International Journal of Science and Research (IJSR)
ISSN: 2319-7064
Impact Factor 2024: 7.101

A Novel Context-Aware Framework for Intelligent and Automated Data Augmentation in Machine Learning

Putty Srividya[1], Dr N. Subramanyam[2]

[1]Research Scholar, Annamalai University, Annamalai Nagar, Chidambaram - 608002, Tamil Nadu, India

[2]Professor, Department of CSE, Annamalai University, Annamalai Nagar, Chidambaram - 608002, Tamil Nadu, India

Abstract: ***Automated data augmentation has become an important technique for improving the performance and generalization of machine learning models by creating diverse training samples. This article examines major approaches to automated data augmentation, including analytical transformations, deep learning methods, generative models, and optimization techniques within Automated Machine Learning (AutoML) frameworks. It discusses how reinforcement learning, Bayesian optimization, and evolutionary algorithms help identify suitable augmentation strategies. Particular attention is given to generative artificial intelligence, including large language models and diffusion models, which can produce synthetic data for computer vision, natural language processing, and other applications. Dataset distillation is also examined as an alternative approach for reducing training data while preserving useful information. Comparative findings from established image classification datasets show that automated augmentation methods generally perform better than traditional manual techniques. However, high computational requirements, complex optimization processes, and the risk of altering the original meaning of data remain major challenges. The discussion highlights the need for efficient, reliable, and context-sensitive augmentation methods that can support machine learning applications across different domains.***

Keywords: Automated Data Augmentation, Automated Machine Learning, Generative Artificial Intelligence, Synthetic Data Generation, Dataset Distillation

Automated data augmentation Data augmentation involves creating variations of the original training data by applying context-appropriate transformation operations. For example, in computer vision, these transformations may involve image scaling, shearing, rotation, reflection and blurring operations. Text augmentation methods include synonym replacement, random insertion and deletion. The creation of such variations lead to a more representative data for the given task, and enhances the machine learning model's ability to generalise well on unseen data. Fig. 12 summaries the common data augmentation operations for commonsensical earning tasks. Typically, for a particular task, data augmentation may involve applying many of these basic operations together with more advanced techniques. As explained earlier, in most cases data augmentation is achieved by manipulating existing data in such a way as to create adequate variability. Most state-of-the-art methods for automated data augmentation rely on AutoML techniques. There are, however, practical situations where training data does not exist in any form, and augmentation methods are required to synthesize novel data from scratch. Generative artificial intelligence techniques have excelled in this area. Presently, for data manipulation, automation has largely been successful in computer vision and image understanding domains. It has been particularly challenging to extend these methods to time series forecasting and natural language processing domains owing to the difficulty in achieving semantic-preserving transformations without human supervision. Only a very few works have reported successful application of automated data augmentation techniques in natural language processing domains. Recently approaches for generative AI techniques have achieved remarkable results in text generation.

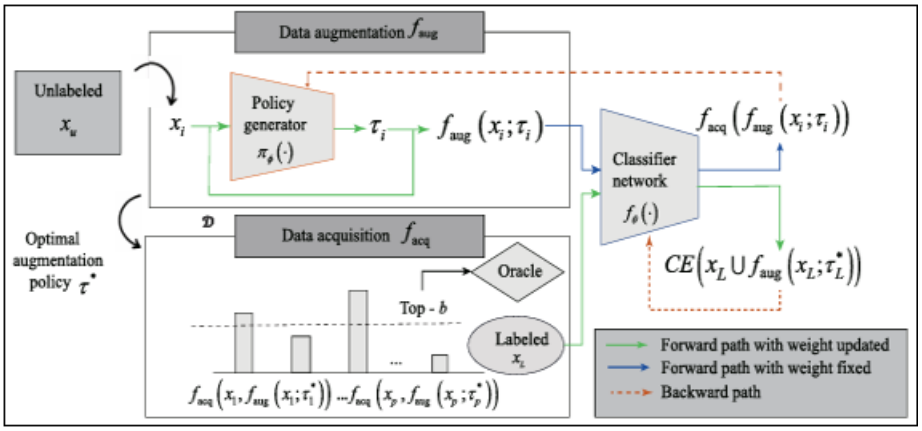


Figure 11: Data labelling by LADA: The LADA framework proposed in Ref. acquires unlabeled training data from the real world. An augmentation policy is then used to carry out initial preprocessing and generate additional data by learning relevant transformation operations. With the aid of a classification network, the optimum policy is determined using the classification loss. Finally, an oracle is used to label the unlabelled raw data.

