AGID | Agenzia per l'Italia Digitale

Bozza di Linee Guida per lo sviluppo di sistemi di Intelligenza Artificiale nella pubblica amministrazione

Ai sensi del D.P.C.M. 12 gennaio 2024, recante "Piano triennale per l'informatica nella pubblica amministrazione 2024-2026"

Sommario

1. Ambito di applicazione

1.1 Ambito soggettivo

Le presenti Linee guida per lo Sviluppo di Intelligenza Artificiale nella Pubblica Amministrazione (di seguito Linee guida) sono rivolte ai soggetti di cui all'articolo 2, comma 2, del CAD. Tali soggetti sono indicati nel documento con l'acronimo PA.

1.2 Ambito oggettivo

Le presenti Linee guida concernono le modalità di sviluppo dei sistemi di Intelligenza Artificiale con particolare riferimento agli aspetti di conformità normativa e di impatto organizzativo.

Esse si applicano a tutte le componenti di applicazioni e infrastrutture tecnologiche che impiegano tecnologie di Intelligenza Artificiale, sia come componente integrata sia come supporto alle funzionalità principali.

1.3 Riferimenti e sigle

1.3.1 Note di lettura del documento

Le presenti Linee guida adottano i termini chiave: «DEVE», «DEVONO», «NON DEVE», «NON DEVONO», «DOVREBBE», «NON DOVREBBE», «PUÒ», «POSSONO» e «OPZIONALE», definiti dalle norme ISO/IEC Directives, Part 2 "Principles and rules for the structure and drafting of ISO and IEC documents" per la stesura dei documenti tecnici. L'interpretazione di tali termini nell'ambito delle Linee guida è descritta di seguito.

DEVE o DEVONO indicano un requisito obbligatorio;

NON DEVE o NON DEVONO o NON PUÒ o NON POSSONO indicano un assoluto divieto;

DOVREBBE o NON DOVREBBE indicano che le implicazioni devono essere comprese e attentamente pesate prima di scegliere approcci alternativi;

PUÒ o POSSONO o l'aggettivo OPZIONALE indica che il lettore può scegliere di applicare o meno la specifica.

Le presenti Linee guida fanno riferimento a concetti e principi contenuti in norme tecniche definite a livello nazionale, europeo e internazionale. Tali norme non sono vincolanti, salvo diversa indicazione espressamente riportata nel testo.

1.3.2 Struttura del documento

Considerata la velocità dell'innovazione, le Linee guida devono garantire un adattamento costante ai cambiamenti imposti dall'incessante rivoluzione digitale. Di qui la scelta di corredare le Linee guida di "Strumenti", ovvero documenti a supporto dell'attività operativa di applicazione delle Linee guida.

L'elenco aggiornato degli strumenti delle Linee guida IA è disponibile al link: <<da definire in fase di pubblicazione>>.

Gli Strumenti potranno essere periodicamente aggiornati per tener conto dell'evoluzione normativa e tecnologica e delle buone pratiche emergenti.

1.3.3 Riferimenti normativi

Sono riportati di seguito i principali atti normativi di riferimento per le presenti Linee guida.

[CDFUE] Carta dei diritti fondamentali dell'Unione Europea 2012/C 326/02

[AI Act] Regolamento europeo (UE) 1689/2024 del 13 giugno 2024 del Parlamento Europeo e del Consiglio, che stabilisce regole armonizzate sull'Intelligenza Artificiale e modifica i regolamenti (CE) n. 300/2008, (UE) n. 167/2013, (UE) n. 168/2013, (UE) 2018/858, (UE) 2018/1139 e (UE) 2019/2144 e le direttive 2014/90/UE, (UE) 2016/797 e (UE) 2020/1828 (regolamento sull'Intelligenza Artificiale).

[Data Act] Regolamento europeo (UE) 2023/2854 del 13 dicembre 2023 del Parlamento Europeo e del Consiglio, riguardante norme armonizzate sull'accesso equo ai dati e sul loro utilizzo e che modifica il regolamento (UE) 2017/2394 e la direttiva (UE) 2020/1828 (regolamento sui dati).

[DGA] Regolamento europeo (UE) 2022/868 del 30 maggio 2022 del Parlamento Europeo e del Consiglio, relativo alla governance europea dei dati e che modifica il regolamento (UE) 2018/1724 (Regolamento sulla governance dei dati).

[CRA] Regolamento (UE) 2024/2847 del 23 ottobre 2024 del Parlamento europeo e del Consiglio, relativo a requisiti orizzontali di cibersicurezza per i prodotti con elementi digitali e che modifica i regolamenti (UE) n. 168/2013 e (UE) 2019/1020 e la direttiva (UE) 2020/1828 (Cyber Resilience Act).

[Reg. UE 2018/1725] Regolamento (UE) 2018/1725 del Parlamento europeo e del Consiglio, del 23 ottobre 2018, sulla tutela delle persone fisiche in relazione al trattamento dei dati personali da parte delle istituzioni, degli organi e degli organismi dell'Unione e sulla libera circolazione di tali dati, e che abroga il regolamento (CE) n. 45/2001 e la decisione n. 1247/2002/CE.

[GDPR] Regolamento (UE) 2016/679 del 27 aprile 2016 del Parlamento europeo e del Consiglio, relativo alla protezione delle persone fisiche con riguardo al trattamento dei dati personali, nonché alla libera circolazione di tali dati e che abroga la direttiva 95/46/CE (regolamento generale sulla protezione dei dati).

[Reg. UE 2012/1025] Regolamento (UE) n. 2012/1025 del 25 ottobre 2012 del Parlamento europeo e del Consiglio, sulla normazione europea, che modifica le direttive 89/686/CEE e 93/15/CEE del Consiglio nonché le direttive 94/9/CE, 94/25/CE, 95/16/CE, 97/23/CE, 98/34/CE, 2004/22/CE, 2007/23/CE, 2009/23/CE e 2009/105/CE del Parlamento europeo e del Consiglio e che abroga la decisione 87/95/CEE del Consiglio e la decisione n. 1673/2006/CE del Parlamento europeo e del Consiglio.

[Direttiva Open Data] Direttiva (UE) 2019/1024 del Parlamento europeo e del Consiglio, del 20 giugno 2019, relativa all'apertura dei dati e al riutilizzo dell'informazione del settore pubblico (rifusione).

