IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Digital Object Identifier 10.1109/ACCESS.2017.DOI

Partially-Supervised Metric Learning via Dimensionality Reduction of Text Embeddings using Transformer Encoders and Attention Mechanisms

RYAN HODGSON[1], JINGYUN WANG[1], ALEXANDRA I. CRISTEA[1], and JOHN GRAHAM[2]
[1]Durham University, Durham, DH1 3LE, United Kingdom (e-mail: ryan.t.hodgson@durham.ac.uk)
[2]Reveela Technologies, Newcastle Upon Tyne, NE1 6UF, United Kingdom

Corresponding author: Alexandra I. Cristea (e-mail: alexandra.i.cristea@durham.ac.uk).

This work was supported in part by the European Regional Development Fund, Intensive Industrial Innovation Program.

ABSTRACT Real-world applications of word embeddings to downstream clustering tasks may experience limitations to performance, due to the high degree of dimensionality of the embeddings. In particular, clustering algorithms do not scale well when applied to highly dimensional data. One method to address this is through the use of dimensionality reduction algorithms (DRA). Current state of the art algorithms for dimensionality reduction (DR) have been demonstrated to contribute to improvements in clustering accuracy and performance. However, the impact that a neural network architecture can have on the current state of the art Parametric Uniform Manifold Approximation and Projection (UMAP) algorithm is yet unexplored. This work investigates, for the first time, the effects of using attention mechanisms in neural networks for Parametric UMAP, through the application of network architectures that have had considerable effect upon the wider machine learning and natural language processing (NLP) fields - namely, the transformer-encoder, and the bidirectional recurrent neural network. We implement these architectures within a semi-supervised metric learning pipeline, with results demonstrating an improvement in the clustering accuracy, compared to conventional DRA techniques, on three out of four datasets, and comparable SoA accuracy on the fourth. To further support our analysis, we also investigate the effects of the transformer-encoder metric-learning pipeline upon the individual class accuracy of downstream clustering, for highly imbalanced datasets. Our analyses indicate that the proposed pipeline with transformer-encoder for parametric UMAP confers a significantly measurable benefit to the accuracy of underrepresented classes.

INDEX TERMS Dimensionality Reduction, Attention Mechanisms, Clustering, Transformer Networks, Metric Learning

I. INTRODUCTION

HIGH quality embeddings produced through techniques such as transformer networks [1] ensure the provision of large amounts of information encoded in highly dimensional vectors, which can be measured using the GLUE benchmark [2], [3]. This has contributed to significant improvements across a range of natural language processing (NLP) tasks, including question answering [2], sentiment classification [4], and text clustering [5]. However, the high degree of dimensionality in embeddings produced by these methods can present difficulties in downstream analysis tasks, due to the prevalence of the "curse of dimensionality" [6], which introduces a multitude of problems, as dimensionality increases. Firstly, the volume of computational memory required for storage and processing of high-dimensional vectors is large, and grows exponentially. Secondly, a greater computational complexity is observed in algorithms, as the number of dimensions is increased [6]. Thirdly, the distance measurements necessary for determining distances between embeddings tend to become meaningless in high dimensional spaces, wherein the ratio between nearest and farthest points approaches one, such that the points essentially become equidistant from each other [7]–[9]. In analysis tasks relying upon such distance measures, such as clustering, this can lead to a detrimental effect upon the result [10].

To address these limitations, the technique of dimensionality reduction (DR) may be applied, which seeks to represent the global and local structure of highly dimensional data

in a smaller feature space [8]. The current state of the art in dimensionality reduction, Uniform Manifold Approximation and Projection (UMAP) [11], has been demonstrated to improve accuracy in a number of clustering algorithms, when used as a DR technique [12], while also considerably improving the computation speed of clustering. Moreover, an extension of UMAP into Parametric UMAP [13] was proposed, which introduces a neural network into the UMAP pipeline. However, there remain a number of open questions to be addressed regarding the use of UMAP and DR in general, prior to clustering, as well as questions related to Parametric UMAP, which we formulate as the following research questions:

- **RQ1** *How do current dimensionality reduction algorithms affect accuracy in text clustering tasks?*
- **RQ2** *How is dimensionality related to performance in text clustering tasks?*
- **RQ3** *Can small portions of labelled data used in the metric learning of dimensionality reduction contribute to improvements in downstream clustering accuracy?*
- **RQ4** *Can the introduction of attention mechanisms within neural networks (further) improve the metric learning of dimensionality reduction algorithms, in terms of clustering accuracy?*

To answer these questions, we target two objectives. Firstly, we perform an empirical investigation of existing DR techniques, and how they impact upon the performance of text clustering tasks. Secondly, we propose *two new architectural pipelines*, both using attention mechanisms in metric-learning dimensionality reduction as a preprocessing technique, constructed from parametric UMAP [12], [13], and each pipeline including one of two network architectures that have been demonstrated to perform well in sequence-to-sequence tasks: the transformer-encoder, and recurrent neural network (RNN) architecture with attention mechanisms.

The main contributions of this work are as follows. (1) To demonstrate, for the first time, to the best of our knowledge, the effectiveness of the transformer-encoder as an architecture in the metric learning of lower dimensionality embeddings with parametric UMAP for text clustering. We demonstrate it achieves the highest accuracy across three of the four datasets investigated, with no loss in accuracy when our proposed transformer-encoder is compared with the current SoA, UMAP, on the fourth dataset. We showcase this through both a visual analysis of the clustering solution on two datasets, and an evaluation of clustering accuracy based on four datasets. (2) We present the outcomes of the first empirical study into the outcomes of DR, and metric learning. I.e., we contribute with an empirical evaluation of all variants of UMAP, as well as of the traditional techniques Principle Components Analysis and Linear Discriminant Analysis, by comparing performance when applied to text-clustering tasks across a range of dimensionalities. (3) We demonstrate, for the first time, the effectiveness of applying attention mechanisms within architectures in the parametric UMAP pipeline, when combined with metric-learning. (4) We provide a public repository of the implementations of the best in class (P-UMAP Transformer) and runner up, i.e. both the transformer-encoder and the RNN with attention architectures, which can be easily accessed and applied by researchers to their own domains using parametric UMAP, or can be compared with the algorithms investigated within this work, using a script within the repository [1]; additionally, we also provide all scores attained by our experiments.

II. RELATED WORKS

A. DIMENSIONALITY REDUCTION

Dimensionality reduction may be defined as the transformation of high-dimensional data into a meaningful representation with reduced dimensionality [14]. As noted, this is necessary in many domains, where highly dimensional data can negatively impact computing efficiency, and accuracy. In clustering tasks, this problem is best represented through the effect on distance measures, such as k-Means, where it becomes clear that the distance measure becomes meaningless, as dimensionality increases [7], [8]. Empirical investigations have demonstrated that this phenomena appears for dimensionalities greater than 10 [7]. The reason being that, as dimensionality increases, the distance to the nearest data point approaches the distance to the farthest data point. This presents difficulties in any downstream tasks that apply nearest-neighbour searches, such as k-Means clustering, or systems that use distance measures, such as *cosine* similarity in nearest neighbour searches.

Amongst techniques proposed for dimensionality reduction, traditional ones include linear techniques, such as Principal Component Analysis (PCA) [15]. Another technique, Linear Discriminant Analysis, provided a supervised approach to dimensionality reduction, through a generalisation of Fischer's linear discriminant [16], seeking to identify linear combinations of features, as a means to characterise or separate objects, or documents. This technique, however, failed to perform suitably when applied to complex, non-linear data.

More recently, t-distributed stochastic neighbor embedding [17] was proposed, as a nonlinear means of dimensionality reduction for the purpose of visualisation. t-SNE was based upon Stochastic Neighbour Embedding [18], wherein a Gaussian is centered over high-dimensional objects, ensuring that a probability distribution may be defined over potential neighbours of the object. t-SNE expanded upon this, through the implementation of a *Student-t* distribution in place of a Gaussian, when computing similarity between points in low-dimensional space. In this method, reduction was typically performed to a dimensionality of 2 or 3, with the resulting vectors being applied as coordinate points in the visualisation. t-SNE observed a significant decrease in performance as dimensionality increased [19].

[1] Provided upon Acceptance

An extension to t-SNE [20] works upon the assumption that a neural network possessing sufficient hidden layers is capable of achieving an approximation of the non-linear functions employed by t-SNE, when mapping a high-dimensional representation to a lower-dimensional representation. In this work, the authors discuss that directly training a neural network through backpropogation is not feasible, due to the tendency for backpropogation to encounter a local minimum, given the complex interactions between layers in the network, which entail a large number of parameters. To address this, the authors applied a training strategy involving the training of autoencoders based upon Restricted Boltzmann Machines (RBMS). In this process, a stack of RBMs is trained, and then used to generate a pre-trained feedforward network, which can subsequently be fine-tuned using backpropogation. The resulting network represents an approximation of the functions of t-SNE. The work demonstrates through experimentation that the parametric model can outperform PCA and an autoencoder in the dimensionality reduction of the MNIST [21], and 20 Newsgroups datasets [22].

Another experiment into dimensionality reduction [12] focused upon the application of the Uniform Manifold Approximation and Projection (UMAP) algorithm to improve clustering performance in image classification tasks; as representing the current state of the art, albeit in a different field, this approach is briefly presented in sections II-A1 and II-A2. Authors applied their experiment across four clustering algorithms; k-Means [23], HDBSCAN [24], [25], Gaussian Mixture Models [26] and Agglomerative Clustering [27]. Results indicated a significant improvement in accuracy across multiple datasets, achieving an improvement of 60% when applied to HDBSCAN on the United States Postal Service [28] digit classification dataset. However, there was no reporting of parameter configuration for both UMAP and the clustering algorithms applied. Most notably, the dimensionality selected for the experiments was not disclosed. This presents a necessity for disclosure of information for future researchers, and forms the basis of our initial experiments into dimensionality reduction, which are detailed in section III-B.

As discussed, dimensionality reduction algorithms have been demonstrated to be an effective preprocessing tool, which can contribute to downstream clustering [12], [29]–[31]. Given their promise, we seek to investigate whether a novel pipeline based onthe cross-domain implementation of neural network architectures, facilitated by the parametric UMAP framework would lead to improvements. Within the literature, we have not identified any evidence of the transformer encoder architecture, or experimentation with any specific neural network architectures within the parametric UMAP framework.

1) Uniform Manifold Approximation and Projection

Uniform Manifold Approximation and Projection (UMAP) [11] was one of the most recent techniques proposed for the task of dimensionality reduction. The algorithm sought to better represent the local structure, while additionally preserving the global structure. Similarly to other dimensionality reduction techniques, the algorithm serves as a suitable tool for visualisation of high dimensional data; however, it has been demonstrated as an efficient tool for general purpose dimensionality reduction for use in machine learning, with it being applied to topic modelling [29], text clustering [12], and genetics research [30], [31]. Most notably, the algorithm provided a significant improvement in scalability, when compared with t-SNE, making it more accessible for use in machine learning pipelines.

Functionally, UMAP is a manifold learning technique based upon Reimannian geometry and algebraic topology. UMAP performs two key steps in the computation of low-dimensionality vectors: Firstly, the computation of a graph representation of data; and secondly, the optimisation of a low-dimensionality representation of the graph through stochastic gradient descent. In the first stage, UMAP performs the construction of a fuzzy simplicial complex (a topological representation of a local neighbourhood graph), containing a weighted graph where edge weightings represent the likelihood that two points are connected. The connectedness of this graph is determined through a radius drawn from each point, with points being connected when radii overlap. This radius length is assigned locally, based on the distance from a point to the n^{th} nearest neighbour, where n is a hyperparameter. The likelihood of connections being formed is decreased as the radius grows, with each point being required to be connected to at least its closest neighbour, thus ensuring to maintain local structure [11]. For the second stage, a stochastic gradient descent optimisation is applied, to identify a low-dimensional representation that provides the closest similarity to the original high-dimensionality input, similar to t-SNE. Currently, UMAP (and derivatives of UMAP) are the state-of-the-art in dimensionality reduction, and, as such, UMAP is the principal DR algorithm investigated in this work.

2) Parametric UMAP

A subsequent extension of UMAP is the *parametric UMAP* [13]. While UMAP performed optimisation of the low-dimensional representation using stochastic gradient descent, parametric UMAP introduced a neural network in its place, which learns a parametric relationship between the original high-dimensional data and the embedding [13]. This provides a significant improvement in the speed of inference of new embeddings, once the parametric model has been trained. The authors also analysed the performance of parametric UMAP in clustering tasks, through the evaluation of normalised mutual information (NMI) in k-Means clustering, with findings indicating this to be comparable to UMAP.