Automated data augmentation in AutoML pipelines AutoML has emerged as a powerful approach that has achieved significant success in data augmentation for large-scale machine learning and big data applications. AutoML-based data augmentation methods generally aimed to automate the process of creating and applying appropriate transformation operations on the training data. In addition to data augmentation, the AutoML models perform a number of additional tasks in line with machine learning model development (e.g., model selection, hyper parameter tuning, performance validation, etc.). In an AutoML framework, a large number of augmentation operations are usually created manually or in an automated way, and the data augmentation task reduces to a simple search for the best transformation operations and their associated hyper parameters. The basic concept of AutoML approach to data augmentation is shown in Fig. 13.

Generation of augmentation operations: The first step in the augmentation process is to generate diverse transformation operations to manipulate the input data. In image augmentation, for example, the basic transformations are typically standard geometric and photometric image processing functions like rotation, flipping, scaling, shearing, color jittering, solarizaion, noise addition and contrast adjustment.

Machine learning task	Basic data augmentation methods
Image processing and computer vision (image classification, object detection, visual tracking, etc.)	• ***Rotation***: Rotating images by various angles to simulate different viewpoints • ***Flipping***: Horizontally flipping images to create mirror images • ***Cropping***: Randomly cropping images to focus on different parts of the scene • ***Zooming***: Applying random zoom to create variations in image scale • ***Color jittering***: Modifying the brightness, contrast, saturation, and hue of images • ***Gaussian noise***: Adding random Gaussian noise to images to simulate real-world variations • ***Blurring***: Applying Gaussian blur or other blurring techniques to images
Text recognition & language understanding (machine translation, text-to-voice conversion, etc.)	• ***Synonym replacement***: Replacing words with their synonyms to create variations. • ***Random deletion***: Randomly removing words to change sentence length and structure • ***Random insertion***: Randomly inserting additional words into a sentence to simulate natural language variation • ***Text paraphrasing***: Rephrasing sentences while preserving the original meaning • ***Character-level perturbations:*** Swapping, inserting, or deleting characters to simulate typographical errors or variations in spelling • ***Contextualized word embedding:*** Replacing words with alternatives that are contextually relevant
Time series forecasting (e.g., stock price prediction, economic performance forecasting, sales forecasting, etc.)	• ***Dynamic time warping***: Stretching or compressing the time axis of a time series to induce variations in the temporal scale of the data • ***Window slicing***: Dividing the original time series into overlapping or non-overlapping windows of varying lengths can provide the model with multiple perspectives of the data • ***Noise addition***: Introducing random noise to the time series data • ***Data interpolation***: Filling missing data points using interpolation techniques • ***Outlier injection***: Introducing outliers or anomalies into the time series data can help the model learn how to identify and handle unexpected events. • ***Time shifting***: Shifting the entire time series or specific portions of it forward or backward in time to create new instances with shifted patterns.

Figure 12: Common data augmentation functions for various machine learning tasks. Note that this list is by no means complete

Transformation magnitudes may include scale factors, rotation angels, translation offsets, color intensity and brightness levels, etc. The various transformation operations and their corresponding magnitudes constitute a search space. The effectiveness of automated augmentation strategies is based largely on the ability to compose a comprehensive set of operations that are representative of the tasks under consideration. The general approach to automating data augmentation tasks involves creating the set of required transformation operations and then applying the created operations on data and algorithmically determining the most useful augmentations using various optimization techniques. We briefly discuss common approaches to creating augmentations as well as techniques for optimizing the augmentation strategies. Analytical methods. The most basic approach to automating data augmentation is the use of semi-automated or analytical methods to create augmented data. The analytical method involves defining a set of explicit mathematical relations or heuristics to perform required transformations on the training data. These relations can be domain-specific and depend on the characteristics of the given data type. This class of methods generally perform standard augmentation functions such as rotation, shearing, flipping, and color jittering. Many works (e.g. Refs. [138–142]) involve defining different types of basic transformations functions to apply, as well as specifying the degree or magnitudes of these operations within an applicable range. While simple, the approaches can be effective for certain tasks, although they may not be able to account for unknown transformations.

