EUROPEAN LAWYERS
EUROPEAN BARS
CCBE
BARREAUX EUROPÉENS
AVOCATS EUROPÉENS

Council of Bars and Law Societies of Europe
The voice of European lawyers

Rue Joseph II, 40/8 - B-1000 Brussels
+32 (0)2 234 65 10 | ccbe@ccbe.eu | www.ccbe.eu

CCBE technical guide on the use of AI tools and models by lawyers

27 March 2026

Table of contents

Introduction

The present guide aims to equip lawyers with foundational technical knowledge allowing them to select, evaluate, and use the available AI tools in compliance with the applicable laws and professional obligations. As such, it will focus on the most popular functionalities in legal practice and shed light on the underlying architectures of different AI solutions. A technical glossary of key terms accompanies this guide for the benefit of its legal audience. They terms explained in the **glossary** are indicated in **blue** throughout the text. The guide does not constitute product endorsements or step-by-step tutorials for specific products. However, given that in some cases there are only a few providers on the market[1], some references to the existing brands are unavoidable.

This guide is to be read together with the CCBE guide on the use of generative AI by lawyers (GenAI guide)[2] and the CCBE guidelines on the use of cloud computing by Bars and lawyers.[3] The focus of the GenAI guide was to outline the key characteristics, the definition of generative AI (GenAI) and to elaborate on the benefits, risks and professional obligations regarding the use of GenAI by lawyers. The focus of the cloud computing guide was to elaborate on the types of solutions available and the professional obligations engaged when using cloud services.

Lawyers primarily interact with AI systems as end-users and may not possess a comprehensive understanding of their internal technical functioning. However, a lawyer is under a professional duty to possess a sufficient understanding of the essential functioning of the technologies they use. This duty is reflected in the principle of professional competence of the CCBE Charter of core principles of the European legal profession ("Charter of core principles").[4] The commentary to this principles states, among other things, that: 'A lawyer should be aware of the benefits and risks of using relevant technologies in his or her practice.'

Technological competence is also a requirement under the EU AI Act[5] (Article 4) which obliges the providers and deployers of AI systems (which includes lawyers) to ensure 'a sufficient level of AI literacy of their staff and other persons dealing with the operation and use of AI systems on their behalf [...].'[6]

Having foundational knowledge of the functioning of AI tools is essential for lawyers, as it enables them to identify the technological solutions appropriate for their practice and to assess the implications of these choices for compliance with applicable legal and professional obligations. One such obligation is the protection of confidentiality of lawyer-client communications. Lawyers are custodians of client data and have deontological obligations to protect their clients and their data. Such technological awareness enables lawyers to distinguish between situations where they have a choice among tools, where no viable options exist, and, most critically, where specific technologies should not be used and why. It also allows lawyers to recognise when they need to consult a specialist, such as an IT professional, for further assistance.

---

[1] This is, for example, the case with the GPU providers where NVIDIA is the dominant leader in the AI GPU market, with an estimated 85% to over 90% market share in AI-optimised GPUs.

[2] CCBE guide on the use of generative AI by lawyers (2 October 2025): https://www.ccbe.eu/fileadmin/speciality_distribution/public/documents/IT_LAW/ITL_Guides_recommendations/EN_ITL_20251002_CCBE-guide-on-the-use-of-the-use-of-generative-AI-for-lawyers.pdf

[3] CCBE guidelines on the use of cloud computing by Bars and lawyers (27 February 2025): https://www.ccbe.eu/fileadmin/speciality_distribution/public/documents/IT_LAW/ITL_Guides_recommendations/EN_ITL_20250227__CCBE-guidelines-on-the-use-of-cloud-computing-by-lawyers.pdf

[4] Charter of core principles of the European legal profession and Code of conduct for European lawyers: https://www.ccbe.eu/fileadmin/speciality_distribution/public/documents/DEONTOLOGY/DEON_CoC/EN_DEON_CoC.pdf. See also: Section 4.2 of the CCBE guide on the use of generative AI by lawyers (see supra note 2)

[5] Regulation (EU) 2024/1689 of the European Parliament and of the Council of 13 June 2024 laying down harmonised rules on artificial intelligence and amending Regulations (EC) No 300/2008, (EU) No 167/2013, (EU) No 168/2013, (EU) 2018/858, (EU) 2018/1139 and (EU) 2019/2144 and Directives 2014/90/EU, (EU) 2016/797 and (EU) 2020/1828 (Artificial Intelligence Act)

[6] The full text of Article 4 reads: "Providers and deployers of AI systems shall take measures to ensure, to their best extent, a sufficient level of AI literacy of their staff and other persons dealing with the operation and use of AI systems on their behalf, taking into account their technical knowledge, experience, education and training and the context the AI systems are to be used in, and considering the persons or groups of persons on whom the AI systems are to be used."

Current Generative AI Solutions

The market for AI tools is evolving rapidly and offers numerous options, though sometimes not easily visible. As a result, users may be inclined to adopt whichever tools are most immediately accessible, often those that are inexpensive or heavily promoted, even where such tools are not specifically designed for legal practice.

For most legal professionals who have not undertaken advanced technical education, currently available generative AI systems from various providers may appear indistinguishable. The user interface typically comprises a chat window, predominantly browser-based, enabling users to interact with the AI system through conversational input. Users can pose queries and upload files, such as text, images, and videos, and get responses that vary in quality and reasonableness. Importantly, very few of these systems afford users insight into their underlying mechanisms or provide transparent information regarding the processing of user-provided data.