Most notably, the introduction of a neural network in learning of low-dimensionality representations in parametric UMAP presents the opportunity for the specification of tailored neural network architectures. This opens up the

possibility to tailor networks to specific domains, and forms thus the basis of our investigation.

Moreover, the robust mathematical foundation of UMAP permits the extension to supervised learning, which is discussed briefly in UMAP [11] and extended by Parametric UMAP [13]. In the case of Parametric UMAP, the introduction of labelled, or partially labelled data allows training of the network using both classifier loss for labelled data, or UMAP loss for unlabelled data. Through the use of *partially labelled data*, semi-supervised learning allows the joint learning of data structure with unlabelled data, with labelled data being used for the optimisation of a supervised objective function. For our work, we thus construct a novel pipeline including the partially-supervised methodology of Parametric UMAP in conjunction with our proposed neural network architectures, aiming at improving the clustering accuracy.

B. RECURRENT NEURAL NETWORKS AND ATTENTION

The term *Recurrent Neural Network* (RNN) refers to an artificial neural network, wherein neurons send feedback signals to each other [32]. This subsequently allows for the output of some nodes to influence an input of the same node. *Long Short-Term Memory* (LSTM) networks [33] have provided significant contributions to several domains, including speech recognition [34], handwriting recognition [35] and machine translation [35]. In speech recognition tasks, bidirectional RNNs [36], [37] have been applied, where both a forward and a backward RNN is present, with each respective RNN reading the input sequence in opposite directions.

As such, LSTM networks have demonstrated success in sequence to sequence learning tasks, and it is this which we would seek to evaluate as part of this study, as we can consider the task of a neural network in parametric UMAP to be able to be described as the modelling of a shorter sequence based upon a longer input.

Subsequent improvements to RNN architectures in sequence to sequence modelling tasks have involved the inclusion of *attention mechanisms* [38], [39] in encoder-decoder networks. Attention mechanisms perform the computation of a context vector, representing the relationship between the layer output and inputs, where the context vector is a weighted sum of the hidden states of the network at each time-step. This guides a model to focus on specific components of the input sequence, rather than the whole vector sequence. In language modelling tasks, this allows a model to focus upon specific words within a sentence or speech, which may provide the most contextual information.

Recently, the use of attention mechanisms was further developed [1], by proposing the concept of *transformer network*. This architecture is based solely on attention mechanisms, with no convolutional or recurrent layers, instead leveraging the proposed "Scaled Dot-Product Attention", and Multi-Head Attention, to achieve improved performance in machine translation tasks [1]. The transformer architecture has subsequently contributed to improvements in benchmark performance across a number of tasks in NLP, including question answering, sentence continuation, named entity recognition, and language understanding [2] [40]. This has been advanced through the introduction of the pre-trained transformer in works such as BERT [2], where a transformer model is trained upon a large corpus, through the omission of certain words and prediction of the correct word.

Of particular value to our investigation is how the introduction of the transformer architecture led to an improvement in downstream NLP tasks, without the need for recurrent or convolutional layers, which in turn allows for a reduction in training time [1].

III. METHODOLOGY

A. DATA

To demonstrate that the outcomes of dimensionality reduction, when applied to text embeddings, are generalisable, we performed experiments on three datasets: 20 Newsgroups [22], Text REtrieval Conference (TREC) [41] and AG's News[2]. They were selected, as each of these provide a textual representation of the data from a range of domains. In the case of the 20 Newgroups, data is arranged into 20 "newsgroup" categories, with each document being assigned to a single category, representing the topic of the document. The 20 Newsgroups dataset is a widely cited dataset, and was selected for the ease of implementation in comparing the results of this work with any future investigations. The original work presenting this dataset is widely cited [22], and the dataset is accessible easily through the scikit-learn framework[3]. The TREC Question Classification dataset provides 5500 training documents, and 500 test documents consisting of labelled textual questions, resulting in 6000 labelled documents overall. Labels for this dataset are provided as coarse-grained and fine-grained, wherein the coarse-grained set has 6 class labels, with the fine-grained having 47 class labels. We evaluate both labelling formats, to estimate the influence of the number of classes upon accuracy. This dataset was originally intended as a classification task [41], however we investigate this dataset from a clustering perspective for two reasons. Firstly, the dataset is imbalanced, particularly for the finer-grained labels, which presents opportunities to analyse how the DR algorithms in our investigation are affected by such an imbalance. Secondly, the dataset is relatively small by modern standards, which presents the opportunity to again investigate how this affects DR algorithms. The AG's News corpus provides 127,600 news articles, consisting of titles and description fields of articles collected from more than 200 news sources over a year through the ComeToMyHead search engine, with each document being assigned one of four class labels from either "World", "Sports", "Business" or "Sci/Tech" news categories. This dataset features no class imbalance; however, it is significantly larger than the others used in our study, which again presents an opportunity in

[2]http://groups.di.unipi.it/~gulli/AG_corpus_of_news_articles.html
[3]https://scikit-learn.org/stable/index.html

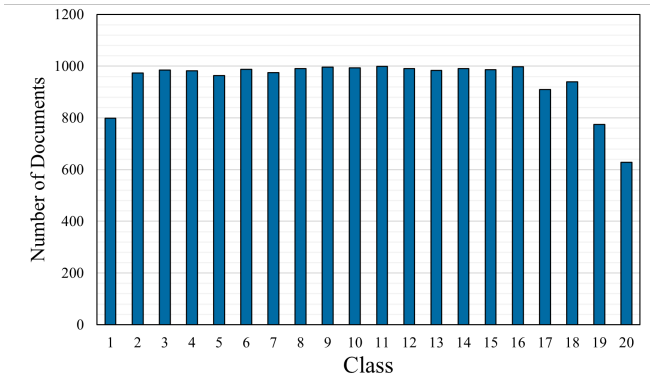
comparing how size influences downstream clustering following DR. These datasets were selected to provide a range in corpus size and class numbers, as we seek to investigate how these may impact upon the proposed methodology. Furthermore, this ensures that additional comparisons of DR algorithms on different datasets are provided. An overview of these is provided in Table 1. For each dataset, we leverage the RoBERTa pre-trained transformer network [40], for the computation of embeddings of the documents within the dataset. These may then be passed to the dimensionality reduction algorithms used in the experiments. The embeddings produced by RoBERTa are contextual, and therefore it is not necessary to perform "traditional" preprocessing, such as stopword removal, or lemmatisation. This is due to the fact that RoBERTa, which is built upon the work of BERT [2], adopts a masked language-model (MLM) strategy for pre-training, where tokens are randomly masked, with the objective of the training being the prediction of the masked term based only on its context [2], [40]. Thus, no preprocessing is performed, as this would affect the contextual information entailed within the text. One exception to this, however, is the 20 Newsgroups dataset, where the original dataset, consisting of data extracted from online forums, also contains header and footer information, with personal information, such as email addresses and names of the users. We remove these, due to both ethical considerations, as well as due to them contributing any useful information to the clustering task. All preprocessing steps can be found in the provided repository[1].

Dataset	20 Newsgroups	TREC-6	TREC-50	AG News
Documents	18846	6000	6000	127,600
Classes	20	6	50	4

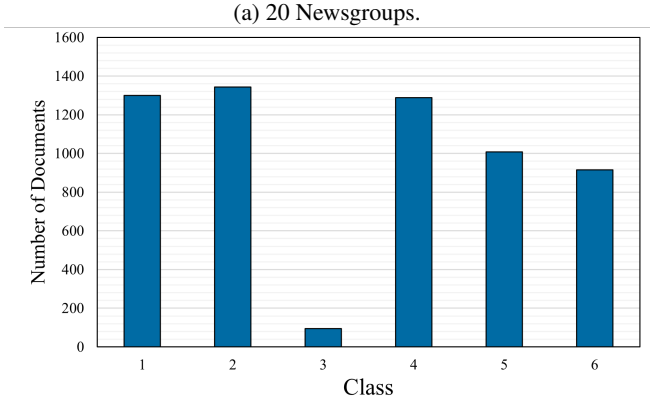
TABLE 1: Total number of records and classes present in the 20 Newsgroups, TREC and AG's News datasets

Figure 1 demonstrates the number of documents present in each class of the selected datasets. In the 20 Newsgroups dataset (Figure. 1a), a significant portion of the classes feature a small degree of class imbalance, with the smallest class being class 20 with 628 documents, and the largest, class 16, having 997 documents. In comparison, in Figure 1b, it is clear that for the TREC-6 dataset, there is a significant imbalance in class sizes, with the smallest (class 3) containing only 95 documents. This imbalance is even more prevalent in the TREC-50 dataset (Figure 1c), which uses the same corpus as TREC-6, however, with a finer-grained labelling scheme. In this dataset, the smallest class contains only 4 documents. Finally, the AG's News dataset (Figure 1d) features no class imbalances, with each class having 31,900 documents.

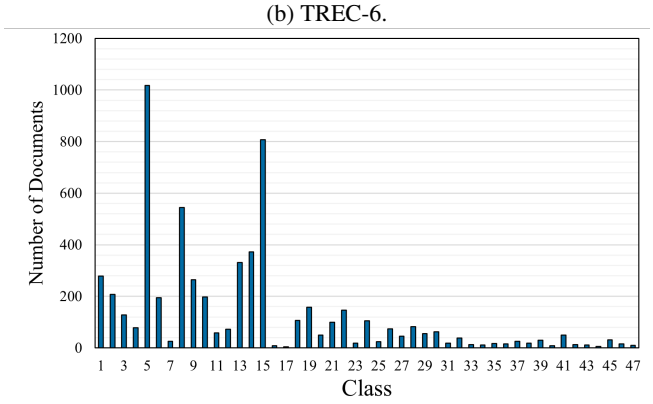
For the investigation of a supervised learning methodology with UMAP, it is important to consider the impact of class imbalance on the dimensionality reduction process and, subsequently, the downstream clustering task. Anticipated effects include reduced class-specific accuracy in underrepresented classes, such as class 20 in the 20 Newsgroups dataset or class 3 in the TREC-6 dataset (as shown later in Results



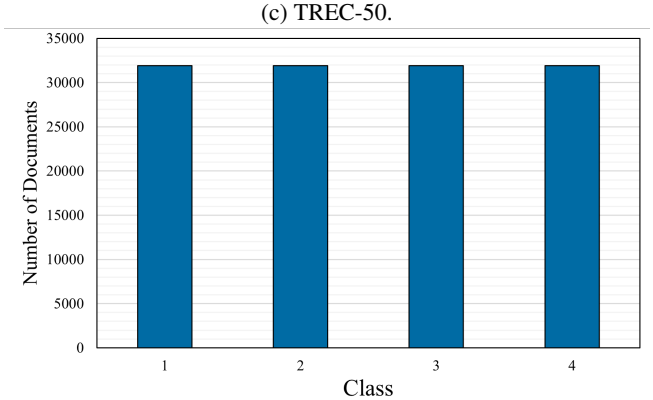
(a) 20 Newsgroups.



(b) TREC-6.



(c) TREC-50.



(d) AG's News.

FIGURE 1: Number of Documents Assigned to Each Class.

Figure 5). We present an analysis of these implications in Section IV. In downstream clustering, class imbalance has been demonstrated to have a considerable impact upon the clustering result for the k-Means algorithm [42]. k-Means was found to tend to produce clusters of uniform size [43], even in diverse datasets with imbalanced data, which leads to sub-optimal clustering results. To address this, a variety of strategies are available to mitigate the effects of class imbalance, broadly classified into three primary groups: re-sampling techniques, algorithm-level adjustments, and hybrid methods [44]. Resampling techniques seek to address the imbalance at a data-level, typically using over-sampling, or under-sampling methods, where the data is adjusted in order to decrease prevalence of the skewed class distribution within the dataset [45]. Of these, one of the most prevalent is the SMOTE algorithm, wherein synthetic minority class examples can be introduced to the dataset to address the imbalance [46]. Algorithm-level adjustments incorporate adaptations to algorithms, such that they take into account the skew in data. For k-Means, examples of this include the introduction of artificial neural networks for the determining of the initial cluster centroids [47], or the introduction of 'multicenter' clustering variant, where multicenters are used to determine each cluster, rather than one centroid per cluster [42]. Additionally, cost-sensitive methods [48] combine algorithm and data-level techniques, to assign a misclassification cost for each class based on evaluation methods [49].

