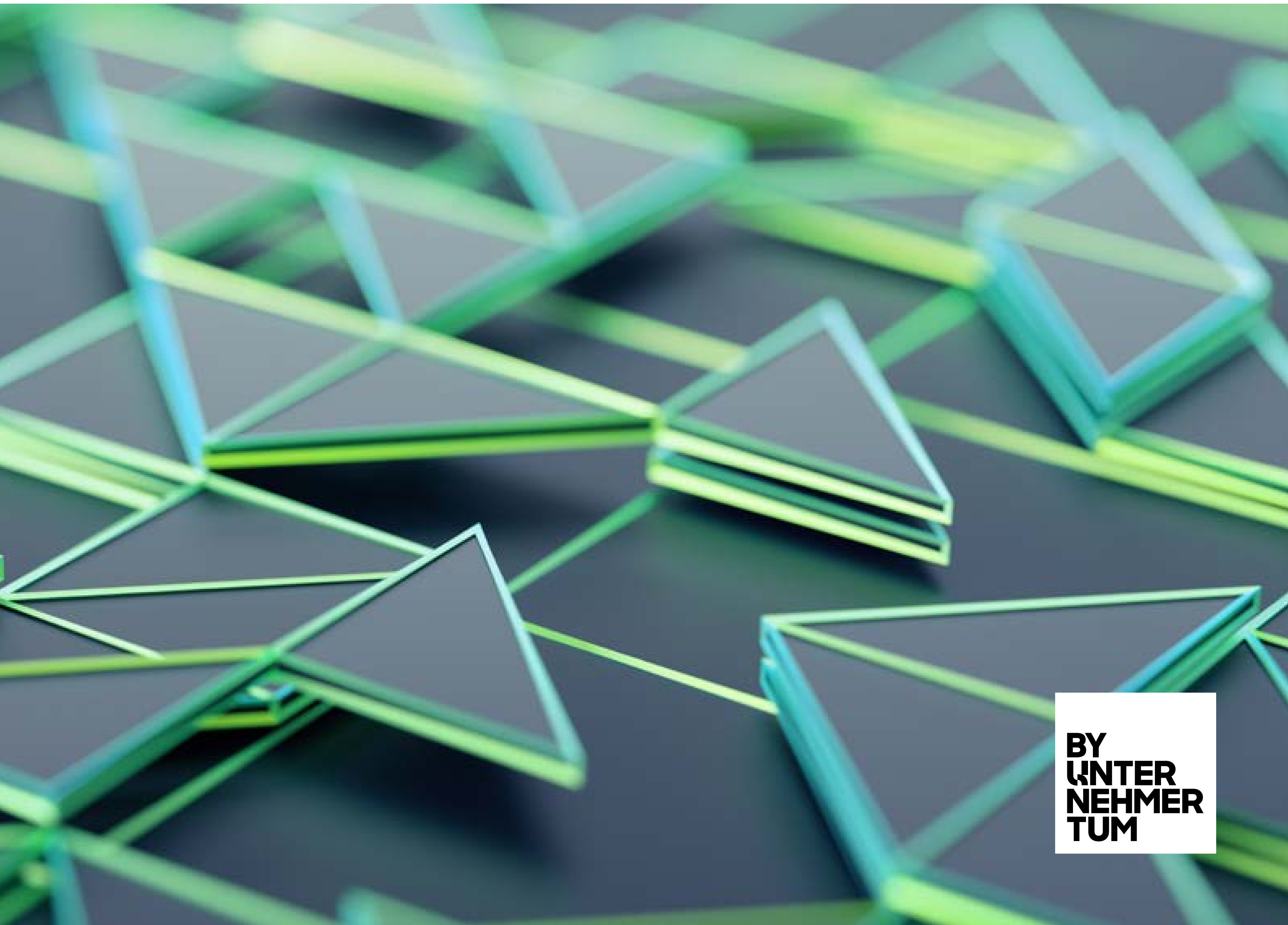
INITIATIVE FOR APPLIED ARTIFICIAL INTELLIGENCE

Applying AI: Enterprise Guide for Make-or-Buy Decisions



BY UNTERNEHMERTUM

“The field of AI is developing at a rapid pace, and hardly any company is (or should be) able to tackle all the issues on its own. A systematic approach to the make-or-buy decision is needed. However, to date most companies have not approached this question systematically at all or, even worse, they have simply delegated this decision to their standard IT purchasing process.”

Elements of a comprehensive AI strategy

There is little doubt that AI will become relevant for all companies, regardless of their industry or size. When it comes to creating value from AI, several pitfalls can be observed in practice – including the isolation of AI use cases, the lack of resources and capabilities, and a poor understanding of use cases and applications.

To avoid this, a systematic approach towards AI is needed. Therefore, from the very beginning, you need to be clear on the overarching objectives or purpose of your company: What is its goal? Furthermore, it is necessary to understand how AI can help to achieve your objectives.

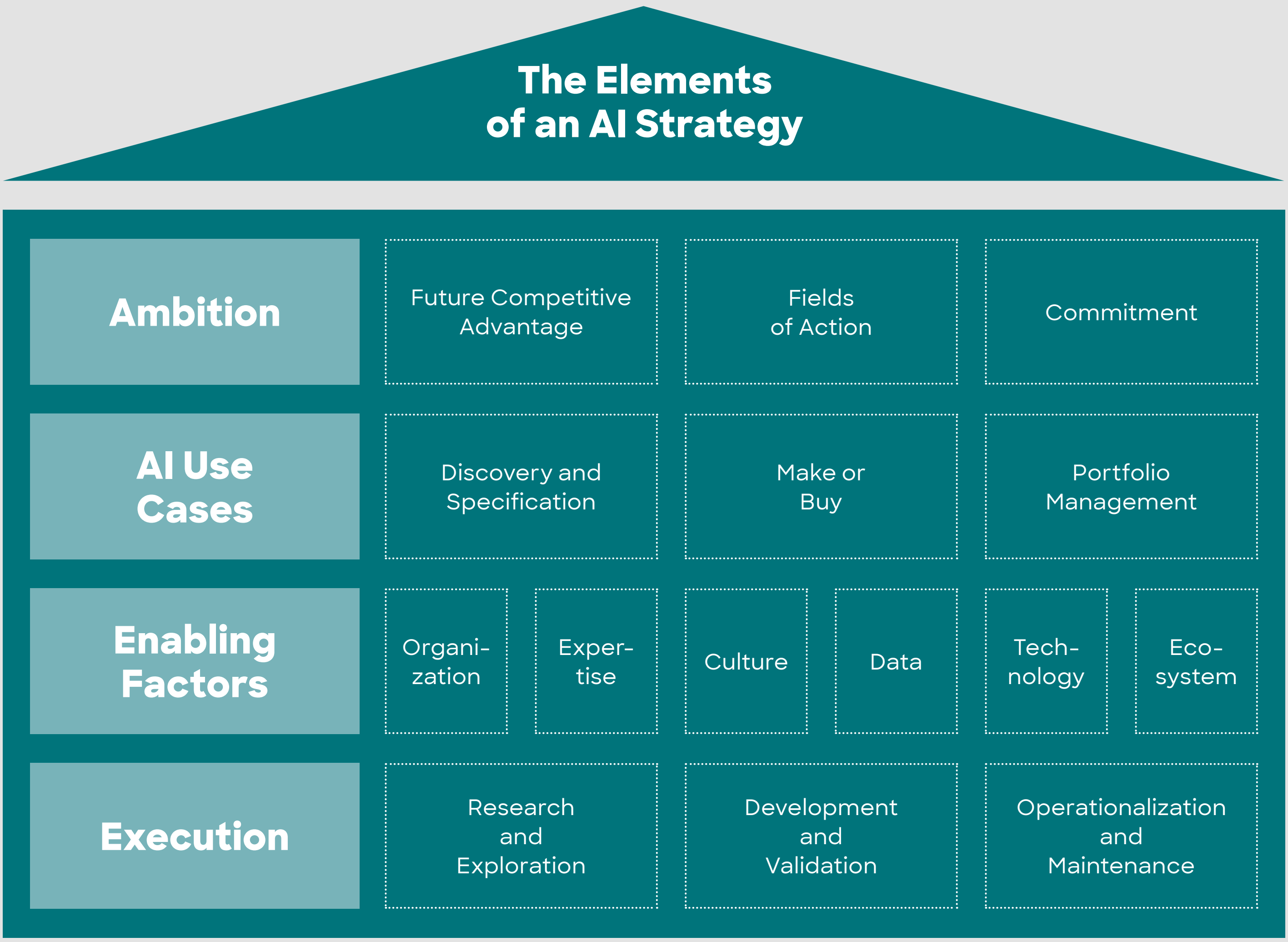
A comprehensive AI strategy consists of four parts: an AI ambition, a portfolio of AI use cases, the required enabling factors, and a clear strategy for execution.