This guide will provide some insight into various forms of generative AI systems and will help determine the specific category with which a user (a lawyer) is engaging. First, it is essential to understand the distinction between an '**AI model**' and an [7]'**AI system**.' An AI model is just one technical component within a wider system: the end result of the complex machine learning process called the training, and which will serve as the single most valuable part of the broader system. In contrast, an AI system refers to the complete, operational application that integrates one or more AI models with additional components, such as data pipelines, user interfaces, and underlying infrastructure, to deliver a functional and accessible service.

Most of the contemporary AI models that are used by lawyers are **large language models (LLMs)**. They rely on a specific architecture of neural network language models that follow the transformer architecture. Instead of reading text one word at a time like earlier systems, transformers "look" at all the words in a broader text at once. This helps them understand context much more effectively, for example. which words relate to each other, even across long pieces of text. It became a worldwide success after OpenAI implemented it in a decoder-only approach via an architecture called GPT (Generative Pre-trained Transformers) which uses the decoder portion of the transformer architecture (designed to predict the next word in a sequence). This is the part specialised in generating text.

More recently, multi- or omni-modal models have become more commonplace and are trained not only on text, but on images and videos, and they are not always called LLMs.

Large models are enormous both during training, when they are created and refined, and during everyday use, known as the **inference phase**. Their computational demands are so high that they cannot run on laptops, phones, or typical local servers. Instead, our devices send queries to powerful remote servers that process the request and return the result. The biggest and most advanced models require such substantial computing power, energy, and cooling that they can operate only in specialised data centres, not on consumer or even standard corporate hardware.

These high computational power demands also come with costs. In 2025, buying hardware to run the most capable models as fast as the way commercial providers do would rise to magnitudes of €6,000,000 as one-off cost, with monthly costs of around €50,000 for power consumption. The combination of high costs and high computational power needs means that these models are usually accessed only remotely as a service while being run from the largest data centres, of which only very few are located within the EU (**SaaS, or 'Software-as-a-Service'**).

[7] See also in the technical glossary

However, recently smaller LLMs[8] have become increasingly more powerful. Moreover, the end devices have also become increasingly more equipped with chips that make the inference costs lower. Therefore, it is now possibility for a lawyer to deploy and operate AI systems with large models in more accessible, affordable and controllable environments. Such options enable a lawyer to use large models in a way that allows them to control their operations and the processing of data. This, however, comes with more responsibility on the user's side in terms of ensuring proper IT security, both physical and logical.

The considerations above, i.e. costs, computational power, the required expertise and responsibility, can be presented as a spectrum of available options. This guide identifies **four major categories** which are key for understanding how lawyers can use AI models:

1. **On-premises (self-hosted) devices**
2. **Self-hosted in colocation/private data centre devices**
3. **Bring-your-own model on infrastructure-as-a-service (IaaS), virtual machines or similar cloud infrastructure use**
4. **Fully managed SaaS/API (vendor-hosted)**

1. On-premises (self-hosted) devices

An on-premise AI model runs entirely on hardware owned and controlled by a user's organisation, rather than in the cloud or on a vendor's remote servers. The user is granted maximum control over their data flows, logging, data retention, with maximum responsibility for the security and uptime of the devices. All data stays within user's infrastructure which means that nothing is sent to an external cloud provider. The user's organisation is responsible for security, updates, maintenance and compliance. Users have maximum control over how the model runs and how data is handled.

1.1 Advantages

On-premise deployment is often preferred when the users deal with confidential or sensitive data. **For lawyers, this is the most secure option regarding the respect for confidentiality.** The data is stored, processed and reviewed at the lawyer's premises, and remains within the control of the lawyer. There is no transfer of data to a third person. This option may come with additional protections (depending on the local laws): a lawyer's office might be granted extra protection against police search warrants (e.g. a special approval is needed from court, a public prosecutor or a Bar representative has to be involved). Such a solution also allows for customisation and full auditability.

1.2 Disadvantages

The trade-off is cost: running large models locally requires specialised hardware, significant power consumption and technical expertise. If the model is not connected to the internet, the primary cybersecurity risk is limited to the potential data loss. If it is connected, the system may be exposed to additional cybersecurity threats. If the user mitigates this additional risk, they can still remain confident in having retained full control over their data.

[8] Some of them are so small (below the so-called 1 billion parameter size) that they are not even presumed under the EU AI Act to be "general purpose AI models" – so they should probably not even be called "large" language models. Nevertheless, these models can still be surprisingly capable, e.g. Qwen 2.5 0.5B etc. – even if they are not practically usable unless further trained or finetuned for a specific task, like classification.

2. Self-hosted (own boxes) in colocation/private data centre

A self-hosted colocation or private data-centre AI model runs on servers that a user owns or controls, but which are physically hosted outside of their premises (for example, in a colocation facility which is a professional data-centre where users rent physical space, power, cooling, and connectivity or user's private data centre). In this scenario, the user maintains full control over the AI model, the servers it runs on, the data flowing through the system and the security, access policies, and auditability.

2.1 Advantages

With this option, the lawyer is still in ownership of the full hardware and has nearly the same control profile as with the on-premises solution.

