Dynamic Mode Decomposition based feature for Image Classification

Rahul-Vigneswaran K
Department of Mechanical Engineering
Amrita Vishwa Vidyapeetham
Amritapuri, India
rahulvigneswaran@gmail.com

Sachin-Kumar S, Neethu Mohan, Soman KP
Center for Computational Engineering and Networking (CEN)
Amrita School of Engineering, Coimbatore
Amrita Vishwa Vidyapeetham, India
{sachinnme, neethumohan.ndkm, kpsoman2000}@gmail.com

Abstract—Irrespective of the fact that Machine learning has produced groundbreaking results, it demands an enormous amount of data in order to perform so. Even though data production has been in its all-time high, almost all the data is unlabelled, hence making them unsuitable for training the algorithms. This paper proposes a novel method of extracting the features using Dynamic Mode Decomposition (DMD). The experiment is performed using data samples from Imagenet. The learning is done using SVM-linear, SVM-RBF, Random Kitchen Sink approach (RKS). The results have shown that DMD features with RKS give competing results.

Index Terms—limited data, few shot learning, dynamic mode decomposition, dmd, singular value decomposition, svd, support vector machine, svm, random kitchen sink algorithm, unlabelled data

I. INTRODUCTION

The human race is generating more data than any point of time in history. Still, they are unsuitable for training an algorithm because almost all the data generated are either biased or unlabelled. Due to these reasons, data scientists are being restricted to use only a minuscule portion of the generated data, that has been preprocessed and cleaned. That makes the almost 99% of the generated data unusable and on the other hand, today’s state-of-the-art Deep learning algorithms are designed to be data thirsty in its training stage. Recent days, scientist have started designing architectures that can learn the distribution with less data, unlike their counterparts. Largely, these new waves of algorithms can achieve this by the following ways,

- Semi-supervised Learning [1]–[13]
- Weak supervised Learning [14]–[20]
- Active Learning [21]–[23]
- Transfer Learning [24], [25]
- Multi-task Learning [26], [27]
- Few-shot Learning [28]–[30]
- Data Augmentation [31], [32]
- Reinforcement Learning [33], [34]

[1], [3] comes under the category of generative models of data. As the name suggests, [1] make use of the generative a classifier through a generative model by iterative Expectation-Maximization (EM) techniques, a variant of Deterministic Annealing whole and [3] use the unlabelled data to make the synthetically generated labelled data, less synthetic by use of Generative Adversarial Nets (GANs) [4]. [5] uses a method called co-training (When a set of data is divided into parts by nature and this trait is exploited by algorithms, they are categorized into co-training) where [5] finds the weak indicator from labelled data and finds the corresponding unlabeled data to strengthen it. Like the methods discussed so far, there are several techniques used for learning with limited labelled data. Table I gives a detailed summary of methods from each category mentioned previously.

Section I gives a brief introduction on the existing methods for limited labelled data learning, Section II provides an elaborate explanation of the concepts used in the proposed approach like Dynamic Mode Decomposition (DMD) and Random Kitchen Sink (RKS) algorithm. Section III details the proposed approach and Section IV elaborates the obtained results and draws the underlying commonalities which are interesting. Finally Section V gives an essence of the proposed approach’s findings and concludes with the future scope of this research.

II. MATERIALS AND METHODS

A. Dataset

The dataset used for benchmarking is the Tiny Imagenet Dataset which is a miniature version of the Imagenet Dataset. It contains 200 classes and each class contains 500 Images each. Each Image is 64x64 pixels in size.

B. Dynamic Mode Decomposition (DMD)

It’s a way of extracting the underlying dynamics of a given data that flows with time. It is a very powerful tool for analysing the dynamics of non-linear systems and was developed by Schmid [35]. It is also used for forecasting [49], natural language processing [50], salient region detection from images [38], etc. It was inspired by and closely related to Koopman-operator analysis [36]. The popularity gained by DMD in the fluids community is majorly due to its ability...
to provides information about the dynamics of flow, even when those dynamics are inherently non-linear. In short, DMD is a method driven by data, free from the equation which has the capability of providing a precise decomposition of a system which is highly complex into respective coherent spatio-temporal structures, that can be fashioned for predicting for few timestamps into the future. A typical DMD algorithm involves the following enumerations,

1) Compute Singular Value Decomposition (SVD) of $X_a$ as,

$$X_a \approx Q\Sigma B^{H}$$

There $Q \in C^{N \times J}$, $\Sigma \in C^{J \times J}$, $B \in C^{M \times J}$, $J$ refers to the SVD approximation of $X_a$ which is reduced.