[NIS 2] Direttiva (UE) 2022/2555 del Parlamento europeo e del Consiglio del 14 dicembre 2022, relativa a misure per un livello comune elevato di cybersicurezza nell'Unione, recante modifica del regolamento (UE) n. 910/2014 e della direttiva (UE) 2018/1972 e che abroga la direttiva (UE) 2016/1148.

[CER] Direttiva 2022/2557 del Parlamento europeo e del Consiglio del 14 dicembre 2022 del Parlamento europeo e del Consiglio relativa alla resilienza dei soggetti critici e che abroga la direttiva 2008/114/CE del Consiglio (Critical Entities Resilience).

[Dir. (UE) 2016/680] Direttiva (UE) 2016/680 del 27 aprile 2016 del Parlamento europeo e del Consiglio relativa alla protezione delle persone fisiche con riguardo al trattamento dei dati personali da parte delle autorità competenti a fini di prevenzione, indagine, accertamento e perseguimento di reati o esecuzione di sanzioni penali, nonché alla libera circolazione di tali dati e che abroga la decisione quadro 2008/977/GAI del Consiglio.

[Dir. Prodotti] Direttiva (UE) 2024/2853 del Parlamento europeo e del Consiglio del 23 ottobre 2024 sulla responsabilità per danno da prodotti difettosi, che abroga la direttiva 85/374/CEE del Consiglio.1

[L. 90/2024] Legge 28 giugno 2024, n. 90 recante "Disposizioni in materia di rafforzamento della cybersicurezza nazionale e di reati informatici."

[D.L. 135/2018] Decreto-legge 14 dicembre 2018, n. 135 recante "Disposizioni urgenti in materia di sostegno e semplificazione per le imprese e per la Pubblica Amministrazione", convertito in legge, con modificazioni, dall'art. 1, comma 1 della legge 11 febbraio 2019, n. 12.

[Accessibilità] Legge 9 gennaio 2004, n. 4 e s.m.i. recante "Disposizioni per favorire e semplificare l'accesso degli utenti e, in particolare, delle persone con disabilità agli strumenti informatici".

[Codice privacy] Decreto legislativo 30 giugno 2003, n. 196 e s.m.i. recante "Codice in materia di protezione dei dati personali, recante disposizioni per l'adeguamento dell'ordinamento nazionale al regolamento (UE) n. 2016/679 del Parlamento europeo e del Consiglio, del 27 aprile 2016, relativo

alla protezione delle persone fisiche con riguardo al trattamento dei dati personali, nonché alla libera circolazione di tali dati e che abroga la direttiva 95/46/CE".

[CAD] Decreto legislativo 7 marzo 2005, n. 82 e s.m.i. recante "Codice dell'amministrazione digitale".

[D.Lgs. 36/2006] Decreto legislativo 24 gennaio 2006, n. 36 recante "Attuazione della direttiva (UE) 2019/1024 relativa all'apertura dei dati e al riutilizzo dell'informazione del settore pubblico che ha abrogato la direttiva 2003/98/CE".

[Codice dei contratti] Decreto legislativo 31 marzo 2023, n.36 e s.m.i recante "Codice dei contratti pubblici in attuazione dell'art. 1 della legge 21 giugno 2022, n.78, recante delega al Governo in materia di contratti pubblici".

[D.lgs 200/2021] Decreto legislativo 8 novembre 2021, n. 200 recante "Attuazione della direttiva (UE) 2019/1024 del Parlamento europeo e del Consiglio, del 20 giugno 2019, relativa all'apertura dei dati e al riutilizzo dell'informazione del settore pubblico (rifusione)".

[D.lgs 82/2022] Decreto legislativo 27 maggio 2022, n. 82 recante "Attuazione della direttiva (UE) 2019/882 (CER) del Parlamento europeo e del Consiglio, del 17 aprile 2019, sui requisiti di accessibilità dei prodotti e dei servizi".

[D.Lgs. 134/2024] Decreto legislativo 4 settembre 2024, n. 134. recante "Attuazione della direttiva (UE) 2022/2557 (CER) del Parlamento europeo e del Consiglio, del 14 dicembre 2022, relativa alla resilienza dei soggetti critici e che abroga la direttiva 2008/114/CE del Consiglio".

[D.lgs. 138/2024] Decreto legislativo 4 settembre 2024, n. 138 recante "Recepimento della direttiva (UE) 2022/2555 (NIS2), relativa a misure per un livello comune elevato di cibersicurezza nell'Unione, recante modifica del regolamento (UE) n. 910/2014 e della direttiva (UE) 2018/1972 e che abroga la direttiva (UE) 2016/1148".

[D.P.R. 81/2022] Decreto del Presidente della Repubblica 24 giugno 2022, n. 81. Regolamento recante individuazione degli adempimenti relativi ai piani assorbiti dal piano integrato di attività e organizzazione (PIAO).

[Piano Triennale] D.P.C.M. 12 gennaio 2024, recante "Piano triennale per l'informatica nella pubblica amministrazione 2024-2026"

[PT agg. 2025] D.P.C.M. 3 dicembre 2024 recante "Aggiornamento 2025 del Piano triennale 2024-2026"

[Digital Decade] Dichiarazione europea sui diritti e i principi digitali per il decennio digitale (2023/C 23/01)

[Strategia IA] Strategia Italiana per l'Intelligenza Artificiale 2024-2026.

[L. 132/2025] Legge 23 settembre 2025, n. 132 recante "Disposizioni e deleghe al Governo in materia di intelligenza artificiale"

[AI_ACT_PROHIB] C(2025) 884 Guidelines on prohibited artificial intelligence practices established by Regulation (EU) 2024/1689 (AI Act)

[Direttiva (UE) 2023/1791] DIRETTIVA (UE) 2023/1791 DEL PARLAMENTO EUROPEO E DEL CONSIGLIO del 13 settembre 2023 sull'efficienza energetica e che modifica il regolamento (UE) 2023/955 (rifusione)

[Reg. UE 2024/1364] REGOLAMENTO DELEGATO (UE) 2024/1364 DELLA COMMISSIONE del 14 marzo 2024 sulla prima fase dell'istituzione di un sistema comune di classificazione dell'Unione per i centri dati.

[Reg. UE 2019/424] REGOLAMENTO (UE) 2019/424 DELLA COMMISSIONE del 15 marzo 2019 che stabilisce specifiche per la progettazione ecocompatibile di server e prodotti di archiviazione dati a norma della direttiva 2009/125/CE del Parlamento europeo e del Consiglio e che modifica il regolamento (UE) n. 617/2013.