2.2 Disadvantages

The colocation provider would ensure the availability of the facilities and the physical security, sometimes with rentable "remote hands" (to carry out update needing physical intervention etc.). From the lawyer's perspective, this means the provider and the location must be trusted and chosen with according to national deontological rules governing the use of external IT providers. Such use leads to a new possible attack vector, i.e. physical tampering with the devices which can also enable access to logs or sensitive data. Also, the national laws that would afford additional protection for law firms in case of search warrants may not apply in this case as the data is not physically stored in lawyer's premises.

3. Bring-your-own model on infrastructure-as-a-service (IaaS), virtual machines or similar cloud infrastructure use

Under this scenario, a user brings their AI model, and the cloud provider supplies the computing infrastructure. This is different from SaaS because the cloud provider does not supply the model but only the hardware layer. A user would use equipment owned by third parties, e.g. a virtual machine that is running on a physical machine owned by a third party. The cloud or the virtual/rented machine provider handles hardware and baseline security under a shared responsibility model.

3.1 Advantages

This option gives users advantages similar to running their own servers, but without the cost of physical data-centre hardware. Under this model, the user can still retain full control over the data processing and how the model operates. The user can – in most cases – also decide on the location of the data centre where the virtual machine is running (or at least, the geographical region), and most importantly, when and where the data itself can travel, with encryption and full custody of the encryption keys, etc. Because the user is operating the AI model on the remote servers, they will remain in control over what AI model and model version they are using.

An important advantage of this option is independence as user avoids lock-in to a single AI vendor's ecosystem. The user can also tailor the model to their preferred use cases (e.g. lawyers can fine-tune the model on their internal materials and in areas in which they work). Also, the cloud GPUs can be scaled up or down as needed. Finally, the price of such virtual environments is usually a lot more attractive than on-premises or private cloud solutions (option 1 and 2 above, respectively).

3.2 Disadvantages

This option usually does not allow the user to have a full audit right as often such rights, as specified by providers, mean that users will be shown relevant certificates upon request. From the lawyer's perspective, it is necessary to consider the national professional rules on the use of external IT providers. These may include the obligation for providers to inform the lawyer immediately in the case of search and seizure measures. This is an obligation that small and medium providers may provide or be willing to negotiate, but large providers are often not willing to or do restrict partially. It is worth mentioning that there also exist relevant laws regulating the notification of investigatory measures to those who are subject to them.[9]

4. Fully managed SaaS/API (vendor-hosted)

Under this option, a user can access the model via a web-based user interface or via an **application programming interface (API)**. The vendor controls the entire AI system and runs the AI model, the servers and storage, all updates, safety filters, guardrails, as well as scaling, performance and availability.

4.1 Advantages

The advantages of this option are the ease of use as users are able to immediately access the service through a user interface (UI) or API. The users are not responsible for the infrastructure. This is the easiest, and sometimes the only, option for users to access the best and most powerful frontier models. It is also cost-effective, especially compared to a on-premise or IaaS solution.

4.2 Disadvantages

This option comes with disadvantages, especially from the point of view of lawyers. Users cannot influence how the model is configured internally, how often the provider updates or replaces the model or what safety filters or limitations are applied. The data leaves user's environment which, in case of lawyers, can raise confidentiality issues. Also, users may not be able to conduct full audits as vendors usually only show relevant certificates. Finally, the users face vendor lock-in risks and limited transparency. This architecture means having minimal control over how the internals work, because this is a fully shared service that is geared towards serving the highest number of customers at the lowest possible cost. Contractual obligations that would grant the lawyer further control or audit rights are in those cases often non-negotiable in these cases, as those providers are often hyperscalers who can and/or will not provide special treatment to individual customers or even groups of customers. **From the lawyer's perspective, it is thus crucial to understand the details of the vendor contract and their implications for the lawyer's work.** In this context, there are many questions that lawyers must consider:

- Do the providers describe in their contractual terms how client data will be processed?
- What is the role of the provider in the data processing flow? If they assume the role of a processor, do they provide a data processing agreement and are the provisions thereof compliant with the GDPR and other applicable regulations?
- Will the providers train their models via uploaded data? (which means the lawyer's instructions and responses, via the context and chat history)
- Will the data be accessible to the providers' personnel and if so, under what conditions?
- Will the data be made accessible to third parties, including to the law enforcement authorities (based on warrants or production orders)? If yes, for how long?

[9] For example, Article 13 of the Regulation (EU) 2023/1543 of the European Parliament and of the Council of 12 July 2023 on European Production Orders and European Preservation Orders for electronic evidence in criminal proceedings and for the execution of custodial sentences following criminal proceedings, OJ L 191, 28.7.2023, available at: http://data.europa.eu/eli/reg/2023/1543/oj

- Will the lawyer be informed by the provider before or at least immediately after disclosure or transfer of data to third parties, including to the law enforcement authorities?
- Do the providers retain the data after the user deletes it, for example, in their backups for disaster recovery purposes? For how long?
- What happens if the provider is exposed to a cyber-attack, and they are no longer able to comply with the contractual commitments agreed with the user?
- What is the physical location of the provider's data centre? What laws will govern the processing of data and cooperation with law enforcement authorities? Will the data processed be subject to different, even competing jurisdictions?
- Also, if the model is trained on the lawyer's data or instructions, the lawyers cannot know what parts of their input will reappear for a different user and how will that data appear for them.
- What are the acceptable use policy restrictions (which might be crucial for criminal or human rights cases)?

