A prototype-based model for set classification

A PROTOTYPE-BASED MODEL FOR SET CLASSIFICATION

M. Mohammadi
University of Groningen
The Netherlands
mohammadimathstar@gmail.com

S. Ghosh
Technical University of Eindhoven
The Netherlands
ghosh.sreejita@gmail.com

ABSTRACT

Classification of sets of inputs (e.g., images and texts) is an active area of research within both computer vision (CV) and natural language processing (NLP). A common way to represent a set of vectors is to model them as linear subspaces. In this contribution, we present a prototype-based approach for learning on the manifold formed from such linear subspaces, the Grassmann manifold. Our proposed method learns a set of subspace prototypes capturing the representative characteristics of classes and a set of relevance factors automating the selection of the dimensionality of the subspaces. This leads to a transparent classifier model which presents the computed impact of each input vector on its decision. Through experiments on benchmark image and text datasets, we have demonstrated the efficiency of our proposed classifier, compared to the transformer-based models in terms of not only performance and explainability but also computational resource requirements.

1 Introduction

Due to the success of linear subspaces in modelling variations present in data sets, they have extensively been applied in the machine learning (ML) community. In Computer Vision (CV), they have been used as a learning scheme for image-set classification, to model variation stemming from lightning conditions, pose and facial expression(s) within a set of images as illustrated in the works of [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11]. In Natural language processing (NLP), linear subspaces have been first used to model variations of topics within a large corpus of texts. Later, with the advancement in word-embedding models, they found new applications in modelling texts, such as phrases, sentences and paragraphs [12, 13]. This idea has recently been employed in document classification where a document is considered as a set of words[14, 13, 15]. These applications demonstrate the capability of subspace representation for set classification where the goal is to assign a label to a set of vectors.

Interestingly, it has also been shown that such linear subspaces form a manifold, called the Grassmann manifold. This has created a mathematical framework to study subspace-based representations and create models that are well-suited for learning on this manifold. However, prior works, mentioned above, suffer from two major drawbacks. First, their reliance on strategies such as the 'Nearest Neighbor' or 'kernel-trick' limits their applications to small data sets. Second, they require to be tuned with an appropriate value for the dimensionality d of subspaces. These limitations necessitate the development of more effective approaches for learning on the Grassmann manifold.

The Generalized Learning Vector Quantization (GLVQ) is a group of distance-based classifiers that uses a set of prototypes as typical representatives of the classes within the data space [16]. They classify a sample based on its proximity to the learned prototypes, following the Nearest Prototype Classification (NPC) strategy. To increase its flexibility and interpretability, several variants of GLVQ have been proposed with different adaptive distances introducing different levels of complexity and transparency such as, [17, 18]. This leads to the application of GLVQ in different areas where interpretability is a must [19, 20, 21, 22], from anthropocentric sectors like healthcare [23, 24] and education [25], to astronomy [26].

Recently, an extension of GLVQ called the Generalized Relevance Learning Grassmann Quantization (GRLGQ)[27], addresses the aforementioned challenges of subspace-based learning. This has been achieved by the introduction of a set of subspace prototypes, summarizing major characteristics of classes, and an adaptive distance, addressing the

[1]Preprint. Under review.

adverse effect of high-dimensionality d on the performance of the algorithm. This however comes by partially trading off GLVQ's inherent explainability. Specifically, it struggles to precisely quantify the effect of input vectors on the final prediction.

In this context, our objective is to develope a transparent model that elucidates the contributions of individual input elements (images or words) to the model's prediction, while still retaining the advantages of GRLGQ. The major contributions of this paper are as follows: (1) We extend the family of GLVQ classifiers by introducing a new adaptive distance measure. (2) We demonstrate how the new model explains its decision, thus addressing the aforementioned limitation of GRLGQ. Finally, through experiments on six benchmarks, (3) we illustrate its performance and interpretability concerning both image and text data.

This paper is organized as follows: In section 2, we provide the necessary background information, including a short introduction about the Grassmann manifold and the problem setting. In section 3, we first present our proposed model, and then we show how it provides details about its decisions. Section 4 contains the application of the new method to several benchmarks, including both images and texts. Finally, we conclude our work in section 6.

2 Background

2.1 Grassmann manifold

The extensive usage of linear subspaces in different fields has necessitated the study of mathematical structures driven by them, such as the Grassmann manifold.

Definition 2.1 *The Grassmann manifold $\mathcal{G}(D, \mathsf{d})$ consists of all d-dimensional subspaces in $\mathbb{R}^D$.*

A conventional representation of a point (i.e., subspace) on the Grassmann manifold involves using an orthonormal matrix $P \in \mathbb{R}^{D\times \mathsf{d}}$, where its columns serve as a basis for the subspace. Notably, this representation is not unique, meaning any orthonormal matrix spanning the subspace can be employed for representation. It has been shown that two orthonormal matrices P_1 and P_2 span the same subspace if and only if there exist rotations matrices $Q_1, Q_2 \in \mathcal{O}(\mathsf{d})$[1] such that $P_1Q_1 = P_2Q_2$.

In the context of performing a classification task on the Grassmann manifold, it is necessary to have a measure that assesses the (dis)similarity among points on the manifold. Various measures based on canonical correlation (and principal angles) have been proposed. Canonical correlation can be seen as a generalization of the cosine similarity measure to d-dimensional subspaces, allowing for the comparison of d pairs of vectors instead of just one.

Definition 2.2 *(Canonical correlation)* Let consider two subspaces $\mathcal{L}_1$ and $\mathcal{L}_2$ with the dimensionalities of d_1 and d_2, respectively. Then, d *principal angles*

$$0 \leq \theta_1 \leq \theta_2 \leq \cdots \leq \theta_{\mathsf{d}} \leq \frac{\pi}{2} \ , \quad \mathsf{d} = \min\{\mathsf{d}_1, \mathsf{d}_2\}$$

are defined recursively by the following conditions:

$$\cos\theta_i = \max_{\substack{\mathsf{u}'_i\in\mathcal{L}_1,\\ \mathsf{v}'_i\in\mathcal{L}_2}} \mathsf{u}'^T_i \mathsf{v}'_i = \mathsf{u}^T_i \mathsf{v}_i \ \text{, s.t.} \begin{cases} \|\mathsf{u}'_i\| = \|\mathsf{v}'_i\| = 1 \ , \\ \mathsf{u}'^T_i \mathsf{u}'_j = \mathsf{v}'^T_i \mathsf{v}'_j = 0 \ , \quad \forall j < i \ , \end{cases} \forall i = [1, 2, \cdots, \mathsf{d}] \tag{1}$$

Then, the cosine of the principal angles is referred to as the *canonical correlation*, and the elements of $\{\mathsf{u}_k\}_{k=1}^{\mathsf{d}}$ and $\{\mathsf{v}_k\}_{k=1}^{\mathsf{d}}$ are known as *principal vectors*[28].

In simpler terms, a pair of principal vectors, denoted by $(\mathsf{u}_i, \mathsf{v}_i)$, consists of unit vectors (from $\mathcal{L}_1$ and $\mathcal{L}_2$, respectively) with the smallest angle between them while being orthogonal to previously paired vectors. A common method for computing principal angles and vectors is through Singular Value Decomposition (SVD). Given two orthonormal matrices P_1 and P_2, spanning subspaces $\mathcal{L}_1$ and $\mathcal{L}_2$ respectively, the SVD of $P_1^TP_2$ provides us with pairs of principal vectors and their corresponding canonical correlations[29]:

$$P_1^T P_2 = Q_{1,2}(\cos\Theta)Q_{2,1}^T \ , \tag{2}$$

where $\cos\Theta$ is a diagonal matrix with canonical correlations, and rotation matrices fulfil the condition: $Q_{1,2}Q_{1,2}^T = Q_{1,2}^TQ_{1,2} = Q_{2,1}Q_{2,1}^T = Q_{2,1}^TQ_{2,1} = I$. Similarly, the principal vectors are given by:

$$U = P_1Q_{1,2} = [\mathsf{u}_1, \cdots, \mathsf{u}_{\mathsf{d}}], \qquad V = P_2Q_{2,1} = [\mathsf{v}_1, \cdots, \mathsf{v}_{\mathsf{d}}] \ . \tag{3}$$

[1] $\mathcal{O}(\mathsf{d})$: the group of $\mathsf{d}\times\mathsf{d}$ orthonormal matrices.