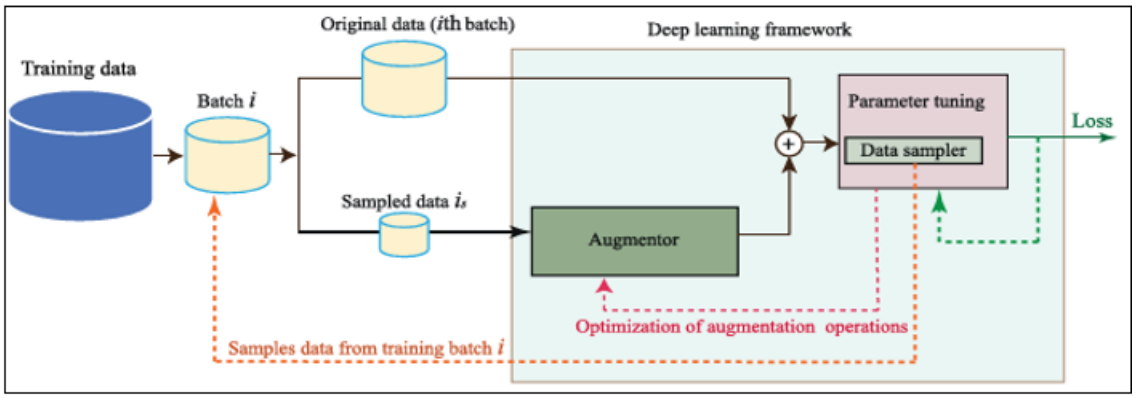


Figure 13: General scheme and procedure for automated data augmentation. A portion of the training set is sampled by a specified algorithm, a so called data sampler, for augmentation.

The selected sample undergoes a series of transformations by an augmenter sub-module. The augmented sample is then added to the original data batch i for training. The training loss is fed to both the sampler and augmenter to fine-tune their performance. Over time, this results in better sampling and augmentations, leading to overall improved performance. Deep-learned transformations. Some recent works [46,145–148] have suggested leveraging learned mechanisms in favor of pre defined augmentation operations. The goal is to allow augmentation operations to be learned from input data rather than relying on explicitly formulated relations. The basic idea is to implement data transformation sub-modules within deep learning pipelines whose parameters and hyperparameters can be leaned and applied on input data in the training process. This method offers a higher degree of automation than approaches based on explicit analytical formulations. It also allows arbitrary or unknown augmentation styles to be achieved. A common technique widely employed for learning transformations is the spatial transformation network (STN) [45] (Fig.14(a)). 4.1.1.3. Generative machine learning techniques. Another common way to automate the construction of data augmentation operations is based on generative modeling [145,149–151]. This approach creates transformed samples without explicitly requiring analytical for mulas, or even requiring implicit transformation parameters to be learned. The basic idea is to first learn the distribution of real data by training a discriminator-generator pair on the original dataset. Knowledge about the distribution can then help to generate new but slightly varied samples that simulate data diversity. Thus, in the automated data processing pipeline, the generative model is employed as an intermediate data transformation unit. Its parameters are learned jointly with the CNN model in a bi-level optimisation scheme (see Fig. 14(b)). Generative machine learning approaches based on GANs are particularly common in computer vision domains, where they enable realistic image samples to be generated with desired visual features. This family of methods is also useful in applications where it is necessary to transfer knowledge from an existing domain to a different domain (e.g., in Ref. [152]). Generative methods are also used to align the distributions of data from different sources, enabling deep learning models to perform better when applied to new data that may come from a slightly different distribution. However, despite the usefulness of the approach, generative models themselves require careful tuning, thereby greatly increasing the for human supervision and limiting the full benefits of automation.

Optimisation of augmentation strategies Optimisation techniques are typically employed to search for the best augmentation strategy. The search task involves finding not only relevant transformations but also the optimum levels of transformations. For image augmentation these levels may be rotation angles, translation offsets, saturation values, etc.

Because the search space is inherently complex and discontinuous, many works utilize black-box optimisation techniques such as reinforcement learning (e.g., in Refs. [138,140,142,153,154]), Bayesian optimisation (e.g., in Refs. [142,155,156]) and evolutionary computation algorithms (e.g., in Refs. [157–160] to search for good augmentation strategies. These techniques have produced impressive results but are noticeably slow when dealing with very large data. To overcome this limitation, approaches for optimising the discrete augmentation search space using the concepts of approximation gradients [161] have also been proposed [149,162–164]. The use of these concepts makes it possible to develop methods for solving optimization problems for discontinuous functions present in the search space. Approaches based on approximate techniques are generally more efficient than black-box search techniques. Alternative search spaces providing the ability to find effective augmentation policies without extensive search have also been considered [165,166]. In some cases (e.g., Ref. [167]) these approaches eliminate the need for the search stage altogether. For instance, researchers in Ref. [166] contend that it is not necessary to conduct exhaustive search for extensive range of possible augmentations and hyperparameters in a combinatorial manner and, instead, propose to reduce the search space to a linear search space of uniform probabilities and augmentation parameters that can be traversed by a simple grid search. The approaches, despite the significant reduction in compute time, have shown competitive performance. However, there seems to be little room to further extend their predictive performance.