In this context, it is crucial to explain how the data input by the user to the LLM-based AI system is processed. When a prompt is submitted to a large language model, it is first tokenised and converted into numerical representations, processed within the model's context window, and the next-token predictions are generated that form the response. This processing is typically transient, meaning the input tokens are held in working memory only for the duration technically necessary to compute the output and is not automatically stored in a retrievable database or incorporated into the model's parameters.

Whether a prompt is later used for model training depends not on the model architecture itself but on the provider's contractual terms, technical configuration, and governance policies. Enterprise or API services often contractually exclude training use, whereas consumer-facing services may reserve the right to use inputs for service improvement unless the user opts out.

Even where prompts are used for training, this does not mean the exact text becomes accessible to others; training involves weight adjustments across billions of parameters rather than just storing training corpus, so the model is supposed to learn abstract patterns rather than retaining a searchable copy of the prompt.

However, while verbatim memorisation (of prompts) is not the design goal, residual risks (e.g. logging, human review, security breaches, or rare memorisation phenomena) arise at the AI system level.

If a matter is sensitive, a lawyer may need to use a more controlled model because professional secrecy, privilege, regulatory duties, and client confidentiality all depend on who can access and copy the data. The less we control the **stack**, the more we must ensure the control by contractual obligations to the provider and rely on promises from a vendor. The lawyer should verify not only the provisions of the contact, but also the likelihood of effectively enforcing the contract.

This specific infrastructure has to be kept in mind, because it means that the data that is entered into the user interface of the AI system has to be sent to the AI model. If the AI system provider and the AI model provider are not identical, this means also transfer of data to a third person, possibly situated in a third country.

For data at rest, that is, data being processed by the model running in a specific environment, the provider may unilaterally change its general terms and conditions, or even just the default settings. A vendor can be acquired by an unreliable party or go insolvent. A provider may also promise deletion or non-disclosure of data but they can still be compelled by mandatory legal provisions to retain or hand over data, sometimes under secrecy restrictions that prohibit notifying the user. Cross-border data flows can amplify this risk,

because the data will fall under multiple jurisdictions. Unfortunately, severe security breaches can affect even the largest provider, exposing one's data to malicious third parties.[10]

For data in transit, traffic to a managed service can traverse networks the lawyer has no control over. While a provider may promise 'end-to-end encryption' often, neither parties will be able to detect or ensure at what point a given communication is actually decrypted and who may have access to the data beyond that point. The encrypted message or data stream may still terminate at nodes that are outside that party's control, often there is no user control over it, with many third-party subcontractors involved in logging and monitoring. In such environments, it is almost impossible to acquire information regarding any lawful interception and encryption backdoor capabilities present.

Table 1 – Summary of the features, advantages and disadvantages of different deployment models

Deployment model	Who provides the model	Where the model runs	Who controls infrastructure	Data residency and confidentiality	Control level	Advantages	Disadvantages
Own boxes on-Premise (Self hosted)	User's organisation	Local/internal servers	User's organisation	Data held internally	Very High	Maximum confidentiality	High cost, maintenance
Own Boxes in Colocation (Self-Hosted)	User's organisation	Colocation/private data centre	Shared (facility infrastructure vs organisation's servers)	Data on user's organisation hardware	High	Confidentiality, auditability	Maintenance, cost
Bring-your-own-model on IaaS	User's organisation	Cloud virtual machines (VMs)	Cloud provider	Region-selectable, encrypted	Med-High	Flexible, scalable	External infrastructure, expertise needed
Fully Managed SaaS/API	Vendor	Vendor servers	Vendor	Data sent externally	Low	Easy, low cost, access to frontier models	Low control, confidentiality risks

[10] Such as the Microsoft SharePoint ToolShell zero-day exploit from 2025 (CVE-2025-53770) or the 2024 Snowflake Inc. breach.

Technical considerations for options outside of the SaaS

This section of the guide seeks to explain which open-weight (free) models are currently available, what they can do, and what type of hardware is required to run them, including an indicative cost of purchasing such hardware. It is intended primarily for readers who wish to explore alternatives to SaaS solutions, where the underlying hardware remains entirely outside the user's control. For those considering an IaaS model, the information provided is relevant mainly for determining what level of virtual infrastructure (for example, vCPUs, RAM, and GPUs) would be sufficient for their needs.

For informational purposes, the guide provides some price estimates below that were valid as of September 2025 and do not include any installation, deployment or labour costs. It would be more difficult to understand and compare these solutions if we omit all such figures. It is important to highlight that market trends constantly and considerably affect these prices. Unfortunately, also hardware prices recently are highly volatile, for example prices for Random Access Memory (RAM) have doubled during 2025, also resulting in a significant increase in the costs of local AI inference capabilities during the last months of the year.[11]

To guide the reader through the second, more technical, part, this paper will give a general description of each of the relevant parameters. Lawyers, as end-users, typically engage AI models during the **inference phase** (generating outputs from trained models) rather than further training them, which is far more resource intensive. Inference involves using the model's trained **weights**, processing input data (such as prompts, images, or texts) through those weights, and retrieving the resulting output.

Most of the models used every day are rather large: while the smaller ones may just take 1 GB of storage space, the largest ones (freely accessible ones) can take more than 2 TBs.