Based on canonical correlations, several distances have been proposed including the following:

Definition 2.3 *Let $\mathcal{L}_1,\ \mathcal{L}_2$ be on the manifold $\mathcal{G}(D, \mathsf{d})$. Then, the chordal distance is defined as:*

$$\text{(Chordal)} \quad d_c(\mathcal{L}_1, \mathcal{L}_2) = \|\sin\Theta\|_2 = \left(\sum_{i=1}^{\mathsf{d}} \sin^2\theta_i\right)^{1/2} = \left(\mathsf{d} - \sum_{i=1}^{\mathsf{d}} \cos^2\theta_i\right)^{1/2} . \tag{4}$$

2.2 Problem setting - set classification

In this contribution, we aim to construct a classifier capable of assigning a label y to any given set of vectors. In essence, we assume a data point as a collection of vectors, denoted by $\{\mathsf{x}_1, \mathsf{x}_2, \cdots, \mathsf{x}_n\}$, and the classifier's task is to assign an appropriate label to it. This scenario is common in both CV and NLP. For instance, let $g = (w_1, w_2, \cdots, w_n)$ denote a text of length n where w_i is the i^{th} word. Given a word embedding model, it assigns vector $\mathsf{x}_i \in \mathbb{R}^D$ to represent the corresponding word w_i. By stacking the word vectors, we can represent the text in a matrix form: $X = [\mathsf{x}_1, \mathsf{x}_2, \cdots, \mathsf{x}_n]$ .However, this representation is still incapable of addressing the challenge of comparing sets of variable lengths. A common way to address it is to compute their low-rank approximations. More precisely, we assume that X is a low-rank matrix represented by its SVD as:

$$X = PSR^T \ , \tag{5}$$

where S_i is a $(\mathsf{d} \times \mathsf{d})$ diagonal matrix containing singular values in descending order, and columns of P and R (of size $D \times \mathsf{d}$) are left and right singular vectors, respectively. The columns of P form the basis for the generated subspace by X's columns. This leads to the modified training set: $\{(P_i, y_i) \mid i = 1, \cdots, N\}$, where $P_i \in \mathbb{R}^{D\times\mathsf{d}}$ is an orthonormal matrix. This representation implies the assumption that all training examples live on the Grassmann manifold $\mathcal{G}(D, \mathsf{d})$. Therefore, we need a classifier that learns on this manifold.

3 Method

In this section, we present an overview of our novel contribution, which builds upon the foundations of GRLGQ via the incorporation of a distinct dissimilarity measure to enhance its explainability. Subsequently, we provide details about its optimization process, based on the Stochastic Gradient Descent (SGD) algorithm. Finally, we demonstrate how the resulting model offers insights into their decision-making processes, shedding light on the role of each input in its predictions.

3.1 Adaptive Chordal Distance and Subspace-based LVQ (AChorDS-LVQ)

Let $\{(P_i, y_i)\}_{i=1}^N$ be the training set on the Grassmann manifold $\mathcal{G}(D, \mathsf{d})$. Following the GLVQ algorithm, we use a set of labelled prototypes $\{(W_i, c_i)\}_{i=1}^p$ to model the distribution of classes on the data space (i.e. Grassmann manifold). Here, prototypes are orthonormal matrices on $\mathcal{G}(D, \mathsf{d})$, representing the corresponding classes by capturing their respective major characteristics. Learning takes place by minimization of the cost-function Eq. (6).

$$E(\mathcal{W}) = \sum_{i=1}^{n} \phi\left(\frac{d_c(P_i, W^+) - d_c(P_i, W^-)}{d_c(P_i, W^+) + d_c(P_i, W^-)}\right) \text{ where,} \tag{6}$$

$$d_c(P, W) = 1 - \sum_{k=1}^{\mathsf{d}} \lambda_k \cos\theta_k = 1 - \sum_{k=1}^{\mathsf{d}} \lambda_k \mathsf{u}_k^T \mathsf{v}_k, \quad \sum_{i=1}^{\mathsf{d}} \lambda_i = 1 \ , \tag{7}$$

where $d_c(P, W)$ is the adaptive distance between a prototype and a data point, $\{\lambda_i\}_{i=1}^{\mathsf{d}}$ are the relevance factors capturing the contribution of the canonical correlations towards differentiating the classes, and $\mathcal{W} = \{W_1, \cdots, W_p\}$ is the set of all prototypes. Note that W^+ and W^- are the closest prototypes to P_i with the same and different labels, respectively. Moreover, ϕ is a monotonically increasing function.[2] As the definition of d_c is motivated from the chordal distance, we refer to our newly introduced method hereafter as **A**daptive **Chor**dal **D**istance and **S**ubspace-based LVQ (AChorDS-LVQ for short).

During the training process, we optimize the cost-function and find the ideal locations of the prototypes on the manifold and the optimum values of the relevances. In the prediction phase, it follows the NPC strategy to assign the sample the

[2] Sigmoid and identity functions are two common choices for ϕ.

label of the prototype nearest to it:

$$\tilde{c}(X) = c(W_k) \ \ , \text{ s.t. } k = \underset{i=1,\cdots,p}{\arg\min}\, d(X, W_i) \ \ , \tag{8}$$

where $\tilde{c}(X)$ is the predicted label of X. Note that as the number of prototypes p is much smaller than the number of data N, NPC strategy provides an efficient model, both in terms of computations and memory cost during the test phase, in comparison to classifiers following the Nearest Neighbor strategy or those using kernel trick.

3.2 Optimization

To optimize the cost-function, we use SGD, requiring the computation of the first-order derivatives. Thus, in the following, we compute the (Euclidean) gradient of the cost-function Eq. (7). Thereafter, one can use this gradient to estimate the Riemannian gradient of the cost-function (details in [30]). Let P be a randomly selected training example. Together, Eq. (6) and (7) show that the explained cost value only depends on the winner prototypes $W^{\pm}$ and the relevance factors λ, i.e., only these terms need to be updated. Thus, we calculate the derivative of E_s with respect to these parameters. From the computation of $d_c(P, W^{\pm})$, we obtain principal vectors of P and W, denoted as U and V, respectively (see Eqs. (2) and (3)). As V is just a rotation of W (generating the same subspace), W and V represent the same point on the Grassmann manifold. This means that we can interchangeably compute the derivative with respect to W and V. Thus, instead of W, we compute the derivative of the cost-function with respect to the principal vectors as follows:

$$\frac{\partial E_s}{\partial V^{\pm}} = \frac{\partial E_s}{\partial \mu} \frac{\partial \mu}{\partial d_c(P, V^{\pm})} \frac{\partial d_c(P, V^{\pm})}{\partial V^{\pm}} \ \ , \text{ where } \frac{\partial E_s}{\partial \mu} = \begin{cases} \mu(1-\mu) & \text{if } \phi \text{ is sigmoid} \\ 1 & \text{if } \phi \text{ is identity} \end{cases} \ \ , \tag{9}$$

$$\frac{\partial \mu}{\partial d_c(P, V^{\pm})} = \pm \frac{2 d_c(P, V^{\mp})}{\left(d_c(P, V^{+}) + d_c(P, V^{-})\right)^2} \ \ , \tag{10}$$

$$\frac{\partial d_c(P, V)}{\partial \mathsf{v}_k} \overset{\text{Eq.(7)}}{=} -\lambda_k \mathsf{u}_k \ \ . \tag{11}$$

This results in the following gradient:

$$\frac{\partial E_s}{\partial V^{\pm}} = \pm \frac{2 d_c(P, W^{\mp})}{\left(d_c(P, W^{+}) + d_c(P, W^{-})\right)^2} (UG)^{\pm} \ \ , \tag{12}$$

where G is a $(\mathsf{d} \times \mathsf{d})$-diagonal matrix containing relevance factors. Similarly, we can compute the derivative of the cost-function with respect to the relevance factors:

$$\frac{\partial E_s}{\partial \lambda_k} = \frac{\partial E_s}{\partial \mu_i} \left[\frac{\partial \mu_i}{\partial d_c(P, V^{+})} \frac{\partial d_c(P, V^{+})}{\partial \lambda_k} + \frac{\partial \mu_i}{\partial d_c(P, V^{-})} \frac{\partial d_c(P, V^{-})}{\partial \lambda_k} \right], \ \text{ where } \frac{\partial d_c^{\lambda}(P, V)}{\partial \lambda_k} = \mathsf{u}_k^T \mathsf{v}_k \tag{13}$$

Using the computed derivative in Eq. 12 and 13, we define the following updating rules:

$$W^{\pm} \leftarrow V^{\pm} - \eta_{W^{\pm}} \frac{\partial E_s}{\partial V^{\pm}} \tag{14}$$

$$\lambda \leftarrow \lambda - \eta_{\lambda} \frac{\partial E_s}{\partial \lambda} \tag{15}$$

where $\eta_{W^{\pm}}$ and η_{λ} are their respective learning rates. We set $\eta_{\lambda} \ll \eta_{W^{\pm}}$ as relevance factors will be updated (much) more often than prototypes. Note that after any update, we orthonormalize $W^{\pm}$[3] and normalize $\{\lambda_i\}_{i=1}^{\mathsf{d}}$.

