Machine learning product key performance indicators and alignment to model evaluation

Ioannis Bakagiannis
Department of Digital Systems
University of Thessaly
Larissa, Greece
ibakagiannis@uth.gr

Vassilis C. Gerogiannis
Department of Digital Systems
University of Thessaly
Larissa, Greece
vgerogian@uth.gr

George Kakarontzas
Department of Digital Systems
University of Thessaly
Larissa, Greece
gkakaron@uth.gr

Anthony Karageorgos
Faculty of Technology
University of Thessaly
Larissa, Greece
karageorgos@uth.gr

***Abstract*—Machine Learning has seen amazing progress the past years with increasing commercial use from industries across the business spectrum. Businesses strive for alignment of vision and mission statement to the actual products they sell. For that reason tools like the Key Performance Indicators exist in order to monitor such progress. Nevertheless, products that embed a machine learning component are being optimized with other objective functions and are being evaluated in a vacuum with specific performance evaluation metrics that often have nothing to do with the business vision. In this position paper, we highlight this gap in different instances of the machine learning life cycle, explore and critically evaluate the current available solutions in the literature and introduce Key Performance Indicators in the machine learning development process. The paper also discusses representative machine learning KPIs in the development and deployment process.**

***Index Terms*—machine learning, key performance indicators, performance measuring, performance monitoring**

I. Introduction and Problem description

For the past years we are seeing a Machine Learning revolution with an intense focus in research and commercialization of Machine Learning applications. Currently almost all commercial products have something ”smart” embedded or even the ”smart” component is the actual product. These ”smart” products find a natural home on the internet since technology products provide the algorithms with a lot of data that so much need. Examples of such products are the recommendation system of YouTube [1] and the pin recommender from Pinterest, which is called Pixie [2]. YouTube recommendations are responsible for helping more than a billion users discover personalized content from an ever-growing corpus of videos [1]. Pixie on the other hand, is the recommender system from Pinterest which is a is a visual catalog with several billion pins, which are visual bookmarks containing a description, a link, and an image or a video. A major problem faced at Pinterest is to provide personalized, engaging, and timely recommendations from a pool of 3+ billion items to 200+ million monthly active users [2].

Companies on the web or in the physical space though have the same goal: to succeed and remain successful for a long time. This success is described by the company’s vision and mission statement, and without them there would be no need for the company at all. Consequently, each company strives to achieve its strategic goals with their products and daily activities. Unfortunately, a company’s strategic goals are hardly ever measurable. Take the Apple’s mission statement as example: ”the company is committed to bringing the best user experience to its customers through its innovative hardware, software, and services”. However, it is often difficult or almost impossible for a company to know when its products/services ”bring the best user experience”. So companies, in order to measure and monitor their success, have invented proxies that are targeted to measurable components of their ultimate business goals. Nowadays, business monitoring is typically supported by an information system that provides information about several Key Performance Indicators (KPIs) [3], spread across one or more Balanced Scorecards [4] and multiple dashboards. Traditionally, business monitoring has been based on the evaluation of the aggregated values of KPIs by users who regularly check the scorecard in order to ensure that everything is in order. For example, the KPI “Customer retention increased by 3%” considers the customers retained throughout all stores of a company in the current year. In order to provide a more complete view for KPI monitoring, dashboards provide detailed information. For example, if the abovementioned KPI is failing, what has been its trend for the past months [5]?

Nowadays companies in order to achieve their vision and maximize their KPIs leverage machine learning and artificial intelligence in their products. From the opposing perspective machine learning applications have their own performance metrics such as accuracy, recall [6] and depending on the application their own KPIs. For example in [1] YouTube optimizes their recommender system for the KPI ”Click Through Rate”. As we mentioned above YouTube’s purpose is to help users discover personalized content from an ever-growing corpus of videos. This purpose is getting reduced to measuring precision and recall during development and user engagement proxies such as Click Through Rate and watch time during the live A/B testing experiments [1]. These layers of abstraction from the vision to the user engagement to the actual metrics we argue that add a lot of noise and assumptions and may not lead to the desired success.