To be able to run a model, one will need a computer that can keep the full model in its memory. Furthermore, to be able to run these at speeds that are acceptable for professional use, one will also need specialised, and rather expensive, hardware.

That is why for the larger models, one cannot rely on the normal central processing unit (**CPU**) of one's computer but one needs to buy graphics processing units (**GPUs**). Highspeed memory for CPUs and GPUs is very costly and requires substantial capital investment, and the associated operations are highly energy intensive, which in turn is driving numerous datacentre construction projects worldwide.

Conversely, advances in state-of-the-art commercial models have also had a positive impact on the capabilities of smaller models, which can now be operated by budget-constrained small enterprises or even private individuals on local hardware or on an infrastructure provider of their choice, under their full and exclusive control. There may therefore be a 'sweet spot' in which a smaller model performs a given task comparably to the commercial frontier systems, while remaining relatively inexpensive to operate. The resource requirements of an AI model, however, depend on **numerous factors**, including the following:

1. Model capabilities

There are now various conversational chatbot models that can answer user queries, but the out-of-the-box performance of smaller models remains significantly below that of state-of-the-art, so-called frontier LLMs

[11] This considerable price increase is due to a shortage of memory and to the reservation of memory-manufacturing capacity from the world's top producers years in advance to build the so-called "AI gigafactories".

such as Claude Sonnet 4.5 or GPT-5.1. Nonetheless, even these smaller systems can be fine-tuned for specific linguistic tasks or used for basic classification of sentences and similar routine operations. By contrast, somewhat larger LLMs underlie so-called reasoning models, which are designed to generate more considered, stepwise outputs for end-users, typically at the cost of slower response times.

A different type of model, known as an embedding model, converts texts (such as documents, reports, or legislation) into a numerical form that can be searched using semantic similarity rather than only exact keywords or regular expressions. These models are typically not used in isolation, but as components within more complex information-retrieval systems, such as search engines like ElasticSearch. Another possible use is turning recorded audio into transcriptions, called automatic speech recognition (ASR). These are single purpose models but often integrated with LLMs.

Another type of model worth mentioning are visual language models (VLMs) which, unlike traditional text-only LLMs, can generate textual descriptions from images to enable image search and filtering. They have also significantly improved performance in classical **optical character recognition (OCR)**, reliably extracting legible handwriting, mathematical formulae, and similar content.

The capabilities one expects from a model heavily define the type of hardware one will need to operate that model.[12]

Lawyers have to be able to estimate what tasks could be done locally and at approximately what costs, and what are the functionalities and use cases where a professionally managed service is still a better option. But to be able to estimate this, lawyers have to be aware of the capabilities and some of the technical details behind local models. It is a bit similar to the 1980s, where having a personal computer by many enthusiastic people was essential for the diffusion of computers into all segments of life. Local AI models enable lawyers (and people in general) to have this first-hand experience and decrease their dependency on large corporations, even if their use is less comfortable or powerful.

2. Size of the model

The second key consideration is understanding the size of models compatible with a given hardware setup. Model sizes are typically expressed in terms of their parameter count (from millions to trillions) where an 8B model has 8 billion parameters (i.e., 8 billion distinct 'weights and biases'). For models within the same family or from the same provider, size generally correlates with both capability and hardware demands. For example, DeepSeek-r1:14b (a 14-billion-parameter DeepSeek model) outperforms DeepSeek-r1:8b (8 billion parameters).

However, the above does not always hold due to various factors. For instance, Google's newer Gemma 3 4B model outperforms the older, larger Gemma 2 27B model on many benchmarks. Similarly, different publishers of models may offer superior performance within the same parameter range, owing to advances in architecture, training corpora, or other techniques. For this reason, practitioners typically rely on comparable capability evaluations published as benchmarks on public leaderboards.[13]

---

[12] One cand find a very rich repository of downloadable, free-to-use models at HuggingFace: https://huggingface.co/models

[13] https://huggingface.co/spaces/DontPlanToEnd/UGI-Leaderboard or https://artificialanalysis.ai/leaderboards/models

https://huggingface.co/spaces/hf-audio/open_asr_leaderboard

https://huggingface.co/spaces/TIGER-Lab/MMLU-Pro

https://huggingface.co/spaces/mteb/leaderboard

In terms of hardware requirements, a VLM of 8B parameters or an image generation Stable Diffusion 8B model will not run at acceptable speeds without a machine with a large enough GPU, while an LLM of 8B can provide its answer at an acceptable speed in conversations using CPU-only. This means that parameter sizes are not comparable across different types of models.

Also, one can often find the same model with different **quantisations**. These are a simplification of the trained model weights to speed up inference and decrease memory size but that may also result in some loss in capability. For example, running OpenAI's free GPT-OSS-20B model's different versions, such as FP16 and with INT4 (the latter requiring almost four times less memory).

Table 2 – Quantisation – rule of thumb regarding its impact on model size and quality

Precision	RAM use	Speed effect	Quality impact	Good for
FP16	1.0×	Baseline	Full fidelity	Highest accuracy, safety-critical outputs
INT8	~0.5×	+10–30%	Close enough to original on most tasks	General chat/RAG; good default for local servers
INT4	~0.25×	+20–50%	Moderate loss	Long context and cost-saving

3. Speed and the number of concurrent users

The speed of model output is usually measured in '**tokens** per second', where token is a conversion of the original text or an image that the AI model can natively understand. By way of illustration, the previous sentence of 173 characters is 35 different tokens (using OpenAI's GPT-4o tokeniser).[14] In this tokenisation method in English, an average 4 characters make up a single token. Importantly, the number of token per character is language dependent: even if the Hungarian translation of the same sentence in Hungarian takes only 171 characters, that will already mean 53 tokens instead of 35 in English.

There are two kinds of token speed measurements, both given in token per second (or **tps**):

- the input processing speed (also called 'prefill'); and
- generation (or decoding) phase.

Different tasks place different demands on a model: chatbots usually take short inputs and generate long outputs, whereas information-retrieval tasks rely on long inputs and short responses. As models grow

[14] The tokens are [976, 7733, 328, 2359, 4733, 382, 6971, 26489, 306, 392, 64329, 777, 3099, 672, 1919, 6602, 382, 261, 22165, 328, 290, 4756, 2201, 503, 448, 3621, 484, 290, 20837, 2359, 665, 297, 11594, 4218, 13]. See https://platform.openai.com/tokenizer

larger, their output speed slows down, and they require increasingly powerful and costly hardware to maintain acceptable performance.[15]

Usually, the generation speed below 5 tps is too slow for any interactive use while the speed above 20 tps outpaces an average lawyer's reading speed. Speed of around 100 tps exceeds typical skim read speed although it may still feel slow if the task involves, for example, searching through a long model output. Importantly, if multiple users are relying on the same local system, its concurrent use will further divide and reduce the effective generation speed.[16]

4. Length of inputs and outputs

The final technical term one must be familiar with is the 'context length' which is also measured in tokens. A typical length for most models is 4,096 tokens, that includes both the prompt and attached files ('input tokens') and the length of the response ('output tokens'). As described above, an average printed page in English can take up 500 tokens, so if a user wants the chatbot to search all the applicable laws and judicial decisions in a country, they would need around 3.5 billion tokens in that hypothetical country. That is simply impossible and, in these cases, a different approach is suggested, like information retrieval (including **RAG - retrieval augmented generation** - tools).

Even if one just wants to search through court documents for a specific case, look through the evidence or witness statements, one can easily reach the size of 100 pages, which could be around an estimated size of 50,000 tokens.

Using an example of a very fast and expensive machine such as the latest Nvidia RTX Pro 6000 (costing now €108000), one can see that it reads input tokens at 10,000 tps and generates an answer at a fast 215 tps where the context length is 2,048 tokens only. At this rate, processing 50,000 tokens will take two minutes for reading the input and the generation speed will most likely be less than 10 tps. So, even for this size, more complex information retrieval pipelines are suggested (chunking the input etc.) A larger context window will also increase the memory requirements.

Table 3 – What is the priority for a use case: length and speed

Task	Input length	Output length	Input speed (prefill priority)	Output speed (decoding priority)
Chat Q&A	Short	Short–Med	●	●●
Legal RAG (long prompt)	Long	Short–Med	●●●	●
Bulk OCR + VLM captioning	Long (images/PDF)	Short	●●	●
Batch summarization (short docs)	Short	Short	●	●
Transcription (ASR)	Long (audio)	Long	●	●●●

[15] You can try out what speed is acceptable for your use case on this simulator: https://kamilstanuch.github.io/LLM-token-generation-simulator/

[16] See a list of typical available speeds depending on hardware here: https://llm.aidatatools.com/results-windows.php.

What models can be run locally?

Based on these explanations, this paper will now examine what can be feasible for a lawyer to operate locally. The paper breaks down the available options in terms of budget-spending: from using existing computers, to spending amounts that are not really practicable for small firms.

The budget option would be to **run a small model on an existing computer** (even if a couple of years old). These can include conversational chatbots, such as deepseek-r1:1.5b, or involve embedding models and information retrieval on a regular Windows computer with as little as 8GB of RAM. A 16GB machine can already run more capable models like a deepseek-r1:14b at the patient speed of 2.5 tps or be used, albeit very slowly, for automated speech recognition (ASR) purposes. The costs involved are minimal save for the time needed to configure the relevant tools. It is worth noting that there already exist accessible 'inference engines', such as Ollama, LMStudio or AnythingLLM, which can help with downloading an appropriate model.

The next tier and budget option is a **dedicated machine to use with local AI models**. Using the September 2025 prices as a benchmark, one should foresee a one-off expense of approximately €2,000 (excluding VAT). This price will include all the components needed for a computer, an inference machine, including motherboard with 128GB of RAM[17] and a fast CPU, a couple of inexpensive GPUs and 24GB VRAM (e.g. two to four GPUs when using an appropriate motherboard). This makes it possible to run 20-40B parameter text-only models at a comfortable speed.[18] At this price category, one will not want to run models without a GPU any longer.[19]

The next and more costly option is to buy a more expensive GPU, such as Nvidia's RTX Pro 6000 with 96GB of VRAM (approximately €8,000). It can be used with the existing computers as long as they have sufficient space and power, or with a dedicated local inference machine. With the local inference machine, it is possible to run OpenAI's current best open weight model called GPT-OSS-120B. The aforementioned deepseek-r1:14b can reach a speed of 114 tps using this GPU (at limited token length).

Beyond these options, one would need to consider buying a server or workstation-specific configurations. This is because consumer-grade motherboards rarely have enough bandwidth for more than one GPU (except for some very limited use cases such as the €2,000 **motherboard** example above). Since these motherboards allow up to 1 or 2 TB of memory, they can be fitted with up to four very expensive GPUs.

A budget of €20,000 would allow one to run some of the most capable open weight models (like an 8-bit quantized 671B DeepSeek V3 or a Qwen3-235B-A22B, even if slowly)[20] or share a well-rounded GPT-OSS-120B simultaneously with several of concurrent users, using larger context windows. The important point

[17] Please note the introductory remarks regarding the extreme volatility of RAM prices.

[18] Most consumer motherboards can only house one full-speed GPU, but there are some that can mechanically receive even four at the same time (having four mechanical PCI Express x16 slot, like a Gigabyte B650 EAGLE AX), even if the last three GPUs will not be able to use the full speed to the processor (running in a so-called x1 mode instead). Nevertheless, this still makes it possible to provide improved speed for such a low budget, and is popular among enthusiasts. However, this is not practically usable for VLM use cases.

[19] Please note this is a strong simplification. Once one has a large enough RAM (VRAM) to host a model, the next crucial question is memory bandwidth. While server and workstation grade motherboards can provide fast bandwidth, they tend to be a lot more expensive than consumer grade components, so GPUs provide a lot better value and speed up to a certain memory size. However, one can insert 4 or more GPUs only in very few and expensive motherboards, and these also require expensive switching solutions. Moreover, only very expensive GPUs contain more than 32 GB VRAM (like a 96 GB version for €8000 each). Considering that the largest open weight models need 1-2 TB of memory, these models would not fit into budgets smaller than €150,000 when using GPU only inference. At the same time, one can run these on €20,000 CPU-only machines or partly on relatively inexpensive GPUs, albeit slowly (5-8 tps).

[20] However, the currently available largest open-weight model, Kimi K2 Thinking of 1000B parameters will not fit into such budget.

is that the language understanding capabilities of these models are said to exceed even that of GPT-4o from 2024 and these can be run easily.[21]

However, machines beyond the €20,000 mark are usually specific and dedicated appliances, such as Nvidia's DGX H100 (around €350,000 each) or the GB300 NVL72 (costing up to €3 million). The latter will require specialised power and cooling capacities that would make it difficult to install them in a neighbouring data centre (even if one could afford the hardware and the ongoing electricity costs, which can reach tens of thousands of euros per month). If one genuinely needs this level of capability, building it within one's own office would no longer be a practical option.

Table 4 – Sample models suitable for certain local AI use cases

Local AI Tasks	**Model type**	**Minimum specifications to be practical**
Drafting and revision (suitable for simpler changes in supported languages)	LLM 7B–8B	CPU sufficient
Long-context RAG (200–500 pp)	LLM 13B–34B + embeddings	GPU ≥16–24 GB or fast CPU w/ INT4
PDF OCR + captions	VLM 8B–14B + OCR	GPU ≥12 GB
Transcription (ASR)	ASR base-large (<1.5B)	CPU sufficient
Batch summarisation (100 docs)	LLM 7B–13B	GPU ≥12 GB
Clause extraction / QA over contracts	LLM 7B–13B + embeddings	CPU sufficient (GPU preferred)

[21] https://huggingface.co/spaces/TIGER-Lab/MMLU-Pro.

Future developments

AI technology is advancing at a rapid pace, and new developments are emerging continuously. One such field is agentic AI which are autonomous systems capable of executing multi-step tasks, accessing external data sources, and taking actions on behalf of users with limited or no human intervention. While such technologies are not yet mature enough for the CCBE to assess their specific impact on legal services and professional obligations, their increasing accessibility means that lawyers will inevitably encounter them, whether in their own practice or in the cases they handle. The general principles outlined in this guide, such as the considerations relating to the protection of confidentiality, data control, and the risks inherent in cloud-based and third-party services, provide a relevant framework for evaluating these and other emerging AI technologies. The CCBE will continue to monitor these developments and intends to address them in future updates to this guide.

Technical glossary

AI model	An AI model is a programme that has been trained on a set of data to recognise certain patterns or make certain decisions without further human intervention. Artificial intelligence models apply different algorithms to relevant data inputs to achieve the tasks, or output, they have been programmed for. More: 'What is an AI model?' IBM
AI system	AI system is a machine-based system that can, for explicit or implicit objectives, infer from input data how to generate outputs such as predictions, content, recommendations, or decisions.
Application Programming Interface (API)	A set of programmable commands that let external software (such as a legal-tech platform) interact with an AI model hosted elsewhere.
Automated Speech Recognition (ASR)	Automatic Speech Recognition (ASR) is a technology that converts spoken language into text, enabling computers to understand and process human speech.
Attention mechanism	The attention mechanism lets the model decide which words in the input are important and how strongly each word relates to every other word. This is why these models can generate coherent text and follow complex instructions.
Central Processing Unit (CPU)	A general-purpose, central processor on a computer which executes the operating system and defines the sequential logic of programmes run. It manages the resources of the computer, coordinates data between storage, cache, memory, GPUs (tells the other controllers and processors integrated on the computer what to do).
Embedding	Embeddings in AI are numerical representations (vectors) of data, such as text, images, or audio. They capture their semantic meaning and relationships and transform high-dimensional, complex information into dense, lower-dimensional, continuous vector spaces, allowing machine learning models to identify patterns and similarities.
Fine-tuning	Adjusting a pre-trained model on a narrower, domain-specific dataset (e.g. legal contracts) so it produces more relevant outputs.
Graphics Processing Unit (GPU)	Graphics Processing Unit (GPU) is specialised hardware used for parallel data processing. It was originally developed for fast, improved computer graphics (very popularly for gaming). But their programmability allows them to accelerate not just graphics, but other data-heavy workloads like mining cryptocurrencies, and now, training and inference of large neural networks.
Infrastructure as a Service (IaaS)	Infrastructure as a Service (IaaS) is a cloud computing model providing on-demand, virtualized IT resources (such as servers, storage, and networking) over the internet on a pay-as-you-go basis. It removes the need for organisations to manage physical data centres, allowing them to rent infrastructure from providers.
Inference (phase)	AI inference is the crucial 'doing' phase where a trained artificial intelligence model uses its learned knowledge to analyse new, unseen data and generate real-world outputs like predictions, classifications, or decisions, essentially applying its skills to solve problems, such as recognising faces, recommending movies, or powering self-driving cars. It's the step after training (learning) that brings AI to life, turning complex patterns into actionable results.