1.3.4 Linee guida di riferimento

Di seguito sono elencate le Linee guida emesse da AgID ai sensi dell'art. 71 del CAD e altra documentazione regolamentare, che sono richiamate, anche indirettamente, nel presente documento. Le linee guida AgID sono disponibili tramite il sito istituzionale al seguente indirizzo: https://www.agid.gov.it/it/linee-guida, dove sono pubblicati anche i relativi aggiornamenti in conseguenza dell'evoluzione tecnologica o della necessità di adeguamento alla normativa di riferimento.

[LLGG_IA] Linee guida per l'adozione di sistemi di intelligenza artificiale nella Pubblica Amministrazione

[LG_OPENDATA] Linee guida Open Data.

[LG-RIUSO] Linee guida su acquisizione e riuso di software per le Pubbliche Amministrazioni.

[LG_ACCESS] Linee guida sull'accessibilità degli strumenti informatici

[LLGG_DES] Linee guida di design per i siti internet e i servizi digitali della Pubblica Amministrazione

[LG_DOC_INF] Linee guida sulla formazione, gestione e conservazione dei documenti informatici

[LG_PDND_INTER] Linee guida sull'infrastruttura tecnologica della Piattaforma Digitale Nazionale Dati per l'interoperabilità dei sistemi informativi delle basi dati

[LG_SIC_INTER] Linee guida Tecnologie e standard per la sicurezza dell'interoperabilità tramite API dei sistemi informatici

[LG_INTER_TEC] Linee guida sull'interoperabilità tecnica delle Pubbliche Amministrazioni

1.3.5 Acronimi

Le definizioni tecniche e gli acronimi sono contenuti nello Strumento A.

2 Come sviluppare sistemi di Intelligenza Artificiale

2.1 Principi generali per lo sviluppo di sistemi di Intelligenza Artificiale

Le presenti Linee Guida definiscono un insieme di principi per orientare lo sviluppo dei sistemi di Intelligenza Artificiale nella Pubblica Amministrazione.

Tali principi mirano a garantire controllo, sostenibilità, interoperabilità e autonomia operativa, lungo l'intero ciclo di vita dei sistemi di IA.

Lo sviluppo di sistemi di Intelligenza Artificiale nella Pubblica Amministrazione DEVE essere orientato alla realizzazione di capacità e servizi governabili, e non di soluzioni monolitiche o tecnologicamente rigide.

I sistemi di IA DEVONO essere considerati come sistemi complessi e stratificati, progettati tenendo conto dell'intero stack tecnologico dell'IA, dall'infrastruttura energetica ai servizi applicativi.

Lo sviluppo DEVE favorire una separazione chiara e il disaccoppiamento dei componenti, consentendo l'evoluzione o la sostituzione di modelli, infrastrutture e servizi senza impatti sull'intero sistema. In tale prospettiva, le Pubbliche Amministrazioni DEVONO privilegiare architetture agentiche e a servizi, supportate da un orchestratore di servizi di IA, in grado di coordinare modelli, dati e risorse computazionali in modo flessibile e controllabile.

I sistemi di IA DEVONO essere progettati secondo principi di neutralità hardware, evitando dipendenze strutturali da specifici acceleratori computazionali o architetture proprietarie. Le attività di deployment ed erogazione dei servizi devono tenere conto, inoltre, di fattori non prevedibili, quali shortage, ovvero un disequilibrio del mercato tra domanda e offerta che genera ritardi, rincari e instabilità nelle catene di fornitura, o modifiche nelle dinamiche di pricing; pertanto, in queste architetture la Pubblica Amministrazione DEVE prevedere attività di rollback e fallback.

Infine, lo sviluppo DEVE essere coerente con i principi di efficienza, sostenibilità, continuità del servizio e autonomia operativa, garantendo che i sistemi di IA possano essere gestiti, adattati ed evoluti nel tempo dalla Pubblica Amministrazione.

2.2 Correlazione tra i principi per l'Adozione, lo Sviluppo e il Procurement di sistemi di IA nella Pubblica Amministrazione

Il capitolo fornisce una declinazione operativa dei principi di adozione dell'Intelligenza Artificiale nelle Pubbliche Amministrazioni, enunciati nelle Linee Guida per l'Adozione dell'Intelligenza Artificiale nella PA par. 3.4, mettendo in relazione ciascun principio con le azioni da adottare nelle fasi di sviluppo e di procurement dei sistemi di IA. Tale raccordo consente di tradurre i principi generali in requisiti tecnici, architetturali e contrattuali concreti, coerenti con un approccio basato sul rischio, con il ciclo di vita dei sistemi di IA e con i principi architetturali delineati nella presente Linea Guida.

In totale, sono individuati 20 principi per l'adozione, lo sviluppo e il procurement di sistemi di IA. Essi sono descritti a seguire e analogamente sintetizzati nella Tabella 1.

Principio 1 – Conformità normativa

Secondo il principio di conformità normativa, le PA adottano i sistemi di IA nel pieno rispetto delle normative nazionali e dell'Unione Europea, assicurando l'aderenza al quadro legislativo vigente. Inoltre, le PA monitorano l'evoluzione normativa e aggiornano costantemente i propri sistemi per assicurarne la conformità alle disposizioni vigenti.

In fase di sviluppo, i sistemi di IA DEVONO essere progettati secondo il principio di compliance by design, includendo requisiti normativi (AI Act, GDPR, New Legislative Framework - NLF) lungo l'intero ciclo di vita. Lo sviluppo DEVE inoltre privilegiare architetture modulari e agentiche, che consentano l'aggiornamento selettivo dei componenti (modelli, servizi, orchestrazione) senza riprogettare l'intero sistema.

In fase di procurement, i contratti DEVONO prevedere obblighi di conformità normativa evolutiva e clausole di adeguamento in caso di modifiche legislative. Il procurement DEVE inoltre evitare soluzioni che rendano tecnicamente o contrattualmente complesso l'adeguamento normativo nel tempo.

Principio 2 – Rispetto dei valori fondamentali dell'UE

Secondo il principio di rispetto dei valori fondamentali dell'UE, le PA adottano i sistemi di IA garantendo i principi definiti dalla Carta dei diritti fondamentali dell'Unione Europea (cfr. Art. 1 e 2 AI Act).

Lo sviluppo DEVE integrare valutazioni d'impatto sui diritti fondamentali e meccanismi di controllo umano, anche attraverso l'uso di agenti con ruoli e responsabilità chiaramente definiti. I modelli DEVONO essere selezionati e orchestrati in modo proporzionato al contesto, evitando soluzioni sovradimensionate.