Machine learning performance metrics and KPIs already exist in the industry and academia. Classic machine learning

performance metrics such as accuracy and recall are used extensively during the development process and contribute to the model selection decision [6]. Other performance metrics are specialized towards the domain of the problem when the classic metrics are not useful or even feasible. One example would be precision@k when recommender systems are being evaluated. This metric requires a value of k - usually 5 or 10 -, picks the k most likely recommendations for the specific model and measures the precision of those recommendations against the users preference [7]. Also machine learning KPIs exist such as "Click Through Rate" that are being measured when the model has gone through development and has been deployed into production [1].

Another aspect of a successful KPI is the continuous monitoring of the KPI value and its meaning [5]. In the industry performance monitoring is a crucial part of the machine learning life cycle. A common practice that [1] and [6] mention is A/B test for model selection during the deployment phase. A step further on that direction would be to use exploration vs exploitation techniques such as multi armed bandits [8] or using reinforcement learning directly [7].

Even though machine learning model evaluation and monitoring methods exist, there exists no practice or literature for connecting the business value to these methods. In the rest of the paper we explore this issue of non-alignment of business and model KPIs and discuss possible directions of resolution. Specifically we present KPIs that currently are being tied to machine learning applications and why are not adequate to ensure business success. Also we introduce three different approaches for aligning business value to the machine learning development, while analyzing their strengths and weaknesses.

The rest of the paper is organized as follows. In section II we present the notion of the Key Performance Indicator as a measure of business success. In section III we introduce the mechanics of training a machine learning model with an objective function. We present our contributions in section IV where we propose an alignment of business KPIs and machine learning objective functions. Also we analyze the potential solutions to this novel problem and we conclude in section V.

II. KPIs as measure of business success

Indicators or key performance indicators (KPIs) in business environment are mostly quantitative information; it illustrates structures and processes of a company. KPIs are very important for planning and controlling through supporting information, creating transparency and supporting management decision makers [9].

Normally, KPIs are defined together within a company's strategic plan in order to help analysts measure the success of their goals. While it is important to know why a past goal has failed or succeeded, it is even more important to monitor what the trend is for a current goal. Is the goal going to be successful or failed eventually? For example financial institutions such as Royal Bank of Canada have year over year sales growth in each quarter as an important goal and KPI. In order to understand if the company is performing well on that KPI RBC monitors the KPI quarterly and compares the result with past results and results of the competitors. At the same time this analysis helps define what would be our goal for the desired year over year growth in the future. Therefore, formally defined goals refer always to the future. This means that target values for KPIs associated with the measurement of goal, are always defined for the future [5]. Which enforces the need to monitor continuously the defined KPIs and make sure that the company is moving towards the right direction.

This also holds true for companies that leverage machine learning in their products. Since most machine learning applications are designed to predict something in the future or understand a behavior and act optimally they are measured primarily with leading indicators such as YouTube and Pinterest in their recommendation engines [2] [1]. Specifically YouTube leverages a recommendation engine to promote videos to users is using "Click Through Rate" and "Engagement Time" as main KPIs [1]. In some cases it is not that straightforward to define KPIs for a machine learning application. For example credit lender companies do not have an instantaneous feedback mechanism to know if their proposed rate is optimal. As a result the machine learning models suffer also since there is no way to measure the performance of the proposition. But again it is possible to define approximate KPIs such is "Percentage of Defaults" - which is a defensive KPI - and Percent Net Profit Gain [10]. Also it would be useful if companies could compare KPIs against the competition. Victor Basili has established the Goal-Question-Metric (GQM) [11] approach to software metrics and later expanded this to the GQM+Strategies approach [12]. We see an analogy here with the proposed work. More specifically a Goal can be a business goal as it is expressed through a KPI. The Questions are used for the definition of models of the object under study that are connected to the KPIs. Therefore, answering the questions can connect the machine learning algorithm performance to the KPI performance. In order to answer the question we need to invent metrics of the machine learning algorithm performance. We view therefore our proposed research agenda as an instantiation of these more generic software engineering approaches to AI. This link can provide a methodological foundation and a scientific approach already tested in the software industry in the field of software engineering, and transfer this knowledge and apply it in AI.

