Home > AI/ML

Lessons learned and takeaways from our first production-ready Generative AI Project

10 May 2024 - 10 min. read

Generative AI

"My CEO told me to" is not a strategy.

In the last few years, we have witnessed an explosion of interest in Artificial Intelligence (AI) in all its forms.

Tools like ChatGPT, Bard, Claude, and Bing AI have made this technology accessible to a vast audience, highlighting its potential, especially in Generative AI.

Many companies have already ventured into commercial applications. Some examples are the personalization of corporate and non-corporate content, the generation of highly targeted and personalized promotional materials, the production of technical documentation, the real-time translation of texts, and even the improvement of Help Desk and customer support systems.

The hype generated, however, triggered a "GenAI rush," causing the so-called fear of missing out (*FOMO*): It is more and more common to fit Generative AI to the business problems, reversing the cause-effect relationship with a serious risk of incurring considerable investments just for the sake of it.

Despite the extraordinary possibilities offered, it is important to consider that Generative AI is not always the solution to all business challenges: it is necessary to evaluate its use from multiple perspectives to understand whether it can truly offer a competitive advantage. This advantage should be unique, tangible, and valuable enough to be perceived and justify the effort and investment, which are still very high today.

We found ourselves making these and other important considerations when implementing our first (real) Generative AI project.

We discuss it in this article, starting from why we believed that Generative AI was the best tool to maximize results and retracing all the valuable lessons we learned during the project implementation.

Next-generation planner

For a client of ours, a player in the food industry, we faced the need to develop a planning and recommendation engine. Starting from an input carrying personalized criteria, the solution autonomously composes a daily and/or weekly menu, planning meals by selecting dishes based on user preferences, available credits, and users' diet needs.

The main challenges were:

- Eliminating the risk of users being left "high and dry" because of forgetfulness or out-of-time lunch orders, particularly on Fridays for Monday delivery;
- Reducing shopping cart abandonment during the ordering phase due to the time required to browse the entire menu and select dishes.

The goal was twofold: on the one hand, to increase customer retention and loyalty while simultaneously reducing friction; on the other hand, to minimize the risk of revenues lost for the company.

Many Natural Language Processing (NLP) algorithms were already available. They can process natural language and interpret it to provide outputs according to specific decision criteria.

By combining NLP algorithms, recommendation systems, and best-fit techniques, we could achieve the same goal as we did using Generative AI.

BUT

We ended up choosing Gen AI to implement our solution. Generative AI brought us much closer to the original objectives for several reasons:

- **Greater customization**
  Traditional engines rely on content labeling and collaborative filtering, which may not fully capture the complexity of individual user preferences.

On the other hand, those based on Generative AI can generate recommendations that consider a wider range of factors, including those expressed in natural language, offering deeper personalization.

Thanks to the use of Gen AI, it was possible to leverage the data provided by the client without manual labeling. The model automatically utilizes all details in the dish description, expressed in natural language, to categorize it and decide when and if to propose it to the user according to the prompt provided.

- **Natural Language Understanding**
  Traditional engines have numerous limitations in understanding natural language requests.

  Gen AI, on the contrary, allows processing and interpreting colloquial language, accurately responding to complex queries.

- **Content Generation**
  Common engines are limited to suggesting existing content or products based on known metrics and models. Generative AI engines have the potential to create new content, such as personalized product descriptions, further enhancing the user experience.

- **Independence from Historical Data**
  Traditional recommendation engines may rely heavily on historical data and past interactions. Without these details, their ability to discover new trends or recommend new products may be limited. Gen AI-based systems can theoretically overcome such limitations by generating innovative recommendations, even without large volumes of historical data.

Now, let's dive deep into the **main takeaways** of implementing the solutions that will be essential for the success of all future projects involving Generative AI.

Successful Generative AI projects: our lessons learned and Takeaways

Quality of outcome and go-to-market times are balanced against effort

We should consider that Using LLMs can greatly simplify our lives in all contexts where extreme precision is not a critical aspect since providing a new service to the user is already a competitive advantage.

In such scenarios, there is no need to invest time and resources in identifying a specific model to solve a particular problem and invest in training it. Model training, in fact, remains one of the most expensive practices in AI.

LLMs come with training that is already broad enough to be independent from advanced adjustments. This allows us to achieve a satisfactory result in relatively short time frames compared to classical algorithms.

The other side of the coin is that an acceptable result may not be the best we could achieve.

Given that, we should always follow a Continuous Improvement approach, maybe relying on popular "classic" algorithms to work afterward to optimize the output and achieve the best match.

Perfection has no limits!

Tangible benefits for the end-user and the company

When it comes to user experience, end-user benefits are often underestimated, but, in this case, the planner has a concrete impact on two dimensions: time and money.

The time spent by the user in planning meals for the following week varies between 5 and 15 minutes (depending on needs, dish selection, and comparison of specific nutritional values).

With the first test alone, the model took 38 seconds to present a plan to the user. As mentioned, various aspects can be improved to reduce time and enhance output. However, in less than 2 minutes, the user can adjust the proposed plan and end up with a complete menu.

This means a **time savings of 60%-86%.**

Moreover, it automatically solves the problem of users needing to remember to order from one week to the next, thus avoiding the need to go out for lunch or order through delivery services. While the discomfort of not having anything to eat for lunch can't be quantified, we can measure that, on average, having lunch at a restaurant costs 30%-45% more, considering the same amount of food consumed. Therefore, there is an improved user experience factor and indirect cost savings.

From the company's perspective, the planner creates a direct relationship between customer satisfaction and profit. Every time a user forgets to order, a lose-lose scenario is created: the user experiences discomfort, and the company incurs a loss of revenue.

