Abdur Rafay Saleem
ab464825@ucf.edu

***Abstract*—This review covers a new technique introduced by a paper for Deep Metric Learning. We review how technical details of this technique are different and why it is more efficient than others. The paper will be in three sections: the research problem, technical details, and a critical analysis.**

I. RESEARCH PROBLEM

Multimedia retrieval is very common in applications that exist today. Users are performing visual based searches many times a day without even realizing it. The technology that goes behind making this possible involves many different theorems and algorithms. One such technique is called Deep Metric Learning (DML). This technique involves training the model on existing image datasets to predict visual results for multimedia queries. It has been useful in computer vision related applications. However, improving the accuracy of these models is a huge challenge due to the vast variety and volume of visual data. Previous practice has been to perform full fine-tuning of the model on large image datasets. But, this is not always ideal as it requires huge computational resources, time, and can result in overfitting on inappropriate data. Moreover, there is a huge gap in the raw data the model was trained on and the local domain data it is applied on. If the model is trained on the local data, it might lose its knowledge provided by the larger pre-train data. Some research efforts have been made to find alternate ways of improving the model by tuning only certain parameters. However, the authors claim that parameter-efficient tuning has not been fully investigated for DML tasks.

The paper is about making a system smarter in understanding and categorizing images. This system uses something called Vision Transformers (ViTs) to help the system understand images better. But these ViTs are complex and need fine-tuning to work well. For this, the authors suggest the use of Visual-Prompt Tuning (VPT). Furthermore, the authors propose a new method to gradually mix in more information (semantic embeddings) from the same category of images. This is done by feeding the new images and the information they contain into a system called recurrent neural network.

The paper shows that this new method works better than the old method in terms of learning efficiency and performance. It also shows that by adjusting only a small part of the system, their new method can perform as well as the most advanced methods that adjust the whole system. This is a big deal because it means less computing power is needed and that saves time and resources. Finally, comparison is done between their new method and several other ViT tuning methods. They prove that their method works well in improving the system's ability to understand and categorize images.

II. TECHNIQUES

The following are some important concepts that are core to the proposed methodology.

Deep Metric Learning (DML)

It is a method used to make sense of and sort images. It uses a function to project the input data samples to a hidden embedding space. The goal of DML is to refine this projection function. This function is usually constructed with CNN or Transformers. Once the data’s features are projected, we can easily measure it with a specific distance metric that is based on the semantic similarity between image samples.

The paper talks about using proxies to make DML efficient. Proxies are learnable representations of subsets or categories of data samples. The paper discusses different methods such as Proxy-NCA and Proxy-Anchor to optimize these proxies. Eventually, it adopts the Proxy-Anchor as the main loss because it works robustly on various extended works.

Vision Transformers (ViTs)

ViTs work by breaking down an image into smaller pieces, called patches, and then analyzing these patches to understand the image. One way to fine-tune them is Adapter Fine Tuning. It involves adjusting a group of models alongside the Transformer. Another way is Bitfit Fine Tuning. It involves only the Bias factor of each linear projector. This tuning has been included in the framework due to its plug-and-play nature.

Visual Prompt Tuning (VPT)

This method involves using additional learnable prompts as extra data input. These prompts are like hints or clues that help the system understand the images better. The prompts are combined with the original visual embeddings and are processed within the transformer blocks.

Proposed Methodology

The authors methodology is designed around optimizing the visual prompts before sending them between layers of the ViT. In the context of ViTs, the model processes an image through several layers known as blocks for understanding and representation. The final block layer is the last layer in this process, and it outputs a set of prompts. These prompts are called VPTs. The prompts from the final block layer are then sent to the decision-making part of the model. This part is called the head layer. The head layer is a part of the model that makes the final decisions based on the information it receives from the previous layers. It's like the brain of the model.

