Saturn Platform: Foundation Model Operations and Generative AI for Financial Services

Antonio J. G. Busson
antonio.busson@btgpactual.com
BTG Pactual

Rennan Gaio
rennan.gaio@btgpactual.com
BTG Pactual

Rafael H. Rocha
rafael-h.rocha@btgpactual.com
BTG Pactual

Francisco Evangelista
francisco.evangelista@btgpactual.com
BTG Pactual

Bruno Rizzi
bruno.rizzi@btgpactual.com
BTG Pactual

Luan Carvalho
luan.carvalho@btgpactual.com
BTG Pactual

Rafael Miceli
rafael.miceli@btgpactual.com
BTG Pactual

Marcos Rabaioli
marcos.rabaioli@btgpactual.com
BTG Pactual

David Favaro
david.favaro@btgpactual.com
BTG Pactual

Abstract

Saturn is an innovative platform that assists Foundation Model (FM) building and its integration with IT operations (Ops). It is custom-made to meet the requirements of data scientists, enabling them to effectively create and implement FMs while enhancing collaboration within their technical domain. By offering a wide range of tools and features, Saturn streamlines and automates different stages of FM development, making it an invaluable asset for data science teams. In this white paper, we discuss the expected impacts of Saturn on the financial sector.

Keywords: Foundation Model, Generative AI, FMOps, Saturn

1 Introduction

An emerging Artificial Intelligence (AI) paradigm called the Foundation Model (FM) has shown great potential due to its ability to learn universal representations that can be applied to diverse tasks [20]. From a technological point of view, foundational models consist of deep learning models that are pre-trained in a self-supervised/semi-supervised manner on a large scale and then adapted for various downstream tasks [2].

The development of FMs relies on several significant challenges related to infrastructure, development kits, governance, security, etc. To address these challenges, we propose Saturn, a platform to help the process of building, managing, and serving FMs. Saturn combines advanced technologies, intelligent automation and robust infrastructure to empower professionals to pursue accurate and reliable models. Saturn's core was generically designed to facilitate its implantation to any application domain. However, in this work, we will focus specifically on the impacts of its use in the financial sector.

The Saturn Platform is proprietary software. All rights are reserved and protected by the BTG Pactual.

In: XXII Workshop de Ferramentas e Aplicações (WFA 2023), Ribeirão Preto, Brasil. Anais Estendidos do Simpósio Brasileiro de Sistemas Multimídia e Web (WebMedia). Porto Alegre: Sociedade Brasileira de Computação, 2023.

ISSN 2596-1683

2 Foundation Models and Generative AI

We gathered requirements for the Saturno platform by analyzing the state-of-the-art in the field of foundational models and generative AI.

2.1 State-of-the-Art

Foundational models are grounded by two techniques: (1) transfer learning and (2) self-supervised learning. The idea of transfer learning is to apply the knowledge that was learned in training from one task to another different task. On the other hand, in self-supervised learning, the pre-training task is automatically derived from unannotated data. For example, the masked language modeling task used to train BERT [7] is to predict missing words in a sentence.

Self-supervised tasks are more scalable and potentially helpful than models trained in a limited space by label annotation. There has been considerable progress in self-supervised learning since word embedding [10], which associated each word with a context-independent vector, and provided the basis for a wide range of NLP models.

Shortly after, self-supervised methods based on autoregressive language modeling (predicting the next word given the previous words) [3] became popular. This produced models representing contextualized words such as GPT [15], ELMo [14], and ULMFiT [4]. The second generation of models based on self-supervised learning, BERT [6], GPT-2 [16], and RoBERTa [9] were based on the Transformer architecture, incorporating deeper and more powerful sentence bidirectional encoders.

Inevitably, foundational models underwent a process of homogenization of architectures since the last generation models are all Transformer derivatives [19]. While this homogenization produces hugely high leverage (improvements

to the basic models can bring immediate benefits for most of the other foundational models), all AI systems can inherit the same problematic biases from some foundational models [1].

In addition to the NLP area, methods have been homogenized among different research communities in recent years. For example, similar Transformer-based modeling approaches have been applied to images [12], speech [8], tabular data [5], organic molecules [17]. These examples point in a direction where we will have a unified set of tools to develop foundational models for a wide range of modalities [18].