A company's AI ambition sets the high-level goals of any AI application to be developed or deployed. It includes an understanding of the current position of the company, its competitive position and industry dynamics, including potential changes to the industry's business model. On this basis, it can be decided where the organization could benefit most from AI - within a specific product or service and/or by improving processes.

The ambition needs to be translated into a portfolio of AI use cases. To build this portfolio, you need to identify and prioritize relevant use cases.

To execute the use cases, a set of enabling factors is required concerning the organization, the people, the technology, and the AI ecosystem.

All of these aspects need to be taken into account when it comes to the development of a comprehensive AI strategy and are further detailed in our report **"Elements of a comprehensive AI strategy"**.



Introduction

With the increasing adoption of AI applications across all areas of a company – from marketing and customer service to production control – one question is becoming increasingly pressing: Should we develop AI solutions in-house or purchase commercial software? In short, this is the “good ole” question of make or buy.

This question involves many factors, including the availability of skills in-house, the relative costs, and the demands of continuously monitoring a deployed model. But one thing is certain: Making the wrong decision can be expensive. And that doesn’t just mean direct costs, but also losses in time-to-market, innovation capability, and competitive advantage.

You might expect that this question has already been answered; after all, a similar decision has been made numerous times in regard to purchasing IT applications. But AI is not IT. AI has certain characteristics which differentiate it from traditional software; accordingly, the make-or-buy decision must also be approached differently.

In fact, the question to make or buy does not exist as an either-or option for AI; it is more a continuum than a binary decision. At one extreme, even when taking full ownership, your teams will never build algorithms from scratch. They will always use packages, often pre-trained on some Machine Learning (ML) capabilities. At the other extreme, even the most productized AI (think of a spam filter) needs to adapt to you, so there must be at least some agreement on data access, confidentiality, and more – and in a business context, the required adaptation is typically quite extensive.

AI is a vast technology field consisting of many subfields. The following report focuses on the subfield of ML, as this is one of the most important and most rapidly evolving areas. The vast majority of raw ML algorithms are (still) open source, available for free, but without immediate business value. Only after training on data (often at least in part your own company data) does the trained algorithm exhibit intelligence. This is quite distinct from the mere parametrization of traditional enterprise software. Furthermore, ML predictions are inherently uncertain, and many algorithms work as a black box, making it very difficult to interpret and validate the results.

The correct reformulation of make or buy for ML is this: To what degree do you prefer doing it yourself versus partnering in regard to building and managing the AI application? With whom should you partner? And how should you structure the partnership or contract? In addition to the usual challenges that arise with any complex make-or-buy decision, there are unique challenges related to AI. As data and software are intertwined, the topic of IP ownership becomes more complicated. In addition, the inherent uncertainty and black-box nature of ML can lead to difficult situations when it comes to liability concerns. Efficient control and prevention mechanisms for bias and malfunctioning need to be put in place.

At the same time, the field of AI is developing at a rapid pace, and hardly any company is (or should be) able to tackle all the issues on its own. In short, a systematic approach to the make-or-buy decision is needed.

However, to date most companies have not approached this question systematically at all or, even worse, they have simply delegated this decision to their standard IT purchasing process.

With these challenges in mind, this enterprise guide is intended to provide helpful guidance for make-or-buy decisions with AI. The remainder of this paper is organized as follows:

We will first discuss the general structure of the make-or-buy question for AI.

The subsequent chapter provides a framework together with decision criteria for solving the make-or-buy question for an individual use case from a lifecycle perspective.

We then focus on the selection of the optimal partner, potential pitfalls of different partner types, and aspects of a good partnering strategy.

The final chapter identifies important elements for contracting in the context of AI use cases and provides a set of guiding questions.

To complete the picture, the report concludes with an interview with a vendor: Dr. Georg Wittenburg, founder and CEO of automated analytics provider Inspirient, explains why you should buy their solution (instead of building it yourself) and how best to work together.

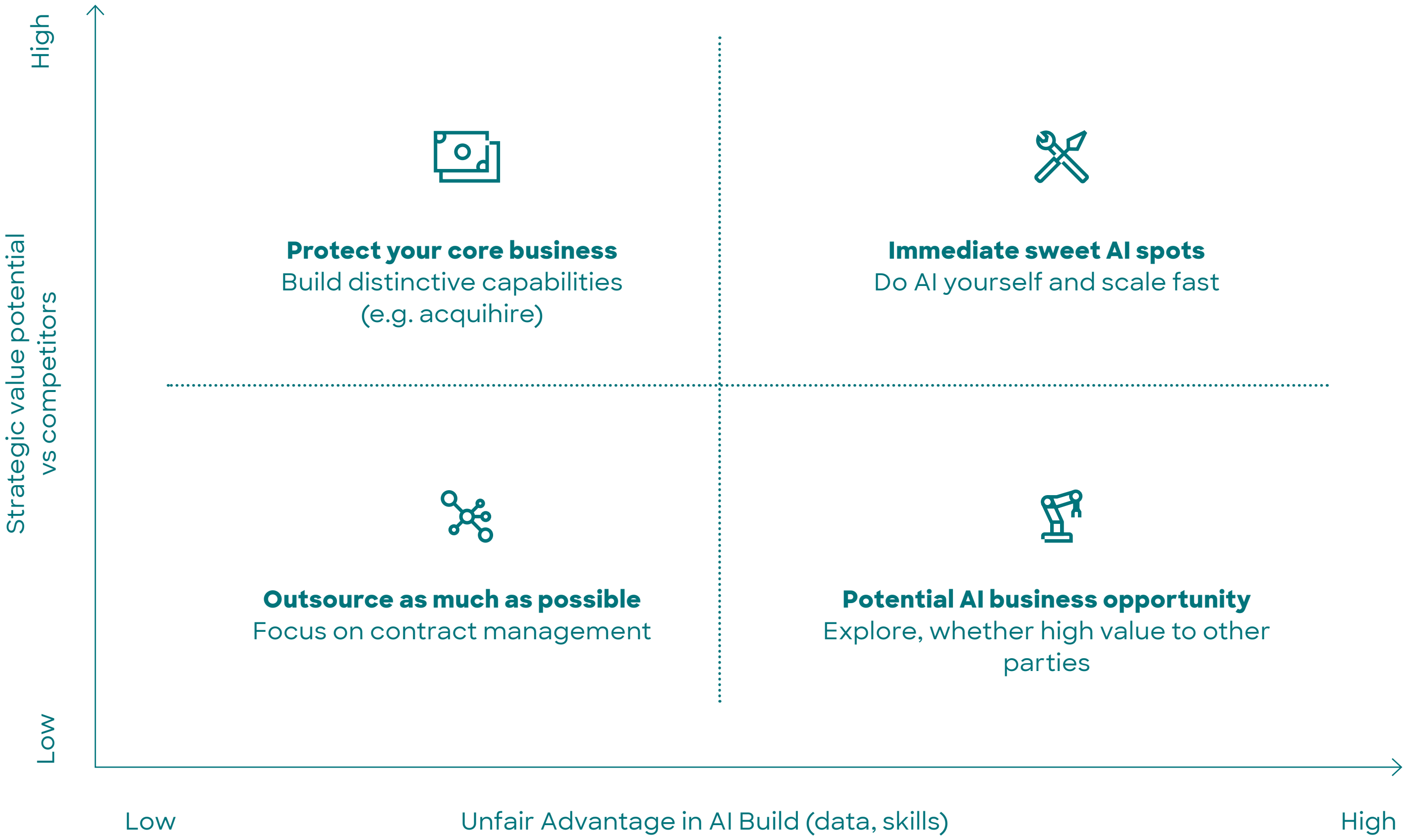
Enjoy the read!