III. Machine Learning objective functions

Machine learning is trying to empower the industries and researchers towards more intelligent applications and, as we mentioned, has become a common tool that gives companies competitive edge in the past years. Generally, machine learning algorithms are either trying to predict an outcome or model some type of behaviour [13]. When the model is trying to predict a class label such as a zebra in a picture or a continuous number such as the price of a given stock, the model is doing supervised learning [14]. There is some known quantity or class from experience or past data that we want to replicate. On the other hand when the model is trying to find patterns

in the data while not having a target in mind, the model is doing unsupervised learning [14]. Unsupervised learning algorithms experience a dataset containing many features, then learn useful properties of the structure of this dataset.

How do the models actually learn though? Machine learning models, like humans, learn from experience. This experience is called training data and depending on the task at hand could take a lot of different forms [13]. For example in [2] Pinterest is trying to predict which pins - items from a visual catalog - each user is likely to interact with. For that reason they collect training data from past user choices where the user clicked - positive label - and did not click - negative label - a pin upon observation. So based on this experience Pixie learns through supervised learning how to predict positive and negative labels for each user.

But machines are not like humans, in the sense that they don't have an innate capability of learning. They need something that they could understand and a learning process that is feasible for them. That is the purpose of the objective function and the gradient descent in most cases. The most common principle is maximum likelihood estimation. Consider a set of m examples $\mathbb{X} = \{x^{(1)}, x^{(2)}, ..., x^{(m)}\}$ drawn independently from the true but unknown data-generating distribution $p_{data}(x)$. The maximum likelihood estimator for θ is then defined as [14]

$$\theta_{ML} = \arg\max_{\theta} p_{model}(\mathbb{X}; \theta) = \arg\max_{\theta} \prod_{i=1}^{m} p_{model}(x^{(i)}; \theta)$$

Maximizing the likelihood is exactly the same as minimizing the cross-entropy between the data distribution and the estimate [14]. Maximum likelihood is an attempt to make the model distribution match the empirical distribution p_{data}. For example, mean squared error is the cross-entropy between the empirical distribution and a Gaussian model [14]. In [1] cross-entropy loss is minimized with gradient descent on the output of the sampled softmax. All these functions do not have closed form solutions and the only way to approximate the global optimum is to iteratively work our way closer to it. The most common method for this type of learning is gradient descent, where on each step the gradients of each parameter are calculated and are used to update the previous value of the parameter.

How do we know that the model performance is satisfactory? How can we choose between different model configurations and model archetypes? Since we are optimizing based on the objective function, we could choose based on the objective function value of an out of sample set [6]. There are many ways to measure performance in out of training sample sets and evaluate a performance metric of a machine learning model such as k-fold [15] [16]. In this setting there is an immediate problem: machine learning algorithms could have different objective functions. The principle of minimizing (maximizing) an objective function is the same across all models but the actual function differs. So if we have to compare two models with different objective functions we can't measure their respective out of sample objective function values. For example Variational Autoencoders and Generative adversarial Networks are both generative models and are being used on the same applications. But Variational Autoencoders have KL-Divergence as the core of their objective function [17] and on the other hand Generative Adversarial Networks use a game theoretical minmax reward function [18].

In order to be able to compare machine learning models, the research community came up with task specific performance metrics. These metrics are divided into two broad categories: metrics for classification task and metrics for regression tasks. [19]. For classification tasks examples of metrics are:

- Accuracy
- Precision
- Recall
- Precision@k
- F-Measure

And for regression tasks:

- Mean Squared Error
- Mean Absolute Error

The above mentioned cross-model performance evaluation metrics are the basis of model selection in the majority of machine learning applications [20], [2], [1]. One thing that has to noticed is that all these performance metrics are available offline and online. Offline means that it is possible to collect data and measure these performance metrics on any candidate model without loss of generality. Online tests involve experiments with live interactions and feedback loops [21]. Both methods of performance measurement, online and offline, are extremely relevant in model selection and performance monitoring. A performance metric has to be consistently measured offline with the collected data in order to optimize over different model architectures and hyper-parameters. At the same time the offline performance of the model should be able to translate to online experiments, and not deviate significantly, which are conducted based on client feedback.

