the magnitude of gradients during training to prevent exploding gradients. This innovation helps stabilize the training process, especially in deep networks. Batch Normalization normalizes the inputs of each layer, reducing internal covariate shift. This innovation allows for the use of higher learning rates and helps with faster convergence.

Layer-wise Adaptive Rate Scaling (LARS) adapts the learning rate for each layer based on the ratio of the norm of the weights to the norm of the gradients. This approach helps prevent vanishing or exploding gradients and improves convergence The Swish activation function, proposed by researchers at Google, has been shown to improve training efficiency and convergence in certain situations compared to traditional activation functions like ReLU (Xu, Zhang, 2005: 337).

Population-Based Training (PBT) PBT is an evolutionary optimization algorithm that dynamically adjusts hyperparameters during training. It combines elements of genetic algorithms with reinforcement learning to efficiently explore the hyperparameter space. Neuroevolution algorithms evolve neural network architectures and parameters using genetic algorithms or other evolutionary strategies. This approach explores the space of possible neural architectures to find effective models (Castells, 2022: 129).

Differentiable Architecture Search (DARTS) is a method that leverages gradient-based optimization to search for optimal neural network architectures. It allows for the automatic discovery of architectures that perform well on a given task.

Conclusion

The following innovations play an important role in the use of artificial neural networks in data mining:

- RNNs continue to be a crucial tool in various applications, including natural language processing, time-series analysis, and sequential pattern recognition. Ongoing research focuses on addressing challenges, improving training efficiency, and extending the capabilities of RNNs in handling diverse types of sequential data.

- Autoencoders continue to be a versatile tool for feature learning, dimensionality reduction, and generative tasks in various domains, including image analysis, natural language processing, and anomaly detection. Ongoing research aims to refine these architectures for even more efficient and effective representation learning.

- Explainable AI, enabling data scientists, domain experts, and end-users to understand and trust the decisions made by neural networks in data mining tasks. As AI systems become more complex, the development of effective XAI techniques is crucial for ensuring accountability, transparency, and ethical use of artificial intelligence.

- The choice of optimization algorithm depends on factors such as the dataset, model architecture, and specific requirements of the data mining task. Researchers continue to explore new approaches and enhancements to improve the efficiency, stability, and generalization of neural network training in the context of data mining. Transfer learning involves training a neural network on one task and then using the knowledge gained to improve performance on a related task. This is particularly beneficial when labeled data is scarce for the target task. Combining multiple neural networks or models can enhance predictive performance and robustness. Techniques like bagging, boosting, and stacking have been applied to neural networks, leading to improved generalization and accuracy.

As technology continues to advance, it is likely that the use of artificial neural networks in data mining will see further innovations and refinements.

References

1. Rabunal, J., Dorado, J. (2006). “Artificial neural networks in real-life applications”. 340 p.
2. Bakardjieva, M. (2019). The use of artificial neural networks in data mining. London, Sage Publications Ltd. 296 p.
3. Arens, W.F., Weigold, M.F. (2020). Artificial neural networks. Boston, McGraw Hill. 736 p.
4. Chaffey, D., Ellis F.C. (2019). Artificial neural networks. London, Pearson Education Limited. 576 p.
5. Daniel, W.B., Barbara, C. (2020). Artificial neural networks. Boston, SAGE Publications Ltd. 671 p.
6. Brown, S. (2018). The use of artificial neural networks in data mining. Boston, Marketing Book. 421 p.
7. Castells, M. (2022). Artificial neural networks. New Jersey, Wiley-Blackwell. 265 p.
8. Xu, S., Zhang, M. (2005). Data mining - an adaptive neural network model for financial analysis. Information Technology and Applications, IEEE, ICITA. 340 p.

Göndərildi: 29.02.2024 Qəbul olundu: 10.04.2024