How to approach the make-or-buy decision

At the top level, two questions are important to approach the make-or-buy decision: First, what is the strategic value for you? (Remember, you can only excel in what you own.) Second, can you build it? (Dependant on your AI capabilities and your preferred access to data).

This results in four generic strategies: There is a general consensus that you should refrain from building your own AI solutions where software / solution providers (like SAP, Salesforce, Microsoft or ServiceNow) are already integrating AI capabilities of the data handled in their systems into their suite. You are towards the lower left side of the matrix (at a disadvantage for building the AI solutions) and, often, these are not at the core of your competitive differentiation. The "sweet spot" are of course AI applications that have a high strategic value and that you as a company have the right skills and data to build - these are typically applications that you will develop internally. Of course, the question becomes more ambiguous when either the value is high but the skills are lacking or you could build something but it does not necessarily have a high strategic value. In the former case, it is often advisable to think carefully about how to develop the skills and data - for example, through acquisitions. In the latter case, an interesting business opportunity can arise - but it is worth considering carefully whether you should really pursue this use case or if there are not more worthwhile cases in the other quadrants.

Figure 1: High-level make-or-buy strategy matrix



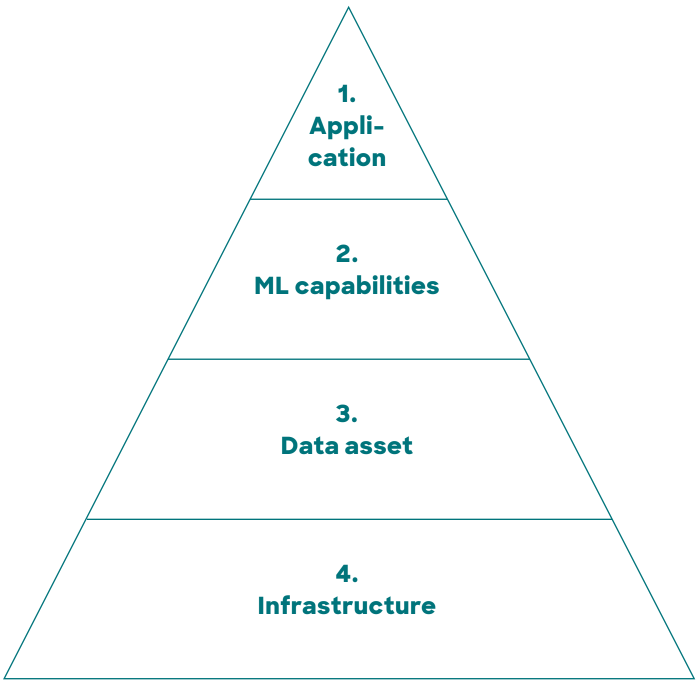
While this matrix provides a first orientation, a systematic decision must take into account many more aspects:

First, it is important to recognize that there is not merely one decision but multiple, layered decisions.

Second, there is not a binary make-or-buy decision but instead a range of options between the two extremes of (complete) internal development or (complete) external purchase.

Each use case is enabled by a series of layers (Figure 2). Make-or-buy-decisions must be made for each layer, but these decisions are interdependent, so decisions at one layer depend on and affect those at the other layers.

The top layer is the **application** itself. The application somehow uses ML either visible or not visible to the user. The decisions that can be made regarding the applications depend on the other layers. If there is a fundamental lack of data or infrastructure for a specific use case, "make" may not be an option at all – unless you want to build up the necessary assets. This report focuses on the make-or-buy decision from the application perspective and considers the other three layers only insofar as they affect the decision.



The second layer refers to **ML capabilities**. "Capability" is defined here as the types of basic technical problems with which AI is able to cope – think of the AI building blocks. We distinguish eight basic capabilities (Figure 3). With regard to ML capabilities, the company needs to determine for which capabilities it wants to build up skills and resources internally and for which it does not. For example, the capabilities necessary for autonomous driving (like computer vision) will be crucial for an automotive company to own, whereas computer linguistic technologies such as NLP models for language translation will probably not be. Companies should try to bundle suppliers for repeatedly required capabilities as part of a systematic make-or-buy strategy.

The third layer is **data assets**. Access to high-volume and high-quality data is certainly the main prerequisite for ML. Clarifying whether internal data is sufficient and, if not, where to source it should be one of the first issues addressed. Sourcing data is not limited to buying commercial data sets but may also include the creation of synthetic data or partnerships with suppliers or competitors. Other important considerations include the curation of data, the setup of data pipelines, and decisions as to who will be responsible for maintaining them.

The lowest layer refers to the **infrastructure** necessary for AI use cases, which includes the required systems and processes for developing, training, deploying, and maintaining AI applications. In general, the types of use cases your company expects to encounter determines your infrastructure decisions. For individual use cases, independent alternatives are possible nonetheless.

When these four layers are taken into account, the options for the make-or-buy decisions result from "composing" the decisions for each layer.

Figure 2: Layers of AI in the context of the make-or-buy question

Figure 3: Overview of AI capabilities

Interactive Intelligence			Motion / Creative Int.
Computer Vision	**Computer Audition**	**Computer Linguistics**	**Robotics and Control**
Process visual data and recognize objects > Understand the semantics of images or video sequences	Process and interpret audio signals	Process, interpret and render text and speech	Analyze, interpret and learn from data representing physical systems (incl. IoT) and control its behavior
Machine Capabilities			
Process large amounts of data and find patterns and 'logical' relationships	Look for optimal solutions to problems with large solution space	Make predictions about future course of time series or likelihood of events	Generate images, music, speech and more based on sample creations
Discovery	**Planning and Search**	**Forecasting**	**Creation**
Analytic Intelligence			**Motion / Creative Int.**

Three broad approaches can be distinguished:

1. End-to-end development comprises all approaches relying on development from scratch. In the case of ML, this means that the ML model is developed in-house and trained by the company. The resources for this approach need not be exclusively internal, however. As noted earlier, even when taking full ownership, your teams will likely not build the algorithms from scratch but rather use packages, often pre-trained on some AI capabilities. Hiring external developers for development from scratch or partnering with academia for research-heavy topics are subsumed under this category as well.

2. The hybrid approach involves the use of pre-trained ML models or entire development modules for realizing a use case. Examples of providers of these components include AWS or Google with their Cloud ML offerings. The amount of internal development varies with the degree of sophistication of the pre-trained component. An example of a hybrid approach would be implementing a cloud-based computer vision solution like Amazon's Rekognition or Google's Vision AI into an application.