IV. Research proposal: Aligning machine learning and KPIs

As it was stated in the previous subsections, businesses have their goals and KPIs but their machine learning applications have completely different objective functions. There are a couple of reasons for this phenomenon. Firstly, machine learning algorithms have been developed in academia where maximization if business value is not the primary concern. Academia needs well defined problems and datasets in order to run experiments that have research significance. On the other hand business problems have too many moving parts and variables that make the performance of the machine learning solution harder to identify. Another reason is that even though KPIs are measurable, they are not differentiable. For example we can measure a "Click Through Rate" but there is no direct link from the metric to the parameters of the underlying machine learning model, therefore we cannot take gradients to update the model weights iteratively. Furthermore, in some

cases, it is not feasible to measure KPIs offline and industry researchers result in using specific metrics, such as accuracy or precision, to determine the efficacy of their models. Also a business could strive to optimize multiple KPIs that ensure success but machine learning models often have only one objective function.

But why is it important for the business KPIs and the machine learning performance metrics to be aligned? We believe that it is important as in any other area where the company's products needs to be in line with the goals of the company. As an analogy, let's assume that the business at hand is a paper company and the product is an A4 sheet of paper. The purpose of an A4 paper is to go through the printer and display the intended user information. If the A4 paper was the machine learning application, the businesses optimize on how thin the paper can be in order to be more efficient when it is produced. There is no line of sight to the actual use of the product and in fact, the thinner paper might crumble when it is processed by a printer or it is too transparent for a user to read. Therefore, we argue that machine learning applications need to be inline with the business KPIs as any other facet of the company and even more if it is the core product that is being sold.

In particular, we argue that business KPIs and machine learning objective functions, performance metrics and model selection rules should be aligned and interconnected. Currently there isn't a direct way to identify the relationship between business KPIs and machine learning models. In our research, we intent to propose and validate specific KPIs, machine learning objective functions and respective performance metrics that could work in the following ways:

1) Define business KPIs from machine learning objective functions or performance metrics. In this case, measurable offline and online performance can be monitored and business value change can be quantified.
2) Incorporate KPIs in machine learning objective functions or performance metrics. An interesting goal would be to directly optimize the business value within the machine learning development process. Such an approach would be highly desirable since the business value would be being optimized throughout the development process.
3) Develop agents on top of the machine learning models that ensure the business alignment. That way we keep the machine learning development unchanged and have the capability of model selection based on business criteria in an online fashion.

A. Machine learning objective functions define business value

The first option is taking more of a brute force approach to the problem. There are use cases such as stock portfolio optimization that the business objective is the exactly the same as the respective objective function of the machine learning algorithm which is to maximize the return/reward [22]. Taking this as a blueprint, one might extrapolate the thought process and apply it to multiple business metrics. This can be proven trickier than it might seem. For example, assume we have a model for predicting the next day's price of a stock in NYSE. Applying the same reasoning as above, we could use Mean Squared Error as the objective function and the business KPI. The model that yields the best Mean Squared Error performance approximates the price of the stock pretty well but cannot handle large jumps/drops in intraday price. Leveraging Mean Squared Error as the business KPI implies that the company does not care about the large jumps/drops in price which is not the case. These jumps reflect the opportunity for large gains or the risk for substantial losses that a company in this domain has a lot of interest in. Generally, we observe that Reinforcement Learning methods fit naturally in this category since the reward function is intuitive and drawn from real game theoretical problems such as in [22].