3.3 Explainability

With the increasing real-world applications of ML models, performance metrics alone do not suffice for trustworthiness, highlighting the critical need for model interpretability and explainability [20, 21, 31]. Even though model explainability is demanded in anthropocentric applications, it is important in other sectors as well to identify the sources of uncertainties and noise in the model training and deployment pipelines, and strategize appropriately to address these [31], minimize algorithmic bias and ensure fair and responsible use of AI[19, 20, 32]. Local interpretable model-agnostic explanations (LIME) [33] and (SHapley Additive exPlanations) (SHAP)[34] are among the most widely used XAI tools which provide post-hoc explanations of decisions by even complex black-box models, such as any model with a deep neural

[3] As we need orthonormalized matrices for computing canonical correlations.

network (DNN) architecture [32]. [32] also demonstrate the susceptibility of these XAI techniques, which function by (i) perturbing the region close to a sample-of-interest and (ii) applying simpler (often linear) surrogate models, to being fooled by adversarial classifiers, thereby misleading model designers, end-users and stakeholders resulting in severe ramifications [35]. Intrinsically interpretable ML models (e.g., linear and logistic regressors, decision trees (DT)s, and NPCs) circumvent these risks since they provide direct access to their respective decision-making logic. In this section, we elucidate what makes AChorDS-LVQ intrinsically interpretable.

AChorDS-LVQ performs distance-based classification following the NPC strategy. Thus, one can measure the effect of different input vectors on the model's decisions by evaluating their impact on the distance measure. Following this, here we show how the same applies to the newly introduced distance (Eq. (7)), when computing the influence of each input on it, and thereby on the prediction. Let X be a matrix containing a set of vectors. Using the SVD, one can derive its corresponding d-dimensional subspace, i.e. P: $X = PSR^T$. By computing its distance to a prototype W, we obtain its principal vectors $U = PQ_P = XM$, where $M_{n\times\mathsf{d}} = RS^{-1}Q_P$. Let m_i denote i^{th} columns of M. Then, we have $\mathsf{u}_i = X\mathsf{m}_i$, where elements m_i capture the contribution of input vectors (i.e. columns of X) on the construction of the i^{th} principal vectors. Additionally, we define x_j as the j^{th} row of X which is the j^{th} coordinate of the input vectors, resulting in:

$$u_i^j = \mathrm{x}_j \mathsf{m}_i = \sum_{k=1}^{n} x_j^k m_i^k, \tag{16}$$

where u_i^j is j^{th} coordinate of i^{th} principal vector u_i. In Eq. (17) the right-hand term within

$$d_c(P, W) = 1 - \sum_{i=1}^{\mathsf{d}} \lambda_i \mathsf{u}_i^T \mathsf{v}_i = 1 - \sum_{i=1}^{\mathsf{d}} \sum_{j=1}^{D} \lambda_i u_i^j v_i^j = 1 - \sum_{k=1}^{n} \left[\sum_{j=1}^{D} \left(\sum_{i=1}^{\mathsf{d}} \lambda_i m_i^k v_i^j \right) x_j^k \right] \tag{17}$$

parenthesis represents the impact of the j^{th} element (pixel) of the k^{th} input vector on the computed distance. Consequently, this allows for these elements to be ranked according to their influence on this distance, and enabling more precise estimation of possible sources of uncertainty and noise, among others. The terms within the square brackets in Eq. (17), i.e., the cumulative sum of elements of an input vector $\vec{x}_k$ (for $k = 1, \cdots, n$), reflect the overall impact of this input vector on the computed distance between the input set X and a prototype.

To decide about an input set X, the model computes its distance to the nearest (winner) prototypes $W^{\pm}$. Thus, as formulated in the numerator of the cost-function Eq. (6), we use the difference between these distances to find the most influential elements for the input set's prediction:

$$d(P, W^-) - d(P, W^+) = \sum_{k=1}^{n} \left[\sum_{j=1}^{D} \left(\sum_{i=1}^{d} \lambda_i m_i^k (v_i^{j+} - v_i^{j-}) \right) x_j^k \right] , \tag{18}$$

Thus, Eq. (18) illustrates how AChorDS-LVQ can define with element-level precision, the impact of the input vectors on the final prediction. Note that the larger the computed value in Eq. (18), the better is the generalization ability of AChorDS-LVQ, a trait similar to its predecessors from the GRLVQ family, since they all follow the large margin principle as proven in [17].

3.4 Complexity

Given a training example, we should first compute its distance to prototypes. This requires prototypes to be orthonormal matrices. To fulfil it, we need to orthonormalize winner prototypes after their update, using (14),. This needs the application of the SVD. Additionally, we use the SVD for the computation of distances. This demonstrates that the SVD plays a central role in the AChorDS-LVQ training process. While the computational complexity of SVD for a $(D \times \mathsf{d})$- matrix is $O(D\mathsf{d}^2)$, several strategies were proposed to reduce it to $O(D\mathsf{d} \log \mathsf{d})$[36]. However, in the presence of a big dataset (large N) with limited amount of computational resource, one has to select a small dimensionality d. Note that the AChorDS-LVQ only requires access to p prototypes during inference. In contrast, methodologies like DCC, GDA, GGDA, PML, AMLS(L) and G(G)PLCR need access to all N training examples due to their dependency on strategies 'Nearest Neighbor' or 'kernel tricks'.

4 Experiment

We demonstrate the efficiency of AChorDS-LVQ on both image and text data, for image set classification and variable-length document classification tasks. The first task further comprises of face recognition, object detection and activity recognition. For the following experiments, we set the learning rates to $\eta_{W^\pm} = 0.1$, and $\eta_\lambda = 10^{-5}$.

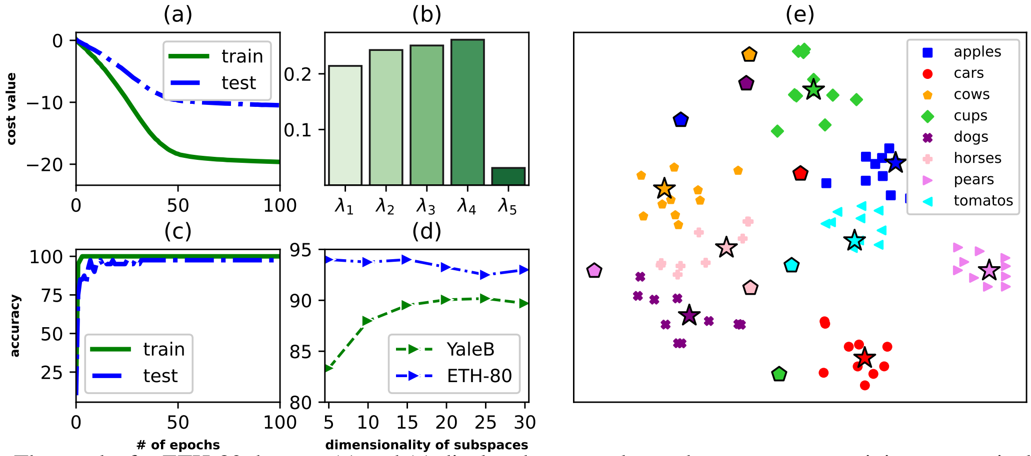

Figure 1: The results for ETH-80 dataset: (a) and (c) display the cost value and accuracy over training, respectively; (b) shows the relevance factors, (d) the performance for different dimensionalities d. Finally (e) presents the visualization of data points with prototypes using t-SNE. Note that the pentagons represent the initial locations of the prototypes while stars denote the learned prototypes.

4.1 Image set classification

In the following, we compare our method with previous works across three datasets: (a) ETH-80 for object recognition, (b) Extended YaleB for face recognition, and (c) UCF for activity recognition. To ensure the fairness and reproducibility of our comparison, we followed the preprocessing steps explained in [10]. To have a fair comparison to GRLGQ, we set the dimensionality d to the ones specified in [27], which are 5, 25 and 22 for the ETH-80, YaleB and UCF datasets, respectively. Additionally, we assume ϕ to be an identity function.