3. Buying external AI applications refers to complete external solutions that need only minor adjustments or that can be deployed immediately. Examples are solutions provided by startups such as the translation solution developed by Lengoo, ready-made voice interfaces such as Alexa, or AI solutions sold by the large IT vendors.

Of course, in practice there are various gradations between these three approaches. So we are not talking about discrete choices, but rather a combination of different decisions, with different proportions of 'make' and 'buy' distributed across the different layers (Figure 4).

Figure 4: Three make-or-buy approaches for AI

Approach	Layers: Infrastructure ↔	Data assets ↔	ML capability ↔	Application
End-to-End Development	"Make" or "Buy"	"Make" or "Buy"	"Make and Buy"	"Make"
Hybrid Development	"Typically buy"	"Make" or "Buy"	"Buy"	"Make"
External AI Application	"Typically buy"	"Make" or "Buy"	"Buy"	"Buy"

The make-or-buy decision for a use case

In the following section we will provide a step-by-step guide for the make-or-buy decision for a specific use case. Of course, in reality the decision is often not so straightforward: not all options are known, and for many criteria there is no binary yes / no decision but rather shades of grey.

Despite this, or precisely because of it, the decision paths listed below are helpful for systematic review of the make-or-buy decision and awareness of the trade-offs involved.

Decision criteria for AI use cases

Six factors influence the make-or-buy decision in a significant way.

1. Strategic value
Strategic value refers to the value potential of the use case with regard to the competitive advantage. Sources for strategic value can be efficiency gains, cost reduction, or AI-based product features / services.

Selected probing questions:
- How does the use case ensure / enhance my strategic positioning in the market?
- Does this create new growth opportunities?
- Does this create differential cost benefits for my organization?

2. Importance of ML model ownership and control
Common reasons for the importance of ML model ownership and control are competitive advantage, safety, or regulatory concerns.

Selected probing questions:
- Do I face the risk of lock-in effects to a specific vendor with strategic and commercial downsize?
- Do I block potential vendor migration or multi-vendor approaches?
- Do regulatory requirements exist which make owning the ML model necessary?

3. Potential for learning
To what extent offers the use case an opportunity to learn from internal development? Learning can be broadly defined and refers not only to acquiring knowledge about a specific ML model or technique but also to gaining experience in executing use cases more generally.

Selected probing questions:
- Does this use case focus on an AI capability for which I foresee additional application areas within the organization and thus should build my own learning journey and scale?

4. Unfair advantage in AI build
This is defined as access to unique resources that offer a significant advantage over the competition. Sources for unfair advantage in AI build can be skills or data.

Selected probing questions:
- Do I have access to unique data sets on which I can train models that no competitor can replicate?
- Which capability is differentiating in my market? Does the solution create a sustainable USP or capability vs. competition?
- Do I have the internal resources / capabilities necessary for realizing the use case?
- If not, do I want to and am I able to develop the resources / capability internally with my own team?

5. Performance of external solutions
What is the (minimum) performance an external partner is able to provide? The performance needs to meet the requirements of the company for various factors such as technical performance, delivery quality and

speed, or customizability. Additionally, the long-term performance must be kept in mind. Usually, the model is trained on a generic benchmark, which does not directly allow for the comparison of the performance with company data.

Selected probing questions:
- How does the solution perform when trained with my own company's data?
- How does the performance of the solution improve over time through learning?

6. Total Cost of Ownership (TCO)
TCO comprises all costs necessary for the development, deployment and maintenance of the use case over the course of its lifespan (e.g., compute costs, developer wages, or costs for accessing third-party data).

Selected probing questions:
- How does this approach impact my company's P&L / balance sheet over the short term and the long term?
- How does TCO compare in each model over a two-year, five-year, or even ten-year period?
- How can the trade-off between costs and ownership of "value"/ IP be solved?

The make-or-buy decision along the use case lifecycle

The development process of ML-based applications and products is characterized by a high amount of uncertainty in the beginning. Defining all specifications up front is hardly possible in most cases and, for this reason, a completely informed decision might not be possible in advance of exploration and initial development steps. While the above considerations for decision making should be applied to all use cases, different factors may be weighted more heavily depending on the lifecycle of a use case.

The development of AI use cases typically follows four process phases. Uncertainty regarding the use case is highest in the beginning and decreases over the course of development. With decreasing uncertainty, the assessment of feasibility and value potential becomes more precise and meaningful. In contrast, the scope of the development and the resources deployed are typically lowest in the beginning and increase continuously.

The four phases of development are as follows:

1. Use case ideation
At the first stage of the process – the use case ideation phase, during which ideas are generated and prioritized – the make-or-buy topic is only partially relevant. The ideation of a use case should always be grounded in an actual need or opportunity of a company rather than simply in the availability of a solution. When it comes to prioritization, however, the existence of an external ML solution simplifies and thus increases the ease of implementation.

2. Proof of concept (PoC)
The main goal of the PoC phase is to check whether the use case is feasible in practice. Resource deployment is usually very low and uncertainty still high, as it is often still unknown whether the use case is technically feasible at all. The make-or-buy question can simplify the evaluation of feasibility in the event that an external solution already exists.

For PoCs (and also MVPs) the potential for learning should be additionally considered in the decision: You may wish to work on a use case in order to build up internal capabilities or to better understand the complexity of a particular solution. At the same time, other factors such as a thorough TCO analysis are typically less important when the feasibility of a specific use case is still in question.

3. Minimum viable product (MVP)
If the PoC phase is completed successfully, the next step is the development of an MVP. During this phase, the value potential of the use case is assessed and further uncertainties regarding the technical feasibility are reduced. If the MVP proves to be a significant value add for the firm, the use case moves on to the last phase of the use case lifecycle. Similar to the PoC phase, the existence of an external alternative can significantly shorten the evaluation process.

4. Scaling product
The last phase of the use case lifecycle is scaling the product. The main focus of this stage is the productionization and scaling of the use case. This means that the use case is rolled out across the complete company and the topics of maintenance and monitoring move into the spotlight. The make-or-buy question plays a central role at this stage of the lifecycle. Companies must focus especially on the long-term costs of different options and partnering approaches.

Mapping the decision path for the scaling product phase

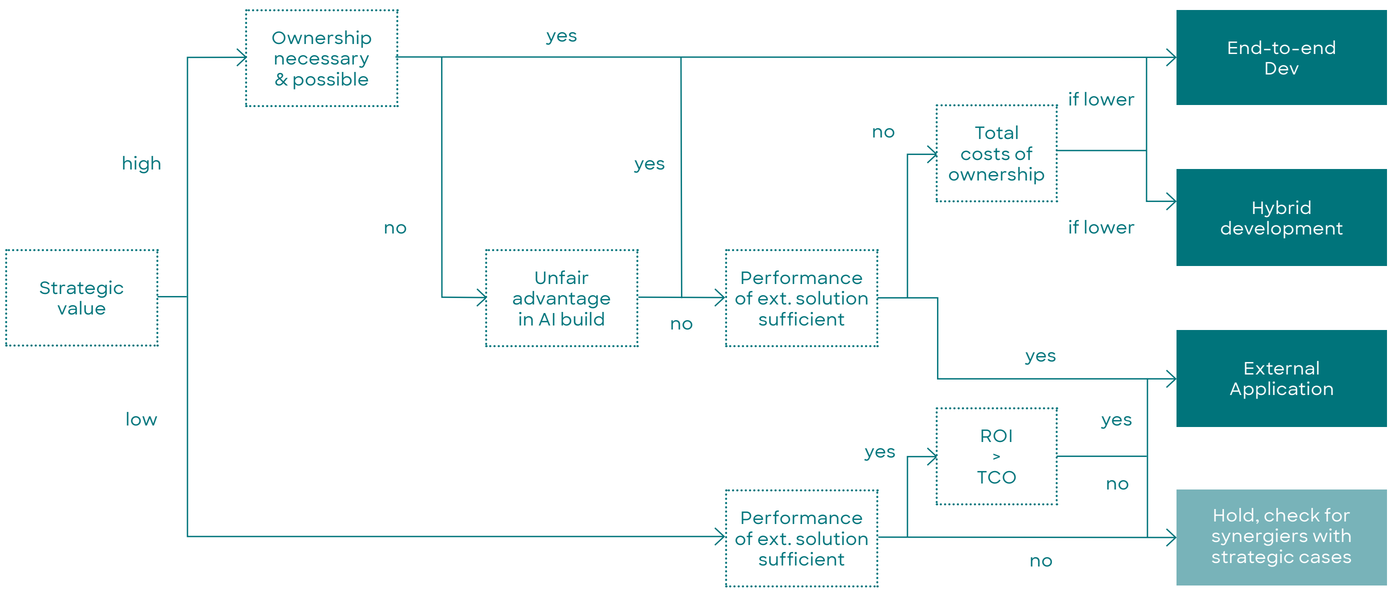
Considering the decision in an advanced phase – you want to bring an AI use case into production – it is helpful to follow the following ideal decision path (Figure 5).