B. Business KPIs as objective functions

Current business KPIs are more difficult to be incorporated in the machine learning pipeline for more obvious reasons. The first obstacle would be the inability of differentiating the KPIs in order to collect gradients and leverage gradient descent methods for optimization. On the other hand, most of the performance metrics of machine learning methods are not differentiable. These metrics are summary statistics for a set of data and there is no way to connect the model's parameters to a result in order to update them iteratively. The most common performance metric, at least in data science competitions, is accuracy in an out of sample test set. Accuracy, which is defined as

$$Accuracy = \frac{TruePositive + TrueNegative}{NumberOfTestSamples}$$

is not differentiable. We measure the accuracy of a model with an out of sample test set in an offline manner, meaning that we have collected the data beforehand. The model with the highest test set accuracy is selected as the best fitted model to the data [6]. Then why can't we use the business KPIs in the same fashion that we are using accuracy? Business KPIs are directly related to an outcome from the interaction with clients or potential clients - something that moves the business towards its goal. Even "Click Through Rate" which is a very common KPI that is used in many online applications and integral component to the success of many advertisement focused internet applications cannot be used in an offline manner. Imagine if we want to test a new ads recommendation engine. Also assume that we consider the user's profile as input and some other available metadata. We train the machine learning model with a traditional objective function as the ones in the literature [1] [2] and we have prepared a test set with input data, suggested ads and the outcome of click or no-click. How can we measure the efficacy of an ad that has not been suggested to the user yet? The user could dismiss the ad or buy the product on the spot. At the same time, how can you measure the effectiveness of an ad for an new product that is not in the training or test set? It is not feasible to measure such outcomes in an offline manner when we are optimizing the model, at least not with the current available business KPIs.

C. KPI model on top of machine learning model

The last alternative is being used currently, in some capacity, from large internet companies. Even in the literature when "Click Through Rate" is measured like in [1], the measurements and optimization were made online through extensive A/B tests or multi-armed bandits [8]. Setting extensive monitoring systems for machine learning performance measurement is a good practice in all deployment settings since more information is always better. On top of that it is possible to select the optimal machine learning model through some exploration with multi-armed bandits as in [8]. Although these solutions, like the one suggested in [7], are a step towards the alignment of business KPI and machine learning model, they come in effect at the very last stage of the product process development, after the model has been deployed into production. Imagine if we want to launch multiple iterations of the product through time and we want to optimize the business KPIs, we would have to wait for the full development circle to find out if the product release is performing well. We argue that agents like these should be involved throughout the development process and not only at the end or during monitoring. A direction that we find intriguing is the possibility of two independent models that work in tandem, one predicting and one aligning to the desired KPI. The model that is doing the prediction would be a standard machine learning model with an objective function that we already discussed. The second model - or agent - would work on top of the first model tuning its hyperparameters so that the model is optimized for the business KPI. This approach would ensure the best fit for the training data given the current model parameterization and the parameters that optimize the target business KPI.

In some cases none of the aforementioned options would be feasible due to theoretical or practical limitations. In such cases it would be very useful if we could understand the relationship between the KPIs and the machine learning models. What an x% accuracy means for this business KPI? Is there an increasing or decreasing relationship between a measurement in development and a business outcome? Even if weak correlations can be made they could have massive effect on business value.

V. Conclusion

Product goals and business goals should be aligned in any industry and this holds true for machine learning and A.I. applications too. We established that this is not the case though, since machine learning applications are optimized for objective functions and performance metrics that are disconnected from the business. This fact stems from the complexity of real world applications and the uncertainty of human behaviour. So far the emphasis was on applications that have clear assumptions and data, such as natural language comprehension approximation with machine learning models using the performance metric BLUE [23]. Moving forward however, to less structured problems with less clear assumptions will require alignment of KPI metrics to machine leaning tuning parameters. In this paper, we proposed approaches on how we could mitigate this disparity and align business KPIs with machine learning model selection and optimization tools. The goal of this effort would be for researchers and practitioners to start rethinking their approach to model optimization and selection when it comes to real world applications. When we do research for real world applications - A.I. research has first and foremost the human reality in the center - we should be concerned with the real world limitations and objectives.

References

[1] Paul Covington, Jay Adams, and Emre Sargin. Deep neural networks for youtube recommendations. In *Proceedings of the 10th ACM Conference on Recommender Systems*, RecSys '16, page 191–198, New York, NY, USA, 2016. Association for Computing Machinery.

[2] Chantat Eksombatchai, Pranav Jindal, Jerry Zitao Liu, Yuchen Liu, Rahul Sharma, Charles Sugnet, Mark Ulrich, and Jure Leskovec. Pixie: A system for recommending 3+ billion items to 200+ million users in real-time. In *Proceedings of the 2018 World Wide Web Conference*, WWW '18, page 1775–1784, Republic and Canton of Geneva, CHE, 2018. International World Wide Web Conferences Steering Committee.