2) Compute matrix $C$ from,

$$X_a \approx X_b C \Rightarrow Q\Sigma B^{H} C$$

$$C \Rightarrow B\Sigma^{T}Q^{H}X_b$$

3) Compute similar matrix of $C$ which is $\tilde{C}$ by,

$$\tilde{C} \Rightarrow Q^{H}X_b\Sigma^{T}B$$

4) Compute the Eigen Decomposition of $\tilde{C}$ by,

$$AQ\Sigma B^{H} = X_b \Rightarrow AQ = X_b B\Sigma^{T}$$

Pre-multiply by $Q^{H}$ on both sides,

$$Q^{H}AQ = Q^{H}X_b B\Sigma^{T} = \tilde{C}$$

$$AQ = Q\tilde{C} = Q(T\Omega T^{T}) \Rightarrow A(QT) = (QT)^{T}$$

There, $T\Omega T^{T}$ is the eigen decomposition.

5) Compute the Dynamic Modes matrix by,

$$\Phi = QT \Rightarrow \Phi = X_b B\Sigma^{T}T$$

C. Random Kitchen Sink algorithm

The aim of Random Kitchen Sinks (RKS) algorithm and the methods similar to it, is not to perform inference but rather aim at overcoming the limitations of other kernel-based algorithms. Kernel-based algorithms perform well in almost all the settings but heavily depend on matrix manipulation. If a matrix is $(n \times n)$ then naively the computation cost is $O(n^3)$ which bottlenecks them to applications that have limited samples. One of the general ways to overcome this limitation is by use of low-rank methods (even though other approaches like Bayesian committee machines and Kronecker based methods exist).

Random Fourier features [37] aims at sampling subset components of kernel Fourier to generate low-rank-approximations of kernels that are invariant to shifts. Due to the reason that the Fourier spaces are invariant to shift, this property is not changed. But now a kernel Hilbert space which is reproduced by a finite dimensional space by these Fourier components’ union. As a result, the RKS which was infinite dimensional once is approximated by the degenerate approximate kernel.

The epitome of supervised machine learning approaches are to obtain the knowledge of an approximate function $f$ (priori) which can map the input variable $\tilde{x}$ to output variable $\tilde{y}$ (i.e.) $\tilde{y} = \tilde{f}(\tilde{x})$. The idea of finding such a function is that when a new data comes, the function can predict the corresponding output. In real-world applications, the input data can be image, 1-D signal, text data etc and the output will be the corresponding labels. The learning of mapping function often involves finding the best parameters $\theta$ for the function $f(\tilde{x}; \theta)$ to get the maximum performance. Kernel methods are the best examples of supervised approaches which extensively used for several machine learning problems. It requires to compute a Kernel matrix $R_{M \times M}$ $(M$ signifies the count of the input vectors). However, the above mentioned computation
suffers badly when the dataset size is large. There has been an effort to reduce the dimension of the Kernel matrix using smart sample selection [39], Eigen decomposition via Nystrom [40], low-rank approximations [41]. In [42], [43], the authors proposed an alternative approach via randomization, known as Random Kitchen Sinks (RKS) algorithm, to compute the Kernel matrix even when the dataset size becomes large. The idea is to provide an approximate kernel function via explicit mapping

\[ K(x_a, x_b) = (\phi(x_a), \phi(x_b)) \approx (z(x_a), z(x_b)) \]

Here, \( \phi(\bullet) \) denotes the implicit mapping function (used to compute kernel matrix) and \( z(\bullet) \) denotes the explicit mapping function. The RKS method approximates the kernel trick [44], [45]. This explicit mapping function can be written as [46]– [48].