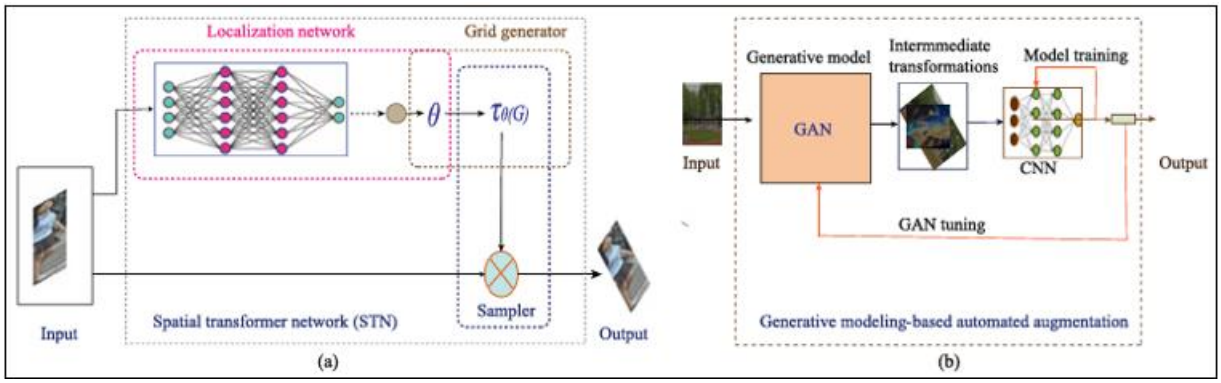


Figure 14: Approaches to generating basic transformations within automated data augmentation frameworks: (a) learned transformation using spatial transformation networks (STN), and (b) generative modelling- based synthesis of transformed image data

Data synthesis by large language models (LLMs) and diffusion models. The automated data augmentation methods covered in Section 4.1.1 utilize generative modeling techniques to perform intermediate data processing in AutoML pipelines. While generative methods can be used to perform data transformation operations in that manner, approaches based on variational autoencoders [168], generative adversarial networks [169], diffusion models [170], autoregressive models [171,172] and large language models (LLMs) [173] are commonly designed to directly synthesize data by themselves, thereby bypassing all intermediate processing steps. Recent advances in deep generative AI-based data synthesis methods, especially diffusion models and large language models (LLMs), have enabled the possibility of generating clean data from scratch or from noisy data in an end-to-end manner. Data generated this way can be used to augment existing data (e.g., in Refs. [174–176]) or completely replace natural data in situations where datasets are inadequate or are inaccessible for training machine learning models (e.g., in Refs. [177,178]). Large language models are a special type of deep learning systems based on the concept of the transformer [179] and are primarily designed to perform NLP tasks. On the other hand, diffusion models are a particularly versatile class of generative methods that synthesize data in a similar way to generative adversarial networks– a technique that allows deep neural networks to generate data using random noise. They are based on non-equilibrium thermodynamics [180]. Diffusion models are realized by first defining a Markov chain of diffusion steps that gradually increase the random noise component to input data, and then implement a reverse diffusion process to re-create the desired data samples from the input noise (Fig. 15). Generative methods based on LLMs [174,181] and diffusion models [182–184] have become very useful in augmenting data for natural language processing and computer vision applications, respectfully. In NLP applications, additional text data commonly generated by prompting LLMs such as ChatGPT [185], LLaMA [186] and BERT [187] to complete input sentences with missing words or phrases so that different variations of the original sentences could be created. For instance, Ubani et al. in Ref. [188] propose to manipulate text data with the help of ChatGPT to produce augmented data for training deep learning networks for NLP-based tasks by giving appropriate prompts to generate syntactically different variations of the original sentences. In computer vision applications, generative AI-based data augmentation typically involves manipulating existing images to generate novel styles, poses, background contexts and views. Diffusion CLIP [184] introduces an approach to perform various image transformations using text-guided prompts with a diffusion model developed on the basis of CLIP [189]. The approach achieved photorealistic stylizations of different images in a wide range of novel contexts. Fig. 16 shows the result of various image stylizations methods using DiffusionCLIP [184] and two other text-based generative data manipulation models– StyleCLIP [190] and StyleGAN-NADA [191]