The decision starts with an assessment of the strategic value of the use case. Use cases with high strategic value are then analyzed as to the importance of owning and controlling the ML model. If ownership is critical, the only possible option for realizing the use case is to develop it internally end-to-end. When ownership and control of the ML model are not critical, more decision options become available. The next step is to check for an unfair advantage in AI build. If an unfair advantage exists, you should probably utilize the advantage and build the ML model in-house end-to-end. But if your company has no unfair advantage to build it yourself, the external options should be considered. If no external solution is available, two options exist: The use case can either be realized through end-to-end development or through a hybrid approach, where pre-trained components are used. The decision between these two alternatives depends on the results of the Total Cost of Ownership (TCO) analysis. In contrast, if an external solution does exist, the company should buy the application from the external provider.

For use cases with low strategic value the decision path is simpler and shorter: Do not develop a solution internally if there is no high strategic value! So, for these use cases you should always begin by questioning whether the performance of the existing external solutions is sufficient for your needs or whether you can at least expect the performance to be sufficient at some future point, as well as whether these solutions have a positive ROI. If no suitable solution is available on the market or if the ROI is smaller than the TCO, then the use case should be put on hold until either a cost-efficient external solution becomes available on the market or synergies with other strategic use cases emerge. In the latter scenario, the use case can be jointly assessed and realized with the high strategic use case.

Even though the decision-making process in reality may differ from the ideal path, it helps to check each decision against this ideal path to ensure that one does not go in the wrong direction.

Figure 5: Make-or-buy decision path for use cases—Scaling product phase



Partnering

Selecting the right partner

The three approaches defined earlier can each be realized with the help of different partners. There is certainly no “right” partner but rather a “suitable” partner, depending on the respective approach and the specific situation.

Overall, there are seven types of partners that can be divided into two groups (Figure 6). First, there are partners that extend the internal resources of the company and support the “make” decision. The second group of partners correlates to different variants of the “buy” decision.

Figure 6: Overview of potential partners for use case execution

Make with	Buy from
In-house development team	Cloud provider
External development team	Product start-up
Academia	Solution provider
	Embedded feature

- Strictly speaking, **in-house development** is not a partnering approach for the company as a whole but is instead an internal partnership driven by the need to pool specific AI resources. That said, such an internal partnership may still prove challenging operationally. End-to-end development of the use case offers the closest integration of the use case and the strongest control of both the data and the deployed model; therefore, this option should be selected when ownership and control are critical for the company. The main disadvantages of in-house development teams are the high resource requirements and the relatively long time-to-market, as the complete use case must be managed and developed from scratch

- **External development teams** describes the development of use cases by the company with the support of external developers, for example, classic body leasing from IT consulting firms. Due to the newness of the technology and the breadth of possible required skills, many companies do not have the necessary internal resources available. An advantage of external development teams is their flexibility, which allows them to substitute for a general lack of internal resources or a temporary shortage due to tight deadlines. One challenge of working with external development teams is IP concerns, which might arise when critical elements of the use case (e.g., data or the ML-model) are provided by research partners and not sourced in-house.

- **Academia** as a partnering approach refers to the establishment of research partnerships for use cases requiring very complex or immature AI techniques and applications that first need to be explored and refined. Academic partnerships are often less expensive than partnerships with commercial developers, and in some cases this partnering approach can be a source for new talent acquisition. However, academic partnerships often suffer from long development times, so they are typically best utilized when time-to-market is less important. Also, as with external development teams, the question of who owns the IP can be difficult to resolve in academic partnerships.

- **Cloud provider** such as Microsoft, AWS, or Google offer a range of pretrained-components with varying degrees of commodity. A company can, for example, rely on the NLP model of Google or on a vendor-supplied, streamlined development framework for select types of use cases matching the company’s own use case. This flexibility of access to user-friendly capabilities that are often superior in regard to model quality is one of the main reasons for selecting components from an established software vendor. Common problems of this approach include unclear ownership of the data and—over the long term—difficulty of integration into the company’s infrastructure and domain landscape.

- **Product start-ups** are young companies that have only recently entered the market and that offer innovative but often relatively immature AI applications. This immaturity of the application can be an advantage, however, as product startups tend to be more flexible and able to react to changing requirements faster than large vendors. Startups thus offer greater opportunity for experimentation and are often more willing to adapt to the customer's needs. Moreover, if the solution provided by the startup is particularly promising and relevant to the company, it may even be possible to invest in or buy the startup outright. As for disadvantages of this approach, besides the difficulties mentioned above that often come with utilizing software vendors (i.e., unclear data ownership and struggles when integrating their solution into the IT infrastructure), the immaturity of startups may pose additional challenges. Startups often lack the experience necessary for realizing large scale projects or for scaling up their solutions, and the long-term viability of their business is typically less clear than the viability of more mature, established companies. Also, risk and liability sharing are much more complicated with startups, as the quality and long-term maintenance of the solution is much less certain. Therefore, companies need a systematic evaluation strategy for ML-based applications that specifically takes into account their uncertainty and black-box nature.

- In contrast, **solution providers** are usually established companies that have already been active suppliers for traditional software products but that now extend their offerings to also include commercial off-the-shelf AI applications. For certain use cases, these ready-to-use applications offer a lower time-to-market, higher quality, and easier implementation. Typical examples are pretrained, commercial translation models or vision models that have been trained on a task similar to the use case in question. This partnership approach often overlaps with that of the external development teams and established software vendors described above, as most larger companies offer a range of services and products to their customers that includes consulting and external development projects, customizable modular components, and full AI products.

In common with those partnership approaches, managing risk and liability sharing is often a problem when relying on solution providers as well. AI solutions must be understood as dynamic, given that their performance is related to the environment and to the input information. This and the black-box nature of AI pose significant challenges for risk and liability management. Another potential problem of solution providers is the lack of cost transparency, especially for SaaS solutions.

> There is certainly no "right" partner but rather a "suitable" partner, depending on the respective approach and the specific situation.

- The last partnering approach, **embedded feature**, describes AI-powered features of already implemented traditional software systems (e.g., intelligent recommendations for contacting specific accounts as provided within the Salesforce suite). This approach is most promising for low complexity and domain-knowledge intensive problems, where ease of implementation is the main driver for the decision. Note, however, that the embedded feature approach limits the flexibility and customizability of the solution, a disadvantage that is to a certain extent inevitable when relying on standardized features.