In the PA loss method, the computer's understanding of an image (the data sample) is compared with a reference point (the proxy anchor) for each category of images. This is different from other methods where each data sample is compared with every other data sample. So instead of comparing each image with every other image, they compare each image with a reference image for its category. This makes the process more efficient and manageable. However, there is a problem with the way these proxies are randomly initialized. They don't have any meaningful information, called semantics, about the images. This means that the updating of these proxies is not very effective. This issue is more significant during the early training. To solve this problem, the authors tried to give the proxies a head start by initializing them with the same input data that the sample encoder uses. The sample encoder is the part of the system that first processes the images. So, instead of starting the proxies from scratch, they tried to start them off with some basic understanding of the images. This way the proxies would already have some semantics right from the start. However, this approach introduced a new problem where the proxies ended up being too similar for the same image samples. This made it hard for the system to distinguish between different images. They solved this by adding additional prompts based on the type of images.

Images are processed in batches and at various training periods. The quality of their semantic proxies can vary from image to image which makes it difficult to combine them and produce a common proxy for a class of images. The solution is built upon exponential moving average (EMA). It is a way of averaging data that gives more weight to recent data. The EMA method is used to blend each proxy with the average of its class to get a better proxy. However, it has some disadvantages. If the information (or vector) decreases at a steady rate, it might not be able to understand the structure of the data. This is because it might not have enough time to learn from all the variations in the data. Also, when new proxies are created, they might contain random or irrelevant information that could make the learning process less efficient and accurate. To solve this, the authors suggest a new recurrent way to gather and update semantic proxies. In each iteration, the current state and relationship with new information is used to update the proxy. They use a modification of the gated recurrent unit (GRU) as a method to carry this out. GRU is a model that can remember information over time. This is how it works:

1. It starts with a proxy to update.
2. In each step, they update this proxy using a set of equations. These equations involve several parameters that can be learned and adjusted as needed. These parameters help to project the input and the previous proxy to a hidden space. This hidden space is a place where the computer can process and understand the information it has learned so far.
3. These learnable parameters along with the new class proxy and hidden state from the previous proxy are used to update the current proxy. A class proxy is like a representative for a group of images that belong to the same category.
4. Then the ReLU function is applied instead of the Tanh function because it keeps the vector scale. The vector scale is a measure of how much the proxy changes in each update.
5. The gradient is not passed between batches. The gradient is a measure of how much the error changes with each update, and not passing it means they don't use this information to adjust the parameters.
6. Groups of images are taken from the same category. They are shuffled in random order added one by one into a special container called an accumulator. They do this for each group of images. The accumulator is like a container where they collect all the updated proxies.
7. The output is constructed by adding the original proxies in a certain ratio. The output representation is the final result that the computer uses to understand and categorize the images.
8. They then test out different ways of gathering and updating these proxies to see which method performs best.

Moreover, the number of tweakable settings is important. They noticed that having more settings in the early stages of the model was more beneficial, so they adjusted it to decrease with each step. Also, as more types of images are categorized, more memory is needed for the prompts. To manage this efficiently, they store these hints in a buffer in the GPU and move them from the CPU memory before each training round.

III. CRITICAL ANALYSIS

Improving the accuracy of machine learning models is a challenging task, especially for visual search applications. The authors introduced a smart technique to overcome the existing challenges and suggest a more practical and efficient solution. They proved several steps of the methodology using theorems and equations.

The experiments conducted by the authors follow industrial standards. They tested their modification on several large scale datasets, compared it against conventional benchmarks, and used existing evaluation metrics for ranking. They follow existing studies to apply various configurations on the training models. After producing the results, they performed a detailed comparative analysis and showed how their method is better in terms of performance and the number of parameters adjusted. The impact of this increased efficiency also reflects in the reduced computational costs and memory requirements.

However, the technique has some limitations and areas of improvement. The DML's performance heavily depends on the quality and diversity of the input data. If the data is biased or not diverse enough, the model might not perform well. Moreover, while the paper uses Proxy-Anchor, there might be other methods that could potentially yield better results. Also, like many deep learning models, DML might suffer from a lack of interpretability. It might be hard to understand why the model makes certain decisions, which could be a problem in certain applications. Future research focusing on integration of VPT and ViT with other machine learning techniques could lead to the development of interesting hybrid models with improved performance.

In summary, the framework introduced by the authors is quite impactful and has been nicely explained in detail. Every small challenge was considered in its implementation and the final method

was extensively tested against existing state-of-the art techniques. This will contribute a great deal towards making deep learning a more practical approach for multimedia retrieval problems.