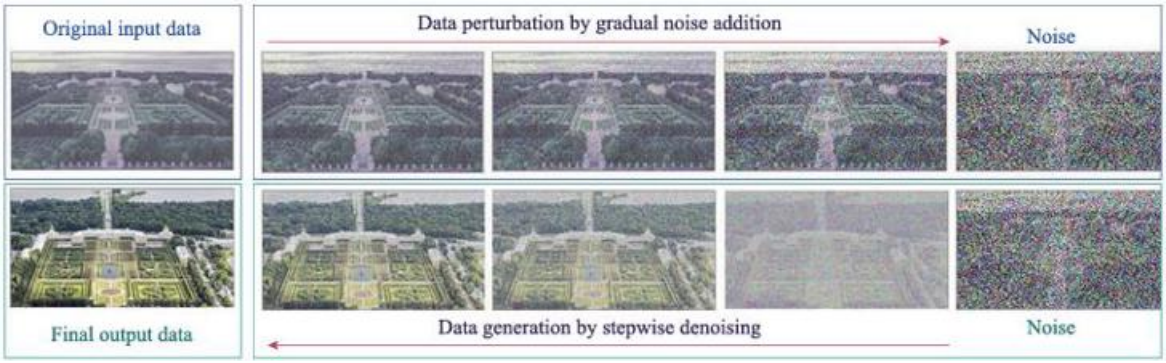


Figure 15: The idea of diffusion models is to learn to generate data by gradually adding random perturbations to the input data, and then perform the reverse process of denoising the data until a clean data output is achieved

While diffusion models and LLMs can both perform data processing– e.g., text manipulation [188] and image quality enhancement (super-resolution) [182,192], denoising [183] and styling [193,194], their most powerful use-case has been the task of automatically generating high quality data from "scratch" [195,196]. Modern generative AI techniques based

on diffusion models can be used to synthesize many different types of data– including images [197,198], videos [199–201], text [188,202], audio [203,204], time series [205] and tabular data [206–208]– using only text prompts without additional input data. Language models, when trained on pairs of data (e.g., images) and corresponding descriptive texts (e.g., captions), can learn the association between the textual descriptions and the underlying data. The trained model can then be used to generate desired images based on descriptive text inputs. In this regard, large language models and diffusion models play a complementary role in the data generation process. Indeed, the popularity of large-scale diffusion models has been driven largely by the development of powerful generative tools that generate data based on intuitive text prompts enabled by LLM techniques. Some of the most prominent data generation models in this category include Dalle-2 [209], stable diffusion [210], Glide [193] and Imagen [211]. Fig. 17 depicts pho to realistic images generated by different diffusion models based on user's supplied prompts. 4.3. Dataset distillation methods While the approaches discussed in Section 4.1 rely on generating extra data to improve machine learning models, an alternative approach, known as dataset distillation [214,215], seeks to select an informative subset from a large-scale training dataset that retains the important properties and, importantly, the generalization performance of the original dataset. That is, the aim is to produce a leaner dataset that still performs satisfactorily on the target task. The approach can also be used to distill data labels from a large set of labels (e.g., in Refs. [216,217]) or from noisy labels (e.g., in Ref. [218]). While traditional approaches to dataset distillation rely on manually engineered procedures, some recent works [219,220] have proposed mechanisms for automating the process. Although these ap proaches are well motivated, and can potentially lead to high performing but leaner deep learning models, at present their level of automation is limited. Consequently, their popularity is relatively low.

4.4. Performance of automated data augmentation methods Results from several studies show that models trained on data generated by automated techniques outperform those trained on data generated using manual approaches. In Table 3 we compare the performance of ten state-of-the-art automated data augmentation techniques and ten of the best traditional augmentation methods across various datasets (CIFAR-10 [221], CIFAR-100 [221] and ImageNet [222]). All results are obtained using wide ResNet (WRN-28-10) [223] backbone trained until convergence– roughly about the same number of epochs (300). This provides a fair ground for comparison of the different methods. In Fig. 18 we summarize the comparative performance by averaging the classification scores as well as the percentage improvement over the baseline (i.e., models trained without augmentation). These results convincingly demonstrate the effectiveness of automated data augmentation approaches (see Table 4). While the performance of automated augmentation techniques is in no doubt, an important challenge is the enormous computational resources typically required for training these models. Also, since the augmentation process is not guided by human intuition, some augmentation operations may unduly alter the original semantic meaning of the data, which may not be observed at training time and can later lead to catastrophic failure. For instance, modifying the label of an image or applying overly aggressive transformations in such a way that they transform its semantic meaning can result in erroneous training signals with potentially harmful implications.