Supplier qualification

After choosing a partner type, you need to select a specific supplier.

Of course all the "normal" qualification requirements need to be evaluated (e.g. company size, financial stability, reference customers). However, there are also AI-specific aspects that should be considered. As companies often are not experienced in sourcing AI solutions, many lack an understanding of which "questions" to ask and which information to require. Additionally, the criteria must be adjusted to the goals and requirements of the individual AI use case lifecycle phases (Figure 8).

When selecting a partner for a specific PoC or MVP, potential suppliers need to provide a range of different documents and information for verifying their fit and suitability for the case. Additionally, open questions on who owns which components (e.g. data, model) of the solution need to be clarified before the actual development starts. AI-specific legal and regulatory documents are certifications depending on the use case and proof of specific insurances.

Figure 8: Checklist for assessing supplier for the AI use case lifecycle model

Deliverables	PoC & MVP phase		Scaling Product phase
Case-specific information			• Hardware • Distribution of responsibilities
AI-related information	• Data-related tools & processes • Model-related tools & processes • Quality assurance processes & tools	+	• Maintenance processes & tools (e.g. retraining)
Legal & regulatory agreements	• (Certifications) • Proof of insurance	+	• Certifications • Data retention, documentation & protection agreements
Documents	• Implementation guidelines • Documentation • Source code	+	• Hand-over & training materials

Apart from these documents, background information about the supplier should be requested. Especially getting to know the team and their experience with AI in general and with the specific subtype necessary for the use case is important. Suppliers should provide details about their internal processes, methods and tools with regards to the development process, including:

- **Data**
  - Expectations regarding data quality and how data is to be handled
  - Data management processes: definition of ETL processes, data pipelines tools
  - Definition of data quality, evaluation and monitoring of data quality → e.g. mechanisms for identifying data drift, "unit testing" for data

- **Model**
  - Process and tools for ML pipeline
  - Model management: Versioning processes and tools
  - Evaluation of feature performance
  - Processes for evaluating and comparing models
  - Documentation of (performance) differences between model versions
  - Model quality measures
  - Model explainability and transparency: documentation of features and impact of features

- **Licences** the supplier relies on

Furthermore, the supplier should provide implementation guidelines and comprehensive documentation. Implementation guidelines cover the solution approach the supplier plans to rely on for realizing the use cases (e.g. open-source / pre-trained vs. from scratch / new) but also questions regarding where the data is processed or what type of AI is the most suitable for the case. The documentation offers information about the actual development and implementation, for example the used training process, details about the data set, the model versioning or protocols on training. The documentation should cover the complete development process and include the source code. The complete replication of the development should be possible on the basis of the documentation and the source code.

In the product scaling phase, all deliverables from the PoC / MVP phase, and additionally detailed plans for the rollout and information regarding the long-term support are required. Suppliers should provide information on how they plan to execute the maintenance and monitoring. This includes retraining processes, model performance monitoring and KPIs and processes for detecting model degradation due to e.g. data or concept drift. The rollout plans must include decisions on the long-term data storage and the hardware (e.g. computing units) and clarification of who is responsible for which part of the execution and later maintenance. Also part of the rollout plan are the locations where the use case will be deployed and used (e.g., all production sites or offices), again detailed documentation of all steps and aspects, and – if the complete use case is to be handed over to the customer – a detailed handover and training plan.

Depending on the particular use case and industry, further requirements might be necessary to clarify and fulfill. Data retention, documentation and protection policies can vary for different industries. For safety-critical systems often additional certifications, supplier checks or insurances are necessary and limitations with regards to data storage and handling exist.

An important aspect of the partner qualification process is the analysis and evaluation of the performance of external solutions.

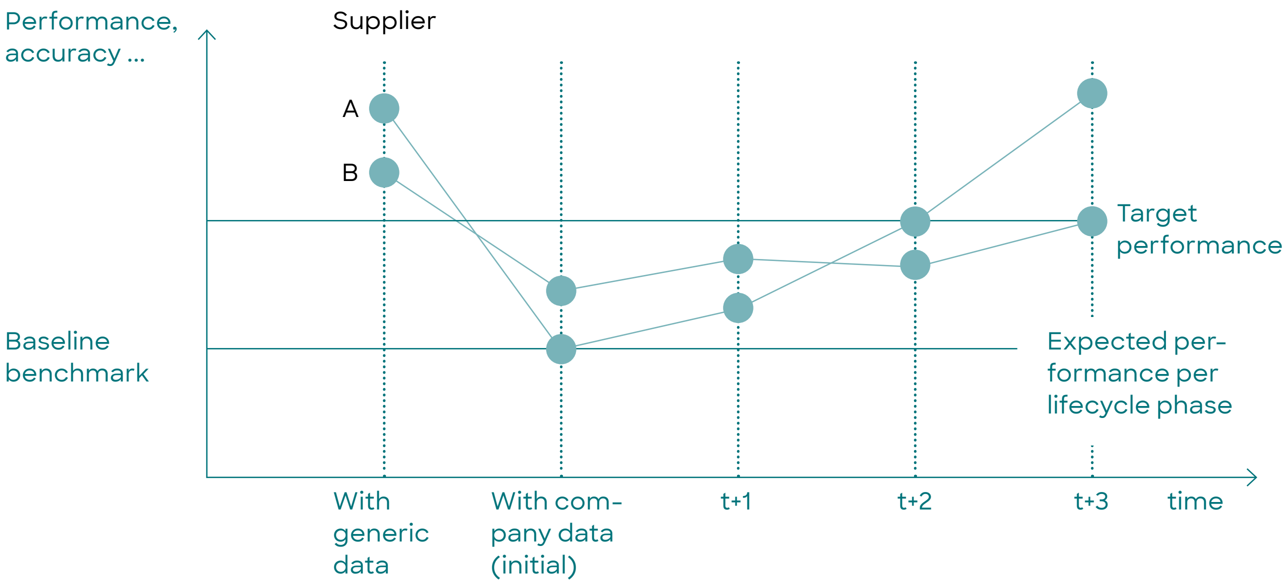
When analyzing and comparing the performance of external solutions, it is important to keep three things in mind (Figure 9).

First, the benchmark performance is usually based on generic data, so the same performance will not necessarily be achieved when training on company data or with particular business problems. Companies need to evaluate external solutions with their own data and cannot simply rely on publicly available benchmarks or results from reference projects provided by the vendor.

Second, the performance of a potential external solution will likely improve over time due to learning in production. Learning is not homogeneous and linear for the different solutions. This means that the best solution may not be the one that performs best in the beginning but the one that improves most over time.

Third, it is important to avoid vendor lock-in. The company should own the data or ideally the trained model itself so as to ensure transferability when switching to a different solution. This question will be discussed in more details further down.

Figure 9: Development of performance of different applications over time



Management of partnership

A comprehensive AI partner strategy should be concerned not only with selecting a specific partner but also with achieving overarching objectives:

Figure 10: Overview of conditions for successful AI partnerships and supporting actions

	Actions for achieving objectives
New perspectives and ideas	• Technology scouting & startup management • Market studies • Ideation funnel • Academic partnerships
Overview of suitable partners for specific problems & co-creation	• Systematic supplier research and qualification process • Defining / agreeing on quality levels or KPIs and monitoring them • Checking quality before contracting • Finding out who provides what data and is willing to share • Storing data in a sharable way
Trust-based, long-term collaboration	• Honesty regarding capabilities / needs • Alignment of goals • Structured and intensive onboarding process • Checking for similar company values / ways of working • Contracting in a win-win situation

First, AI partnerships can offer insights into other companies' experiences with and approaches to AI or specific unsolved problems. These partnerships may also inspire new ideas and new perspectives, and in some cases offer the opportunity to discover new talent. Fostering openness for new ideas and perspectives and actively working on integrating them is vital, given that AI changes and advances at a rapid pace.

Second, the complexity and vastness of AI as a field means that most companies will not have the capacity to develop all the necessary capabilities in-house and will lack crucial internal assets such as sufficient data. Entering into partnerships with suppliers or even competitors will be necessary, therefore, in order to be successful in the field of AI. A prerequisite for selecting the optimal partner for a specific problem or even for co-creation is the development of an exhaustive information overview of all potential partners. This increases the chances of forming partnerships that will, over the long term, prove optimal in regard to factors such as delivery speed, service offerings, and price.

Third, the development process of AI is experimental, and project success often requires the ability to tolerate and manage a comparatively high degree of uncertainty. Additionally, sensitive and valuable assets such as data often need to be shared. Therefore, successful AI partnerships typically entail the development of a long-term relationship based on trust and a shared understanding of the nature and importance of effective collaboration (e.g., as when each partner is familiar with the communication patterns and problem solving culture of the other partner).

How to structure an AI contract

When contracting for AI companies, it is important to keep in mind the unique characteristics of the technology that can increase the complexity of contract negotiation. Questions regarding to what extent partners can profit from the final solution (e.g., who can reuse the ML model) or how partners can protect themselves from potential risks of a black-box technology-solution pose special challenges for the contracting process.

Following are seven elements of contract negotiation that are crucial when contracting for ML.

1. Contract design / structure
refers to the general arrangement of the contracting process, for example, whether one comprehensive contract is signed in the beginning of the project or sequential contracts are negotiated and signed after successful completion of each development stage. Another important factor to consider here is the use of stage gates for managing project progress and success. Due to the experimental and iterative nature of AI development projects, it is impossible to assess all unknowns and uncertainties up front; the different forms of sequential contracting or stage gates are mechanisms for managing this inherent uncertainty. Although the specific gates and the number of gates can vary for the individual use case, most contracts should be orientated on the following five basic gates: (1) data, (2) feasibility, (3) ROI or business case, (4) quality & scope and (5) enterprise integration. The first stage and gate focus on the familiarization, preparation and exploration of the available data. Next follows the development of a first rough solution for assessing the feasibility of the use case. After the feasibility is validated, the next stage is proving that the use case is a profitable business case. The gate should be a rough ROI target the case has to surpass. If the business value is validated, the solution is fully developed with focus on quality and scope. The last stage should aim at the full integration of the solution in the broader enterprise environment (e.g. IT infrastructure, domain processes).

Central questions:
- Which phases and stage gates do I need to define?
- How do I handle the exploration phase?

2. Performance measures
apply in this case to the performance of the finalized development project. As contracting for AI can be carried out for different lifecycle phases of AI use cases, individual performance measures can vary. PoCs and MVPs require a different set of criteria for measuring project success than do Scaling Products. For fully productionized AI systems, traditional performance measures such as performance, accuracy, availability, and so forth are applicable to a limited extent. In contrast, the performance of PoCs needs to be evaluated on an individual basis, depending on the specific agreed upon objective. In general, a rough estimation of the minimum ROI the case has to generate to be profitable is a good measure at this stage. After a better understanding for the case has been developed, typically a few months into a bigger PoCs or at the MVP stage, roughly estimated accuracy and performance improvements can be used as measures. Also, not only defining levels of acceptable performance for each stage can be beneficial, in joint development projects, clarifying who is responsible for reaching them can also be helpful.

Central questions:
- What is a suitable minimum level of performance for my use case?
- Which factors are most suitable for measuring and monitoring the performance of my use case?
- What are the different performance indicators relevant during development?
- How can I benchmark the performance of suppliers before learning and after learning has occurred?

3. Data appropriation rights (IP)
are the first new element relevant for ML contracting. This element is derived from the classic IP management agreements, but certain adjustments need to be made for it to be applicable to AI. As already explained, AI depends on data, and data is intertwined with software. Therefore, data and the developed (or trained) ML model are a strategic asset and must be managed in a way similar to traditional IP. Contracts need to include a detailed explanation of the value each partner is creating and owning based on

the provided data (e.g., uniqueness) and the resulting ML model (e.g., value-add through training with unique data). Furthermore, clear agreements about who is allowed to profit from what need to be negotiated. For example, is a product startup allowed to reuse in another project an ML model that has been trained with the customer's internal data?

Central questions:
- Are there security or privacy issues which make it especially critical for me to own the data and model?
- Who owns the (trained) model?
- Who can profit from additional value created? What rights does the supplier have (e.g., reuse of data)?
- What is an acceptable trade-off between fees and rights?
- How can I protect unfair advantage with regard to the data (e.g., uniqueness of the data set)?

4. IP buy-out rights
are closely related to the data appropriation rights and can be interpreted as an additional specification. The main concern here is the definition of clear criteria and the determination of situations in which a company should be legally allowed to override the agreed upon IP clauses and have the right to buy back full ownership of the IP.

Central question:
- In which cases do I want to or need to buy out the rights to use my model / data?

5. Data protection
needs to be considered because, as mentioned above, data is a strategic asset for AI. Difficulties regarding data protection can arise out of regulatory requirements (e.g., financial data in the banking industry), the sensitive nature of certain data sets (e.g., customer data), or complex ownership structures (e.g., third-party data with limited access and usage rights). Adequate data protection actions, such as anonymization of specific data sets, and mechanisms for monitoring and enforcing these mandated actions need to be defined in the contract.

Central questions:
- What protection requirements are needed for different data types (e.g., personal data vs. machine data)?
- What are the minimum requirements for the handling of sensitive data (e.g., GDPR, regulation)?
- What special clauses are needed to ensure the anonymization of sensitive data?

6. Data-related liability
is a very important topic, as AI-based solutions possess an inherent uncertainty due to the nature of the technology. Often ML-models are also black boxes; therefore, results cannot be evaluated and explained or only to a limited extent. Also, ML-models cannot be regarded as static. Due to the continuous learning and feedback loops after deployment, the quality of the predictions can silently degrade over time as the underlying target or the data changes (i.e., concept or data drift). Problems can also arise from outliers and edge cases, problems which most often cannot be anticipated fully in advance. Companies need to specify control and countermeasures and also define who is responsible for which cases of malfunction. A possible way to mitigate the liability is relying on insurance. First insurance companies, as for example MunichRE, have extended their product portfolio with new tailor-made solutions for insuring AI-based solutions.

Central questions:
- What measures constitute adequate prevention mechanisms?
- What constitutes optimal risk distribution?
- How do I handle outliers / edge cases?
- How can I ensure accurate performance over time and even when the environment is changing?
- Is a safety margin necessary, depending on the criticality of the case?
- What is the effect on runtime (i.e., termination of cause)?

Depending on the partner type and the position in the use case lifecycle, different ones of these elements should be prioritized. The following checklist provides a high-level overview of the most important points to consider:

Figure 11: Elements of an AI contract

Contract Element / Partner type	PoC / MVP	Scaling Product
In-house development team	• n / a	• n / a
External development team	• Agreements on data protection	• Agreements on data protection • Focus on clarifying model ownership / IP • Definition of performance measures • Agreement on scope of documentation
Academia	• Agreements on data protection • Focus on clarifying model ownership / IP	• Agreements on data protection • Limited application for industrialization & scale up
Established software vendor	• Agreements on data protec-tion / data residency	• Agreements on data protection • Focus on data & model owner-ship / IP • Definition of performance measures • Clarification of liability clauses • Data residency
Product Startup	• Experimentation-focus and sequential contracting • Agreements on data protection	• Agreements on data protection • Focus on data ownership / IP • Clarification of IP buy-out rights / code escrow • Definition of performance measures • Clarification of liability clauses
Solution provider	• Stage-gate-based contracts • Agreements on data protection	
Embedded feature	• Feature available as part of an existing enterprise software solution	

Let a vendor have their say: Interview with Inspirient

This report examines the make-or-buy decision from the perspective of a company looking to implement its AI use case. But how do vendors view this question? Why should you buy their solution and not build it yourself? And what is the best way to work together? We discussed these and other questions with Dr. Georg Wittenburg, founder and CEO of Inspirient (www.inspirient.com). With its AI system, Inspirient helps companies to scale their analytics capabilities beyond humanly possible levels, in order to critically evaluate more data in less time and at reduced cost. This approach has helped banks to lower their risk profile, manufacturers to streamline their processes, and retailers to optimize their prices. Georg founded Inspirient after five years of ICT research and three years as a management consultant at the Boston Consulting Group.

Why should a company buy your solution instead of developing the use case by themselves?
Even in the 21st century, not every company needs to become a full-blown tech player or AI-first company. The big-picture reality is that the pace of innovation is just too fast for companies to do everything in-house, and business incentives for tackling larger questions are limited, e.g., having an AI autonomously analyse business data as in our case. Instead, companies need to clearly define their (future) core competencies and then derive their strategy position with regards to AI. Rather than trying to become the best developer of AI, most companies are likely better off striving towards being the fastest adopter of AI (as well as other emerging technologies). Therefore, firms should most definitely prefer buying commercial solutions whenever they become available. For this strategic approach, agile and timely integration of external innovation is critical: Companies need to be great at (out)sourcing, vendor selection / management, agile integration, and technology transfer – eventually fostering a network of innovation around them. Speeding up the innovation-to-business-impact process is key for staying on top.

What are typical problems / false expectations from companies on the buyer side when they interact / select an AI application vendor?
Upon their first contact with AI, companies often focus on an outdated set of questions that may have been applicable to commercial AI applications five to ten years ago but are not anymore: First of all, AI is at times incorrectly perceived as an emerging, experimental technology with poorly understood business applications. As a matter of fact, the field of AI predates both modern microprocessors and the Internet and is one of the most mature fields of research to transition to commercial applications. More practically, many current AI applications have been commercially deployed in the last years in markets outside of Germany, and thus a lack of experience is rather a regional anomaly than a deficiency of the field.

"Companies really need to bring their A-game project management and outsourcing skills to the table. Like all projects, AI projects need a dedicated project manager and senior sponsor just as they need a budget and defined processes and milestones."

Second, companies may worry about whether an AI project they undertake with a vendor comprises "real" AI. This is motivated out of the need to responsibly allocate their innovation budget, but in the case of AI it's unfortunately misguided. The understanding of what "real" AI is (and what it's not) has been constantly changing: 25 years ago, winning a game of chess against the human world champion may have been regarded as a pinnacle achievement of AI, but today it would hardly be perceived as intelligent. For multi-year corporate innovation programmes this implies that the perception of what "real" AI is may shift between programme inception and application roll-out. It's really best to not worry about the "real" AI-ness of a solution, but rather focus on the business value it delivers. And if the best solution to a business question happens to be one of the more opaque techniques, that is not necessarily a reason to refrain from using it. Companies deal with "black boxes" all the time (suppliers, customers, human employees). It's just like outsourcing: Retain enough capability to manage the external component, avoid vendor lock-in and then, potentially, cut costs.

A third point is exaggerated expectations towards model accuracy. Building on their experience from IT outsourcing, companies may start the discussion expecting 99.999 % accuracy of an AI system. While not impossible in theory, when looking at a real use case with real-world data it is a fact that the final few percentage points in training a model require the most effort, i.e., it is not economically reasonable to try to achieve them. Also, in most cases 99,999 % or even

“Rather than trying to become the best developer of AI, most companies are likely better off striving towards being the fastest adopter of AI.”

Dr. Georg Wittenburg, founder and CEO of Inspirient

99 % accuracy is actually not necessary for establishing a good business case. A more realistic approach is to take the current human precision as a target benchmark and then first realize cost savings instead of trying to be both better and cheaper at the same time.

A last aspect, companies really need to move on from innovating via the “lighthouse” approach. Quite often we meet companies who focus on only one question, project or solution provider at a time. This keeps current costs low and does not overburden the organisation with change, but comes at the expense of higher risk (of an individual project failing) and reduced speed (by looking at ideas sequentially). For innovative topics such as AI, an operational portfolio approach is more efficient at reaching actual outcomes. In other words, you need to race with multiple horses for one to reach the finish line.

What would you as a supplier wish for to ensure a better selection process / contract negotiations / onboarding?
Allow sufficient time to actually understand the options. In particular, really understanding and then deciding upon how a new technical approach may affect a company’s cost structure or even its business model does take more than five minutes, even for seasoned decision makers. For example, autonomous, AI-based analytics allows to efficiently mine company and external data for the unknown unknowns and entirely changes how companies can approach controlling, compliance and risk management. A 5-minute-per-pitch innovation event is not the right format for diving into this kind of new and unfamiliar technology – but going out for lunch together just might be! To be more successful, companies should take their time to really understand where AI can add value for them specifically and also look beyond the three familiar use cases at the top of the current hype cycle.

Finally, companies really need to bring their A-game project management and outsourcing skills to the table. Like all projects, AI projects need a dedicated project manager and senior sponsor just as they need a budget and defined processes and milestones. AI-centric innovation is, for the most part, just another process that can, but also must be managed.

Authors

Philipp Hartmann serves appliedAI as Director of AI Strategy. Prior to joining appliedAI, he spent four years at McKinsey & Company as a strategy consultant. Philipp holds a PhD from Technical University of Munich where he investigated factors of competitive advantage in Artificial Intelligence.

Andreas Liebl is Managing Director at UnternehmerTUM as well as appliedAI. Before joining UnternehmerTUM, he worked for McKinsey & Company for five years and completed his PhD at the Entrepreneurship Research Institute at the Technical University of Munich.

Maria Schamberger is a Senior AI Strategist at appliedAI. She has a rich background within the financial services industry from her former role as Vice President at the Allianz Group as well as consulting and research experience from McKinsey & Company. Maria studied Corporate Innovation at Stanford University and Banking at the Frankfurt School of Finance and Management.

The authors would like to thank Christine Schöber for her invaluable contribution in writing this report and Andrea Rusp for designing this publication.

Contributors

This report is the result of the appliedAI working group “Make-or-Buy descisions in AI” and has drawn on the experience of leading experts from appliedAI partner companies:

Dr. Stefan Dierks, Senior Development Engineer, Corporate R&D, Rohde & Schwarz GmbH & Co. KG

Matthias Neuenhofer, Project Manager Corporate Strategy, BayWa AG

Daniela Rittmeier, Head of AI Hub, Artificial Intelligence Center of Excellence, BMW Group

Thorsten Stein, Purchaser for Data Science, BMW Group

About appliedAI

appliedAI is Europe's largest initiative for the application of cutting edge AI. Our vision is to shape the European AI ecosystem as a trusted enabler and innovator. With partners such as NVIDIA, Google, BMW, Siemens, Deutsche Telekom and many more, we have been strengthening and building the next champions in AI since 2018.

You can find more information about appliedAI at: www.appliedai.de

INITIATIVE FOR
APPLIED ARTIFICIAL
INTELLIGENCE

UnternehmerTUM GmbH
Lichtenbergstraße 6
D-85748 Garching
Deutschland

BY
UNTER
NEHMER
TUM