To mitigate overfitting in the transformer-encoder architecture, instead of using sampling techniques that may not accurately represent the data, randomised dropout layers are incorporated within the network architecture. This is applied through the random omission of units within the neural network, which has been demonstrated to be effective in addressing overfitting [50], [1].

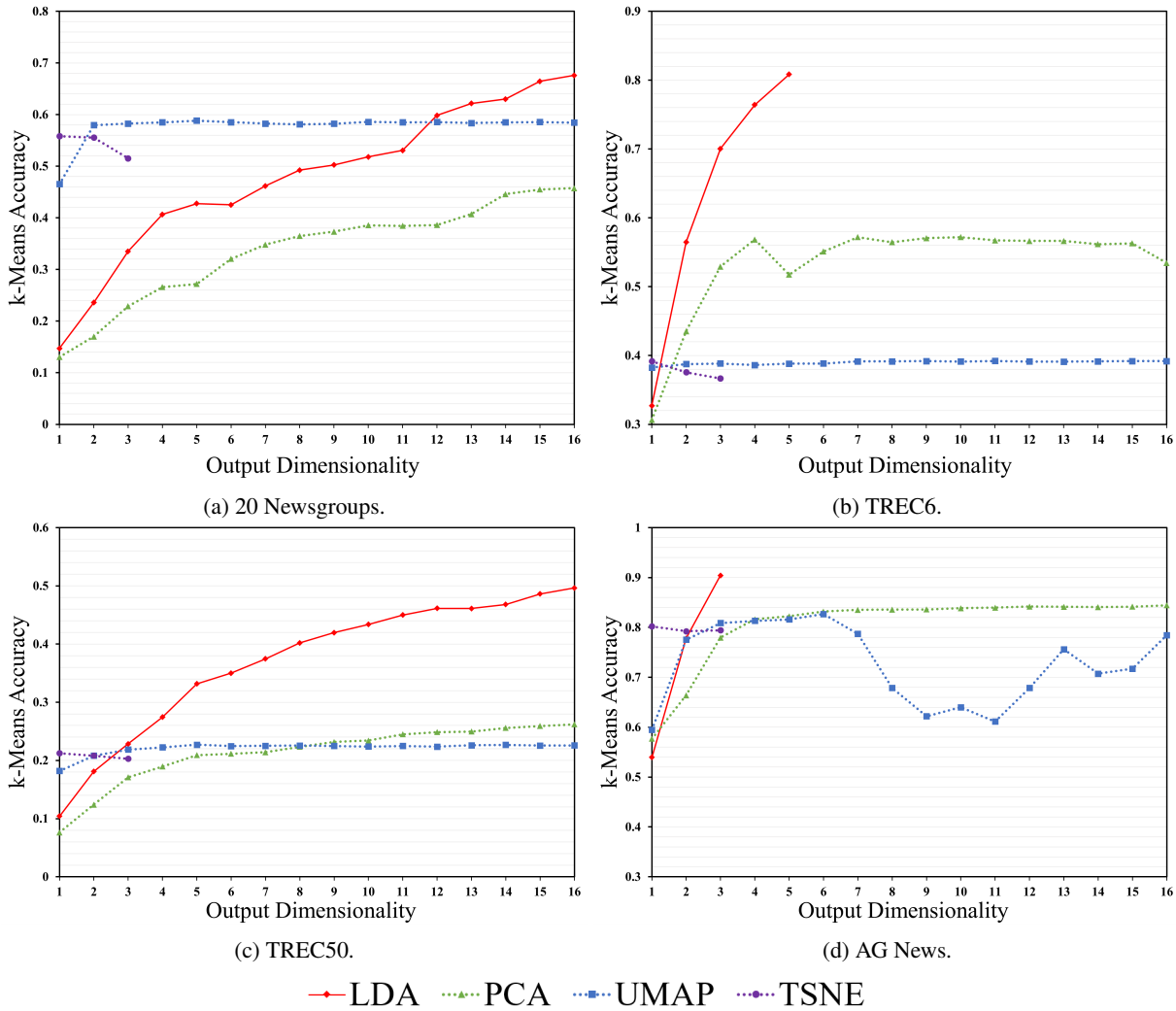
For our task, which differs from the language modelling task the architecture was initially intended for, we apply dropout following the multi-head attention layer of the transformer block, similar to the original transformer encoder [1]. However, a second layer of dropout is then applied to the final feedforward layer of the architecture, prior to the output layer. The amount of dropout in each model is determined through an optimisation strategy. The hyper-parameter optimisation strategy is discussed in Section III-D.

B. PRELIMINARY EVALUATION OF EXISTING TECHNIQUES FOR DIMENSIONALITY REDUCTION

Uniform Manifold Approximation and Projection (UMAP) [11], t-distributed stochastic neighbour embedding (t-SNE) [17], Principal Components Analysis (PCA) [15] and Linear Discriminant Analysis (LDA) [16] are state-of-the-art, respectively benchmark methods for DR, which we evaluate in terms of accuracy when used as a DR technique prior to clustering, addressing our first and second research questions (**RQ1**, **RQ2**). LDA requires the provision of labelled data. Hence, we provide this as a randomly shuffled subset of 20% of the whole dataset, from which we compute the low-dimensionality representations of the full dataset.

The computational complexity for the algorithms analysed in the preliminary investigation differ considerably, and therefore their use is best suited for different situations and datasets. For LDA, the time complexity is $O(Ndt + t^3)$, and memory requirement is $O(Nd + Nt + nt)$, where N is the number of samples, d is the number of features, or the dimensionality of the data, and $t = min(N, d)$. In instances where N and d are large, the algorithm becomes infeasible, as discussed by [51], who also evaluated the algorithm upon the 20 Newsgroups dataset, where they identified a considerable increase in time complexity for large samples of the dataset. For PCA, a time complexity of $O(min(N^3, d^3))$ is outlined by [52]. Regarding t-SNE, there is a significant limitation due to the computational complexity of the algorithm, which scales at a degree of $O(N^2)$, where N is the number of data points [53]. The application of the Barnes-Hut algorithm as an approximation method for the gradient calculation algorithm can enhance efficiency to $O(\log N)$ time complexity, as demonstrated in [54], [55]. However, it is important to note that this approach is applicable only when the output dimensionality is less than or equal to 3 dimensions. Finally, in the case of UMAP, empirical results indicate an approximate complexity of $O(N^{1.14})$, which is bounded by the complexity of the approximate nearest neighbour algorithm, and has, at this time, no theoretical proof [11], [56]. Based on these, it would appear that UMAP presents the best scalability in terms of time complexity. As discussed in our introduction section, the "curse of dimensionality" arises in clustering algorithms, due to the impact that high dimensionalities have upon the distance measurements, which are necessary in determining distances between points, when assigning them to clusters. k-Means has a time complexity of $O(N^{dk+1})$ [57], where d is the dimensionality, N is the number of points to cluster, and k is the number of clusters. Therefore, opting for the lowest-dimensional representation prior to clustering is beneficial when working with large datasets, if it can be proven that a low-dimensionality representation will perform adequately.

Figures 2a-2d demonstrate the clustering accuracy attained by k-Means clustering across dimensionality ranges $d = \{1, ..., 16\}$ for UMAP, PCA, LDA and t-SNE. Notably, LDA indicates a positive correlation regarding an increase in accuracy relative to dimensionality across all datasets, outperforming both UMAP and PCA. However, there are limitations to this method, as LDA relies upon the presence of labelled data. Additionally, the algorithm is not capable of generating embeddings with a dimensionality greater than $(u - 1)$, where u is the total number of unique labels present in the data. Most notably, as demonstrated in Figures 2a and 2c, in some cases, the dimensionality must be large, in order to obtain optimal clustering accuracy. This is not ideal based on the increase in computational complexity that is observed by clustering algorithms when analysing highly dimensional data. PCA outperforms, or performs comparably to UMAP, the established state-of-art, on the TREC6, TREC50 and



(a) 20 Newsgroups.

(b) TREC6.

(c) TREC50.

(d) AG News.

FIGURE 2: Comparison of LDA, PCA, t-SNE and UMAP DR techniques upon accuracy in downstream k-Means clustering.

AG News datasets. As with LDA, accuracy for downstream clustering by k-Means appears to improve as the dimensionality increases. t-SNE performs comparably with UMAP at low-dimensionalities; however, this is evaluated only up to an output dimensionality of 3, due to the Barnes-Hut approximation algorithm used in the algorithm restricting output dimensionality to below 4 dimensions.

C. PARAMETRIC UMAP WITH BIDIRECTIONAL RECURRENT NETWORKS WITH ATTENTION

Following the preliminary evaluation of UMAP, t-SNE, PCA and LDA, a *novel pipeline of a bidirectional RNN with attention mechanism for DR* is proposed, which is facilitated through parametric UMAP [13], using a *metric learning methodology*, wherein the parametric dimensionality reduction model is trained upon a small subset of labelled data. It is to be hypothesised that the introduction of supervised learning to the computation of lower-dimensionality embeddings could improve downstream clustering performance. Furthermore, *this work aims to demonstrate how the definition of a recurrent network with attention could potentially enhance clustering accuracy*, given a small amount of training data. Therefore, a sample of only 20% of the dataset is used in the training of the supervised metric learning model. This performance is compared with other configurations of UMAP, including UMAP itself, supervised UMAP, parametric UMAP and parametric supervised UMAP. Parametric UMAP in this case is configured with a default network architecture of 3 fully connected layers consisting of 100 units each.

We propose a recurrent neural network with self-attention mechanism, to *investigate the impact of recurrent networks and attention upon the parametric learning of low-dimensionality embeddings*. Figure 3 demonstrates this overall architecture, which consists of 2 blocks of stacked RNN-attention layers, followed by a fully connected layer. This results in a total of $35,119,359$ trainable parameters. The configuration of this architecture and the assigned number of nodes for each layer is identified through the Tree-

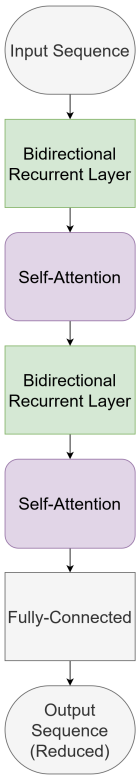


FIGURE 3: Proposed Novel Bidirectional Recurrent Architecture with Attention Mechanism using Gated Recurrent Units for Use in the DR Pipeline via Parametric UMAP

structured Parzen Estimator [58]–[60] hyper-parameter optimisation, facilitated by the Optuna framework [61]. For this optimisation strategy, we set the optimisation objective as the maximising of accuracy score for the k-Means clustering of the whole dataset, with the model being provided with only a 20% subset of the data.

D. PARAMETRIC UMAP WITH TRANSFORMER-ENCODER

We define a transformer-encoder, the main crux of our investigation, based upon the original architecture [1], consisting of a stack of N transformer blocks, where each transformer block comprises two sub-layers. The first of these sub-layers consists of a multi-head self-attention mechanism, while the second is a fully connected feedforward network with ReLU activation [62]. After both the multi-head attention, and feedforward layer, layer addition is performed. This entails the concatenation of the outputs of the previous layer with the original input sequence. This layer addition process, which is also known as residual connection, is intended to mitigate the vanishing gradient [63] problem, wherein during back-propagation, the multiplication of the gradients of each layer, if they are smaller than 1, leads to exponentially decreasing gradients. It is reported that as the sequence length of a model increases, the gradient magnitude typically decreases, which can slow down or even stop the training process [64].

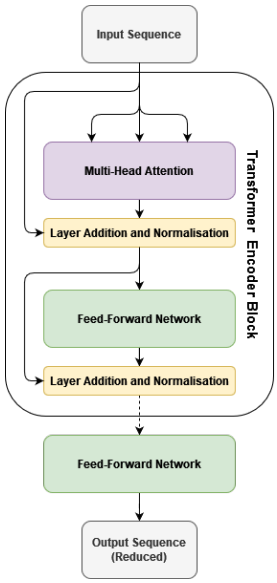


FIGURE 4: Proposed Novel Transformer Encoder Architecture for Dimensionality Reduction for Use in the DR Pipeline via Parametric UMAP

Using the same hyper-parameter optimisation strategy outlined when designing the RNN with attention, we identify an optimal network configuration, consisting of four sequential transformer-encoder blocks, followed by a fully-connected layer. The amount of dropout present within the transformer block, and before the output layer of the model, is also defined through this same hyper-parameter optimisation strategy. Contrary to the implementation of dropout in [1], the strategy for hyper-parameter optimisation pinpoints an ideal setup where the transformer-encoder block does not undergo any dropout. Instead, dropout is exclusively implemented on the model's final feedforward layer, at a rate of 5%. This configuration results in a total of $10,784,272$ trainable parameters, which is more than three times fewer than the total parameters of the RNN with attention architecture. For both the transformer-encoder, and RNN with attention, we investigate our fourth research question (**RQ4**).