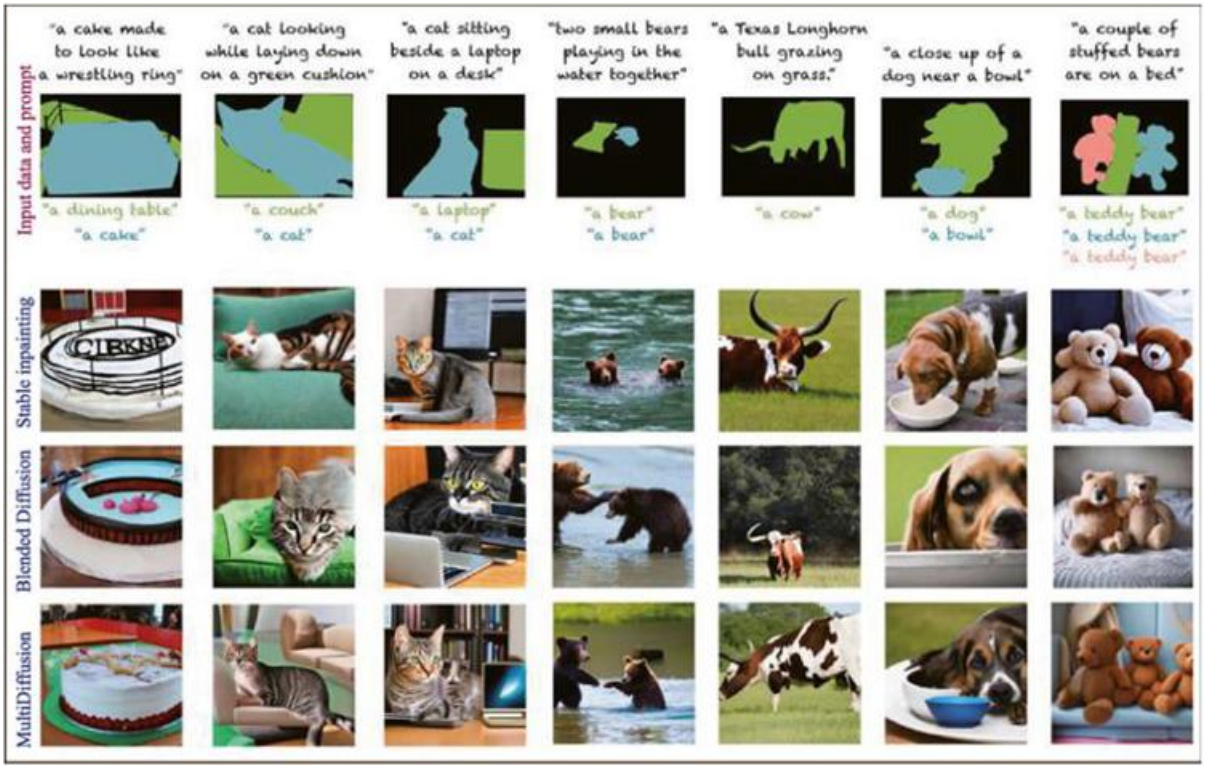


Figure 17: Examples of images generated by state- of- the- art diffusion models. Stable inpainting [210], Blended Diffusion [212] and MultiDiffusion [213]. The images have been generated using text prompts together with sample foreground images as inputs. Image is from MultiDiffusion [213].



Figure 16: Visual Comparison of results of different generative AI- based image stylization methods. DiffusionCLIP [184], StyleGAN- NADA [191] and StyleCLIP [190]. The models attempt to style a church building with respect to different visual context. Image is from Ref. [184]. We have applied some minor changes in the text fonts to make it clear

Table 3
Performance results of automated and traditional data augmentation methods on classification tasks. The values represent percentage accuracies on CIFAR-10, CIFAR-100 and ImageNet datasets. **Bold** is highest score, *italic* is second highest, and underline is third highest.

Method	Classification accuracy on various datasets			
	CIFAR-10	CIFAR-100	ImageNet (top 1)	ImageNet (top 5)
Baseline	96.1	81.2	76.3	92.1
Automated data augmentation methods				
TrivialAA [224]	97.5	84.3	78.1	93.9
DivAug [225]	*98.1*	84.2	78.0	–
AWS [153]	98.0	*84.7*	79.4	*94.5*
MetaAug [226]	97.7	83.8	*79.7*	**94.6**
PBA [159]	97.4	83.3	77.2	93.4
AdvAA [149]	*98.1*	84.5	**79.9**	*94.5*
A2-Aug [227]	98.0	**85.2**	79.2	–
FastAA [142]	97.3	82.8	77.6	93.7
KeepAA [228]	**98.7**	–	78.0	93.0
DeepAA [229]	97.4	83.7	78.3	–
Traditional data augmentation methods				
StochasticDepth [230]	–	–	77.5	93.7
RE [231]	96.9	82.3	77.3	93.3
RICAP [232]	97.2	82.0	78.6	–
SalencyMix [233]	96.0	80.5	78.7	–
GridMask [144]	97.2	83.4	77.9	–
SmoothMix [234]	–	–	77.7	93.7
Manifold Mixup [235]			77.5	93.8
CutOut [236]	96.9	81.6	77.1	
Mixup [237]	97.3	82.1	77.9	93.9
FMix [238]	96.4	82.0	77.7	–