\[
z(x) = \sqrt{\frac{1}{K}} \begin{bmatrix}
\cos (x^T \Omega_1) \\
... \\
\cos (x^T \Omega_k) \\
\sin (x^T \Omega_1) \\
... \\
\sin (x^T \Omega_k)
\end{bmatrix}
\]

III. PROPOSED APPROACH

As mentioned earlier in Section II-B, DMD can be used in data that can flow in time. Contrary to that, the images that we are using are static in nature. Therefore flow is induced [38] to the image as shown in Figure 1 by extracting the different bands of colours by converting it into Lab colour space and permutation of luminescence bands and colour bands into a single matrix. After applying DMD, the sparse and low-rank components are extracted and normalized for being used as features.

These extracted features capture the underlying dynamics of the image. Therefore, these are then given as input to the Random Kitchen Sink Algorithm (Section II-C) for classification.

IV. RESULTS AND DISCUSSION

After obtaining the features, they are given as input to the Random Kitchen Sink algorithm and Support Vector Machine (SVM) with various configurations. Tables II, III, IV contain the accuracies of the proposed approach under various configurations when classified by Random Kitchen Sink algorithm, SVM-rbf and SVM-linear respectively. Column 1 represents the percentage of the total dataset used for setting, column 2 represents the count of eigenvalues taken into consideration for reconstruction of the features, column 3 denotes the type of class (Distinctive - Classes that are easily differentiable from one another; Overlapped - Classes that have overlapping features), and column 4 represents the accuracy of the corresponding configuration. All the accuracies corresponding to their configurations are plotted in Figure 4. It is evident from it that the Random Kitchen Sink Algorithm (RKS), constantly comes on top as compared to the other two algorithms.

It is evident from both the Figure 4 and Table II that, the maximum accuracy is attained when the configuration is as follows:

| Test Data (in %) | Number of Eigen Values | Type of Data | Accuracy (in %) |
|-----------------|------------------------|--------------|----------------|
| 60              | 5                      | Distinctive  | 80.87          |

Fig. 1. Proposed Architecture

- **Table II**: Accuracy of Random Kitchen Sink Algorithm under different configurations.

- **Table III**: Accuracy of SVM-rbf under different configurations.

- **Table IV**: Accuracy of SVM-linear under different configurations.
TABLE III
ACCURACY OF SUPPORT VECTOR MACHINE (RBF KERNEL) UNDER DIFFERENT CONFIGURATIONS

| Test Data (in %) | Number of Eigen Values | Type of Data | Kernel | Accuracy (in %) |
|-----------------|------------------------|--------------|--------|-----------------|
| 70              | 3                      | Distinctive  | rbf    | 51.52           |
| 70              | 3                      | Overlapped   | rbf    | 34.76           |
| 70              | 4                      | Distinctive  | rbf    | 47.43           |
| 70              | 4                      | Overlapped   | rbf    | 34.00           |
| 70              | 5                      | Distinctive  | rbf    | 48.76           |
| 70              | 5                      | Overlapped   | rbf    | 36.86           |
| 60              | 3                      | Distinctive  | rbf    | 32.89           |
| 60              | 3                      | Overlapped   | rbf    | 34.89           |
| 60              | 4                      | Distinctive  | rbf    | 55.00           |
| 60              | 4                      | Overlapped   | rbf    | 36.11           |
| 60              | 5                      | Distinctive  | rbf    | 41.78           |
| 50              | 3                      | Distinctive  | rbf    | 32.40           |
| 50              | 3                      | Overlapped   | rbf    | 34.13           |
| 50              | 4                      | Distinctive  | rbf    | 51.33           |
| 50              | 4                      | Overlapped   | rbf    | 36.67           |
| 50              | 5                      | Distinctive  | rbf    | 54.00           |
| 50              | 5                      | Overlapped   | rbf    | 38.27           |

The reason why the accuracy tops when the number of Eigenvalues is “5” can be explained through Figure 3. After the Eigenvalue index of “5”, the Eigenvalues cease to change and doesn’t contribute much to the underlying dynamics of the image. After the features are extracted, it is given as input for RKS in which it is mapped from 640 to 500.