Reinforcement Learning has been applied to enhance various FMs in NLP tasks. InstructGPT proposes a method called RLHF (Reinforcement Learning from Human Feedback), which involves fine-tuning large models using PPO (Proximal Policy Optimization) based on a reward model trained to align the models with human preferences [13]. This approach is also employed by ChatGPT[1]. The reward model is trained using comparison data generated by human labelers who rank the model outputs manually. Based on these rankings, the reward model or a machine labeler calculates a reward that is then utilized to update the FM through PPO.

A notable advancement in FM technology is GPT-4 [11]. GPT-4 employs a pre-training phase where it predicts the next token in a document, followed by RLHF fine-tuning. GPT-4 surpasses GPT-3.5 in terms of reliability, creativity, and ability to handle more detailed instructions as the complexity of the task increases.

2.2 Requirements

Based on the research in the previous subsection, the requirements gathered are as follows:

1. Pre-defined self-supervised learning pipelines or frameworks;
2. Transfer learning support;
3. Efficient data processing and training pipelines to handle large amounts of unannotated data;
4. Framework that allows data scientists to build and improve upon existing FMs easily;
5. Mechanisms to detect and mitigate problematic biases inherited from FMs;
6. Support a wide range of data modalities, including text, images, speech, tabular data, etc.;
7. Collaborative features, such as version control, model sharing, and experiment tracking, to facilitate collaboration among data scientists;
8. Performance optimization by utilizing parallel computing; distributed training, and hardware acceleration (e.g., GPUs or TPUs);
9. Tools for monitoring model performance in real-world settings and providing insights into potential drift or degradation;
10. Process for deploying trained FMs into production environments. This includes model-serving infrastructure, REST API support, and containerization for easy integration with other applications;
11. Support approaches like RLHF for fine-tuning FMs using reward models aligned with human preferences.

[1]https://openai.com/blog/chatgpt

3 Saturn Platform

Saturn is a cutting-edge platform designed to help in FM building and seamlessly integrate it with IT operations (Ops). It is specifically tailored to cater to the needs of data scientists, empowering them to efficiently develop and deploy foundation models while optimizing the collaboration between their technical expertise. The platform provides a comprehensive suite of tools and features that simplify and automate various aspects of FM development, making it an indispensable resource for data science teams.

Figure 1 shows the architecture of the Saturn platform. The platform is structured in three environments: (1) Saturn Environment; (2) Data Science (DS) Development And (3) Automated FM Operations. Each environment is detailed in the subsections that follows.

3.1 Saturn Environment

Model Zoo offers a centralized repository for FMs, allowing data scientists to access and leverage pre-trained models, architectures, and components. This facilitates knowledge sharing and reduces redundancy, enabling teams to accelerate model development. *Model Zoo* prioritizes data security and governance, ensuring that sensitive data and models are protected throughout the development and deployment lifecycle. It includes robust access controls, encryption mechanisms, and compliance features, adhering to industry standards and regulations.

Embedding Farm provides an efficient storage solution for embeddings generated by FMs. It leverages advanced algorithms and optimized data structures to ensure high-speed retrieval (via vector database). It offers powerful management capabilities, allowing users to organize, categorize, and tag embeddings. Additionally, *Embedding Farm* incorporates access control mechanisms, ensuring that sensitive embeddings are safeguarded from unauthorized access.

Saturn can deploy FMs as API endpoints, enabling other software to access and leverage these models' power easily. Whether it is a console prompt or a *LangChain System*, Saturn offers a versatile solution that can be seamlessly integrated into various financial service applications. In addition, the *RLHF system* collects and manages human feedback data to train reward models used to fine-tune FMs via reinforcement learning.