With the planner, the company ensures that there will be revenue every day and maximizes it, as the model will seek to use the maximum number of credits possible when planning meals.

Great! But is it sustainable?

The consumption of Generative AI in terms of economic aspects, energy, computational power, and CPU usage is considerable. It should be kept in mind that we are using an extremely powerful tool to solve specific and narrow problems that are often potentially solvable with far fewer resources (but with greater implementation effort).

It's like "shooting ants with bazookas": you'll certainly succeed in killing them but with lots of considerable side effects.

This raises the sustainability topic, on which Generative AI has a significant impact from various perspectives:

- **Energy consumption and resource usage**
  Model training and inference require enormous computational power, contributing significantly to the consumption of non-renewable resources and greenhouse gas emissions.

  Furthermore, rapid developments in Generative AI lead to rapid obsolescence of the devices and infrastructures needed to run these powerful models. This increases electronic waste, which is notoriously difficult to recycle, impacting the carbon footprint problem.

  This topic should primarily concern large providers over which we may have little impact. However, we must keep in mind that both system integrators - like us - and the end-users are active parts of this value chain.

- **Economic sustainability**
  Concerning economic sustainability, experimenting with Generative AI entails significant costs. A company approaching this technology must face high, sometimes prohibitive costs from the early stages. For example, in the case of our planner, each call to Generative AI - even in testing phases - brought considerable costs. For companies in their early years of operation, the cost of the service is a factor to be considered.

  To achieve economic sustainability in a Gen AI-based project, one must first handle the testing costs to bring the solution into production. The prod solution must generate a value for the end-user to justify the investments. Today, it is still difficult for providers to make end-users aware of the business cost behind generating the great value they can benefit from with Generative AI, and this is likely due to the proliferation of "AI-powered" tools offering similar benefits… for free!

In conclusion, allocating a sustainable budget and planning long-term is essential to ensure the economic sustainability of Generative AI usage, along with raising user awareness regarding payment for premium plans.

Cost control: joys and sorrows

When using Generative AI, the workload context can be a friend… or not (or, as we will see, even both), making cost prediction uncertain.

From a purely computational and programming perspective, using LLMs is relatively straightforward because the data processing that generates the output occurs almost automatically, like in a black box.

Since advanced computations are not required (at least in the initial phase), most of the effort (and the needle of the scale affecting the costs) shifts to system integration and understanding the logical context in which the LLM operates.

In using LLMs, in fact, costs depend on the number of tokens (that can be considered as similar to the textual syllables) inputted into the prompt and outputted from the generated result. For this reason, limiting uncertainty in prompt usage whenever possible is essential to avoid uncontrolled expenses.

That said, let's go back to our planner: each textual input causes a cost for our customer that can't be overlooked since, for the solution to function properly, each prompt must always include additional details, including the entire menu characterized by a very long text. This information is necessary and provided automatically by the system itself, along with the user's prompt (meta-prompt).

Optimizing the length and variability of the prompt is difficult, so the cost cannot be predicted with certainty.

We managed to avoid uncontrolled cost explosion with a couple of tricks: a forced limit of 200 characters for the user to express criteria, ensuring an almost stable meta-prompt length;

Through the meta-prompt, we could pass the maximum amount of credits available per person (80 credits, 8€) so that we could limit the number of dishes and, as a consequence, the number of tokens in output.

Although LLMs are not made for mathematical calculations, the result we obtained was acceptable for us since the error in using credits was minimal and for the user since the system ensures that any dishes exceeding the budget will be hidden.

What is Language

One aspect we have initially underestimated when using LLMs is undoubtedly the concept of "language".

LLMs are not limited to accepting and generating text in human-like language. They can accept input in any formal language based on some grammar and produce output in the same form.

Programming languages are included in this more comprehensive definition: JSON, for example, is a structured language that can be used as input in a prompt and generated consequently in an output. It was possible for us to use the LLM to propose the menu using JSON

It is possible to customize further by indicating the exact form of the JSON expected in the output.

To Conclude

The potential of Generative AI is now evident to everyone. Thanks to the countless opportunities to experience firsthand the power of Gen AI, end-users largely benefit from it.

However, questions and uncertainties arise for those who - like us - build and operate this kind of solution.

Is avoiding traditional AI complexities enough to justify experimentation through Gen AI?

Also, in the face of increasing interest and investments in the sector, companies, even the largest ones, are still trying to figure out how to monetize AI-based technologies fully.

In conclusion, despite their popularity among users, the path toward complete integration and sustainable use seems rather long and uncertain. Maintaining a critical and balanced approach and carefully weighing the benefits and risks associated with these emerging technologies is essential.

What do you think about this topic?

Have you already implemented a Gen AI-based solution?

Let us know in the comments!

About Proud2beCloud

Proud2beCloud is a blog by beSharp, an Italian APN Premier Consulting Partner expert in designing, implementing, and managing complex Cloud infrastructures and advanced services on AWS. Before being writers, we are Cloud Experts working daily with AWS services since 2007. We are hungry readers, innovative builders, and gem-seekers. On Proud2beCloud, we regularly share our best AWS pro tips, configuration insights, in-depth news, tips&tricks, how-tos, and many other resources. Take part in the discussion!

Alessio Gandini

Cloud-native Development Line Manager @ beSharp, DevOps Engineer and AWS expert.Since I was still in the Alfa version, I'm a computer geek, a computer science-addicted, and passionate about electronics all-around.At this moment, I'm hanging out in the IoT world, also exploring the voice user experience, trying to do amazing (Io)Things.Passionate about cinema and great TV series consumer, Sunday videogamer