Large Language Model (LLM)	A Large Language Model (LLM) is a type of AI trained on massive datasets using deep learning, specifically transformer architectures, to understand and generate human-like text, translate languages, and perform natural language processing tasks.
Motherboard	The printed circuit board of a computer that acts as the central physical and electrical backbone of the system.
Natural Language Processing (NLP)	Natural Language Processing (NLP) is a branch of artificial intelligence (AI) that enables computers to understand, interpret, generate, and manipulate human language (text and speech).
Open weight models	Open-weight models mean that the learned parameters (often billions of numbers, stored in large files) are publicly available, so others can run or fine-tune the model themselves. (see also: weights)
Optical Character Recognition (OCR)	Optical Character Recognition (OCR) is a technology that enables computers to identify and convert text contained in images into machine-readable, editable text. In practice, OCR takes inputs such as scanned documents, photographs of documents or images containing typed or handwritten text and produces outputs in the firm of a structured, searchable text.
Parameter	A numeric weight inside the model that influences how it transforms inputs into outputs. Modern LLMs have billions of parameters which drive their capability.
Prompt	The input text (or other instruction) given to an LLM to steer its output. Prompts can be a single question, a set of instructions, or a few example interactions.
Prompt Engineering	Crafting prompts deliberately to obtain desired, reliable, and legally sound outputs. Often involves adding constraints, examples, or 'guardrails.'
Quantisation	Quantisation is a technique used in AI models to reduce the size of the model and the amount of memory it needs, by storing the model's numerical weights with *lower precision*. Instead of using full-precision numbers (such as FP16 or FP32), a quantised model uses reduced-precision formats such as INT8 or INT4. This makes the model smaller, faster, cheaper to run, usable on less powerful hardware, but sometimes with a slight loss in accuracy.
RAM (Random Access Memory)	The fast storage on the motherboard. RAM is volatile, meaning that is loses its content when power is turned off. It stores the operating system, application programmes and data that are currently in use by the CPU.
Retrieval-Augmented Generation (RAG)	Combining a language model with a searchable database of documents (e.g. statutes) so the model can reference actual source material in its output.
SaaS	Software as a Service (SaaS) is a cloud-based software delivery model where applications are hosted by a provider and accessed by users over the internet, typically via a subscription. Instead of purchasing and installing software locally, users pay-as-you-go, with providers managing all infrastructure, security, and updates
Stack	(or AI tech stack) in the context of AI refers to the layered collection of technologies, frameworks, infrastructure, and software tools that work together to build, train, deploy, and manage AI applications.
Token	AI tokens are the fundamental, smallest units of data, such as words, parts of words, or characters, that Large Language Models (LLMs) like ChatGPT use to process and generate text. They act as the 'building blocks' or language, allowing AI to analyse input, understand context, and predict the next logical piece of information to form a response.

Tokenisation	within Natural Language Processing (NLP) and Large Language Models (LLMs), tokenisation is the fundamental process of breaking down raw input text into smaller, manageable units called tokens.
tps	Token per second
Transformers / Transformer Models	Transformers are a modern breakthrough in AI. Instead of reading text one word at a time like earlier systems, transformers look at all the words in a sentence at once. This helps them understand context much more effectively, such as which words relate to each other, even across long sentences.
Video RAM / VRAM	Video RAM used in the GPUs (as opposed to RAM used on the motherboard for CPUs). (see also: RAM)
Weights	In an AI model, the trained weights are the numerical parameters that encode everything the model has learned from its training data. A modern AI model (such as a large language model) is a huge network of interconnected 'neurons.' Each connection between neurons has an associated weight: a number that tells the model how strongly one unit should influence another. During training, the model repeatedly adjusts these weights to reduce its prediction errors on the training data. It does so through: ▪ gradient descent which is a fundamental first-order optimisation algorithm used in machine learning to train models by iteratively minimising the error (loss function); or ▪ backpropagation (backward propagation of errors) which is the fundamental, iterative algorithm for training artificial neural networks by calculating gradients of a loss function with respect to the network's weights. After training, the final set of weight values determines how the model transforms any given input into an output. In practice, downloading or loading a model means downloading its trained weights. Loading those weights into the model architecture is what allows the system to generate text, classify documents, or perform any other learned task. (see also: open-weight models)