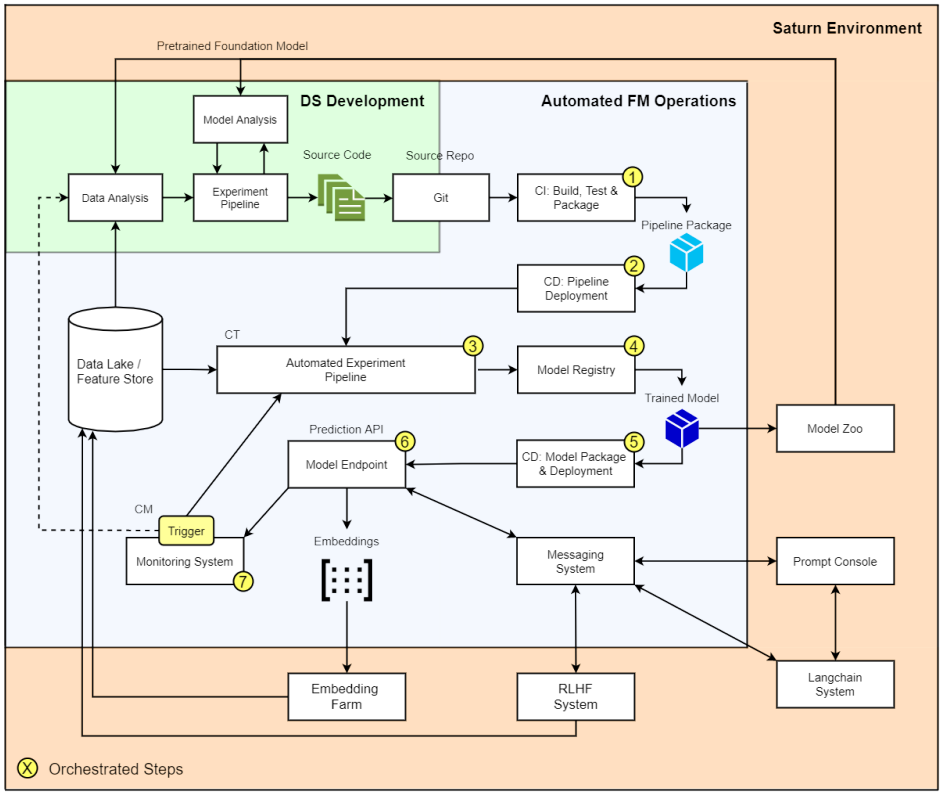

Figure 1. Architecture of the Saturn platform.

3.2 DS Development

The platform provides a collaborative workspace equipped with powerful development tools and libraries tailored for foundation model building. Data scientists can leverage a range of programming languages, frameworks, and visualization tools, ensuring flexibility and compatibility with their preferred workflows.

In the *Data Analysis* module, pre-trained FMs can assist in the initial data exploration and preprocessing stages. They can generate summaries of large datasets, identify patterns, generate data, and perform basic data-cleaning tasks. This can significantly speed up the data preparation phase, allowing data scientists to focus on higher-level analysis.

On the other hand, in the *Model Analysis* process, pre-trained FMs allow data scientists to iterate quickly on their specific task. Instead of training a model from scratch, they can start with a pre-trained FM and fine-tune it, significantly reducing the training time and effort required. This accelerated iteration process facilitates faster experimentation and hypothesis testing.

3.3 Automated FM Operations (FMOps)

The FMOps environment allows FMs to be continuously updated and deployed, ensuring they are always in sync with the latest data and maintaining their performance over time.

Orchestrated processes are structured in the following steps:

- (1-3) The user sends the source code of the machine learning model to a Git server, where the code will be stored and versioned. A continuous integration and continuous delivery (CI/CD pipeline) treadmill is set up to trigger a continuous training (CT) process whenever there is a new source code.
- (3-4) After training, the resulting model artifact is saved and stored in a "model zoo" (model repository). This centralized repository allows the storage and management of trained models.
- (5-6) The trained model is deployed to a production environment. This can be done through a cloud infrastructure, containers, or model-specific deployment services. The model is configured to be accessed through an API endpoint, allowing external users to request inferences.
- (7) A continuous monitoring (CM) system is set up to observe the model's performance in production. It can track relevant metrics. In addition, the system detects changes in input data (data drift) and automatically

starts continuous training to update the model with the latest data.

4 Application in Financial Services

Forecasting and Predictive Analytics. Financial institutions often rely on accurate forecasts and predictions for making informed decisions. FMs can be trained on historical financial data to develop predictive models for various financial metrics, such as stock prices, market trends, and economic indicators. These models can assist in generating forecasts, and scenario analysis, helping financial professionals make more informed investment and strategic decisions.

Financial Report Generation. FMs can generate reports, summaries, and insights based on financial data. They can automatically extract relevant information from financial statements, filings, or market reports and generate concise summaries, reducing the time and effort required for manual analysis. These generated reports can quickly overview key financial metrics, trends, and investment opportunities, facilitating decision-making processes.