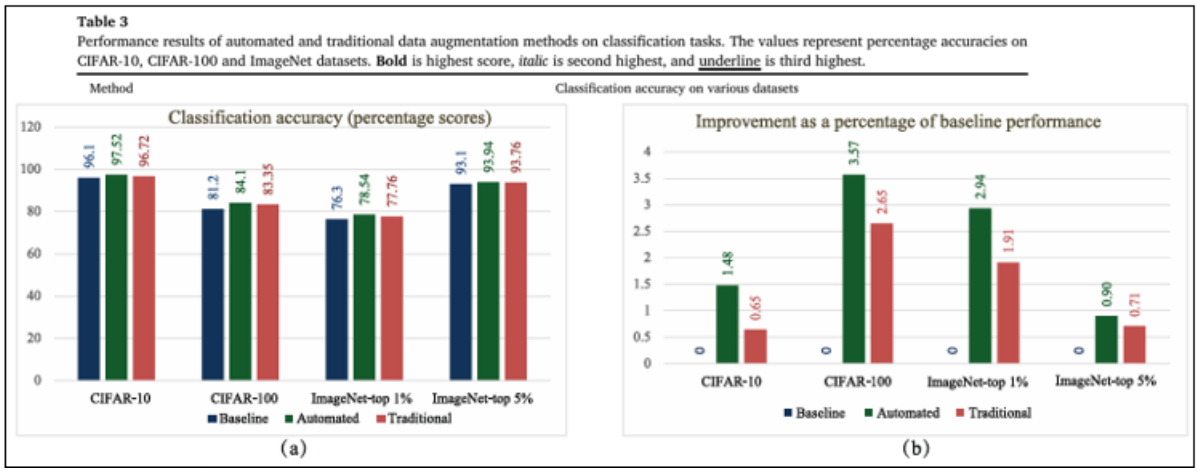


Figure 18: Comparison of automated data augmentation and traditional methods. Results for each class of approaches are based on 10 state- of- the art techniques

References

[1] M. Guillermo, R.K. Billones, A. Bandala, R.R. Vicerra, E. Sybingco, E.P. Dadios, A. Fillone, Implementation of automated annotation through mask RCNN object detection model in CVAT using AWS EC2 instance, in: Proceedings of the 2020 IEEE Region 10 Conference (TENCON), IEEE, Piscataway, 2020, pp. 708–713.