To assess the success of the optimization process, Fig. 1 displays the training process and the resulting model for the ETH-80 dataset. Fig. 1 (a) and (c) present the cost value and accuracy across various epochs, revealing a consistent decrease in the cost value over time. Additionally, to see the trained model, we visualize the position of prototypes within the data space using t-distributed Stochastic Neighbor Embedding (t-SNE). We employed t-SNE on the trained model, utilizing a distance matrix derived from the learned distance measure. While pentagons in Fig. 1-(e) represent randomly initialized prototypes, stars show the learned prototypes (after training). This shows the optimization's success in positioning prototypes closer to class members while maintaining a larger distance from members of other classes.

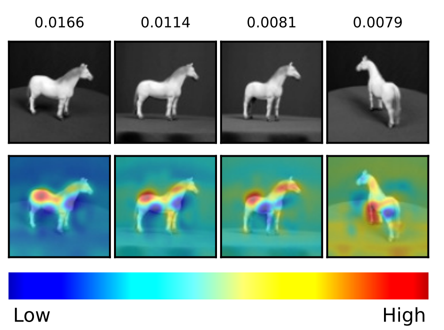

Figure 2: Explainability for ETH-80 dataset:(a) four most important images (impact score above respective image) on prediction; and (b) heatmaps depicting the impact of their constituent pixels.

Table 1: Results (mean accuracy with standard deviations)) of the image-set classification. Methods with (*) are reported from [10] (first two columns.) and [11] (last column).

Methods	ETH-80	Ex. Yale B	UCF
GDA*	$91.00(\pm 2.13)$	$82.80(\pm 0.96)$	70.00
GGDA*	$92.50(\pm 1.16)$	$79.11(\pm 0.92)$	-
PML*	$93.75(\pm 1.80)$	$77.08(\pm 0.18)$	72.67
AMLSL*	-	-	74.00
GGPLCR*	$\mathbf{96.75}(\pm 1.30)$	$88.57(\pm 0.48)$	-
DCC	$90.75(\pm 3.34)$	$77.08(\pm 3.25)$	59.33
MSM	$77.00(\pm 6.21)$	$70.00(\pm 4.12)$	48.67
AHISD	$73.00(\pm 7.24)$	$55.95(\pm 5.54)$	53.33
CHISD	$75.50(\pm 4.38)$	$62.38(\pm 4.97)$	52.00
GRLGQ	$94.50(\pm 1.87)$	$90.12(\pm 2.25)$	**80.00**
AChorDS-LVQ	$94.00(\pm 2.00)$	$\mathbf{90.18}(\pm 1.64)$	79.33

Performance: Table 1 indicates that among the different methods we applied on the aforementioned datasets, GGPLCR, AChorDS-LVQ and GRLGQ achieve the best performances. Among them, GGPLCR has two drawbacks. Firstly, its dependency on the 'Nearest Neighbor' prediction strategy directly links its complexity (number of parameters) to the size of the training set. This limits its applicability to small-size data sets. Secondly, it has been shown in [10] that GGPLCR needs to be tuned with an optimal value for the dimensionality of the subspace d. In contrast to GGPLCR, GRLGQ and AChorDS-LVQ overcome these limitations. For instance, in the ETH-80 experiment, due to the usage of the NPC strategy for prediction, their complexities are independent of data size (one-fourth of GGPLCR). Additionally, the relevance factors minimize the impact of redundant and noisy dimensions, resulting in a more robust classifier (Fig. 2-b). While the deviation from the optimal value ($\mathsf{d} = 3$) in GGPLCR results in its significant performance deterioration ($< 90\%$)[10], GRLGQ and AChorDS-LVQ maintain stable higher accuracies ($> 93\%$) for a wide range of dimensionalities (Fig. 2-d).

Explainability: Although GRLGQ and AChorDS-LVQ yield similar outcomes with equivalent complexity levels, their levels of explainability significantly differ. While GRLGQ offers a degree of transparency by highlighting significant pixels and images for each principal angle individually, it fails to specify precise insights into the influence of pixels and images on its final decisions. Conversely, AChorDS-LVQ allows evaluation of the impact of any image on the model's decisions, by computing its influence on distances. Following Eq. (18), we can sort images (from an image set) based on their impact on the model's prediction. The first row in Fig. 2 shows four images (of a horse) with the highest impact. Through the computation of inner parentheses in Eq. (18), we can find the most influential pixels within each image. As seen in the second row of Fig. 2, the most influential pixels correspond to those representing the mane of the horse, which is a characteristic that differentiates a horse from a dog (the class represented by the closest incorrect prototype). Similarly, the pixels representing the background of the horses have lower relevance than pixels representing the main body, and specifically the characteristic features of the horse. This insight is useful not only to verify the model's decision, but also to strategize camera placement for optimal object recognition.

4.2 Document classification

To evaluate the performance of the proposed approach for document classification, we applied it to three document classification tasks across a range of document lengths: (a) two datasets with shorter texts, i.e. Reuters-8[37] and Hyperpartisan[38], (b) Arxiv-4[39] which is a dataset with long texts (with an average length of 6000 words). In our experiments, we use the GloVe and the Word2Vec models for generating word embeddings, due to their flexibility across varying document lengths. To prevent the effect of commonly used words, we remove the stop words from texts as the preprocessing step. We use the sigmoid function for ϕ (in Eq. (9)). Additionally, we set $\mathsf{d} = 20$ for the first two experiments and $\mathsf{d} = 30$ for the last one (with longer texts).

To assess the performance of our proposed approach, we conducted a comparison against established models. We compared the performance of our proposed model against those of state-of-the-art models, including the traditional yet widely-used SVM, subspace-based models: MSM[13] and GRLGQ, and two transformer-based models: BERT and RoBERTa. For models that are not transformer-based, word embedding models, such as Word2Vec and GloVe have been used for generating vector representations of texts, as specified in Tab. 2. To further explore the influence of different embedding methods on our model's efficacy, we utilized Word2Vec, GloVe, and BERT embeddings. Finally, given the computational cost of transformer-based models, we compared two common practices for their fine-tuning: updating all parameters as well as LoRA [40], a method to reduce training time.

Table 2: The performance of different classifiers on the document classification task.

Methods	Word embedding	N_{params} (training)	Reuters-8	Hyper-partisan	Arxiv-4
MSM[13]	W2V			92.01	-
SVM	GloVe	457k	96.03	70.77	93.64
GRLGQ	W2V	48k	97.67	90.77	96.06
	GloVe	48k	97.93 (± 0.09)	91.28 (± 0.73)	96.65 (± 0.04)
	BERT	184k	**98.17**	90.77	-
AChorDS-LVQ	W2V	48k	97.94	90.77	96.06
	GloVe	48k	97.95 (± 0.07)	**91.80** (± 0.73)	**96.74** (± 0.17)
	BERT	184k	**98.17**	90.77	-
BERT$^{\text{FULL}}$		110 M	98.04	90.77	96.35
BERT$^{\text{LoRA}}$		300k	97.94	90.77	-
RoBERTa$^{\text{FULL}}$		125M	98.08	90.77	95.45
RoBERTa$^{\text{LoRA}}$		892k	98.08	90.77	-
Longformer		-	-	-	95.12
CNN+*+Agg[39]					
• Rand		-	-	-	94.02
• LSTM		-	-	-	94.25
• RAM					94.73
Glimpse[41]					94.18

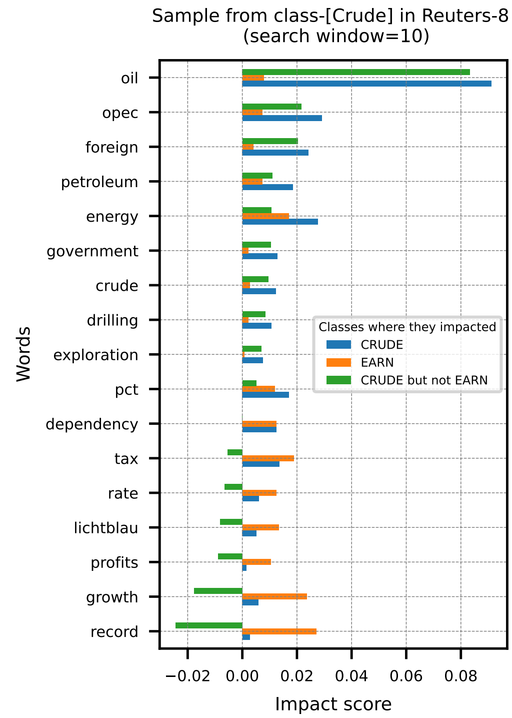

Figure 3: Words with the highest impact on the prediction of a sample from *Crude* class of the Reuters-8 dataset.