E. EXPERIMENTAL SETUP

To evaluate the outcomes of our proposed methods, namely the implementation of the *transformer-encoder*, and *RNN with attention* upon metric learning for DR, we propose the following experiment, comparing clustering accuracy across a range of UMAP variations, where we seek to evaluate both **RQ3** and **RQ4**. These are performed upon the four benchmark datasets applied for the preliminary evaluation in Section III-B. For each dataset, we evaluate UMAP, Parametric UMAP (P-UMAP), UMAP Supervised, Parametric UMAP Supervised, Parametric UMAP with RNN Supervised, and Parametric UMAP with Transformer-Encoder Supervised, with the latter being our two proposed novel architecture pipelines demonstrated in Figure 3 and Figure 4. In the su-

pervised cases, dimensionality reduction models were trained upon the same subset of 20% of the overall dataset, to perform metric learning of the reduced embeddings.

Similarly to the preliminary evaluation (III-B), we compare the accuracy across dimensionalities ranging from $d = \{1, ..., 16\}$ for each algorithm against a baseline score, where k-Means clustering is performed upon the original RoBERTa [40] embeddings, without dimensionality reduction being applied. Results are based on an average taken over 25 separate experiments, which allows for the calculation of statistical significance using a Wilcoxon-Mann-Whitney U Test [65]. We present the accuracy for each algorithm, across each dataset, for each dimensionality, across all 25 iterations of each experiment[1].

1) Clustering Evaluation

Evaluation of the performance relative to dimensionality entails calculation of the accuracy of the downstream clustering task. The integer label assigned by the k-Means algorithm to a cluster may not directly reflect the integer label assigned as the true label, even if the clustering solution is correct. For example, a clustering solution may assign cluster A a label of 0, and cluster B a label of 1, however, the true label present in the dataset is 1 for cluster A, and 0 for cluster B. When we examine the results, we may find that if the error is low, and the clustering solution is correct, then it is necessary to map the predicted clusters to their associated true labels for evaluation. The Kuhn-Munkres Algorithm [66] is hence applied to map the assigned label, by clustering, to the true label. This allows the framing of the evaluation as a supervised learning task, to obtain an accuracy score. We calculate accuracy as:

$$ACC = \frac{TP + TN}{TP + TN + FP + FN} \tag{1}$$

Where TP represents the true positives, TN represents true negatives, FP represents false positives and FN represents false negatives.

2) Testing for Statistical Significance in Results

We select the Mann-Whitney-Wilcoxon (WMW) U test [65] when testing the significance of experimental results. WMW has been demonstrated to function suitably when applied to a smaller sample size, such as a population of 25 [67]. The WMW test is a nonparametric test that makes no assumptions about the distribution of the data, and as such is suitable when data is not normally distributed. For our experiments in section IV-B, we use a population size of 25. While it would be preferable to conduct our experiment across a larger population size, this would require extensive compute requirements and therefore take significantly longer to evaluate across all dimensionalities and datasets. For testing the significance in results, we adopt a significance level $\alpha = 0.01$. The significance between two arguments is computed using the complete set of 25 iterations for each experiment, wherein a run represents the training of the dimensionality reduction algorithm, and subsequent clustering by k-Means. This guarantees that the given scores encapsulate a comprehensive depiction of the performance exhibited by the corresponding algorithms. For example, if testing for significant difference between the accuracy of the UMAP algorithm, when compared to the Supervised Parametric UMAP algorithm, with both having an output dimensionality of 3, for the 20 Newsgroups dataset; Sample 1 would entail the 25 accuracy scores, which were attained by k-Means algorithm when clustering the low-dimensional vectors produced by UMAP with an output dimensionality of 3. Sample 2 would be the 25 accuracy scores for the clustering of the vectors produced by the Supervised Parametric UMAP algorithm, using the same criteria. We form a null hypothesis H_0 that there is no significant difference between two comparisons, and an alternative hypothesis H_a that there is significant difference between results. For each statement we make with respect to statistical significance, we provide the U-statistic U, z-score, and probability p, as well as providing the standard deviation SD, and mean accuracy score M for each individual population.

IV. RESULTS

A. CLUSTER ANALYSIS

Figures 5 and 6 represent a two-dimensional plot of the distribution of vectors produced by the dimensionality reduction techniques of UMAP, supervised UMAP, supervised UMAP with RNN and Attention, and supervised UMAP with transformer network. This is provided as a visual demonstration of the clustering solution, which can aid in evaluating how the different DR algorithms can partition the data. Points are colour-coded, to correspond to the true class label associated with each document. To aid in visual comprehension by the reader, both TREC-50 and 20 Newsgroups were not visualised in this work, due to the large number of classes present, and subsequent difficulty in discerning colour differences. However, these may be produced using the provided reference repository.

We observe a notable distinction in the arrangement of points when comparing *unsupervised* (UMAP and UMAP parametric, Figures 5a and 6c) with *supervised* approaches (UMAP supervised, UMAP parametric supervised, UMAP supervised, RNN with attention, and UMAP supervised transformer, Figures 5b, 6d, 5e and 5f). When trained upon the subset of data, all supervised configurations of UMAP indicate greater effectiveness in the partitioning of documents based upon their class assigned label, in comparison with their unsupervised variants. However, there appear to be only five distinct clusters extracted across all supervised variants, which reflects the class imbalance in the TREC-6 dataset identified in 1b, where class 3 is highly underrepresented, with only 95 documents. When accounting for the 20% sample taken for training, this provides only 19 labelled documents representing class 3. Overall, both the RNN with attention, and transformer architectures, appear to provide improvements in the separation of clusters, with the

transformer providing the clearest separation. However, these appear to form only 5 clusters, which is best demonstrated in Figure 5f. It appears that the class imbalance and under-representation of class 3 in the TREC-6 dataset leads to this cluster failing to be represented by UMAP in all its variants.

B. ACCURACY OF CLUSTERING

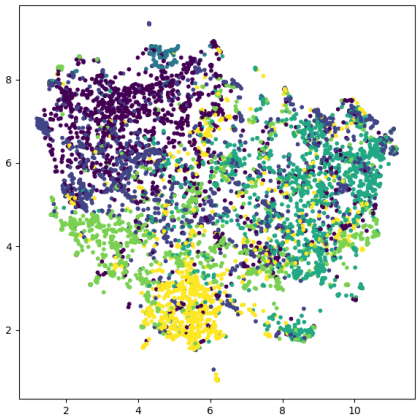
Figure 9a demonstrates the clustering accuracy of the models for the 20 Newsgroups dataset. When testing for significance in subsequent analyses in this section, we apply a WMW test as detailed in Section III-E2. For a two-tailed WMW test with a population size of 25, the critical value of U is 180, such that any U value greater than this is rejected as being statistically significant. It is worth noting that in our analyses, a U value of 0 is frequently observed. This is due to the prevalence of all observations in one population having a score lower than all observations in the other population, and implies a perfect separation between two groups. Individual results used in our experiments can be accessed at the downloadable repository[1].

At a glance, it is evident that all configurations of UMAP provide an improvement in accuracy when compared to the baseline, with a clear "knee" that can be observed between an output dimensionality of 2 and 3. At a dimensionality of 3, which we adopt for all subsequent calculations, comparing UMAP [11] ($SD = 0.007$, $M = 0.582$) with the k-Means baseline score ($SD = 0.012$, $M = 0.517$) indicates a *significant* improvement of 6.5% ($U = 0$, z-score $= 6.054$, $p < 0.0001$) and parametric UMAP (P-UMAP) ($SD = 0.013$, $M = 0.579$) [13] improving *significantly* by 6.1% ($U = 0$, z-score $= 6.054$, $p < 0.001$), compared to the baseline score. The introduction of a metric learning approach, trained upon a subset of data, is demonstrated by the UMAP Supervised ($SD = 0.013$, $M = 0.668$), where we accept the alternative hypothesis, indicating a *significant* improvement compared baseline score of 15% ($U = 0$, z-score $= 6.054$, $p < 0.001$). For the supervised UMAP parametric (P-UMAP Supervised) ($SD = 0.018$, $M = 0.562$) algorithm, we again accept the alternative hypothesis, indicating a *significant* improvement of 4.4% ($U = 0$, z-score $= 6.054$, $p < 0.001$). This supports our investigation into **RQ3**, into *evaluating metric learning as a suitable method to be used prior to clustering*. There is a significant difference between supervised UMAP, and supervised parametric UMAP algorithm, wherein at a dimensionality of 3, the nonparametric algorithm attains an accuracy 10.6% higher than the parametric variant ($U = 0$, z-score $= 6.054$, $p < 0.001$). As the default configuration of the parametric UMAP neural network architecture consists of only fully-connected layers, it is worth evaluating whether the implementation of more complex architectures can contribute to improving the performance of the parametric UMAP model. It is observable in both unsupervised and supervised cases that the nonparametric UMAP algorithm attains a higher accuracy score compared to the parametric version consisting of the default fully connected layers. However, the transformer-encoder (P-UMAP Transformer) ($SD = 0.0083$, $M = 0.698$) model attains the greatest *significant* improvement in accuracy relative to the baseline, of 18.1% ($U = 0$, z-score $= 6.054$, $p < 0.001$). The supervised RNN with attention ($SD = 0.0103$, $M = 0.594$), in comparison, provides a *significant* accuracy improvement of 7.7% ($U = 0$, z-score $= 6.054$, $p < 0.001$) compared to the baseline score.

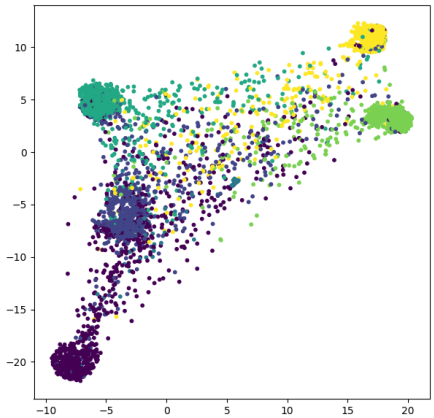
When comparing the transformer-encoder with the next highest scoring algorithm, UMAP supervised, it is evident that the transformer-encoder attains a *overall significant* higher-scoring accuracy in downstream clustering, even at a lower dimensionality. At a dimensionality of 2, the transformer-encoder ($SD = 0.007$, $M = 0.698$) exhibits a *significant* difference 4.7% greater ($U = 0$, z-score $= 6.054$, $p < 0.001$) than UMAP supervised ($SD = 0.013$, $M = 0.65$). At a dimensionality of 3, this *significant* difference is 3% greater ($U = 0$, z-score $= 6.054$, $p < 0.001$) for the transformer-encoder ($SD = 0.008$, $M = 0.698$), compared with UMAP supervised ($SD = 0.013$, $M = 0.668$). This indicates that an *advantage of our proposed pipeline using the transformer-encoder*, is that a higher accuracy can be achieved in downstream clustering at lower dimensionalities, which has benefits with regards to computational complexity, and memory efficiency corresponding to the storage of the smaller vectors. These results are depicted in Figure 9a. In Figure 9b, when applied to the TREC-6 dataset, we observe a decrease in the average accuracy relative to the k-Means baseline for both UMAP, and parametric UMAP. At an output dimensionality of 3, the UMAP algorithm ($SD = 0.002$, $M = 0.039$) is on average 14.7% ($U = 0$, z-score $= 6.054$, $p < 0.001$) poorer than the k-Means baseline score, which is *significant*. Similarly, parametric UMAP ($SD = 0.008$, $M = 0.392$) performs 14.3% worse than the baseline ($U = 0$, z-score $= 6.054$, $p < 0.001$). This decrease is similar for the TREC-50 dataset, where at a dimensionality of 3, UMAP ($SD = 0.005$, $M = 0.22$) is on average 9.1% less accurate than the baseline score ($U = 0$, z-score $= 6.054$, $p < 0.001$), and parametric UMAP ($SD = 0.004$, $M = 0.22$) is 7.1% ($U = 0$, z-score $= 6.054$, $p < 0.001$) less accurate. This is of interest, as it indicates that there is no guarantee that UMAP can contribute to improvements in downstream clustering accuracy. Considering that both TREC-6, and TREC-50 have the lowest number of documents, consisting of only 6000 rows (see Table 1), this may have an influence upon the generalisation of the unsupervised model for UMAP. In comparison, all supervised variants of UMAP demonstrate an improvement in clustering accuracy for both TREC-6, and TREC-50 datasets. Most notably, the transformer-encoder model attains the greatest improvement in accuracy relative to the baseline score. At a dimensionality of 3, this is a significant improvement of 30.3% for TREC-6 ($U = 0$, z-score $= 6.054$, $p < 0.001$), attaining an average accuracy that is greater than all other DR algorithms investigated in our study.

