**1. How can learning from Study 1 enable us to identify and then exploit an opportunity in the Machine learning field?**

The machine learning field has grown rapidly in recent years and more advancements have happened recently. Machine learning applications provide promising and reasonable solutions for many problems. There are so many opportunities in the machine learning field because of its potential and obvious outputs. This study helps us examine some of its major flaws and related issues in the software development phase. It is beneficial to know the common issues in the development because software development for machine learning requires both coding and data understanding skills. On the other hand, machine learning libraries are upgrading to solve many issues by resorting to robust functions but some practitioners don’t know about them. The practitioners who are capable to handle the code smells can implement the models more precisely. In general, companies look for models which are efficient and faster than the others. Most people analyse the efficiency but do not look at the coding level problems. There is a high demand for models which are capable to address the problems with low code smells.

We can increase the efficiency of a model by taking precautions in the development phase, reduction of these issues will enable the model to stand in the industry. The code smells are the least concern while building a model, but the reduction of these smells will help the developers and companies to build more robust and structured models. Machine learning models work better while coding but due to the code smells and anti-patterns the model may become inefficient and less productive. It is very hard to find the code smells at the end of the model. This can be minimized if the developer knows about different kinds of code smells and solutions to reduce them. Code smells affects the model’s efficiency, maintainability, and complexity. The primary code smells are ‘misusing’ and ‘lack of knowledge about better functions'. For example, the pipeline() function is useful for reducing the data leakage and this will help the issues downstream. Machine learning libraries provide less complex and easy-to-learn functions for solving complex problems. Exploring and implementing these functions increase the efficiency and accuracy of the model.

Using in-built pandas library functions will increase the efficiency of the code, for instance, it takes less time for execution. Pandas built-in functions will reduce the iterations and increase the speed. Efficient and proper memory usage plays a crucial role in machine learning models. This can be implemented with the help of different built-in functions such as detach(). There are many cases of misusing the functions and API’s. While implementing any functions, practitioners must take care of these parameters to avoid future problems. While handling the gradients, the developer should think about the order. Ignoring scaling will give inefficient results. Proper code smell handling can reduce the issues in the deployment phase. Controlling the generic code smells and API code smells gives us many benefits such as better performance, less error-proneness, better readability, robustness, and efficiency. So, learning these techniques is an advantage to both the practitioners and the companies.

**2. Write your thoughts about ”Opportunities implementing the process discussed in Study 1.”**

Machine learning code may generate outputs based on the context of the problem, but we need to identify the performance, efficiency, and understandability of the model. The model with code smells may give reasonable output with less efficiency. Many machine learning libraries provide easy solutions to these problems with in-built functions. Applying these solutions to a specific context will increase the performance and robustness of the model. Code smells are a major concern to any Machine Learning community/group. It is a concern to data scientists, machine learning application developers, machine learning library developers, code analysis tool developers, and students. This is a continuous process of learning and implementing different control methods to achieve the desired or optimum output. Implementing new and alternate functions can reduce the complexity and take less execution time. This process will help in increasing the efficiency, understandability, ease of use, and implementation of the machine learning models.

These solutions help us in dealing with the NaN values and none values. These techniques discussed above aid in better implementation of scalers and gradients in the machine learning models. These methods are even useful with memory, parameters, scaling, randomness, masking, training, evaluation, reducing data leakage, and mitigating misuse of functions. The proper use of these methods will succour Machine Learning practitioners in increasing the model's performance and reducing the execution time of the code. The major advantage of these techniques is reduced future problems, manageable, easy to learn, and time reduction. They can be used in different types of domains and for different problems. In addition, this process is more concentrated on the code-level problems and this will also help the model in the long run. This process will also help in building robust and efficient models with machine learning coding standards. This is a continuous process to sharpen the efficiency of the code and ease the problems in the production phase.

While doing software development for machine learning projects, we must also remind ourselves about the code smells and the related future issues that may occur in the long run. Many sources are providing promising solutions for these issues. Exploring these sources will help this machine learning developer community and data scientists in implementing code smell mitigation. The major sources for suggestions are grey literature, paper mining, and library selection. Grey literature is materials different from traditional commercial or academic publishing, for instance, online blog posts. There are many good and efficient sources for suggesting alternatives to these issues available from which more reliable functions can be studied and implemented empirically. There are also blogs written by the data practitioners which will help the students and new practitioners who have less knowledge or experience in the software development field. Sources like stack overflow and GitHub are provided with a lot of useful and practical examples and they are easily accessible publicly. From paper mining and the citations from these papers, we can gather many machine learning-specific code smells and solutions to these issues which are affecting the performance of our machine learning application models. The papers will give a deep understanding of the pitfalls a data scientist faces during the development of an application model. Finally, looking into the machine learning libraries, there are many libraries that can be used for different problems in the development of machine learning applications. Some libraries help more with specific types of issues, for example, Pandas library helps in dealing with iteration problems. Implementing these techniques on or before building a model will benefit the performance of the model.

**3. In the context of Study 2, what complement the theoretical results with an experiment?**

Many data practitioners think that increasing data size can increase the accuracy of the model and subsequently it is going to perform well. This assumption is a barrier for new companies to enter the competition. Most of the big companies accumulate a massive amount of data and they build models from that data to improve business value. Small companies need to build models with limited resources unlike already established big companies because the data is not publicly available. Time dependency of the data affects the scaling of the model and may give negative outcomes. Looking through practical terms in search of data, we are losing relevance over time. If we have a large dataset with less relevant information we cannot get the outputs correctly. In some business cases, the value changes over time when the data is sampled dynamically over time. The amount of information increases over time but it may not be useful for current trends. If the data of primitive telegram used is used for voice calls prediction, this data would not be suitable and creates errors in the model. A perfectly built and evaluated model will fail if we use irrelevant data. Any model built with the required amount of relevant data will give good results and it has the potential to stand in the competition.

In this study, the benefits and efficiency of the model are explained using a word prediction model. The data is collected about Reddit from 2006 to 2018. The data is sampled and checked efficiently using the maximum likelihood estimation method. The results show that increasing the data will increase the diversity and this will increase the cross-entropy. These metrics show that taking the overall data is not a good idea. Based on this experiment, we can understand that removing old and irrelevant data for better results is a preferable and effective way. The loss of memory size is balanced by the increase of relevance in the model. The data practitioners can decide which information has less relevance based on the gain/loss function. The word prediction model shows that using old words gives more cross-entropy. The tastes and slang of the people changes over time. So, building a model based on the current flow of data will give better predictions than building a model with old and new information.

On the other hand, if the old information has relevance to the current trends then it is better to keep it. This evaluation can be carried out using gain functions and the data can be processed using sequential offloading. In this study, the data which is 100 MB and useful during a time is less valued now with only 50 MB or less. So, keeping the relevant information and primarily concentrating on the current data flow increases the robustness of the model. These outputs recommend small firms build a model based on the relevant information. Machine learning and artificial models reduce the cost of prediction. By adopting these practices, small firms can give competition to the big companies and they could predict better.

**4. Briefly summarize your learning from Study 2 and how you can improve the overall process.**

Predictions from data depend on how the relevance of the data related to the context. Change in business values over time affects the predictions. The quantity of the data does not decide the efficiency of the model always but the relevance of data will increase the efficiency. Another important aspect of time-dependent models is the current flow of the data. The trends and nature of the data change over time. Selecting relevant information not only increases efficiency but also reduces errors. Models which change with time require evaluation of data before making the predictions. Newly collected relevant or updated resent data is an essential requirement for creating new models. Having infinitesimal data is not an advantage for a model and having old information sometimes harms the efficiency of the model. A perfectly built model also fails in the long run if the relevant information is not taken appropriately. The data which is very precious for the present model may not be useful or less valued for the same model in the future. This is because of the increase in diversity of the data due to time dependency. Filtering data for analysis play a key role in the efficiency of the model.

Data can be filtered based on the gain/loss functions. If the removal of data is beneficial to the model it is better to do it. Offloading algorithm is used to remove less important data by compromising loss in data with the benefit of gain in relevance. More data leads to higher accuracy of models, in other words, it makes better services. Better services result in higher user engagement which again creates a large amount of data. Historical data should be evaluated first for relevance because it may not be useful for the present model. In the long run, the model needs to adapt to the new trends. The variations and observations based on the time and relevance show promising results. For example, the literature changes with time and there is no need to consider the words used in 1900 for the word prediction model. There are so many literature changes and slang changes over time and the birth of the new words replaces some old words (like the word ‘covfefe’ there is no such word recorded ever before until former US president trump used it in his tweets multiple times). Using recent 3 or 5 years of data for word prediction gives reasonable predictions. These types of problems can be solved by using data relevance and adapting to the new flows. This practice can help in almost all dynamic models. By implementing this approach, small firms can enter the machine learning industry and build models which are efficient with the small size dataset.

This method explains the approach to dynamic models with monotonic losses and the flow of the data. This model is efficient for extendable arguments by changing the probability distribution. These study algorithms and evaluation methods eliminate most of the barriers to conducting accurate and consistent analysis. Training the model with different static learning rates and different batch sizes may improve the results. The major algorithm used for the analysis is Maximum Likelihood Estimation(MLE) but it is better to explore more types of algorithms for analysis because some models need a different approach. The trial and error method will help in this case. There are not many improvements to this model because the chosen algorithms are robust.

References:

|  |  |
| --- | --- |
| [1] | H. Zhang, L. Cruz and A. v. Deursen, “Code Smells for Machine Learning Applications,” Delft.  https://arxiv.org/pdf/2203.13746.pdf |
| [2] | E. Valavi, J. Hestness, N. Ardalani and M. Iansiti, “Time and the Value of Data,” 2020. |

https://arxiv.org/pdf/2203.09118.pdf