Performance: The results for the three text-based datasets are presented under their respective columns in Tab. 2 which demonstrates the competitive performance of AChorDS-LVQ. For the Arxiv-4 dataset containing longer texts, we additionally compared the performance of AChorDS-LVQ against the algorithms proposed in [42, 39, 41] for handling long texts. Specifically, the combination of GloVe embeddings and AChorDS-LVQ demonstrates comparable performance to transformer-based models for the Reuters-8 dataset and outperforms them for the Hyperpartisan and Arxiv-4 datasets. This is achieved with significantly fewer parameters (48k) compared to BERT (with 110M for full fine-tuning and 300k with LoRA) and RoBERTa (with 125M for full fine-tuning and 892k with LoRA). Given the large number of parameters in transformer-based models, we needed GPU for accelerated computations during training. However for training AChorDS-LVQ CPU was enough (Appendix A. 3), which makes it compute resource efficient.

Explainability: Following our discussion in Subsec. 3.3, and similar to in Subsec. 4.1, we can use Eq. (18) to determine the impact of each word on the model's decisions, and rank them accordingly. Fig. 3 provides the influential words for a text from *'crude'* class. The blue and orange bars corresponding to each term indicate the extent of impact of these words from the selected document, in driving the model's decision towards class 'crude', representing the *crude oil* market, and class 'earn', representing the topic of earning through investments. The green bar represents the net impact of this term in identifying 'crude' from 'earn' (Eq. (18)); the negative direction of the green bar indicate that the corresponding terms are more representative of the closest incorrect class (here 'earn'). This also shows that terms related to the 'crude oil' market, such as 'oil', 'opec', 'crude', 'energy', 'petroleum' and 'Premier Composite Technologies (PCT)', a leading global supplier, are correctly identified from the document by our model, as drivers of its decisions about this document. Further, the terms whose corresponding green bar is obscure indicate potential sources of uncertainties for the model (e.g., 'dependency'), as these are equally relevant for both these classes and hence not helpful in differentiating them. Remarkably, despite some words associated with the oil market, such as 'crude', 'opec' and 'petroleum', appearing only once or twice, the model accurately identifies them as important, reflecting their significance in capturing key class characteristics for identifying the topic 'crude oil'.[4] This illustrates that the learned prototypes of our model have successfully encapsulated the general topic characteristics and keywords of each class. Appendix A. 2 contain further examples of such explanations for other samples from 'Earn' class, and samples from the Hyperpartisan and Arxiv-4 datasets.

[4] Fig. A. 5 in appendix contains the full sample text corresponding to Fig. A. 5.

5 Limitation

One of the limitations of AChorDS-LVQ is its complexity, as explained in Subsec. 3.4. This also indicates that under computational resource constraints, when we need to restrict its dimensionality, its performance might be impacted.

6 Conclusion

In this contribution, we extend the GLVQ family to AChorDS-LVQ by introducing a new distance measure, chordal distance. AChorDS-LVQ assumes that data lies on the Grassmann manifold, and it learns a set of subspace prototypes, encoding the characteristics of classes, and relevance factors, capturing the relevance of canonical correlation for distinguishing the classes. In contrast to GRLGQ, the proposed classifier provides detailed information about the effect of different inputs on its final decisions. This makes our model intrinsically interpretable, which means that, unlike the model-agnostic explainability techniques it provides direct access to its internal decision-making mechanism. We applied the new approach to six benchmark data sets, including both images and texts. For image-set classification tasks, even though the AChorDS-LVQ achieves similar performance to GRLGQ, unlike its predecessor, it provides crucial insights into its decision-making, by highlighting the images and corresponding pixels of these images, which had the most influence, on its decision. For textual data, we demonstrated that the performance of AChorDS-LVQ in document classification task (with documents of varying lengths), is comparable (or better) than that of the transformer-based models, i.e. BERT and RoBERTa. Additionally, we show that this new approach is more compute and resource-efficient since it follows the NPC strategy. The limited computing resource requirement of our proposed methodology also makes it an energy-efficient, environment-friendly, and affordable choice.

In both image-set and document classification tasks, the newly proposed model enables detection of sources of uncertainties and noise affecting its decision-making, and estimation of the strength of this uncertainty, which makes it a robust and explainable model. These characteristics allow the end-users of this model to verify if its decision-making mechanism is sensible and fair, thereby making it trustworthy for anthropocentric applications. However, these very explanations provided by AChorDS-LVQ, similar to explanations extracted from any XAI technique, could be also misused to bypass the rules of an existing fair system.

Acknowledgment

This research is part of the research project *EVICT* (`www.eviction.eu`), funded the the European Union's ERC Research Grant under grant agreement No 949316, which supported M.M. S.G. has been supported by ITEA4 DAIsy 21016 (`https://itea4.org/project/daisy.html`).

References

[1] Osamu Yamaguchi, Kazuhiro Fukui, and K-i Maeda. Face recognition using temporal image sequence. In *Proceedings Third IEEE International Conference on Automatic Face and Gesture Recognition*, pages 318–323. IEEE, 1998.

[2] Kazuhiro Fukui and Osamu Yamaguchi. Face recognition using multi-viewpoint patterns for robot vision. In *Robotics Research. The Eleventh International Symposium*, pages 192–201. Springer, 2005.

[3] Tae-Kyun Kim, Ognjen Arandjelović, and Roberto Cipolla. Boosted manifold principal angles for image set-based recognition. *Pattern Recognition*, 40(9):2475–2484, 2007.

[4] Tae-Kyun Kim, Josef Kittler, and Roberto Cipolla. Discriminative learning and recognition of image set classes using canonical correlations. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(6):1005–1018, 2007.

[5] Ruiping Wang, Shiguang Shan, Xilin Chen, and Wen Gao. Manifold-manifold distance with application to face recognition based on image set. In *2008 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8. IEEE, 2008.

[6] Ruiping Wang and Xilin Chen. Manifold discriminant analysis. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 429–436. IEEE, 2009.

[7] Jihun Hamm and Daniel D Lee. Grassmann discriminant analysis: a unifying view on subspace-based learning. In *Proceedings of the 25th international conference on Machine learning*, pages 376–383, 2008.

[8] Mehrtash T Harandi, Conrad Sanderson, Sareh Shirazi, and Brian C Lovell. Graph embedding discriminant analysis on grassmannian manifolds for improved image set matching. In *CVPR 2011*, pages 2705–2712. IEEE, 2011.

[9] Zhiwu Huang, Ruiping Wang, Shiguang Shan, and Xilin Chen. Projection metric learning on grassmann manifold with application to video based face recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 140–149, 2015.

[10] Dong Wei, Xiaobo Shen, Quansen Sun, Xizhan Gao, and Wenzhu Yan. Prototype learning and collaborative representation using grassmann manifolds for image set classification. *Pattern Recognition*, 100:107123, 2020.

[11] Naoya Sogi, Lincon S Souza, Bernardo B Gatto, and Kazuhiro Fukui. Metric learning with a-based scalar product for image-set recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 850–851, 2020.

[12] Jiaqi Mu, Suma Bhat, and Pramod Viswanath. Representing sentences as low-rank subspaces. *arXiv preprint arXiv:1704.05358*, 2017.

[13] Erica K Shimomoto, François Portet, and Kazuhiro Fukui. Text classification based on the word subspace representation. *Pattern Analysis and Applications*, 24(3):1075–1093, 2021.

[14] Hongyu Gong, Tarek Sakakini, Suma Bhat, and Jinjun Xiong. Document similarity for texts of varying lengths via hidden topics. *arXiv preprint arXiv:1903.10675*, 2019.

[15] Erica K Shimomoto, Edison Marrese-Taylor, Hiroya Takamura, Ichiro Kobayashi, and Yusuke Miyao. Subspace representation for text classification with limited training data. In *Proceedings of the 29th Annual Conference of the Language Processing Society of Japan*, 2023.

[16] Atsushi Sato and Keiji Yamada. Generalized learning vector quantization. *Advances in neural information processing systems*, 8, 1995.

[17] B. Hammer, M. Strickert, and T. Villmann. On the generalization ability of GRLVQ networks. *Neural Processing Letters*, 21(2):109–120, 2005.

[18] Petra Schneider, Michael Biehl, and Barbara Hammer. Adaptive relevance matrices in learning vector quantization. *Neural Comput.*, 21(12):3532–3561, dec 2009.

[19] Andreas Holzinger, Chris Biemann, Constantinos S Pattichis, and Douglas B Kell. What do we need to build explainable ai systems for the medical domain? *arXiv preprint arXiv:1712.09923*, 2017.

[20] Alejandro Barredo Arrieta, Natalia Díaz-Rodríguez, Javier Del Ser, Adrien Bennetot, Siham Tabik, Alberto Barbado, Salvador García, Sergio Gil-López, Daniel Molina, Richard Benjamins, et al. Explainable artificial intelligence (xai): Concepts, taxonomies, opportunities and challenges toward responsible ai. *Information Fusion*, 58:82–115, 2020.