The P-UMAP transformer-encoder architecture achieves the greatest accuracy for three of our datasets, 20 News-

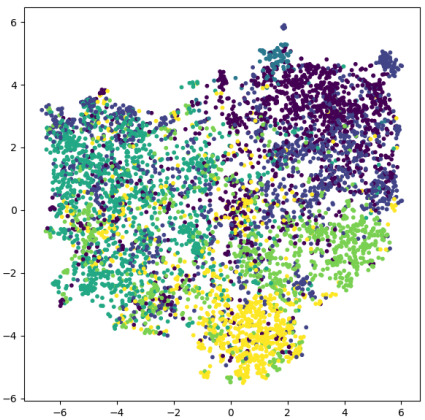
This article has been accepted for publication in IEEE Access. This is the author's version which has not been fully edited and
content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2024.3403991



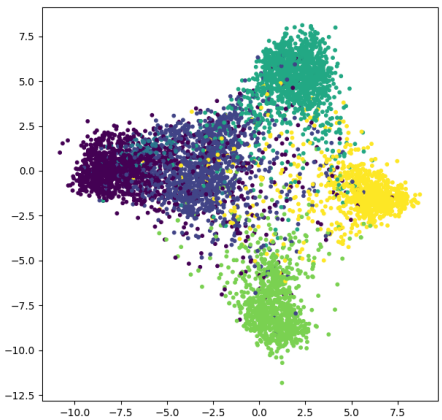
(a) UMAP.



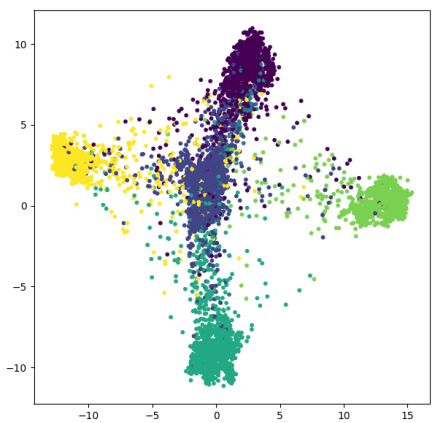
(b) Supervised UMAP.



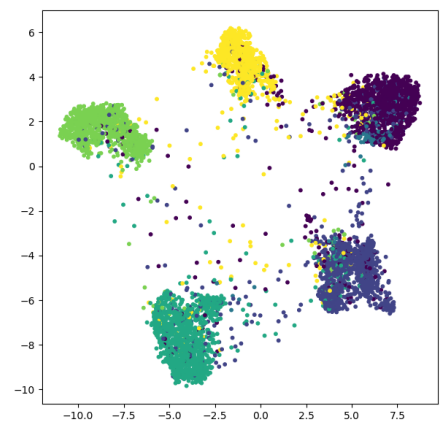
(c) UMAP Parametric.



(d) Supervised UMAP Parametric.



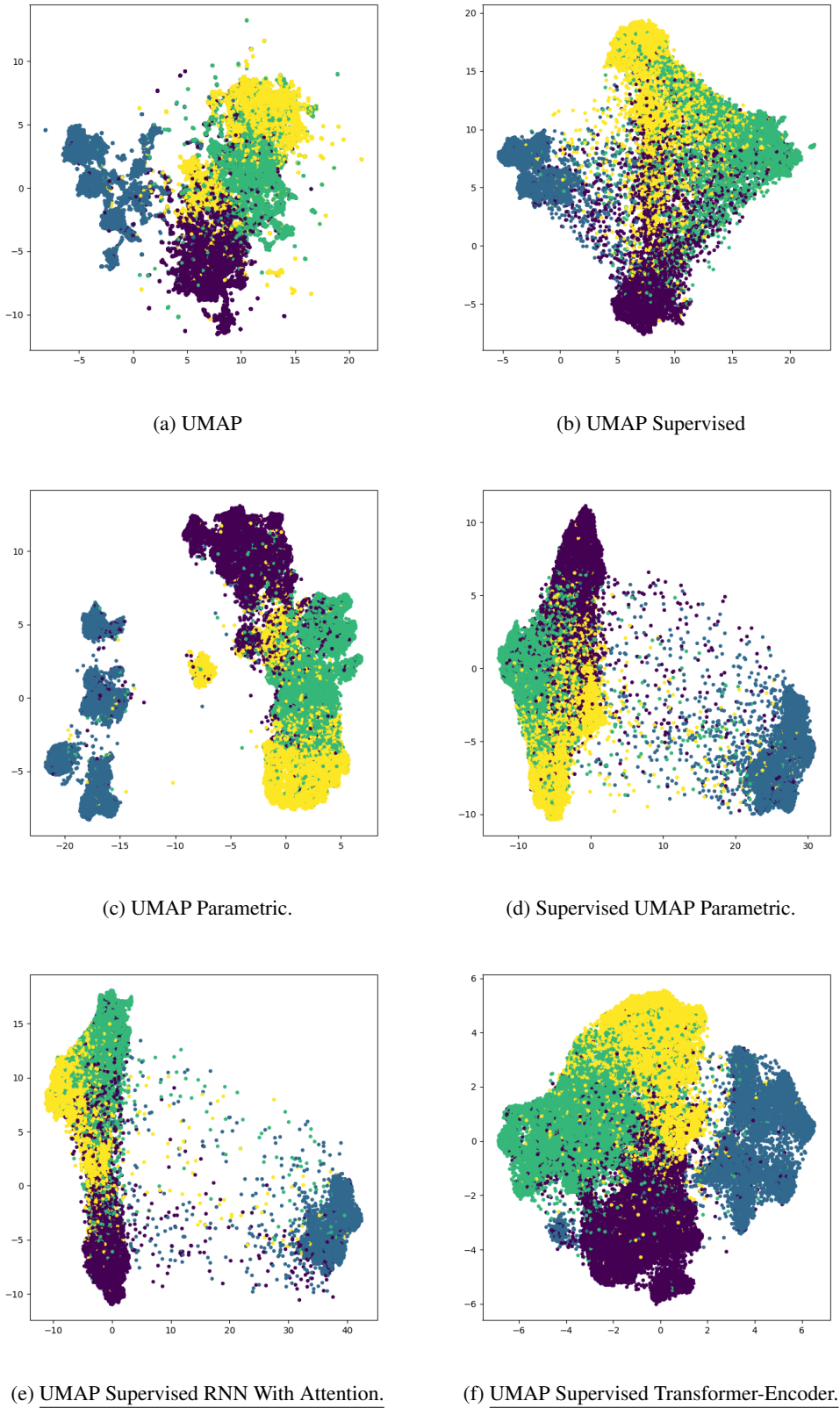
(e) Supervised UMAP with RNN and Attention.



(f) Supervised UMAP Transformer-Encoder.

FIGURE 5: Visualisation of reduced vectors at a dimensionality of 2 for TREC6, for our proposed pipelines (underlined), compared with existing UMAP configurations, with colours representing the true label for each point.

This article has been accepted for publication in IEEE Access. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2024.3403991



(a) UMAP

(b) UMAP Supervised

(c) UMAP Parametric.

(d) Supervised UMAP Parametric.

(e) UMAP Supervised RNN With Attention.

(f) UMAP Supervised Transformer-Encoder.

FIGURE 6: Visualisation of reduced vectors at a dimensionality of 2 for AG's News, for our proposed pipelines (underlined), compared with existing UMAP configurations, with colours representing the true label for each point.

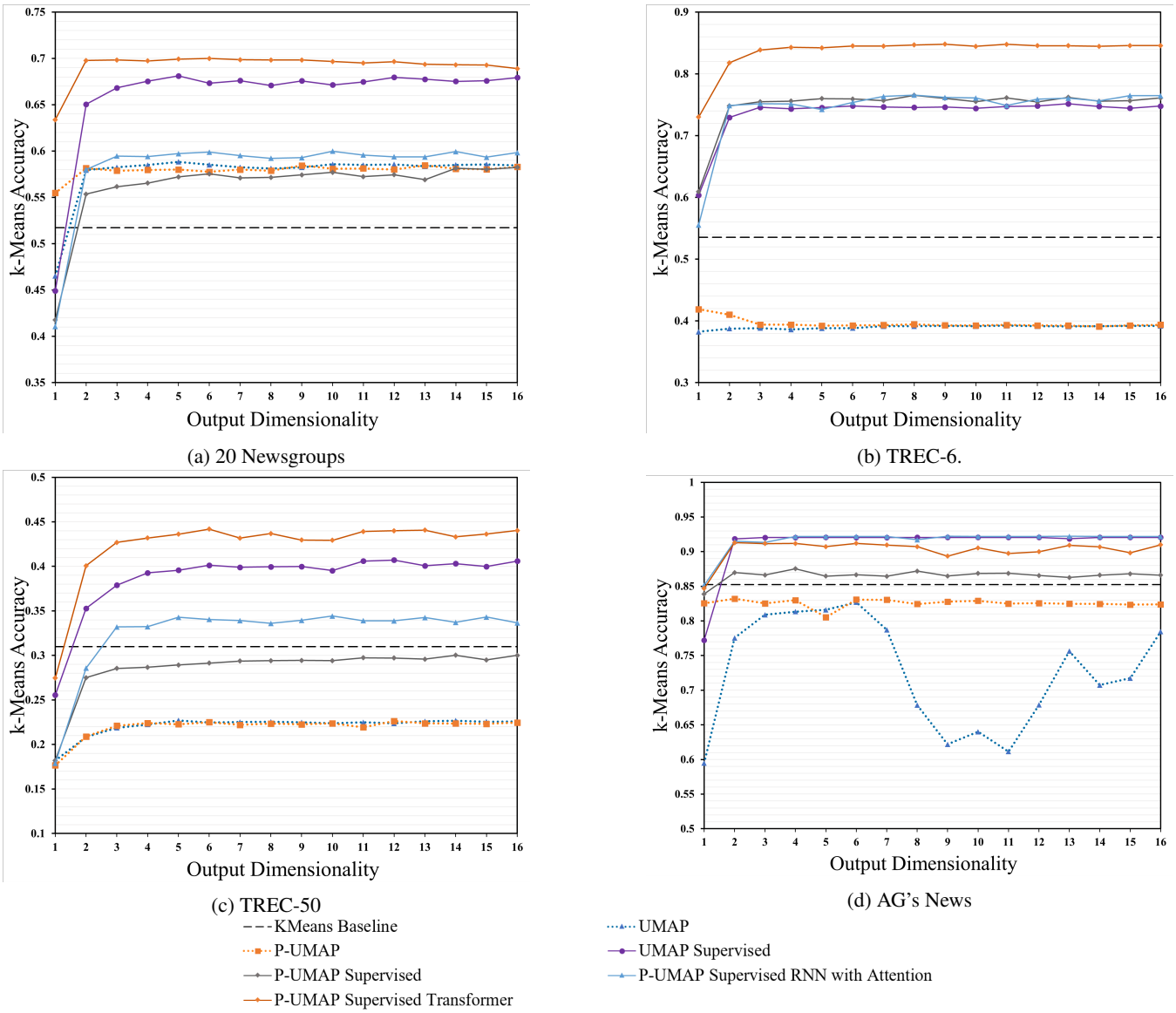


FIGURE 7: Accuracy of k-Means clustering with dimensionality ranging from 1 to 16 for the proposed methods (our proposed pipelines underlined), compared with existing UMAP configurations.

groups, TREC-6, and TREC-50, which reflects the outcomes observed in our visualisation analysis of TREC-6 in Figure 5. However, on the fourth, the AG's News dataset, the P-UMAP transformer-encoder does not outperform UMAP Supervised, the current SoA (Figure 7d). Nevertheless, both of our proposed architectures, the transformer-encoder, and RNN with attention, perform comparably with UMAP supervised for the AG's News dataset.