[3] David Parmenter. *Key performance indicators(KPI): developing, implementing, and using winning KPIs.* Willey, 2015.

[4] RC Dorf Robert Kaplan, David Norton. The balanced scorecard: translating strategy into action. In *Volume 4*. Harvard Business school press Boston, 1996.

[5] Alejandro Maté, Kostas Zoumpatianos, Themis Palpanas, Juan Trujillo, John Mylopoulos, and Elvis Koci. A systematic approach for dynamic targeted monitoring of kpis. In *Proceedings of 24th Annual International Conference on Computer Science and Software Engineering*, CASCON '14, page 192–206, USA, 2014. IBM Corp.

[6] Sebastian Raschka. Model evaluation, model selection, and algorithm selection in machine learning. *CoRR*, abs/1811.12808, 2018.

[7] Minmin Chen, Alex Beutel, Paul Covington, Sagar Jain, Francois Belletti, and Ed H. Chi. Top-k off-policy correction for a REINFORCE recommender system. *CoRR*, abs/1812.02353, 2018.

[8] Samuel Daulton, Shaun Singh, Vashist Avadhanula, Drew Dimmery, and Eytan Bakshy. Thompson sampling for contextual bandit problems with auxiliary safety constraints. *CoRR*, abs/1911.00638, 2019.

[9] Horst Meier, Henning Lagemann, Friedrich Morlock, and Christian Rathmann. Key performance indicators for assessing the planning and delivery of industrial services. *Procedia CIRP*, 11:99 – 104, 2013. 2nd International Through-life Engineering Services Conference.

[10] Irene Unceta, Jordi Nin, and Oriol Pujol. Towards global explanations for credit risk scoring. In *NIPS 2018 Worksop on Challenges and Opportunities for AI in Financial Services: the Impact of Fairness, Explainability, Accuracy, and Privacy*, 2018.

[11] Victor R. Basili, Gianluigi Caldiera, and H. Dieter Rombach. The goal question metric approach. In *Encyclopedia of Software Engineering*. Wiley, 1994.

[12] V. R. Basili, M. Lindvall, M. Regardie, C. Seaman, J. Heidrich, J. Münch, D. Rombach, and A. Trendowicz. Linking software development and business strategy through measurement. *Computer*, 43(4):57–65, 2010.

[13] Christopher M. Bishop. *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer-Verlag, Berlin, Heidelberg, 2006.

[14] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. MIT Press, 2016.

[15] Ji-Hyun Kim. Estimating classification error rate: Repeated cross-validation, repeated hold-out and bootstrap. *Computational Statistics & Data Analysis*, 53(11):3735 – 3745, 2009.

[16] Hawkins, Douglas M., Subhash C. Basak, and Mills Denise. Assessing model fit by cross-validation. *Journal of Chemical Information and Computer Sciences*, 43(2):579 – 586, 2003.

[17] Diederik P. Kingma and Max Welling. Auto-encoding variational bayes. In Yoshua Bengio and Yann LeCun, editors, *2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014, Conference Track Proceedings*, 2014.

[18] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems*, volume 27, pages 2672–2680. Curran Associates, Inc., 2014.

[19] Alejandro Bellogín and Alan Said. *Recommender Systems Evaluation*, pages 2095–2112. Springer New York, New York, NY, 2018.

[20] Abhinandan S. Das, Mayur Datar, Ashutosh Garg, and Shyam Rajaram. Google news personalization: Scalable online collaborative filtering. In *Proceedings of the 16th International Conference on World Wide Web*, WWW '07, page 271–280, New York, NY, USA, 2007. Association for Computing Machinery.

[21] Mingang Chen. Performance evaluation of recommender systems. *International Journal of Performability Engineering*, 13, 12 2017.

[22] J. Moody and M. Saffell. Learning to trade via direct reinforcement. *IEEE Transactions on Neural Networks*, 12(4):875–889, 2001.

[23] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics.