

CS6460 - Project Paper

Adria Romero

adria@gatech.edu

Abstract— This document presents the project paper for the CS6460 course at Georgia Institute of Technology for which a Content track was chosen to teach a *Machine Learning in Production* course using Cognitive Emotional Pedagogy and Distance learning and framed around a connectivist teaching approach using Notion as an educational tool.

0 DELIVERABLES

- Final project: [productize.ml](#)
- Project presentation: [video presentation](#)

1 INTRODUCTION

Given the nature of my previous research and interests in working on a content track where a course about *Machine Learning in Production* would be thought, I considered the main four phenomenons that have allowed the Machine Learning takeoff in both the Academy and Industry sectors.

In a short summary, these have been: (1) the **algorithms** and research advances (Krizhevsky, 2012; Ioffe, 2015; Lan, 2019; Srivastava, 2014; Radford, 2019) that have demonstrated and proven the potential benefits that these technologies could have in society when brought to production; (2) the digitization process and access to new **data sources** (Brickley, 2019) that let the large artificial networks generalize to new unseen cases and therefore automating some of our daily tasks with smarter decisions; (3) the availability of programmable **frameworks** that contain most part of today's state-of-the-art algorithms and methods in order to carry out research experiments (Abadi, 2016; Paszke, 2019; Pedregosa, 2011) and makes them accessible and easy to incorporate to the whole community; and finally but still key in this development of ML (4) the equipment with **computational power** provided by Graphics Processing Unit (GPUs) in order to be able to fed all the vast amounts of data depending on their types (image, audio, text, etc.) and let the execution be parallelizable.

When all these four pieces are put together and taken to production, a need to measure and iterate the product that is presented to real customers is required. This course aims to standardize some of the recommended processes so the community can work and evolve as

a whole with a sense of collaboration. The course provides exemplar guidelines and processes in the field when delivering Machine Learning solutions to real customers such as an accurate description of the product development process, how to properly assess and measure the business impact, and benefits and considerations around ML products.

2 RELATED WORK

Despite the topic around productizing machine learning solutions is quite recent and still under progression and research, the big online learning platforms are delivering a similar form of content that I envision in my course. This includes a course by Google that focuses on the basic ML algorithms and data processing; a Coursera featuring Google Cloud course built around production ML Systems and architectures; and an Udemy course on ML pipeline and deployment technologies.

2.1 Machine Learning Crash Course by Google

The [*Machine Learning Crash Course*](#) is a fast-paced and practical introduction to Machine Learning delivered by Google with the objective of teaching the basics about ML to future developers and practitioners. This course includes not only video lectures with 25 lessons and 15 hours of content but also real-world case studies with more than 30 exercises as hands-on practice.

This content is delivered in the form of video tutorials and a video lecture summary with a short description of each section. The content that is covered in this course tries to answer the main differences between traditional programming and Machine Learning, model optimization and measurement, loss functions, data management and preparation before training, gradient descent as an optimization algorithm, and some concepts about deep neural networks.

The tools that the course uses are mostly based on Google Cloud solutions such as Colaboratory and Tensorflow APIs.

The course is divided into six sections:

1. **Problem Framing:** Presents the basic ML terminology and general methods along with common problems and solutions. It also compares the traditional methods to ML methodologies and when these are suggested over traditional approaches.
2. **Data Preparation:** Tries to describe the importance of data on the ML workflows, how accessible is the data, ways of preparing and cleaning a dataset, as well as common procedures to deal with imbalanced classes.

3. **Clustering:** Defines the common clustering methods such as k-means algorithms and how these compare to supervised machine learning approaches.
4. **Recommendation:** This section describes recommendation systems and their deeper technical components to develop candidate generation, scoring, and re-ranking.
5. **Testing and Debugging:** explains how to debug and optimize an ML model using the main evaluation metrics used during development, launching, and production.
6. **GANs:** describes the technology behind the Generative adversarial networks (GANs) and presents some examples of how to train these challenging networks.

2.2. Production Machine Learning Systems by Coursera

Coursera is also offering a course in collaboration with Google Cloud called [Production Machine Learning Systems](#) with the objective of teaching the main components and best practices of a high-performing ML system when deployed in production environments. The course is delivered in the form of video lectures, and some of the Prerequisites are to have basic knowledge of SQL, and familiarity with Python and TensorFlow.

The course is thought to be completed in two weeks, and its content is divided into six submodules:

1. **Introduction:** a brief introduction to the course and Google Cloud tools.
2. **Architecting Production ML Systems:** a review of production ML system needs and requirements, as well as high-level concepts, design decisions around training and model serving
3. **Ingesting data for Cloud-based analytics and ML:** how to process, manage, and bring your data to the cloud for further processing.
4. **Designing Adaptable ML systems:** recognize the ways that our model is dependent on our data, make cost-conscious engineering decisions, and understand your model's behavior.
5. **Designing High-performance ML systems:** identify performance considerations for machine learning models, as well as computational speed.
6. **Hybrid ML systems:** understand the tools and systems available and when to leverage hybrid machine learning models.

2.3. Deployment of Machine Learning Models by Udemy

Udemy online teaching platform has also developed a course to teach ML in production called [Deployment of Machine Learning Models](#) in which the content is delivered in the form of 9 hours video and 30 complimentary articles along with assignments.

The course targets data scientists wanting to deploy their first ML model, data engineers who build data pipelines to productionise machine learning models, or software developers transitioning into artificial intelligence.

After completing the course, the learner should be capable of deploying ML models into the cloud, being able to make those models accessible via an API; understand how an ML pipeline should look like to be reproducible; discern among an optimal machine learning architectures; and resources, integrations, and production tips to deploy models.

3 PROJECT REVIEW

3.1 Productize ML

The name of the course is *Productize ML*, and it is hosted and accessible for anyone at productize.ml.



Figure 1—Productize ML logo designed with [Figma](#) software.

Productize ML provides guidelines and best practices to the productization of Machine Learning solutions from the experimental stage to the deployment of the solution to customers. Once a solution is moved to production, there are some concerns we might face related to data dependencies and changes in the external world, model complexity, reproducibility, testing, monitoring, model deployment, and quick delivery of new models. All these are covered in the course to the point that the learner will feel confident about how to tackle them after completing the course.

In order to achieve that, the course is offered as an online tool and open-ended regarding the product and technical delivery recommendations. The shared content will get updated periodically in order to guarantee new and state-of-the-art information. The community will

engage with the course in the form of chats and comments, and they will take an important role in the content of the course as well, promoting a shared course built thanks to every learner.

When comparing the course to the existing ones, *Productize ML* is unique not only for the way this novel content is delivered but also regarding the educational technologies that are being used. It extends the delivered materials to the point of delivering the final solution to customers, and not only that, but provides best practices for its continuation. The estimated course time is 8-10 hours, including course-specific content, and recommended readings from other sources like articles and blog posts..

3.2 Content

In general terms, this course will provide the learner with:

- Common and most-used AI, Machine Learning, Deep Learning, and Data Science terminology;
- Realistically what someone can expect from these systems to realistically do and what they actually cannot do;
- Abilities to understand what the opportunities of an AI/ML product are and how these will impact their institution/organization/personal project;
- Computational resources and recommended frameworks;
- Best practices on data collection and curation;
- Training, optimization, and monitoring of Machine Learning models;
- Instructions on how to build reproducible machine learning pipelines;
- Create continuous and automated integrations to deploy models;
- When building a product, what things we want the AI solution to improve or enhance when compared to traditional solutions;
- Recommendations on how to deliver AI/ML-generated predictions to users;
- Understand real-world cases and users data;

The previously mentioned objectives have been divided into four main lectures that are hereafter described.

3.2.1 Introduction

The course introduction debriefs the main objectives of the course, tools and processes that need to be followed, as well as ways the learner can contribute to the course, and a series of tips in order to successfully complete the course.

3.2.1.1 Course Objectives

Accessible from: productize.ml/objectives

The course objectives try to engage the learner by providing reasons why their product should get smarter by using Artificial Intelligence and Machine Learning at the same time this is well controlled and measured from the business side in order to fulfill our business requirements. It also provides some descriptions around the role of a ML Product Manager and some of the challenges they need to solve by collaborating with Data Scientists.

3.2.1.2 Course Guidelines

Accessible from: productize.ml/guidelines

In this section, the learner is introduced to the course structure and tools that will be used, as well as requirements and pre-work that needs to be fulfilled for a successful journey in this course.

3.2.1.3 Course Syllabus

Accessible from: productize.ml/syllabus

In the course syllabus section, a brief description of each of the four lectures is provided in order to provide the learner with a first insight about what the course is offering.

3.2.1.3 Course Schedule

Accessible from: productize.ml/schedule

Although the course is divided into four different lectures containing different learning materials (e.g. lecture, reading, etc.), it is important to provide the learners with the recommended terms of time required to complete, previous course requirements, and topics that will be discussed.



Schedule

This course schedule provides a thorough list of weekly topics, readings, assignments, and exams. Click **All** to switch to week, exam, or calendar view.

Schedule

Lecture	Time (minutes)	Type	Topic
1 Why ML, and why now	5	lecture reading	machine-learning
2 ML terminology	8	lecture	machine-learning
3 Supervised Learning	15	lecture reading	machine-learning
4 Unsupervised Learning	15	lecture reading	machine-learning
5 Deep Learning	15	lecture reading	machine-learning
1 Data Access	10	lecture	data-management
2 Data Collection	8	lecture reading	data-management
3 Data Curation	8	lecture	data-management
1 Framework and Hardware	10	lecture reading	train-evaluate
2 Training Neural Networks	12	lecture	train-evaluate
3 Model evaluation	5	lecture	train-evaluate
1 ML Product Development Process	20	lecture reading	product-management
2 Measuring Business Impact	10	lecture reading	product-management

+ New

COUNT 13

Figure 1—Example of the course schedule that will be presented at the beginning of the course.

3.2.2 Machine Learning

Accessible from: productize.ml/machine-learning

In this section, we will dive into the common terminology and algorithms that the ML community is currently using and will prepare the learner with a strong knowledge of the field for the next sections. We will also briefly discuss what the benefits of these technologies are, and what the level of automation to impact the product we can expect from them.

3.2.2.1 Why ML, and why now

Accessible from: productize.ml/machine-learning/why-now

In this first lesson, a list of reasons behind the raise of AI applications is given, most of them thanks to the development of new algorithms and research advances, the digitization process of data and access to big data sources, advances in programmable framework, and finally the equipment with computational power like Graphics Processing Units.

3.2.2.2 ML Terminology

Accessible from: productize.ml/machine-learning/terminology

This lesson tries to put together a glossary of this first lecture with common terms associated with Machine Learning, training concepts, hyperparameter values, evaluation concepts, and data management terms.

3.2.2.2 Supervised Learning

Accessible from: productize.ml/machine-learning/supervised-learning

A brief review of the most-used supervised learning algorithms such as linear regression, logistic regression, SVM, and decision trees is provided in this lesson, along with examples.

3.2.2.3 Unsupervised Learning

Accessible from: productize.ml/machine-learning/unsupervised-learning

A brief description and review of the most-used unsupervised learning methods such as clustering and dimensionality reduction algorithms is provided in this lesson.

3.2.2.4 Deep Learning

Accessible from: productize.ml/machine-learning/deep-learning

Deep Learning is a type of Machine Learning that has pioneered most of any current application that has been offered to the public. This lesson reviews its fundamentals starting from Neural Networks (NN), types of layers, types of NN such as Convolutional Neural Networks (CNN), characteristics of forward propagation and backpropagation, optimization algorithms such as gradient descent, and the theory behind activation and loss functions.

3.2.3 Data Management

Accessible from: productize.ml/data-management

Brief description and best practices in order to access datasets, availability and challenges, sources and tools for data management, the recommended ways to collect data in case this is not available online and we need to curate our own dataset, and how this data can be managed and curated to successfully train ML models.

3.2.3.1 Data Access

Accessible from: productize.ml/data-management/access

Since data comes before training any ML model, this lesson provides ways of accessing and sources of data from third party sources and possible challenges we might find when requesting access to this.

3.2.3.2 Data Collection

Accessible from: productize.ml/data-management/collection

When data is not available from public or third party sources, a solution is to leverage and collect the data by ourselves. In this lesson, ways of collecting data by labeling techniques for data flywheeling cases, or methods to create synthetic data is provided.

3.2.3.3 Data Curation

Accessible from: productize.ml/data-management/curation

Once you have access to the data, it is recommended to follow some curation processes before this is fed into your training in order to guarantee reproducibility of experiments, such as standardizing your data using conceptual hierarchies, checkpointing your data using versioning controls, and splitting your data properly for the different purposes of the training and evaluation steps.

3.2.4 Train and evaluation

Accessible from: productize.ml/train-evaluate

In this section, we will review the main framework and hardware required to train and evaluate ML models, as well as recommended techniques to follow when training these complex architectures such as data processing techniques, understanding concepts like overfitting and underfitting, when data augmentation is needed, common mistakes and solutions to optimize the performance of these models, as well as ways of evaluating and monitor your model training.

3.2.4.1 Framework and Hardware

Accessible from: productize.ml/train-evaluate/framework-hardware

A brief introduction to the most-used framework and hardware for AI and ML applications such as Tensorflow, Keras and Pytorch, as well as the computational power provided by Graphics Processing Units (GPUs) and Tensor Processing Units (TPUs).

3.2.4.2 *Training Neural Networks*

Accessible from: productize.ml/train-evaluate/train-neural-networks

A extensive description of the main factors to keep in mind when training Neural Networks, such as types of data processing techniques, how to solve problems associated to overfitting or underfitting of your model such as data augmentation techniques, how transfer learning can be applied to new domain problemes and still perform successfully, and ways to make the most of your models by ensembling them.

3.2.4.3 *Model Evaluation*

Accessible from: productize.ml/train-evaluate/model-evaluation

How to assess the evaluation of ML models, types of evaluation metrics and recommended ways to evaluate your models depending on the task.

3.2.5 *Productize it*

In this section, a review of the previous sections will be given but this time keeping in mind the product development process by providing the learner the tools and concepts in order to be able to measure the business impact, benefits and considerations when launching a product with AI/ML features.

3.2.5.1 *ML Product Development Process*

Accessible from: productize.ml/productize/product-development-process

In this first lesson, it is reviewed in detail the whole ML lifecycle that goes firstly from understanding your business needs, depending on these, having access to proper the data, coming up and developing the most appropriate models depending on your problem, how to properly train these models, evaluate their performance before its deployment, and keep iterating over them via constant communication between Data Science and Product Management teams.

3.2.5.2 *Measuring Business Impact*

Accessible from: productize.ml/productize/business-impact

This lesson tries to provide the sense of the business impact analysis and how to effectively iterate over your launched product, monitor and test in order to be ready for any disruption introduced to your product by the AI solution.

3.2.5.3 *PM Terminology*

Accessible from: productize.ml/productize/terminology

This lesson tries to put together a glossary of the lecture with common terms associated with Product Management that the learner should be familiar with.

3.2.6 *Resources*

A recommended list of readings to related blog posts and books, similar courses, and video lectures in order to guide the learner to expand the course experience with other resources.

3.3 Tools

As previously mentioned, this course differentiate from the competitors in the way of delivering the content in order to be free, flexible, and accessible from any platform without a need to sign up on any external platform.

The tool chosen in order to guarantee all these requirements is [Notion](#), a collaboration and note-taking app that integrates kanban boards, tasks, wikis, and databases.

This tool will provide online as well as offline (in the form of PDF formats) access to the course content. Furthermore, Notion will provide the user with the ability to create a copy and edit the course content in its own way for note-taking purposes.

The current URL where the course is hosted, productize.ml, points to a public Notion page. I have used [Cloudflare](#) to perform the URL routing.

The course is also offering a [Slack group](#) in order to create an online community where conversations around the course will be hosted.

3.4 Audience

The target audience the course aims to approach is graduate levels interested in working on a personal project or vocational, and working professionals that might receive the training as a corporate need or purely personal development growth. The audience needs to be familiar with some programming skills but without a strong knowledge about Artificial Intelligence nor Machine Learning. Some of the (optional) prerequisites that could be listed are linear algebra, trigonometry, statistics, Python programming, and Bash terminal.

3.5 Pedagogical theories

The course will be designed using a connectivist learning approach, the learner will be encouraged to understand that the course knowledge does not reside on the individuals, but it does on non-human objects and tools such as the Internet, computers, literature, networks of people, etc. For this reason, they will be required to seek out information within the network when needed, and being able to judge if the information retrieved is valid or not by connecting and comparing it to the others.

The way that the course tries to achieve it is by not setting any deadline and letting the learner take their own pace in order to complete the course. Readings are not mandatory, but suggested and the course tries to get the learner attention firstly with the lectures, and once the learner has completed these, they can extend it with the recommended readings.

4 FUTURE WORK

Productize ML is a course that will keep extending and updating the current content in order to always deliver the best recommended practices at the same time that the technology keeps evolving. Industry experts will be offered the opportunity to contribute to the course as blog posts with real cases explained. As a future work, a better coverage on model deployment will be included, as well as adding hands-on experiences as code assignments for each lecture.

5 CONCLUSIONS

Productize ML helps Developers and Product Managers to build Machine Learning products by learning the technology in depth as well as bridging the gap from the theory and deployment of AI systems to real users.

6 REFERENCES

1. Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., ... & Ghemawat, S. (2016). Tensorflow: Large-scale machine learning on heterogeneous distributed systems. arXiv preprint arXiv:1603.04467
2. Brickley, D., Burgess, M., & Noy, N. (2019, May). Google Dataset Search: Building a search engine for datasets in an open Web ecosystem. In *The World Wide Web Conference* (pp. 1365-1375).
3. Dagger, D., Wade, V., & Conlan, O. (2004). Developing Adaptive Pedagogy with the Adaptive Course Construction Toolkit (ACCT). In *Workshop Proceedings of the 3rd International Conference on Adaptive Hypermedia and Adaptive Web-Based Systems (AH'04)*.
4. Ioffe, S., & Szegedy, C. (2015). Batch normalization: Accelerating deep network training by reducing internal covariate shift. arXiv preprint arXiv:1502.03167.
5. Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems* (pp. 1097-1105).
6. Lan, Z., Chen, M., Goodman, S., Gimpel, K., Sharma, P., & Soricut, R. (2019). Albert: A lite bert for self-supervised learning of language representations. arXiv preprint arXiv:1909.11942.
7. Machine Learning Crash Course | Google Developers. (n.d.). Retrieved from <https://developers.google.com/machine-learning/crash-course>
8. Mattar, J. (2018). Constructivism and connectivism in education technology: Active, situated, authentic, experiential, and anchored learning. *RIED. Revista Iberoamericana de Educación a Distancia*, 21(2).
9. Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., ... & Desmaison, A. (2019). PyTorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems* (pp. 8024-8035).
10. Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., ... & Vanderplas, J. (2011). Scikit-learn: Machine learning in Python. *the Journal of machine Learning research*, 12, 2825-2830.

11. Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. (2014). Dropout: a simple way to prevent neural networks from overfitting. *The journal of machine learning research*, 15(1), 1929-1958.