[21] Finale Doshi-Velez and Been Kim. Towards a rigorous science of interpretable machine learning. *arXiv preprint arXiv:1702.08608*, 2017.

[22] Adrien Bibal and Benoît Frénay. Interpretability of machine learning models and representations: an introduction. In M.Verleysen, editor, *ESANN*, pages 77–81, 2016.

[23] Sreejita Ghosh, Peter Tino, and Kerstin Bunte. Visualisation and knowledge discovery from interpretable models. In *2020 International Joint Conference on Neural Networks (IJCNN)*, pages 1–8. IEEE, 2020.

[24] Sreejita Ghosh, Elizabeth S Baranowski, Michael Biehl, Wiebke Arlt, Peter Tino, and Kerstin Bunte. Interpretable models capable of handling systematic missingness in imbalanced classes and heterogeneous datasets. *arXiv preprint arXiv:2206.02056*, 2022.

[25] I Gusti Made Wahyu Krisna Widiantara, Kadek Yota Ernanda Aryanto, and I Made Gede Sunarya. Application of the learning vector quantization algorithm for classification of students with the potential to drop out. *Brilliance: Research of Artificial Intelligence*, 3(2):262–269, 2023.

[26] Mohammad Mohammadi, Jarvin Mutatiina, Teymoor Saifollahi, and Kerstin Bunte. Detection of extragalactic ultra-compact dwarfs and globular clusters using explainable ai techniques. *Astronomy and Computing*, 39:100555, 2022.

[27] Mohammad Mohammadi, Mohammad Babai, and Michael.H.F. Wilkinson. Generalized relevance learning grassmann quantization. *arXiv preprint arXiv:2403.09183*, 2014.

[28] Jiayao Zhang, Guangxu Zhu, Robert W Heath Jr, and Kaibin Huang. Grassmannian learning: Embedding geometry awareness in shallow and deep learning. *arXiv preprint arXiv:1808.02229*, 2018.

[29] Ake Björck and Gene H Golub. Numerical methods for computing angles between linear subspaces. *Mathematics of computation*, 27(123):579–594, 1973.

[30] Alan Edelman, Tomás A Arias, and Steven T Smith. The geometry of algorithms with orthogonality constraints. *SIAM journal on Matrix Analysis and Applications*, 20(2):303–353, 1998.

[31] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. Model-agnostic interpretability of machine learning. *arXiv preprint arXiv:1606.05386*, 2016.

[32] Dylan Slack, Sophie Hilgard, Emily Jia, Sameer Singh, and Himabindu Lakkaraju. Fooling lime and shap: Adversarial attacks on post hoc explanation methods. In *Proceedings of the AAAI/ACM Conference on AI, Ethics, and Society*, pages 180–186, 2020.

[33] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. " why should i trust you?" explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, pages 1135–1144, 2016.

[34] Scott M Lundberg and Su-In Lee. A unified approach to interpreting model predictions. *Advances in neural information processing systems*, 30, 2017.

[35] Joaquín Borrego-Díaz and Juan Galán-Páez. Explainable artificial intelligence in data science: From foundational issues towards socio-technical considerations. *Minds and Machines*, 32(3):485–531, 2022.

[36] Gene H Golub and Charles F Van Loan. *Matrix computations*. JHU press, 2013.

[37] Ana Margarida de Jesus Cardoso Cachopo et al. Improving methods for single-label text categorization. *Ph.D. dissertation, Instituto Superior Técnico, Portugal*, 2007.

[38] Johannes Kiesel, Maria Mestre, Rishabh Shukla, Emmanuel Vincent, Payam Adineh, David Corney, Benno Stein, and Martin Potthast. SemEval-2019 task 4: Hyperpartisan news detection. In Jonathan May, Ekaterina Shutova, Aurelie Herbelot, Xiaodan Zhu, Marianna Apidianaki, and Saif M. Mohammad, editors, *Proceedings of the 13th International Workshop on Semantic Evaluation*, pages 829–839, Minneapolis, Minnesota, USA, June 2019. Association for Computational Linguistics.

[39] Liu Liu, Kaile Liu, Zhenghai Cong, Jiali Zhao, Yefei Ji, and Jun He. Long length document classification by local convolutional feature aggregation. *Algorithms*, 11(8):109, 2018.

[40] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*, 2021.

[41] Jun He, Liqun Wang, Liu Liu, Jiao Feng, and Hao Wu. Long document classification from local word glimpses via recurrent attention learning. *IEEE Access*, 7:40707–40718, 2019.

[42] Iz Beltagy, Matthew E Peters, and Arman Cohan. Longformer: The long-document transformer. *arXiv preprint arXiv:2004.05150*, 2020.

[43] Bastian Leibe and Bernt Schiele. Analyzing appearance and contour based methods for object categorization. In *2003 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2003. Proceedings.*, volume 2, pages II–409. IEEE, 2003.

[44] Mikel D Rodriguez, Javed Ahmed, and Mubarak Shah. Action mach a spatio-temporal maximum average correlation height filter for action recognition. In *2008 IEEE conference on computer vision and pattern recognition*, pages 1–8. IEEE, 2008.

A. 1 Appendix: Datasets

A. 1.1 Object Recognition: ETH-80

The ETH-80 dataset, comprising 8 object categories, is widely utilized for object classification [43]. Each category contains 10 image sets, with 41 images per set captured from various viewpoints. Following [10], images were converted to grayscale and resized to 20×20. For evaluation, data was split into training and testing sets, with five randomly chosen image sets per object for training. This process was repeated 10 times and then the mean and the standard deviations of accuracies.

A. 1.2 Face Recognition: Extended Yale Face Database B (YaleB)

The Extended Yale Face Database B (YaleB) encompasses 16,128 images featuring 28 individuals captured across varied poses and illumination conditions (with 9 poses and 64 illumination conditions per person). Following the preprocessing steps outlined in [10], we organized the images (of size 20×20) into 9 sets per subject based on their poses. To create training and testing sets, three image sets were randomly chosen for training (per individual), with the remainder reserved for testing. This process was iterated ten times, after which the mean accuracy and standard deviation were computed.

A. 1.3 Activity recognition: UCF

The UCF Sports dataset consists of 150 videos of ten actions. They have been captured in different scenes and from different viewpoints [44]. To prepare videos for training, we followed the same preprocessing steps as [11, 27]. Similar to [11, 27], we used a leave-one-out cross-validation scheme (LOOCV) to report the performance for the UCF data set.

A. 1.4 Document Classification

Reuters-8 is a preprocessed version of the Reuters-21578 and consists of short texts categorized into eight classes: acq, crude, earn, grain, interest, money-fx, ship, and trade. It contains 5,485 training and 2,189 test examples. The second dataset is Hyperpartisan[38] which is labelled based on whether an article is hyperpartisan or not. It consists of 516 documents for training and 64 and 65 documents as validation and test sets, respectively. The third dataset is the Arxiv-4 dataset [39], comprising 12,195 research papers from the arXiv repository, classified into four distinct categories: cs.IT, cs.Ne, math.AC, and math.GR. In contrast to the Reuters-8 and the Hyperpartisan datasets, the Arxiv-4 dataset contains significantly longer documents, with an average length exceeding 6,000 words. Following [39, 41], we use 80% of documents for training and the remaining for testing.

For our proposed model and the one conceptually similar to it, we have also provided the standard deviations over three random initializations. Since running transformer-based models hav been too compute-resource intensive, we have reported their performance for only one run at this moment.

A. 2 Appendix: Explainability

As it has been shown in section 3.3, we can investigate the role of any input vector on the model's final prediction. Following Eq. (18), we first investigate the role of pixels and images on the prediction for ETH-80. Then, for the Reuters-8 and Hyperpartisan datasets, we study the most influential words for two exemplar texts.

A. 2.1 ETH-80

Given an image set of a horse, the model could correctly classify it. Fig. A. 4a shows the sorted images based on their impact where their impact is above figures. Fig. A. 4b displays heatmaps capturing the influence of pixels within each image.