[2] M. Jiu, H. Sahbi, Context-aware deep kernel networks for image annotation, Neurocomputing 474 (2022) 154–167.
[3] G. Yang, J. Zhang, Y. Zhang, B. Wu, Y. Yang, Probabilistic modeling of semantic ambiguity for scene graph generation, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, IEEE, Piscataway, 2021, pp. 12527–12536.
[4] B. Martinez, M.F. Valstar, Advances, challenges, and opportunities in automatic facial expression recognition, Advances in Face Detection and Facial Image Analysis (2016) 63–100.
[5] K. Goyle, Q. Xie, V. Goyle, DataAssist: A machine learning approach to data cleaning and preparation, 2023. https://arxiv.org/pdf/2307.07119v2.pdf.
[6] P. Li, Z. Chen, X. Chu, K. Rong, DiffPrep: Differentiable data preprocessing pipeline search for learning over tabular data, Proceedings of the ACM on Management of Data 1 (2) (2023) 1–26.
[7] J.A. Valeri, L.R. Soenksen, K.M. Collins, P. Ramesh, G. Cai, R. Powers, N.M. Angenent-Mari, D.M. Camacho, F. Wong, T.K. Lu, et al., BioAutoMATED: An end-to-end automated machine learning tool for explanation and design of biological sequences, Cell Systems 14 (6) (2023) 525–542.
[8] S. Krishnan, E. Wu, AlphaClean: Automatic generation of data cleaning pipelines, 2019. https://arxiv.org/pdf/1904.11827v2.pdf.
[9] S. Chen, N. Tang, J. Fan, X. Yan, C. Chai, G. Li, X. Du, HAIPipe: Combining human-generated and machine-generated pipelines for data preparation, Proceedings of the ACM on Management of Data 1 (1) (2023) 1–26.
[10] Y.-Y. Kim, K. Song, J. Jang, I.-C. Moon, LADA: Look-ahead data acquisition via augmentation for deep active learning, Advances in Neural Information Processing Systems 34 (2021) 22919–22930.
[11] D. Sakkos, E.S.L. Ho, H.P.H. Shum, G. Elvin, Image editing-based data augmentation for illumination-insensitive background subtraction, Journal of Enterprise Information Management 36 (3) (2023) 818–838.
[12] S. Schelter, J. Stoyanovich, Taming technical bias in machine learning pipelines, Bulletin of the Technical Committee on Data Engineering 43 (4) (2020) 39–50.
[13] R. Jiang, S. Mei, Polar coordinate convolutional neural network: From rotation-invariance to translation-invariance, in: Proceedings of the 2019 IEEE International Conference on Image Processing (ICIP), IEEE, Piscataway, 2019, pp. 355–359.
[14] B. Hilprecht, C. Hammacher, E.S. Reis, M. Abdelaal, C. Binnig, DiffML: End-to-end differentiable ML pipelines, in: Proceedings of the Seventh Workshop on Data Management for End-To-End Machine Learning, ACM
[15] S. Schelter, J. Stoyanovich, Taming technical bias in machine learning pipelines, Bulletin of the Technical Committee on Data Engineering 43 (4) (2020) 39–50.
[16] R. Jiang, S. Mei, Polar coordinate convolutional neural network: From rotation-invariance to translation-invariance, in: Proceedings of the 2019 IEEE International Conference on Image Processing (ICIP), IEEE, Piscataway, 2019, pp. 355–359.
[17] B. Hilprecht, C. Hammacher, E.S. Reis, M. Abdelaal, C. Binnig, DiffML: End-to-end differentiable ML pipelines, in: Proceedings of the Seventh Workshop on Data Management for End-To-End Machine Learning, ACM, New York, 2023, pp. 1–7.
[18] A. Mumuni, F. Mumuni, Data augmentation: A comprehensive survey of modern approaches, Array 16 (2022) 100258.
[19] T. Niu, M. Bansal, Automatically learning data augmentation policies for dialogue tasks, 2019. https://arxiv.org/pdf/1909.12868v1.pdf.
[20] S. Ren, J. Zhang, L. Li, X. Sun, J. Zhou, Text AutoAugment: Learning compositional augmentation policy for text classification, 2021. https://arxiv.org/pdf/2109.00523v1.pdf.
[21] E.D. Cubuk, B. Zoph, D. Mane, V. Vasudevan, Q.V. Le, AutoAugment: Learning augmentation strategies from data, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, IEEE, Piscataway, 2019, pp. 113–123.
[22] R. Hataya, J. Zdenek, K. Yoshizoe, H. Nakayama, Faster AutoAugment: Learning augmentation strategies using backpropagation, in: Proceedings of the European Conference on Computer Vision, Springer, 2020, pp. 1–16.
[23] S. Lin, T. Yu, R. Feng, X. Li, X. Jin, Z. Chen, Local Patch AutoAugment with multi-agent collaboration, IEEE Transactions on Multimedia 26 (2023), 724-236.
[24] E.D. Cubuk, B. Zoph, J. Shlens, Q.V. Le, RandAugment: Practical automated data augmentation with a reduced search space, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), IEEE, Piscataway, 2020, pp. 702–703.
[25] S. Lim, I. Kim, T. Kim, C. Kim, S. Kim, Fast AutoAugment, in: Proceedings of the 33rd International Conference on Neural Information Processing Systems, Curran Associates Inc., New York, 2019, pp. 6665–6675, 32 (2019).
[26] A. Liu, Z. Huang, Z. Huang, N. Wang, Direct differentiable augmentation search, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, IEEE, Piscataway, 2021, pp. 12219–12228.
[27] P. Chen
[28] R. Hataya, J. Zdenek, K. Yoshizoe, H. Nakayama, Faster AutoAugment: Learning augmentation strategies using backpropagation, in: Proceedings of the European Conference on Computer Vision, Springer, 2020, pp. 1–16.
[29] S. Lin, T. Yu, R. Feng, X. Li, X. Jin, Z. Chen, Local Patch AutoAugment with multi-agent collaboration, IEEE Transactions on Multimedia 26 (2023), 724-236.
[30] E.D. Cubuk, B. Zoph, J. Shlens, Q.V. Le, RandAugment: Practical automated data augmentation with a reduced search space, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), IEEE, Piscataway, 2020, pp. 702–703.
[31] S. Lim, I. Kim, T. Kim, C. Kim, S. Kim, Fast AutoAugment, in: Proceedings of the 33rd International Conference on Neural Information Processing Systems, Curran Associates Inc., New York, 2019, pp. 6665–6675, 32 (2019).

[32] A. Liu, Z. Huang, Z. Huang, N. Wang, Direct differentiable augmentation search, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, IEEE, Piscataway, 2021, pp. 12219–12228.
[33] P. Chen, S. Liu, H. Zhao, J. Jia, GridMask data augmentation, 2020. https://arxiv.org/pdf/2001.04086v2.pdf.