An interesting observation, which is evident only for the AG News dataset, is the high degree of variance in downstream clustering accuracy for the UMAP algorithm. This is prevalent between the dimensionalities of 6 and 16, where a considerable decrease in the accuracy of downstream clustering appears. This presents a decrease in the average accuracy of downstream clustering from 82.6% ($SD = 0.054$) at a dimensionality of 6, to a minima of 61.1% ($SD = 0.08$) at a dimensionality of 11. Furthermore, within the scope of our experiments, it was observed that the population of results for UMAP exhibited the highest standard deviation of accuracy, at a dimensionality of 11 ($SD = 0.08$). This was the most significant fluctuation in accuracy across all conducted tests.

A general observation across all of the experiments (Figure 7) indicates the *presence of a "knee" in accuracy* for all derivatives of UMAP , which becomes apparent between the dimensionalities of $\{2, 3\}$, when the reduced dimensionality embeddings are used in k-Means clustering. Increasing the output dimensionality of embeddings beyond this degree tends to have little or diminishing influence on the quality of the clustering solution for k-Means. This is of interest, and merits a further discussion, as we have, at this time,

found *no evidence of this phenomenon within the literature*. Notably, it is empirically apparent that the k-Means algorithm experiences a decrease in the rate of improvement beyond an output dimensionality of 3 for low-dimensional representations produced by UMAP and UMAP derivatives, for both a supervised and unsupervised training manner. When considering computational complexity, there are numerous benefits to choosing a lower output dimensionality. As discussed in Section III-B, clustering algorithms, such as k-Means, can be affected by data dimensionality. Notably in the case of k-Means clustering, the time complexity scales quadratically with relation to the dimensionality and the number of clusters. Therefore, it is often advantageous to perform clustering using a representation of the data with reduced dimensionality. Considering the marginal accuracy improvements beyond the knee curve point, it could be more efficient to select a small output dimensionality, such as 2 or 3, to optimise the runtime of the clustering solution.

Overall, the proposed architecture based on the transformer-encoder is demonstrated to contribute to significant improvements in clustering accuracy across three of the four experiments conducted. The RNN with attention also contributes to improvements in accuracy relative to existing methods, and outperforms the transformer-encoder by a small margin on the AG's News dataset. This addresses both **RQ3** and **RQ4** outlined in our introduction (Section I).

The maximum average accuracy based upon 25 iterations of each experiment at a dimensionality of 3, for each algorithm across our experiments is summarised in Table 2.

TABLE 2: Average Accuracy at an Output Dimensionality of 3 based on 25 Iterations for Each Experiment (our proposed pipelines are underlined)

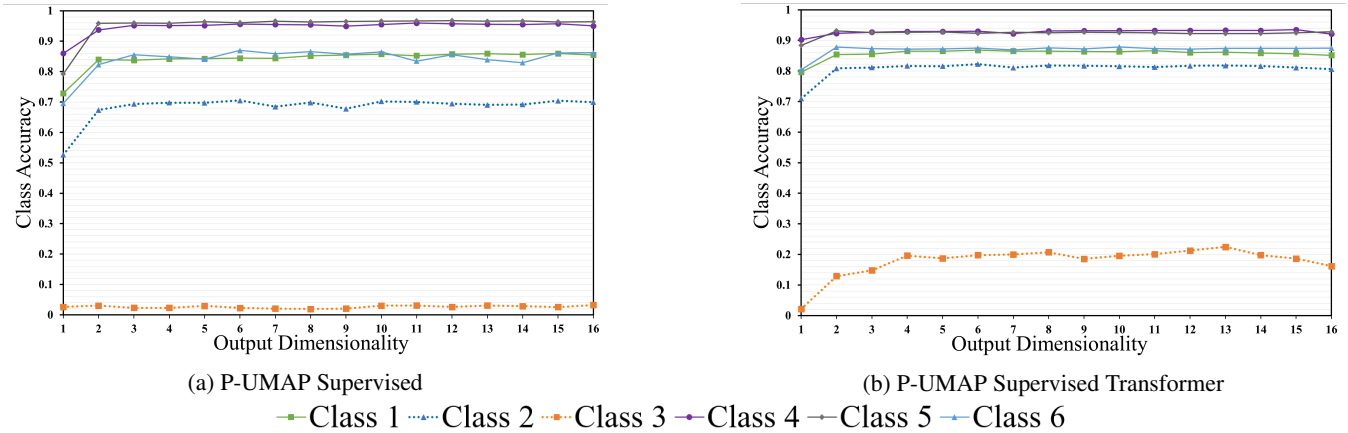
	Dataset			
Algorithm	20-NG	TREC-6	TREC-50	AG News
KMeans (No DR)	0.517	0.535	0.31	0.85
LDA	0.335	0.70	0.228	0.904
PCA	0.229	0.529	0.211	0.779
t-SNE	0.515	0.367	0.203	0.794
UMAP	0.582	0.388	0.219	0.809
P-UMAP	0.579	0.392	0.22	0.84
UMAP Supervised	0.668	0.746	0.379	**0.92**
P-UMAP Supervised	0.562	0.755	0.285	0.866
P-UMAP NN+Attention	0.595	0.752	0.334	0.913
P-UMAP Transformer	**0.698**	**0.839**	**0.427**	0.911

Overall, the results of these experiments are of value to any applications of the parametric UMAP transformer-encoder to dimensionality reduction in downstream tasks, as the model requires fewer trainable parameters, compared to the RNN with attention.

C. ANALYSIS OF PER-CLASS ACCURACY

In section III-A, we discussed the presence of a considerable imbalance in the TREC-6, and TREC-50 datasets. Given the supervised learning nature of our methodology, we consider it essential to explore the performance of the proposed transformer-encoder, given this imbalance. We focus upon the proposed transformer-encoder pipeline, as it is apparent, based upon the outcomes of our experiments, that this architecture attains a higher accuracy when compared to the contending pipeline proposed, based on RNN with attention. In any form of supervised learning, there exists the possibility of overfitting, where a model fails to generalise upon unseen data, based on the training data provided [68], [69]. In this case, any supervised derivative of the UMAP algorithm would be susceptible to this phenomena. This is of particular importance when working with highly imbalanced data, such as the TREC-6 and TREC-50 datasets, where the sampling of a training set from the data, which entails only 6000 rows, would lead to an underrepresentation of many of the classes. To facilitate this, we evaluate the accuracy of the supervised transformer-encoder, and compare this to parametric UMAP supervised. While an analysis of the effects of all configurations of UMAP, and the other algorithms used in our preliminary analysis, would be beneficial, it does not fall within the main focus of our study, which is the investigation of the transformer-encoder. Therefore, we focus upon our comparison with the supervised parametric UMAP algorithm. This algorithm, in its default configuration, consists solely of fully-connected layers, thereby offering the most comparable algorithmic structure. The only point of divergence of our transformer-encoder proposal lies in the architecture of the neural network. In Figure 8, the average per-class accuracy is presented for the transformer-encoder, and the default configuration of parametric UMAP, for the TREC-6 dataset, showing the accuracy for each individual class across a varying output dimensionality. As with all other experiments in this study, this average is calculated based on 25 individual iterations of the dimensionality reduction and clustering pipeline.

Based on an analysis of the accuracy of each class within the TREC-6 dataset, it is apparent that the transformer-encoder confers an improvement to the accuracy of some classes. At an output dimensionality of 3, when comparing the accuracy of class 1 between the transformer-encoder ($SD = 0.024$, $M = 0.85$), and default supervised parametric UMAP architecture ($SD = 0.034$, $M = 0.84$), we identify a p value of 0.02, indicating no significant difference between results ($U = 187$, z-score $= 2.42$, $p = 0.02$). For class 2, there is an improvement of 11.8% for the transformer-encoder approach ($SD = 0.012$, $M = 0.81$) compared to supervised parametric UMAP ($SD = 0.045$, $M = 0.696$), which demonstrates a significant increase ($U = 0$, z-score $= 6.054$, $p < 0.001$). Of particular interest regarding this comparison, is class 3, where the introduction of the transformer-encoder model demonstrates an increase in the average accuracy of this class, after k-Means clustering. For example, at an output dimensionality of 3, this improves from an average accuracy of 2.3% for parametric UMAP ($SD = 0.016$, $M = 0.023$), to 14.8% for the transformer-encoder ($SD = 0.075$,



(a) P-UMAP Supervised

(b) P-UMAP Supervised Transformer

FIGURE 8: Per-Class Accuracy of k-Means Clustering Performed Upon across all Dimensionalities, for the TREC-6, dataset, comparing our proposed pipeline, P-UMAP Supervised Transformer, with the current state of the art, P-UMAP Supervised.

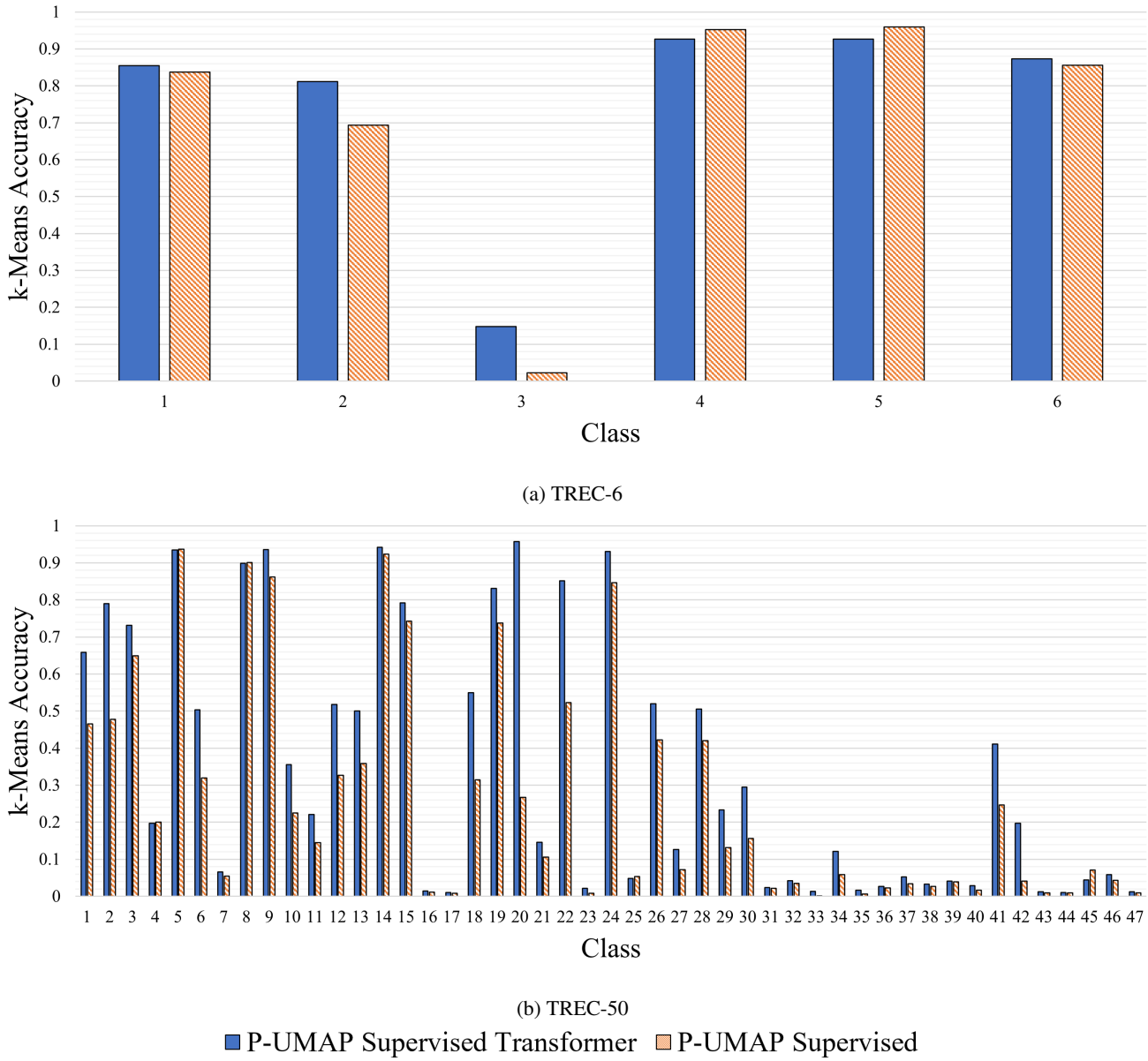
$M = 0.141$) derived model, a significant increase of 12.5% ($U = 102$, z-score $= 4.07$, $p < 0.001$). At an output dimensionality of 4, the transformer-encoder ($SD = 0.043$, $M = 0.19$) demonstrates a significantimprovement relative to parametric UMAP supervised ($SD = 0.016$, $M = 0.023$) of 16.6% ($U = 5$, z-score $= 5.95$, $p < 0.001$). This is a considerable improvement, given that class 3 of the TREC-6 dataset contains only 95 documents and represents the highest degree of imbalance. It is also worth noting that this underrepresented class also observes the greatest degree of change in accuracy as the dimensionality increases for the transformer-encoder, as denoted by the orange line in Figure 8. Returning our focus to an output dimensionality of 3, class 4 observes a decrease in the accuracy of transformer-encoder ($SD = 0.013$, $M = 0.928$) compared to supervised parametric UMAP ($SD = 0.026$, $M = 0.952$) of 2.3% ($U = 110$, z-score $= 3.92$, $p < 0.001$), which is significant based on the analysis using the WMW test. For class 5, the transformer-encoder confers a significant decrease in accuracy compared to supervised parametric UMAP of 3.3% ($U = 25$, z-score $= 5.56$, $p < 0.001$). Finally, for class 6, there is no significant difference observed based on the 25 individual experiments between the transformer-encoder, and supervised parametric UMAP ($U = 291$, z-score $= 0.4$, $p = 0.03$).

Taking the assumption that an output dimensionality of 3 is the typical point at which the "knee" is observed in the UMAP algorithm in Figure 7, beyond which there are diminishing returns in the clustering result when UMAP derivatives are used for DR, we present the per-class accuracy for the transformer-encoder, and default configuration of supervised parametric UMAP at an output dimensionality of 3 in Figure 9. A full reference of the results at other dimensionalities is provided within the repository[1]. As there are too many classes in TREC-50 to analyse within this work succinctly, we sample some which we argue merit discussion. For instance, class 20, which is assigned to only 50 documents within the dataset, observes a significant improvement for the transformer encoder of 69% ($U = 0$, z-score $= 6.054$, $p < 0.001$), based on our average of 25 experiments. Similarly, class 41 also represents 50 documents, and it can be observed that the transformer-encoder leads to a significant increase of 16.5% ($U = 24.5$, z-score $= 5.578$, $p < 0.001$). In comparison, for class 5, the largest within the dataset, there is no significant difference between the accuracy of both models ($U = 311.5$, z-score $= 0.009$, $p < 0.95$).Based on the analyses of individual class accuracy, a hypothesis may be formed that *the introduction of the transformer-encoder architecture within parametric UMAP confers a degree of robustness to the sensitivity of imbalanced data when used in a supervised metric-learning methodology*.

V. CONCLUSION

This work has investigated how two neural network architectures can affect the accuracy of downstream clustering tasks, when used in a parametric UMAP dimensionality reduction pipeline, namely the P-UMAP Supervised RNN with Attention, and the P-UMAP Supervised Transformer. Our analysis highlights several interesting findings. Firstly, we provide an empirical investigation into the effects of "traditional" dimensionality reduction algorithms PCA, LDA, t-SNE, and UMAP upon downstream k-Means clustering upon four benchmark datasets. Through our evaluation of these traditional algorithms across a range of dimensionalities, we demonstrate the effectiveness of each with respect to the output dimensionality, as well as discussing the benefits and disadvantages of each technique in relation to computational complexity. Furthermore, in our subsequent analysis of UMAP, Parametric UMAP, and their supervised-learning alternatives, we empirically identify a consistently observable "knee" curve in relation to the accuracy of k-Means clustering upon the low-dimensional representations produced by variations of the UMAP algorithm. This finding is beneficial for researchers or industry who seek to perform any clustering of embeddings, as we have demonstrated that

This article has been accepted for publication in IEEE Access. This is the author's version which has not been fully edited and
content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2024.3403991



(a) TREC-6

(b) TREC-50

FIGURE 9: Per-Class Accuracy of k-Means Clustering Performed Upon across all Dimensionalities, for the TREC-6 and TREC-50 datasets, comparing our proposed pipeline, P-UMAP Supervised Transformer, with the current state of the art, P-UMAP Supervised.

there is no benefit from opting for a large output dimensionality when using the UMAP algorithm as a preprocessing step prior to downstream clustering, where it is evident that an output dimensionality of 2 or 3 is suitable.

Through an investigation across a range of different dimensionalities, we have identified that attention mechanisms can have a significant effect upon clustering accuracy, when used in a metric learning framework within parametric UMAP. Moreover, our second proposed architecture, entailing a transformer-encoder, achieved the best overall improvement across three of the four datasets used in our investigation. Through a visual analysis of the lower-dimensionality embeddings produced by the transformer-encoder, we are able to demonstrate the effectiveness of the transformer-encoder pipeline for downstream clustering for a parametric UMAP dimensionality reduction pipeline when used for metric-learning, when compared to several other architectures. Additionally, we demonstrate that the transformer-encoder ensures an improvement in accuracy, while also maintaining significantly fewer trainable model parameters compared to an RNN, which extends our findings from investigating **(RQ4)**.

Through an analysis of the accuracy of individual classes within a dataset, it is apparent that the transformer-encoder

architecture confers a benefit when faced with imbalanced data. More specifically, underrepresented classes have been found to benefit from the introduction of the transformer-encoder. These findings open novel avenues of research, particularly in relation to the individual components of the architecture, which confer the robustness of the model in handling underrepresented data.

Our research has demonstrated the feasibility of *our proposed pipeline, employing a transformer-encoder within a Parametric UMAP metric learning framework*. However, there are still unresolved issues and unanswered questions which may affect researchers and real-world applications. Firstly, there is a considerable increase in the training time and computing requirements of the transformer-encoder architecture, which is best represented by the number of trainable parameters of the architecture. While the low-dimensional representations produced by the proposed UMAP architecture can contribute to a reduction in the time required for clustering, it is worth considering that the large number of trainable parameters of the architecture confers a longer time required for preprocessing. This could, however, be mitigated through optimisation strategies such as parallelisation and GPU training of the neural network architecture. Furthermore, our study has focused upon a metric-learning methodology, wherein we have used a small portion of labelled data to contribute to the dimensionality reduction process. For many real-world applications of our methodology, labelled data may not be available. Therefore, it would be beneficial for future works to investigate how the architecture proposed would perform when applied in an unsupervised manner. Additionally, we have conducted our study with a focus upon the downstream clustering of low-dimensionality vectors by the k-Means algorithm. While this clustering technique is widely known, the algorithm requires the prior specification of a known number of clusters, and as such is not suitable for applications, where the number of clusters is not known. Consequently, it remains to be determined if the transformer-encoder architecture can be utilized for calculating low-dimensional representations and subsequently subjected to alternative clustering algorithms. Among these, the potential enhancement of results through the application of density-based clustering methods such as HDBSCAN [24], [25] or DBSCAN [70], which have been demonstrated in tandem with UMAP for topic modelling, is of particular interest [29]. Aside from our contribution of showing how to apply a transformer-encoder to parametric UMAP, we have provided further experimentation into the outcomes of using various existing dimensionality reduction algorithms to contribute to improvements to clustering accuracy. Most notably, we demonstrate how the combination of the transformer-encoder with metric learning, when using parametric UMAP, can provide significant improvements to the clustering solution. Finally, we provide a repository containing our two proposed architectures and all algorithms used in this study, for further studies to validate and compare against the results presented in our paper[1].

VI. FUTURE WORKS

In this study, metric-learning was performed upon a subset of 20% of the overall dataset, which is provided along with the class-labels, such that learning is conducted only upon this sample. However, when using UMAP, one alternative is to provide masked labels for data where a label which is not known. In this case, a sample of 20% of labels could be provided, with a larger sample entailing unlabelled data also being used in the training process. This merits a further investigation in how to address overfitting in undersampled classes, however does not fall into the main scope of the goal with this paper. Thus, in future works it is worth considering the implications of this alternative training strategy in improving the performance of downstream clustering.

To further contribute to the findings that the proposed transformer-encoder pipeline with parametric UMAP confers a benefit to underrepresented classes, and an improvement in downstream clustering accuracy in general, we define two avenues of research. Firstly, from a supervised metric-learning perspective, an in-depth ablation study of the transformer-encoder architecture, evaluated in relation to per-class accuracy, could contribute to reinforcing our findings. This could be applied to domains other than the clustering of text embeddings. Secondly, from an unsupervised learning perspective, it would be beneficial to investigate the performance of the transformer-encoder when applied outside of the metric-learning methodology used in this study. This could be evaluated similarly to the manner of this study, based on accuracy. Alternatively, there remain open questions as to the effects that the clustering method selected, k-Means, has had upon the outcomes of the study. Subsequent investigations could apply the transformer-encoder within parametric UMAP in conjunction with different clustering algorithms.

REFERENCES

[1] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," Advances in neural information processing systems, vol. 30, 2017.

[2] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," arXiv preprint arXiv:1810.04805, 2018.

[3] A. Wang, A. Singh, J. Michael, F. Hill, O. Levy, and S. R. Bowman, "Glue: A multi-task benchmark and analysis platform for natural language understanding," arXiv preprint arXiv:1804.07461, 2018.

[4] M. Munikar, S. Shakya, and A. Shrestha, "Fine-grained sentiment classification using bert," in 2019 Artificial Intelligence for Transforming Business and Society (AITB), vol. 1, 2019, pp. 1–5.

[5] A. Subakti, H. Murfi, and N. Hariadi, "The performance of bert as data representation of text clustering," Journal of big Data, vol. 9, no. 1, pp. 1–21, 2022.

[6] R. Bellman, R. Corporation, and K. M. R. Collection, Dynamic Programming, ser. Rand Corporation research study. Princeton University Press, 1957. [Online]. Available: https://books.google.co.uk/books?id=wdtoPwAACAAJ

[7] K. Beyer, J. Goldstein, R. Ramakrishnan, and U. Shaft, "When is "nearest neighbor" meaningful?" in Database Theory — ICDT'99, C. Beeri and P. Buneman, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 1999, pp. 217–235.

[8] I. Assent, "Clustering high dimensional data," WIREs Data Mining and Knowledge Discovery, vol. 2, no. 4, pp. 340–350, 2012. [Online]. Available: https://wires.onlinelibrary.wiley.com/doi/abs/10.1002/widm.1062

[9] C. C. Aggarwal, A. Hinneburg, and D. A. Keim, "On the surprising behavior of distance metrics in high dimensional space," in International conference on database theory. Springer, 2001, pp. 420–434.

[10] M. Steinbach, L. Ertöz, and V. Kumar, "The challenges of clustering high dimensional data," New directions in statistical physics: econophysics, bioinformatics, and pattern recognition, pp. 273–309, 2004.

[11] L. McInnes, J. Healy, and J. Melville, "Umap: Uniform manifold approximation and projection for dimension reduction," arXiv preprint arXiv:1802.03426, 2018.

[12] M. Allaoui, M. L. Kherfi, and A. Cheriet, "Considerably improving clustering algorithms using umap dimensionality reduction technique: A comparative study," in Image and Signal Processing, A. El Moataz, D. Mammass, A. Mansouri, and F. Nouboud, Eds. Cham: Springer International Publishing, 2020, pp. 317–325.

[13] T. Sainburg, L. McInnes, and T. Q. Gentner, "Parametric umap embeddings for representation and semisupervised learning," Neural Computation, vol. 33, no. 11, pp. 2881–2907, 2021.

[14] L. van der Maaten, E. O. Postma, and J. van den Herik, "Dimensionality reduction: A comparative review," 2009.

[15] S. Wold, K. Esbensen, and P. Geladi, "Principal component analysis," Chemometrics and intelligent laboratory systems, vol. 2, no. 1-3, pp. 37–52, 1987.

[16] R. A. FISHER, "The use of multiple measurements in taxonomic problems," Annals of Eugenics, vol. 7, no. 2, pp. 179–188, 1936. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1111/j.1469-1809.1936.tb02137.x

[17] L. Van der Maaten and G. Hinton, "Visualizing data using t-sne." Journal of machine learning research, vol. 9, no. 11, 2008.

[18] G. E. Hinton and S. T. Roweis, "Stochastic neighbor embedding," in NIPS, 2002.

[19] F. Anowar, S. Sadaoui, and B. Selim, "Conceptual and empirical comparison of dimensionality reduction algorithms (pca, kpca, lda, mds, svd, lle, isomap, le, ica, t-sne)," Computer Science Review, vol. 40, p. 100378, 2021. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1574013721000186

[20] L. Van Der Maaten, "Learning a parametric embedding by preserving local structure," in Artificial intelligence and statistics. PMLR, 2009, pp. 384–391.

[21] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," Proceedings of the IEEE, vol. 86, no. 11, pp. 2278–2324, 1998.

[22] T. Joachims, "A probabilistic analysis of the rocchio algorithm with tfidf for text categorization," in ICML, 1997.

[23] J. MacQueen, "Classification and analysis of multivariate observations," in 5th Berkeley Symp. Math. Statist. Probability, 1967, pp. 281–297.

[24] L. McInnes, J. Healy, and S. Astels, "hdbscan: Hierarchical density based clustering." J. Open Source Softw., vol. 2, no. 11, p. 205, 2017.

[25] R. J. Campello, D. Moulavi, and J. Sander, "Density-based clustering based on hierarchical density estimates," in Pacific-Asia conference on knowledge discovery and data mining. Springer, 2013, pp. 160–172.

[26] C. Rasmussen, "The infinite gaussian mixture model," Advances in neural information processing systems, vol. 12, 1999.

[27] A. Subasi, "Chapter 7 - clustering examples," in Practical Machine Learning for Data Analysis Using Python, A. Subasi, Ed. Academic Press, 2020, pp. 465–511. [Online]. Available: https://www.sciencedirect.com/science/article/pii/B9780128213797000072

[28] J. Hull, "A database for handwritten text recognition research," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 16, no. 5, pp. 550–554, 1994.

[29] D. Angelov, "Top2vec: Distributed representations of topics," arXiv preprint arXiv:2008.09470, 2020.

[30] E. Becht, L. McInnes, J. Healy, C.-A. Dutertre, I. W. Kwok, L. G. Ng, F. Ginhoux, and E. W. Newell, "Dimensionality reduction for visualizing single-cell data using umap," Nature biotechnology, vol. 37, no. 1, pp. 38–44, 2019.

[31] A. Diaz-Papkovich, L. Anderson-Trocmé, and S. Gravel, "A review of umap in population genetics," Journal of Human Genetics, vol. 66, no. 1, pp. 85–91, 2021.

[32] S. Grossberg, "Recurrent neural networks," Scholarpedia, vol. 8, no. 2, p. 1888, 2013, revision #138057.

[33] S. Hochreiter and J. Schmidhuber, "Long Short-Term Memory," Neural Computation, vol. 9, no. 8, pp. 1735–1780, 11 1997. [Online]. Available: https://doi.org/10.1162/neco.1997.9.8.1735

[34] S. Fernández, A. Graves, and J. Schmidhuber, "An application of recurrent neural networks to discriminative keyword spotting," in Proceedings of the 17th International Conference on Artificial Neural Networks, ser. ICANN'07. Berlin, Heidelberg: Springer-Verlag, 2007, p. 220–229.

[35] A. Graves and J. Schmidhuber, "Offline handwriting recognition with multidimensional recurrent neural networks," in Advances in Neural Information Processing Systems, D. Koller, D. Schuurmans, Y. Bengio, and L. Bottou, Eds., vol. 21. Curran Associates, Inc., 2008. [Online]. Available: https://proceedings.neurips.cc/paper/2008/file/66368270ffd51418ec58bd793f2d9b1b-Paper.pdf

[36] M. Schuster and K. K. Paliwal, "Bidirectional recurrent neural networks-signal processing, ieee transactions on," 1998.

[37] A. Graves, N. Jaitly, and A. rahman Mohamed, "Hybrid speech recognition with deep bidirectional lstm," 2013 IEEE Workshop on Automatic Speech Recognition and Understanding, pp. 273–278, 2013.

[38] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," arXiv preprint arXiv:1409.0473, 2014.

[39] M.-T. Luong, H. Pham, and C. D. Manning, "Effective approaches to attention-based neural machine translation," arXiv preprint arXiv:1508.04025, 2015.

[40] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov, "Roberta: A robustly optimized bert pretraining approach," arXiv preprint arXiv:1907.11692, 2019.

[41] X. Li and D. Roth, "Learning question classifiers," in Proceedings of the 19th International Conference on Computational Linguistics - Volume 1, ser. COLING '02. USA: Association for Computational Linguistics, 2002, p. 1–7. [Online]. Available: https://doi.org/10.3115/1072228.1072378

[42] J. Liang, L. Bai, C. Dang, and F. Cao, "The k-means-type algorithms versus imbalanced data distributions," IEEE Transactions on Fuzzy Systems, vol. 20, no. 4, pp. 728–745, 2012.

[43] H. Xiong, J. Wu, and J. Chen, "K-means clustering versus validation measures: a data distribution perspective," in Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining, 2006, pp. 779–784.

[44] M. Galar, A. Fernandez, E. Barrenechea, H. Bustince, and F. Herrera, "A review on ensembles for the class imbalance problem: bagging-, boosting-, and hybrid-based approaches," IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews), vol. 42, no. 4, pp. 463–484, 2011.

[45] G. E. Batista, R. C. Prati, and M. C. Monard, "A study of the behavior of several methods for balancing machine learning training data," ACM SIGKDD explorations newsletter, vol. 6, no. 1, pp. 20–29, 2004.

[46] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "Smote: synthetic minority over-sampling technique," Journal of artificial intelligence research, vol. 16, pp. 321–357, 2002.

[47] O. Sitompul, E. Nababan et al., "Optimization model of k-means clustering using artificial neural networks to handle class imbalance problem," in IOP conference series: materials science and engineering, vol. 288, no. 1. IOP Publishing, 2018, p. 012075.

[48] Y. Qian, Y. Liang, M. Li, G. Feng, and X. Shi, "A resampling ensemble algorithm for classification of imbalance problems," Neurocomputing, vol. 143, pp. 57–67, 2014. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0925231214007644

[49] N. V. Chawla, D. A. Cieslak, L. O. Hall, and A. Joshi, "Automatically countering imbalance and its empirical relationship to cost," Data Mining and Knowledge Discovery, vol. 17, pp. 225–252, 2008.

[50] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: a simple way to prevent neural networks from overfitting," The journal of machine learning research, vol. 15, no. 1, pp. 1929–1958, 2014.

[51] D. Cai, X. He, and J. Han, "Training linear discriminant analysis in linear time," in 2008 IEEE 24th international conference on data engineering. IEEE, 2008, pp. 209–217.

[52] I. M. Johnstone and A. Y. Lu, "Sparse principal components analysis," arXiv preprint arXiv:0901.4392, 2009.

[53] N. Pezzotti, A. Mordvintsev, T. Hollt, B. Lelieveldt, E. Eisemann, and A. Vilanova, "Linear tsne optimization for the web," arXiv preprint arXiv:1805.10817, vol. 2, 2018.

[54] J. Barnes and P. Hut, "A hierarchical o (n log n) force-calculation algorithm," nature, vol. 324, no. 6096, pp. 446–449, 1986.

[55] L. van der Maaten, "Barnes-hut-sne," 2013.

[56] W. Dong, C. Moses, and K. Li, "Efficient k-nearest neighbor graph construction for generic similarity measures," in Proceedings of the 20th international conference on World wide web, 2011, pp. 577–586.
[57] M. Inaba, N. Katoh, and H. Imai, "Applications of weighted voronoi diagrams and randomization to variance-based k-clustering," in Proceedings of the tenth annual symposium on Computational geometry, 1994, pp. 332–339.
[58] J. Bergstra, R. Bardenet, Y. Bengio, and B. Kégl, "Algorithms for hyper-parameter optimization," in Proceedings of the 24th International Conference on Neural Information Processing Systems, ser. NIPS'11. Red Hook, NY, USA: Curran Associates Inc., 2011, p. 2546–2554.
[59] Y. Ozaki, Y. Tanigaki, S. Watanabe, and M. Onishi, "Multiobjective tree-structured parzen estimator for computationally expensive optimization problems," in Proceedings of the 2020 Genetic and Evolutionary Computation Conference, ser. GECCO '20. New York, NY, USA: Association for Computing Machinery, 2020, p. 533–541. [Online]. Available: https://doi.org/10.1145/3377930.3389817
[60] Y. Ozaki, Y. Tanigaki, S. Watanabe, M. Nomura, and M. Onishi, "Multiobjective tree-structured parzen estimator," Journal of Artificial Intelligence Research, vol. 73, pp. 1209–1250, 2022.
[61] T. Akiba, S. Sano, T. Yanase, T. Ohta, and M. Koyama, "Optuna: A next-generation hyperparameter optimization framework," in Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining, 2019, pp. 2623–2631.
[62] K. Fukushima, "Visual feature extraction by a multilayered network of analog threshold elements," IEEE Transactions on Systems Science and Cybernetics, vol. 5, no. 4, pp. 322–333, 1969.
[63] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 770–778.
[64] S. Basodi, C. Ji, H. Zhang, and Y. Pan, "Gradient amplification: An efficient way to train deep neural networks," Big Data Mining and Analytics, vol. 3, no. 3, pp. 196–207, 2020.
[65] H. B. Mann and D. R. Whitney, "On a Test of Whether one of Two Random Variables is Stochastically Larger than the Other," The Annals of Mathematical Statistics, vol. 18, no. 1, pp. 50 – 60, 1947. [Online]. Available: https://doi.org/10.1214/aoms/1177730491
[66] J. Munkres, "Algorithms for the assignment and transportation problems," Journal of the society for industrial and applied mathematics, vol. 5, no. 1, pp. 32–38, 1957.
[67] M. Cohen and J. Arthur, "Randomization analysis of dental data characterized by skew and variance heterogeneity," Community Dentistry and Oral Epidemiology, vol. 19, no. 4, pp. 185–189, 1991.
[68] X. Ying, "An overview of overfitting and its solutions," in Journal of physics: Conference series, vol. 1168. IOP Publishing, 2019, p. 022022.
[69] D. M. Hawkins, "The problem of overfitting," Journal of chemical information and computer sciences, vol. 44, no. 1, pp. 1–12, 2004.
[70] M. Ester, H.-P. Kriegel, J. Sander, X. Xu et al., "A density-based algorithm for discovering clusters in large spatial databases with noise," in kdd, vol. 96, no. 34, 1996, pp. 226–231.

RYAN HODGSON is studying a PhD in data science at Durham University. The main focus of his research is investigating the applications of predictive and prescriptive analytic techniques, and how these can impact upon the media and publication industry. As part of this, a significant focus has been conducted into investigations of unsupervised learning techniques within NLP, which include Topic Modelling, Clustering, and Dimensionality Reduction techniques and their impact upon downstream analysis tasks. In industry, he has contributed to the research and development of information retrieval technologies with the industry sponsor Reveela Technologies.

JINGYUN WANG is an assistant professor in the CS department of Durham University since September 2020. She is a member of Artificial Intelligence and Human Systems Group (AIHS) and a member of Pedagogical Innovation in Computer Science Group(PICS). She is also a core academic member of Centre For Neurodiversity & Development. Before that, she was an Assistant Professor (2014-2020) at the Research Institute for Information Technology, Kyushu University, Japan. Her current research focuses on ontology, visualization learning support systems, meaningful learning environments, personalized language learning support systems, game-based learning, computational thinking education, data science, learning analytics, and AI-based learning support.

ALEXANDRA I. CRISTEA is Professor, Deputy Executive Dean of the Faculty of Science, Director of Research and Founder of the Artificial Intelligence in Human Systems research group in the Department of Computer Science at Durham University. She is Advisory Board Member at the Ustinov College, Alan Turing Academic Liaison for Durham, N8 CIR Digital Humanities team lead for Durham and member of the IEEE European Public Policy on ICT. Her research includes web science, learning analytics, user modelling and personalisation, semantic web, social web, authoring, with over 300 papers on these subjects (over 5700 citations on Google Scholar, h-index 40). Especially, her work on frameworks for adaptive systems has influenced many researchers and is highly cited (with the top paper with over 220 citations). She was classified within the top 50 researchers in the world in the area of educational computer-based research according to Microsoft Research (2015-02-10). Prof. Cristea has been highly active and has an influential role in international research projects.

JOHN GRAHAM is the CEO of Reveela Technologies, a platform providing cutting edge artificial intelligence solutions for the MarComms, PR and Media industries. His work investigates industry changing, hyper personalised experiences for marketers, journalists and trade media across the globe. John is currently partnered through an Intensive Industrial Innovation Partnership (IIIP) with Durham University aimed at addressing issues in the publishing industry through predictive and prescriptive analytics technologies.