Figure 5 shows the reconstructed image with low-rank and sparse matrix with 5 Eigenvalues which clearly captures the skeleton of the image. This extraction of the skeletal structure is the dynamics captured by applying DMD for feature extraction. t-SNE plot in Figure 2 provides a better picture of how the extracted DMD features of images arrange themselves distinctive groups flawlessly. Figure 6 and Figure 7 are the t-SNE plots of 3 classes before and after applying the proposed technique. It is evident from them that, the proposed approach is promising and effective.

The present approach is a novel one and needs more applications-oriented experimental evaluations. There are certain cases like in Figure 8 which has a complex background and where it is difficult to differentiate the foreground of the object of interest from the background, the proposed approach fails to perform. Apart from cases like in Figure 8 the proposed approach is proven to be effective in learning with limited labelled data.

V. CONCLUSION

As the world’s data generation explodes and due to the reason that manual labelling is highly expensive, it is necessary

TABLE IV
ACCURACY OF SUPPORT VECTOR MACHINE (LINEAR KERNEL) UNDER DIFFERENT CONFIGURATIONS

| Test Data (in %) | Number of Eigen Values | Type of Data | Kernel | Accuracy (in %) |
|-----------------|------------------------|--------------|--------|-----------------|
| 70              | 3                      | Distinctive  | linear | 46.38           |
| 70              | 3                      | Overlapped   | linear | 34.00           |
| 70              | 4                      | Distinctive  | linear | 42.10           |
| 70              | 4                      | Overlapped   | linear | 32.86           |
| 70              | 5                      | Distinctive  | linear | 42.95           |
| 70              | 5                      | Overlapped   | linear | 36.10           |
| 60              | 3                      | Distinctive  | linear | 41.78           |
| 60              | 3                      | Overlapped   | linear | 34.00           |
| 60              | 4                      | Distinctive  | linear | 44.33           |
| 60              | 4                      | Overlapped   | linear | 34.22           |
| 60              | 5                      | Distinctive  | linear | 42.44           |
| 60              | 5                      | Overlapped   | linear | 34.44           |
| 50              | 3                      | Distinctive  | linear | 49.20           |
| 50              | 3                      | Overlapped   | linear | 32.00           |
| 50              | 4                      | Distinctive  | linear | 44.80           |
| 50              | 4                      | Overlapped   | linear | 36.53           |
| 50              | 5                      | Distinctive  | linear | 44.93           |
| 50              | 5                      | Overlapped   | linear | 33.07           |
Fig. 4. Accuracy plot - RKS Vs SVM-rbf Vs SVM-linear

Fig. 5. Reconstructed Low-rank (1st row) and Sparse matrix (2nd row) for 5 Eigen Values

Fig. 6. t-SNE plot of 3 classes before applying the proposed approach.

Fig. 7. t-SNE plot of 3 classes after applying the proposed approach.

Fig. 8. Cases where the proposed approach is ineffective
to develop and explore machine learning architectures that can classify with limited labelled data. The proposed approach in this paper has provided a novel direction in which Dynamic Mode Decomposition based feature can be used in conjunction with a classifier for achieving competing results. As a future scope of this research, the shortcomings of the current proposed architecture can be solved and extended to fast paced applications like Intrusion Detection Systems [51], [52] and Data-driven solvers [53] where the data is limited and the classifier must be re-trained frequently on the go.

REFERENCES
[1] Nigam, K., McCallum, A., & Mitchell, T. (2006). Semi-supervised text classification using EM. Semi-Supervised Learning, 33-56.
[2] Cohen, I., & Cozman, F. G. (2006). Risks of semi-supervised learning: how unlabeled data can degrade performance of generative classifiers.
[3] Shrivastava, A., Pfister, T., Tuzel, O., Susskind, J., Wang, W., & Webb, R. (2017). Learning from simulated and unsupervised images through adversarial training. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (pp. 2107-2116).
[4] Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., ... & Bengio, Y. (2014). Generative adversarial nets. In Advances in neural information processing systems (pp. 2672-2680).
[5] Blum, A., & Mitchell, T. (1998, July). Combining labeled and unlabeled data with co-training. In Proceedings of the eleventh annual conference on Computational learning theory (pp. 92-100). ACM.