Risk Assessment. FMs can assist in risk assessment by analyzing various data sources, including financial statements, market data, credit ratings, and news articles. By processing this information, they can help identify potential risks, such as credit defaults, market volatility, regulatory changes, or company-specific risks. This information can support risk management and help financial professionals make informed decisions regarding investment portfolios and risk mitigation strategies.

Financial Data Generation. FMs can assist in generating synthetic financial data that closely resembles real-world data. This can be beneficial for various purposes, including testing and validating financial models, conducting simulations, or training machine learning algorithms in a controlled environment.

5 Final Remarks

Saturn is a groundbreaking platform that combines foundation models and IT operations to empower data scientists in the financial services industry. Saturn enables data scientists to accelerate model building, optimize predictive accuracy, and drive informed decision-making by seamlessly integrating pre-built models, intuitive development tools, and robust infrastructure. With Saturn, financial institutions gain a competitive edge by leveraging cutting-edge technology while maintaining the highest security and compliance standards.

References

[1] Abubakar Abid, Maheen Farooqi, and James Zou. 2021. Persistent anti-muslim bias in large language models. In *Proceedings of the 2021 AAAI/ACM Conference on AI, Ethics, and Society*. 298–306.

[2] Rishi Bommasani, Drew A Hudson, Ehsan Adeli, Russ Altman, Simran Arora, Sydney von Arx, Michael S Bernstein, Jeannette Bohg, Antoine Bosselut, Emma Brunskill, et al. 2021. On the opportunities and risks of foundation models. *arXiv preprint arXiv:2108.07258* (2021).

[3] Andrew M Dai and Quoc V Le. 2015. Semi-supervised sequence learning. *Advances in neural information processing systems* 28 (2015).

[4] Jeremy Howard and Sebastian Ruder. 2018. Universal language model fine-tuning for text classification. *arXiv preprint arXiv:1801.06146* (2018).

[5] Xin Huang, Ashish Khetan, Milan Cvitkovic, and Zohar Karnin. 2020. Tabtransformer: Tabular data modeling using contextual embeddings. *arXiv preprint arXiv:2012.06678* (2020).

[6] Jacob Devlin Ming-Wei Chang Kenton and Lee Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of NAACL-HLT*. 4171–4186.

[7] JDMCK Lee and K Toutanova. 2018. Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805* (2018).

[8] Andy T Liu, Shu-wen Yang, Po-Han Chi, Po-chun Hsu, and Hung-yi Lee. 2020. Mockingjay: Unsupervised speech representation learning with deep bidirectional transformer encoders. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 6419–6423.

[9] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692* (2019).

[10] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781* (2013).

[11] OpenAI. 2023. GPT-4 Technical Report. arXiv:2303.08774 [cs.CL]

[12] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. 2023. Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193* (2023).

[13] Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul Christiano, Jan Leike, and Ryan Lowe. 2022. Training language models to follow instructions with human feedback. arXiv:2203.02155 [cs.CL]

[14] Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. arXiv:1802.05365 [cs.CL]

[15] Alec Radford, Karthik Narasimhan, Tim Salimans, Ilya Sutskever, et al. 2018. Improving language understanding by generative pre-training. (2018).

[16] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. 2019. Language models are unsupervised multitask learners. *OpenAI blog* 1, 8 (2019), 9.

[17] Daniel Rothchild, Alex Tamkin, Julie Yu, Ujval Misra, and Joseph Gonzalez. 2021. C5t5: Controllable generation of organic molecules with transformers. *arXiv preprint arXiv:2108.10307* (2021).

[18] Alex Tamkin, Mike Wu, and Noah Goodman. 2020. Viewmaker networks: Learning views for unsupervised representation learning. *arXiv preprint arXiv:2010.07432* (2020).

[19] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems* 30 (2017).

[20] Ce Zhou, Qian Li, Chen Li, Jun Yu, Yixin Liu, Guangjing Wang, Kai Zhang, Cheng Ji, Qiben Yan, Lifang He, et al. 2023. A comprehensive survey on pretrained foundation models: A history from bert to chatgpt. *arXiv preprint arXiv:2302.09419* (2023).