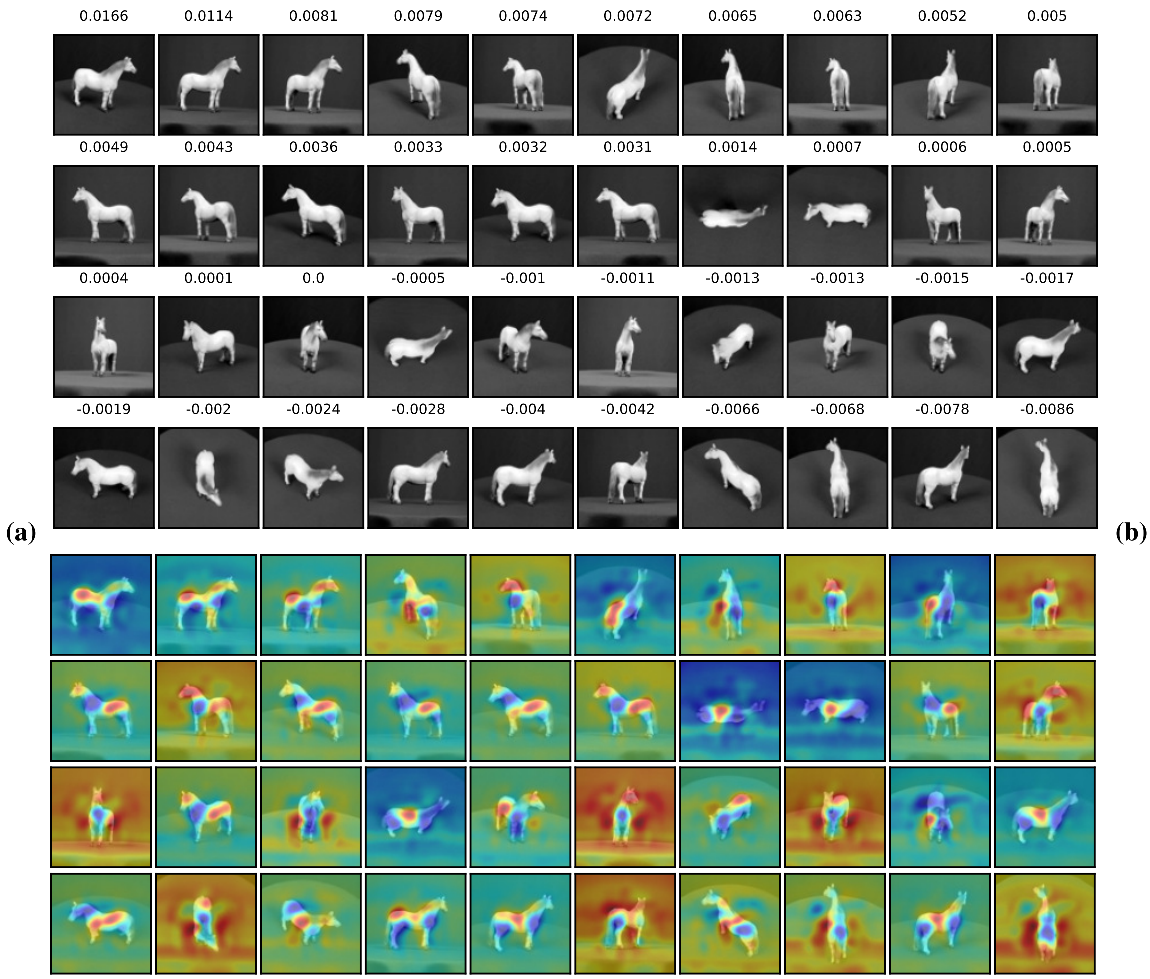

Figure A. 4: a) Sorted images based on their impact on the model decision. b) Highlighted pixels for sorted images.

A. 2.2 Reuters-8

Fig. A. 5 displays a text from the ‘crude’ class.

> US oil dependency seen rising to record level the united states dependency on foreign oil sources may reach record levels by the mid s according to john h lichtblau president of petroleum industry research associates lichtblau speaking at an alternative energy conference here said the US may depend on foreign suppliers for as much as pct of its oil by surpasssing the previous high level of pct in the long term growth in dependency on foreign oil is inevitable lichtblau said as much as pct of US oil imports in could come from opec nations he said lichtblau said the US depended on foreign suppliers for pct of its oil in and predicted that would increase to pct by however the rate of this growth can be affected positively or negatively through government action or inaction lichtblau said he said that one of the government s negative actions is the maintenance of the windfall profits tax which acts as a disincentive to developing existing fields and reduces cash flow for oil exploration lichtblau called for the adoption of an international floor price for crude oil to help stabilize world oil prices an international floor price adopted by all or most industrial countries would clearly be a much more effective measure and would be much less distortive for the US than if we imposed it alone lichtblau said development of alternate energy sources such as synthetic fuels as well as increased development in alaska could lessen US dependency on foreign oil lichtblau said a potential for alternative supplies could limit the willingness of opec nations to raise oil prices he said lichtblau also called for the federal government to offer tax abatements for oil drilling to fill the strategic petroleum reserve at a faster rate and to develop pilot plans for alternative energy reuter.

Figure A. 5: Example text from the ‘Crude’ class in the Reuters-8 dataset, based on which Fig. 3 is generated.

citibank norway unit loses six mln crowns citibank cci norwegian subsidiary based bank made net loss six mln crowns foreign bankers expect show profits two lean years citibank oslo treasury head bjoern sejerstad told reuters citibank foreign bank subsidiaries operating norway lost money restructuring for investment banking commercial banking and economic slump norway last year plunge oil prices foreign banks allowed operate susbidiaries norway since foreign banking analysts oslo access norway second hand securities and equities markets approved this spring and lower primary reserve requirements make profit this year citibank lost crowns norway sejerstad profit likely this year planned liberalisation and economic performance helped steadier oil price dlrs barrel earlier this year chase manhattan bank cmb subsidiary decided stop foreign exchange trading heavy losses and focus fee based merchant banking reuter

Figure A. 6: A sample text from the ‘Earn’ class in the Reuters-8 dataset.

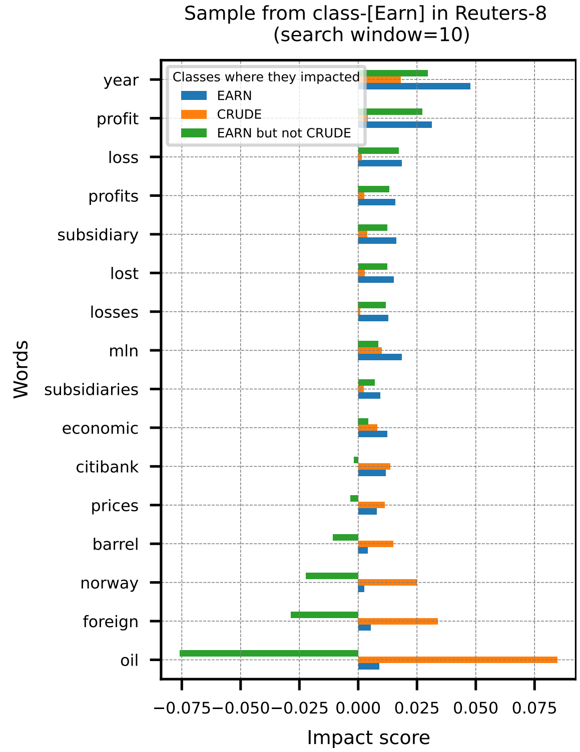

Figure A. 7: Words with the highest impact on the identification of sample from Fig. A. 6 as *Earn*.

A. 2.3 Hyperpartisan

Former FBI director James Comey speaks during a Senate Intelligence Committee hearing on Capitol Hill, Thursday, June 8, 2017, in Washington. WASHINGTON (Sputnik) - The White House initially said Trump fired Comey over his handling of the Hillary Clinton private email server investigation, however, later the US president stated that when he decided to dismiss Comey, he was thinking of the Russia investigation. US President Donald Trump said in a statement on Thursday that he did not fire former Federal Bureau of Investigation (FBI) Director James Comey because of the Russia investigation. Not that it matters but I never fired James Comey because of Russia! The Corrupt Mainstream Media loves to keep pushing that narrative, but they know it is not true! — Donald J. Trump (@realDonaldTrump) May 31, 2018 In an interview with NBC News' Lester Holt in May 2017, Trump said he was thinking about "this Russia thing" when he decided to fire Comey. READ MORE: Twitter on Fire as Comey Strikes Back at Trump "Lying" About FBI Comey's firing ultimately led to the appointment of Special Counsel Robert Mueller to investigate allegations of Russian interference into the 2016 election and collusion between Moscow and the Trump campaign. Russian officials have repeatedly denied allegations of interference, calling the accusations groundless and absurd. Trump has also denied any collusion with the Kremlin and has called Mueller's investigation "a witch hunt."

Figure A. 8: An example text from the 'non-Hyperpartisan' class in the Hyperpartisan dataset.

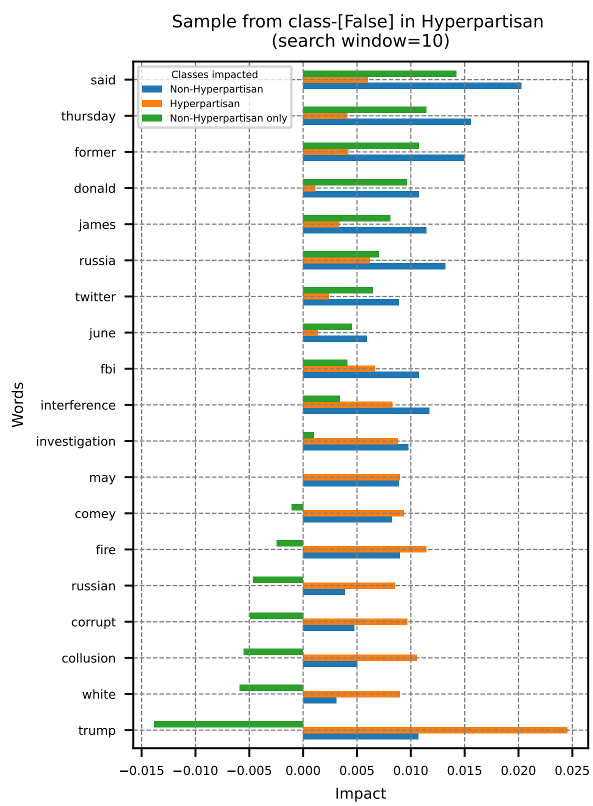

Figure A. 9: Words with the highest impact on the identification of sample Fig. A. 8 as *non Hyperpartisan*.

In his election campaign, Mr Trump was clear about his desire to build a "great great" wall along the border of the US and Mexico which he believes will bring an end to illegal immigration from South and Central America. LBC presenter Ian Collins received a call from Mexican Josephine, who explained why she could not wait for work to begin on the wall. Josephine believes people from her native country are victimised because the Mexican government does not do enough for their people and have no authority when it comes to handling crime, especially in relation to drugs. She said: “We get the blame for all these criminals. A Mexican caller is excited for Donald Trump to build a wall along the American and Mexican border “The Mexican government has turned a blind eye and the war has been lost with the drugs. “Many areas of Mexico have become worse than Chicago. “People are shooting themselves. they look after their family and they have machine guns and they drive around in trucks.” Collins then asked: “So you’re saying Donald Trump wasn’t being a racist then? I don’t want to put words in your mouth I just want to be clear.” Tue, December 13, 2016 Donald John Trump is an American businessman and politician who is President-elect of the United States as well as chairman and president of The Trump Organization “No, no, no,” Josephine replied. Since becoming President-elect has reiterated his intention to crack down on immigration. Speaking to CBS in his first major interview since being elected Mr Trump said: “What we are going to do is get the people that are criminal and have criminal records, gang members, drug dealers, where a lot of these people, probably two million – it could be even three million – we are getting them out of the country or we are going to incarcerate.” Collins, attempting to understand why the Mexican woman was so pleased about Mr Trump’s election success, said: “Was he saying something that lots of people think but are too scared to say?” Josephine said: “Yes, and another thing is that I love the people that are here that I know, I know many people that are Asians, that are Arabs, good speaking English people, but there are very few. “The most people in the area don’t want to mix, and they are very hostile. “Everything has been twisted. I have scrutinised everything in this campaign and newspapers said today he is getting rid of three million migrants. “But they missed a word. Trump said ‘illegal criminals’.”

Figure A. 10: An example text from the ‘Hyperpartisan’ class in the Hyperpartisan dataset.

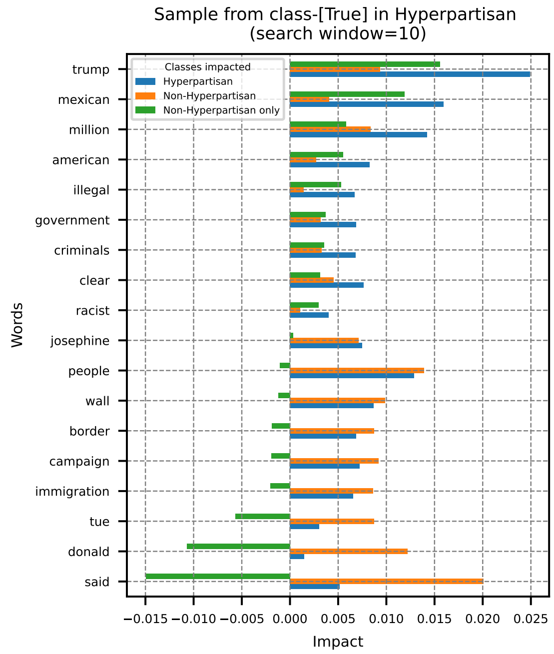

Figure A. 11: Words with the highest impact on the identification of sample from Fig. A. 10 as *Hyperpartisan*.

A. 2.4 Arxiv-4

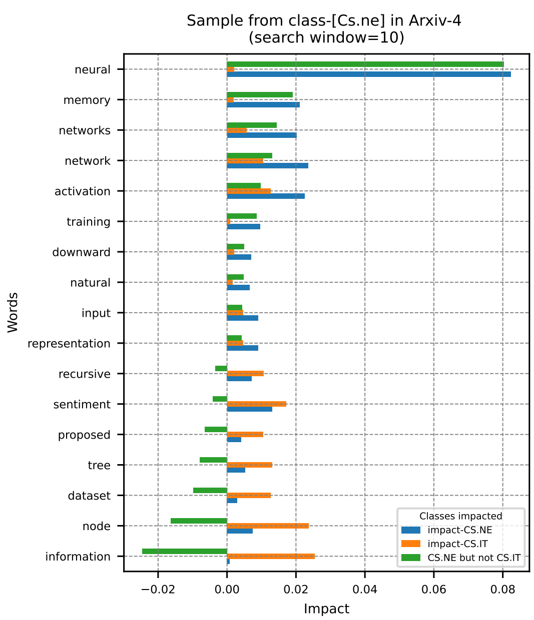

Figure A. 12: Words with the highest impact on the identification of the paper *Structural Attention Neural Networks for improved sentiment analysis* as a sample from class *CS.NE*, of the Arxiv-4 dataset.

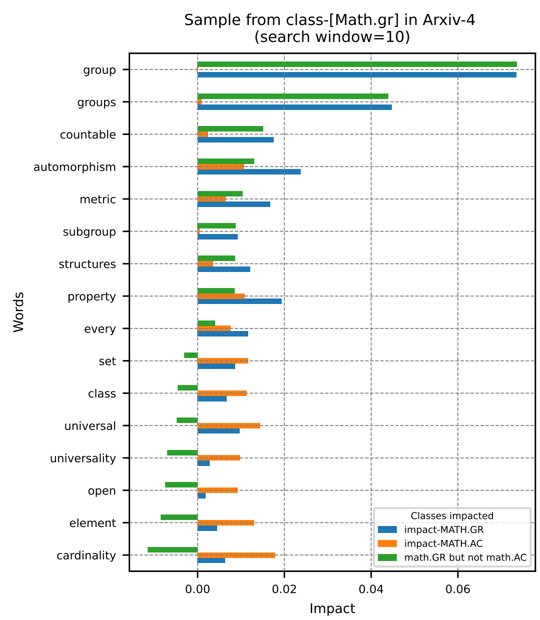

Figure A. 13: Words with the highest impact on the identification the paper *Non-universality of automorphism groups of uncountable ultrahomogeneous structures* as a sample from class *math.GR* of the Arxiv-4 dataset.

Table A. 2: The performance of different classifiers on document classification task.

Model	**Version and documentation link**	**Computing resource used**	**≈ Compute time**
BERT	google-bert/bert-base-uncased	T4GPU (via Google Collaboratory)	8.41min (hyperpartisan), 171min(arxiv-4)
RoBERTa	FacebookAI/roberta-base	T4GPU (via Google Collaboratory)	9.20min (hyperpartisan), 179min(arxiv-4)
Longformer	allenai/longformer-base-4096	A100GPU (via Google Collaboratory)	301min(arxiv-4)
word2vec	word2vec-google-news-30		
GloVe	glove.42B.300d		
SVM	Sklearn SVM	6 cores of CPU	
GRLGQ	Code repository of GRLGQ in Github	6 cores of CPU	55 second(hyperpartisan), 81min(arxiv-4)
AChorDS-LVQ	Proposed method in this contribution. Code repository will be made public soon after acceptance of this work.	6 core of CPU	50 second (hyperpartisan) , 76min (arxiv-4)

A. 3 Appendix: Model and Compute resource specifications