In fase di procurement, le procedure di gara DEVONO valutare l'impatto sui diritti fondamentali dell'intero sistema, inclusi modelli, servizi e componenti di terze parti. Il procurement DEVE inoltre evitare vincoli tecnologici che limitino la sostituibilità dei componenti in caso di criticità.

Principio 3 – Gestione del rischio

Secondo il principio di gestione del rischio, le PA adottano adeguate politiche di gestione del rischio conducendo un'analisi approfondita dei rischi associati all'impiego di sistemi di IA.

I sistemi DEVONO essere sviluppati includendo analisi del rischio tecnico, organizzativo e sistemico, considerando anche la dipendenza da fornitori e infrastrutture specifiche. Le architetture agentiche DEVONO consentire la mitigazione del rischio tramite sostituzione o isolamento dei componenti critici.

Il procurement DEVE prevedere valutazioni del rischio di lock-in, di continuità operativa e di perdita di controllo tecnologico. I contratti DEVONO supportare portabilità, reversibilità e fallback operativo, inclusa l'esecuzione su infrastrutture alternative.

Principio 4 – Protezione dei dati personali

Secondo il principio di protezione dei dati personali, le PA adottano i sistemi di IA nel rispetto delle norme in materia di protezione dei dati personali, garantendo qualità e integrità dei dati.

Lo sviluppo DEVE garantire una netta separazione tra dati, modelli e servizi, favorendo il controllo, la portabilità dei dati e la minimizzazione dei trattamenti. Le architetture agentiche DEVONO consentire un controllo analitico dei flussi informativi.

In fase di procurement, i contratti DEVONO limitare l'uso secondario dei dati, garantire la piena titolarità pubblica dei dataset e prevedere obblighi di restituzione e cancellazione. Il procurement DEVE inoltre evitare soluzioni che impongano la coesistenza forzata tra dati e modelli proprietari.

Principio 5 – Responsabilità

Secondo il principio di responsabilità, la responsabilità ultima delle decisioni adottate dai sistemi di IA rimane in capo alla PA. Le PA identificano chiaramente le responsabilità di tutti gli attori coinvolti.

In termini di sviluppo, i sistemi DEVONO essere progettati con chiara attribuzione delle responsabilità umane, possibilità di intervento e supervisione, e tracciabilità delle decisioni. L'uso di architetture agentiche DEVE rendere espliciti ruoli, funzioni e limiti dei singoli componenti.

In fase di procurement, i contratti DEVONO definire in modo chiaro responsabilità, obblighi di cooperazione, supporto in audit, verifiche e incident handling. Il procurement DEVE inoltre evitare modelli contrattuali che trasferiscano implicitamente responsabilità decisionali ai fornitori.

Principio 6 – Accessibilità, inclusività e non discriminazione

Secondo il principio di accessibilità, inclusività e non discriminazione, le PA assicurano un trattamento equo per tutti i soggetti e gruppi coinvolti nell'adozione dell'IA, promuovendo la parità di accesso, l'uguaglianza di genere e la diversità culturale, adottando misure preventive per evitare bias e discriminazioni.

Lo sviluppo DEVE includere test sistematici su bias e impatti discriminatori, coerenti con il livello di rischio del sistema. I sistemi DEVONO prevedere meccanismi correttivi attivabili dalla PA, inclusa la possibilità di sostituire modelli o componenti responsabili di effetti discriminatori, anche tramite architetture modulari e agentiche.

In fase di procurement, le gare DEVONO richiedere evidenze documentate su accessibilità, inclusività e mitigazione dei bias, incluse metodologie di test e risultati. Il procurement DEVE inoltre evitare soluzioni che rendano tecnicamente o contrattualmente difficile l'intervento correttivo in caso di discriminazioni.

Principio 7 – Trasparenza, spiegabilità e documentazione

Secondo il principio di trasparenza, spiegabilità e documentazione, le PA garantiscono trasparenza, comprensibilità e adeguata documentazione dei sistemi di IA lungo il loro ciclo di vita, secondo un principio di proporzionalità tra trasparenza ed efficacia (cfr. artt. 11 e 13 AI Act).

I sistemi DEVONO essere sviluppati con meccanismi di spiegabilità proporzionati al rischio e al contesto d'uso. Le architetture DEVONO consentire la documentazione separata e aggiornata di modelli, agenti, servizi e logiche di orchestrazione, favorendo la tracciabilità delle decisioni.

Il procurement DEVE garantire accesso continuativo alla documentazione tecnica e funzionale, anche post-contratto mediante apposite clausole in fase di gara (contratto). I contratti DEVONO prevedere obblighi di aggiornamento della documentazione in caso di modifica dei modelli, degli agenti o delle logiche decisionali.

Principio 8 – Trasparenza e informazione

Secondo il principio di trasparenza e informazione, le PA informano gli utenti sull'interazione con sistemi di IA, rendendoli consapevoli delle capacità e dei limiti di tali sistemi (cfr. art. 50 AI Act).

Lo sviluppo DEVE supportare funzionalità di informazione e consapevolezza dell'utente finale, incluse segnalazioni chiare sull'uso dell'IA e sui limiti del sistema. Le applicazioni DEVONO consentire l'aggiornamento delle informazioni in modo indipendente dal fornitore, anche in contesti multi-agente.

In fase di procurement, i contratti DEVONO prevedere obblighi di supporto alla comunicazione istituzionale verso cittadini e imprese. Il procurement DEVE inoltre evitare soluzioni che vincolino i contenuti informativi a interfacce o servizi proprietari non modificabili dalla PA.

Principio 9 – Qualità dei dati

Secondo il principio di qualità dei dati, le PA adottano sistemi di IA garantendo una gestione etica, trasparente e conforme dei dati, assicurandone qualità, integrità, sicurezza e sostenibilità (cfr. art. 10 AI Act).

In fase di sviluppo, i sistemi DEVONO includere meccanismi di validazione, controllo e tracciabilità dei dati lungo l'intero ciclo di vita. Lo sviluppo DEVE prevedere separazione tra dati, modelli e servizi, consentendo il controllo dei flussi informativi e l'uso di pipeline orchestrabili, anche in contesti multi-agente.

In fase di procurement, i capitolati DEVONO definire requisiti minimi di qualità dei dati, incluse completezza, rappresentatività e aggiornamento. Il procurement DEVE inoltre prevedere strumenti di verifica indipendenti e l'accesso ai log di trattamento dei dati, evitando dipendenze da piattaforme chiuse.

Principio 10 – Accuratezza

Secondo il principio di accuratezza, le PA garantiscono che i risultati prodotti dai sistemi di IA siano accurati e coerenti con gli obiettivi prefissati, adottando misure di monitoraggio continuo (cfr. art. 15 AI Act).

Lo sviluppo DEVE prevedere metriche di accuratezza misurabili, documentate e monitorabili nel tempo, in relazione al contesto d'uso e al livello di rischio. I sistemi DEVONO consentire l'aggiornamento o la sostituzione dei modelli (anche tramite orchestrazione agentica) in caso di degrado delle prestazioni.

In fase di procurement, i contratti DEVONO includere SLA e KPI sull'accuratezza, con meccanismi di intervento correttivo, inclusa la sostituzione progressiva dei modelli. Il procurement DEVE inoltre evitare vincoli che impediscano l'adeguamento delle prestazioni nel tempo.

Principio 11 – Robustezza

Secondo il principio di robustezza le PA assicurano che i sistemi di IA siano in grado di operare correttamente anche in condizioni avverse o in presenza di guasti (cfr. art. 15 AI Act).

In fase di sviluppo i sistemi DEVONO essere progettati con architetture modulari, resilienti e degradabili, in grado di mantenere livelli di servizio proporzionati anche in condizioni non ottimali. Lo sviluppo DEVE inoltre prevedere meccanismi di fallback, inclusa l'esecuzione su infrastrutture CPU-only e la riallocazione dinamica dei componenti tramite orchestrazione.

Il procurement DEVE favorire soluzioni componibili e sostituibili, riducendo il rischio di lock-in tecnologico. I contratti DEVONO inoltre prevedere clausole di verifica ed eventualmente risoluzione contrattuale in caso di comportamenti non in linea con quanto previsto a livello contrattuale.

Principio 12 – Sicurezza cibernetica

Secondo il principio di sicurezza cibernetica, le PA garantiscono la sicurezza cibernetica dei sistemi di IA, prevenendo alterazioni, compromissioni o usi illeciti, tutelando integrità, disponibilità e riservatezza di dati e processi (cfr. art. 15 AI Act).

Lo sviluppo DEVE integrare requisiti di sicurezza end-to-end su modelli, agenti, infrastrutture e servizi, includendo controllo degli accessi, segregazione dei ruoli e protezione delle pipeline di dati. Le architetture agentiche DEVONO consentire isolamento e sostituzione dei componenti compromessi senza impatti sull'intero sistema.

In fase di procurement, i contratti DEVONO includere obblighi di sicurezza, procedure di gestione degli incidenti, notifica tempestiva e cooperazione operativa con la PA. Il procurement DEVE inoltre evitare soluzioni che impediscano l'adozione di misure di sicurezza indipendenti dal fornitore o la migrazione verso infrastrutture alternative.

Principio 13 – Sorveglianza umana

Secondo il principio di sorveglianza umana, le PA garantiscono un adeguato livello di supervisione umana dei sistemi di IA, assicurando la possibilità di verifica, correzione o sostituzione delle decisioni automatizzate (cfr. art. 14 AI Act).

In fase di sviluppo, i sistemi DEVONO consentire un intervento umano effettivo, tempestivo e documentabile, anche a livello di orchestrazione dei servizi e degli agenti di IA. Lo sviluppo DEVE inoltre prevedere punti di controllo, modalità di override e meccanismi di arresto o riconfigurazione del sistema.

Il procurement DEVE escludere soluzioni che impediscano il controllo umano sostanziale o che rendano l'intervento umano solo formale. I contratti DEVONO garantire accesso operativo agli strumenti di supervisione, senza dipendenze esclusive dal fornitore.

Principio 14 – Registrazioni (logging)

Secondo il principio di logging, le PA adottano sistemi di IA dotati di adeguati meccanismi di registrazione per tracciare e conservare nel tempo le operazioni svolte (cfr. art. 12 AI Act).

I sistemi DEVONO essere sviluppati con meccanismi di logging interoperabili, strutturati e verificabili mediante audit, che traccino decisioni, flussi di dati, interventi umani e interazioni tra agenti. I log DEVONO inoltre essere indipendenti dall'infrastruttura sottostante e conservabili anche in caso di migrazione o dismissione.

In fase di procurement, i contratti DEVONO garantire alla PA accesso completo ai log, diritti di audit e possibilità di esportazione in formati aperti. Il procurement DEVE evitare soluzioni che rendano i log non accessibili o vincolati a piattaforme proprietarie.

Principio 15 – Adozione di standard tecnici

Secondo il principio di adozione di standard tecnici, le PA tengono in debita considerazione le norme tecniche nazionali, europee e internazionali per garantire interoperabilità, manutenibilità, sicurezza e conformità normativa dei sistemi di IA.

Lo sviluppo DEVE basarsi su standard aperti e riconosciuti a livello UE e internazionale, favorendo interoperabilità, portabilità e manutenibilità. Le architetture DEVONO inoltre adottare Application Programming Interface (API) standard e meccanismi di orchestrazione compatibili con ecosistemi multi-fornitore, evitando dipendenze tecnologiche rigide.

Il procurement DEVE privilegiare standard aperti, API documentate e formati interoperabili. I contratti DEVONO evitare soluzioni che limitino l'adozione futura di standard emergenti o l'integrazione con altri sistemi e fornitori.

Principio 16 – Efficienza e qualità dei servizi

Secondo il principio di efficienza e qualità dei servizi, le PA adottano l'IA per incrementare l'efficienza operativa e migliorare la qualità dei servizi destinati a cittadini e imprese, favorendo automazione, proattività e semplificazione.

I sistemi DEVONO essere progettati in funzione del miglioramento misurabile dei servizi pubblici, in termini di tempi, qualità, accessibilità ed efficacia. L'uso di architetture agentiche DEVE consentire l'automazione modulare dei processi e l'adattamento progressivo dei servizi, anche su infrastrutture eterogenee.

In fase di procurement, le gare DEVONO valutare le soluzioni sulla base dell'impatto reale sui servizi, e non esclusivamente sul costo o sulle prestazioni teoriche. Il procurement DEVE inoltre favorire soluzioni che consentano scalabilità, sostituibilità e continuità del servizio nel tempo.

Principio 17 – Innovazione e miglioramento continuo

Secondo il principio di innovazione e miglioramento continuo, le PA adottano un approccio orientato all'innovazione, al miglioramento continuo e alla collaborazione, promuovendo ecosistemi aperti, riuso, aggregazione e, ove opportuno, l'uso di componenti open source e open weights.

Lo sviluppo DEVE favorire riuso, modularità e integrazione in ecosistemi multi-fornitore. Le architetture agentiche DEVONO consentire la sperimentazione controllata, l'introduzione progressiva di nuovi modelli o servizi e il miglioramento continuo senza interruzioni del sistema.

Il procurement DEVE incoraggiare l'utilizzo di componenti aperte, riusabili e, ove appropriato, open source e open weights, valutandone il contributo in termini di trasparenza, interoperabilità, autonomia operativa e sostenibilità economica. Le procedure DEVONO inoltre favorire forme di collaborazione e aggregazione tra PA, anche attraverso modelli consortili o reti stabili.

Principio 18 – Sostenibilità ambientale

Secondo il principio di sostenibilità ambientale, le PA adottano sistemi di IA secondo un approccio sostenibile, attento alla tutela ambientale e coerente con i principi di sostenibilità energetica.

In fase di sviluppo, i sistemi DEVONO essere progettati tenendo conto dell'efficienza energetica e computazionale, privilegiando modelli proporzionati. L'uso di architetture agentiche DEVE inoltre consentire l'ottimizzazione dinamica dei carichi e la riduzione degli sprechi energetici.

In fase di procurement, le procedure di gara DEVONO includere criteri di sostenibilità ambientale riferiti all'intero stack IA (energia, infrastrutture, modelli, servizi). Il procurement, inoltre, DEVE favorire soluzioni che dimostrino misurabilità e riduzione dell'impatto ambientale nel tempo.

Principio 19 – Formazione e sviluppo delle competenze

Secondo il principio di formazione e sviluppo delle competenze, le PA investono nella formazione del proprio personale e promuovono iniziative di alfabetizzazione digitale per cittadini e imprese, al fine di garantire un'adozione dell'IA consapevole, inclusiva e responsabile.

Lo sviluppo DEVE essere accompagnato da trasferimento strutturato di competenze verso il personale della PA, includendo conoscenze su architettura, orchestrazione, modelli, dati e rischi. I sistemi DEVONO inoltre essere progettati in modo da essere comprensibili, gestibili e governabili dal personale interno.

In fase di procurement, i contratti DEVONO prevedere formazione continua, documentazione completa e affiancamento operativo strutturato. Il procurement DEVE inoltre evitare soluzioni che richiedano competenze esclusive o non trasferibili, riducendo la dipendenza da singoli fornitori.

Principio 20 – Rafforzamento dell'organizzazione e delle infrastrutture

Secondo il principio di rafforzamento dell'organizzazione e delle infrastrutture, le PA ottimizzano il proprio assetto organizzativo e le infrastrutture tecnologiche, anche in forma aggregata o condivisa, per sostenere la trasformazione digitale e migliorare sicurezza e resilienza.

In fase di sviluppo, i sistemi DEVONO essere progettati secondo architetture a servizi, con orchestrazione controllata dalla PA, separazione dei componenti e possibilità di gestione centralizzata o federata. Le architetture agentiche DEVONO supportare modelli organizzativi aggregati, condivisi o cooperativi tra PA.

Il procurement DEVE sostenere modelli architetturali che rafforzino il controllo, la resilienza e l'autonomia operativa dell'amministrazione. Le procedure, inoltre, DEVONO favorire soluzioni riusabili, condivisibili e interoperabili tra più PA, anche attraverso accordi quadro o modelli consortili.

Tabella 1: Principi per l'adozione, lo sviluppo e il procurement di sistemi di IA

Principi LG Adozione	Sviluppo	Procurement
P.1 – Conformità normativa Le PA adottano i sistemi di IA nel pieno rispetto delle normative	I sistemi di IA DEVONO essere progettati secondo il principio di compliance by design, includendo	I contratti DEVONO prevedere obblighi di conformità normativa evolutiva e clausole di

nazionali e dell'Unione Europea, assicurando l'aderenza al quadro legislativo vigente. Le PA monitorano l'evoluzione normativa e aggiornano costantemente i propri sistemi per assicurarne la conformità alle disposizioni vigenti	requisiti normativi (AI Act, GDPR, New Legislative Framework - NLF) lungo l'intero ciclo di vita. Lo sviluppo DEVE privilegiare architetture modulari e agentiche, che consentano l'aggiornamento selettivo dei componenti (modelli, servizi, orchestrazione) senza riprogettare l'intero sistema.	adeguamento in caso di modifiche legislative. Il procurement DEVE evitare soluzioni che rendano tecnicamente o contrattualmente complesso l'adeguamento normativo nel tempo.
P.2 – Rispetto dei valori fondamentali dell'UE Le PA adottano i sistemi di IA garantendo i principi definiti dalla Carta dei diritti fondamentali dell'Unione Europea (cfr. Art. 1 e 2 AI Act).	Lo sviluppo DEVE integrare valutazioni d'impatto sui diritti fondamentali e meccanismi di controllo umano, anche attraverso l'uso di agenti con ruoli e responsabilità chiaramente definiti. I modelli DEVONO essere selezionati e orchestrati in modo proporzionato al contesto, evitando soluzioni sovradimensionate.	Le procedure di gara DEVONO valutare l'impatto sui diritti fondamentali dell'intero sistema, inclusi modelli, servizi e componenti di terze parti. Il procurement DEVE evitare vincoli tecnologici che limitino la sostituibilità dei componenti in caso di criticità
P.3 – Gestione del rischio Le PA adottano adeguate politiche di gestione del rischio conducendo un'analisi approfondita dei rischi associati all'impiego di sistemi di IA.	I sistemi DEVONO essere sviluppati includendo analisi del rischio tecnico, organizzativo e sistemico, considerando anche la dipendenza da fornitori e infrastrutture specifiche. Le architetture agentiche DEVONO consentire la mitigazione del rischio tramite sostituzione o isolamento dei componenti critici.	Il procurement DEVE prevedere valutazione del rischio di lock-in, di continuità operativa e di perdita di controllo tecnologico. I contratti DEVONO supportare portabilità, reversibilità e fallback operativo, inclusa l'esecuzione su infrastrutture alternative.
P.4 – Protezione dei dati personali Le PA adottano i sistemi di IA nel rispetto delle norme in materia di protezione dei dati personali, garantendo qualità e integrità dei dati.	Lo sviluppo DEVE garantire una netta separazione tra dati, modelli e servizi, favorendo il controllo, la portabilità dei dati e la minimizzazione dei trattamenti. Le architetture agentiche DEVONO consentire un controllo analitico dei flussi informativi.	I contratti DEVONO limitare l'uso secondario dei dati, garantire la piena titolarità pubblica dei dataset e prevedere obblighi di restituzione e cancellazione. Il procurement DEVE evitare soluzioni che impongano la coesistenza forzata tra dati e modelli proprietari.
P.5 – Responsabilità La responsabilità ultima delle decisioni adottate dai sistemi di IA rimane in capo alla PA. Le PA identificano chiaramente le responsabilità di tutti gli attori coinvolti.	I sistemi DEVONO essere progettati con chiara attribuzione delle responsabilità umane, possibilità di intervento e supervisione, e tracciabilità delle decisioni. L'uso di architetture agentiche DEVE rendere espliciti ruoli,	I contratti DEVONO definire in modo chiaro responsabilità, obblighi di cooperazione, supporto in audit, verifiche e incident handling. Il procurement DEVE evitare modelli contrattuali che trasferiscano implicitamente

	funzioni e limiti dei singoli componenti.	responsabilità decisionali ai fornitori.
P.6 – Accessibilità, inclusività, non discriminazione Le PA assicurano un trattamento equo per tutti i soggetti e gruppi coinvolti nell'adozione dell'IA, promuovendo la parità di accesso, l'uguaglianza di genere e la diversità culturale, adottando misure preventive per evitare bias e discriminazioni.	Lo sviluppo DEVE includere test sistematici su bias e impatti discriminatori, coerenti con il livello di rischio del sistema. I sistemi DEVONO prevedere meccanismi correttivi attivabili dalla PA, inclusa la possibilità di sostituire modelli o componenti responsabili di effetti discriminatori, anche tramite architetture modulari e agentiche.	Le gare DEVONO richiedere evidenze documentate su accessibilità, inclusività e mitigazione dei bias, incluse metodologie di test e risultati. Il procurement DEVE evitare soluzioni che rendano tecnicamente o contrattualmente difficile l'intervento correttivo in caso di discriminazioni.
P.7 – Trasparenza, spiegabilità e documentazione Le PA garantiscono trasparenza, comprensibilità e adeguata documentazione dei sistemi di IA lungo il loro ciclo di vita, secondo un principio di proporzionalità tra trasparenza ed efficacia (cfr. artt. 11 e 13 AI Act).	I sistemi DEVONO essere sviluppati con meccanismi di spiegabilità proporzionati al rischio e al contesto d'uso. Le architetture DEVONO consentire la documentazione separata e aggiornata di modelli, agenti, servizi e logiche di orchestrazione, favorendo la tracciabilità delle decisioni.	Il procurement DEVE garantire accesso continuativo alla documentazione tecnica e funzionale, anche post-contratto mediante apposite clausole in fase di gara (contratto) I contratti DEVONO prevedere obblighi di aggiornamento della documentazione in caso di modifica dei modelli, degli agenti o delle logiche decisionali.
P.8 – Trasparenza e informazione Le PA informano gli utenti sull'interazione con sistemi di IA, rendendoli consapevoli delle capacità e dei limiti di tali sistemi (cfr. art. 50 AI Act).	Lo sviluppo DEVE supportare funzionalità di informazione e consapevolezza dell'utente finale, incluse segnalazioni chiare sull'uso dell'IA e sui limiti del sistema. Le applicazioni DEVONO consentire l'aggiornamento delle informazioni in modo indipendente dal fornitore, anche in contesti multi-agente.	I contratti DEVONO prevedere obblighi di supporto alla comunicazione istituzionale verso cittadini e imprese. Il procurement DEVE evitare soluzioni che vincolino i contenuti informativi a interfacce o servizi proprietari non modificabili dalla PA.
P.9 – Qualità dei dati Le PA adottano sistemi di IA garantendo una gestione etica, trasparente e conforme dei dati, assicurandone qualità, integrità, sicurezza e sostenibilità (cfr. art. 10 AI Act).	I sistemi DEVONO includere meccanismi di validazione, controllo e tracciabilità dei dati lungo l'intero ciclo di vita. Lo sviluppo DEVE prevedere separazione tra dati, modelli e servizi, consentendo il controllo dei flussi informativi e l'uso di pipeline orchestrabili, anche in contesti multi-agente.	I capitolati DEVONO definire requisiti minimi di qualità dei dati, incluse completezza, rappresentatività e aggiornamento. Il procurement DEVE prevedere strumenti di verifica indipendenti e l'accesso ai log di trattamento dei dati, evitando dipendenze da piattaforme chiuse.
P.10 – Accuratezza Le PA garantiscono che i risultati prodotti dai sistemi di IA siano accurati e coerenti con gli obiettivi prefissati, adottando	Lo sviluppo DEVE prevedere metriche di accuratezza misurabili, documentate e monitorabili nel tempo, in	I contratti DEVONO includere SLA e KPI sull'accuratezza, con meccanismi di intervento correttivo, inclusa la sostituzione progressiva dei modelli.

misure di monitoraggio continuo (cfr. art. 15 AI Act).	relazione al contesto d'uso e al livello di rischio. I sistemi DEVONO consentire l'aggiornamento o la sostituzione dei modelli (anche tramite orchestrazione agentica) in caso di degrado delle prestazioni.	Il procurement DEVE evitare vincoli che impediscano l'adeguamento delle prestazioni nel tempo.
P.11 – Robustezza Le PA assicurano che i sistemi di IA siano in grado di operare correttamente anche in condizioni avverse o in presenza di guasti (cfr. art. 15 AI Act).	I sistemi DEVONO essere progettati con architetture modulari, resilienti e degradabili, in grado di mantenere livelli di servizio proporzionati anche in condizioni non ottimali. Lo sviluppo DEVE prevedere meccanismi di fallback, inclusa l'esecuzione su infrastrutture CPU-only, architetture hardware agnostic e la riallocazione dinamica dei componenti tramite orchestrazione.	Il procurement DEVE favorire soluzioni componibili e sostituibili, riducendo il rischio di lock-in tecnologico. I contratti DEVONO prevedere clausole di verifica ed eventualmente risoluzione contrattuale in caso di comportamenti non in linea con quanto previsto a livello contrattuale.
P.12 – Sicurezza cibernetica Le PA garantiscono la sicurezza cibernetica dei sistemi di IA, prevenendo alterazioni, compromissioni o usi illeciti, tutelando integrità, disponibilità e riservatezza di dati e processi (cfr. art. 15 AI Act).	Lo sviluppo DEVE integrare requisiti di sicurezza end-to-end su modelli, agenti, infrastrutture e servizi, includendo controllo degli accessi, segregazione dei ruoli e protezione delle pipeline di dati. Le architetture agentiche DEVONO consentire isolamento e sostituzione dei componenti compromessi senza impatti sull'intero sistema.	I contratti DEVONO includere obblighi di sicurezza, procedure di gestione degli incidenti, notifica tempestiva e cooperazione operativa con la PA. Il procurement DEVE evitare soluzioni che impediscano l'adozione di misure di sicurezza indipendenti dal fornitore o la migrazione verso infrastrutture alternative.
P.13 – Sorveglianza umana Le PA garantiscono un adeguato livello di supervisione umana dei sistemi di IA, assicurando la possibilità di verifica, correzione o sostituzione delle decisioni automatizzate (cfr. art. 14 AI Act).	I sistemi DEVONO consentire un intervento umano effettivo, tempestivo e documentabile, anche a livello di orchestrazione dei servizi e degli agenti di IA. Lo sviluppo DEVE prevedere punti di controllo, modalità di override e meccanismi di arresto o riconfigurazione del sistema.	Il procurement DEVE escludere soluzioni che impediscano il controllo umano sostanziale o che rendano l'intervento umano solo formale. I contratti DEVONO garantire accesso operativo agli strumenti di supervisione, senza dipendenze esclusive dal fornitore.
P.14 – Registrazioni (logging) Le PA adottano sistemi di IA dotati di adeguati meccanismi di registrazione per tracciare e conservare nel tempo le operazioni svolte (cfr. art. 12 AI Act).	I sistemi DEVONO essere sviluppati con meccanismi di logging interoperabili, strutturati e verificabili mediante audit, che traccino decisioni, flussi di dati, interventi umani e interazioni tra agenti. I log DEVONO essere indipendenti dall'infrastruttura	I contratti DEVONO garantire alla PA accesso completo ai log, diritti di audit e possibilità di esportazione in formati aperti. Il procurement DEVE evitare soluzioni che rendano i log non accessibili o vincolati a piattaforme proprietarie.

	sottostante e conservabili anche in caso di migrazione o dismissione.	
P.15 – Adozione di standard tecnici Le PA tengono in debita considerazione le norme tecniche nazionali, europee e internazionali per garantire interoperabilità, manutenibilità, sicurezza e conformità normativa dei sistemi di IA.	Lo sviluppo DEVE basarsi su standard aperti e riconosciuti a livello UE e internazionale, favorendo interoperabilità, portabilità e manutenibilità. Le architetture DEVONO adottare API standard e meccanismi di orchestrazione compatibili con ecosistemi multi-fornitore, evitando dipendenze tecnologiche rigide.	Il procurement DEVE privilegiare standard aperti, API documentate e formati interoperabili. I contratti DEVONO evitare soluzioni che limitino l'adozione futura di standard emergenti o l'integrazione con altri sistemi e fornitori.
P.16 – Efficienza e qualità dei servizi Le PA adottano l'IA per incrementare l'efficienza operativa e migliorare la qualità dei servizi destinati a cittadini e imprese, favorendo automazione, proattività e semplificazione.	I sistemi DEVONO essere progettati in funzione del miglioramento misurabile dei servizi pubblici, in termini di tempi, qualità, accessibilità ed efficacia. L'uso di architetture agentiche DEVE consentire l'automazione modulare dei processi e l'adattamento progressivo dei servizi, anche su infrastrutture eterogenee.	Le gare DEVONO valutare le soluzioni sulla base dell'impatto reale sui servizi, e non esclusivamente sul costo o sulle prestazioni teoriche. Il procurement DEVE favorire soluzioni che consentano scalabilità, sostituibilità e continuità del servizio nel tempo.
P.17 – Innovazione e miglioramento continuo Le PA adottano un approccio orientato all'innovazione, al miglioramento continuo e alla collaborazione, promuovendo ecosistemi aperti, riuso, aggregazione e, ove opportuno, l'uso di componenti open source e open weights.	Lo sviluppo DEVE favorire riuso, modularità e integrazione in ecosistemi multi-fornitore. Le architetture agentiche DEVONO consentire la sperimentazione controllata, l'introduzione progressiva di nuovi modelli o servizi e il miglioramento continuo senza interruzioni del sistema.	Il procurement DEVE incoraggiare l'utilizzo di componenti aperte, riusabili e, ove appropriato, open source e open weights, valutandone il contributo in termini di trasparenza, interoperabilità, autonomia operativa e sostenibilità economica. Le procedure DEVONO favorire forme di collaborazione e aggregazione tra PA, anche attraverso modelli consortili o reti stabili.
P.18 – Sostenibilità ambientale Le PA adottano sistemi di IA secondo un approccio sostenibile, attento alla tutela ambientale e coerente con i principi di sostenibilità energetica.	I sistemi DEVONO essere progettati tenendo conto dell'efficienza energetica e computazionale, privilegiando modelli proporzionati L'uso di architetture agentiche DEVE consentire l'ottimizzazione dinamica dei carichi e la riduzione degli sprechi energetici.	Le procedure di gara DEVONO includere criteri di sostenibilità ambientale riferiti all'intero stack IA (energia, infrastrutture, modelli, servizi). Il procurement DEVE favorire soluzioni che dimostrino misurabilità e riduzione dell'impatto ambientale nel tempo.

P.19 – Formazione e sviluppo delle competenze Le PA investono nella formazione del proprio personale e promuovono iniziative di alfabetizzazione digitale per cittadini e imprese, al fine di garantire un'adozione dell'IA consapevole, inclusiva e responsabile.	Lo sviluppo DEVE essere accompagnato da trasferimento strutturato di competenze verso il personale della PA, includendo conoscenze su architettura, orchestrazione, modelli, dati e rischi. I sistemi DEVONO essere progettati in modo da essere comprensibili, gestibili e governabili dal personale interno.	I contratti DEVONO prevedere formazione continua, documentazione completa e affiancamento operativo strutturato. Il procurement DEVE evitare soluzioni che richiedano competenze esclusive o non trasferibili, riducendo la dipendenza da singoli fornitori.
P.20 – Rafforzamento dell'organizzazione e delle infrastrutture Le PA ottimizzano il proprio assetto organizzativo e le infrastrutture tecnologiche, anche in forma aggregata o condivisa, per sostenere la trasformazione digitale e migliorare sicurezza e resilienza.	I sistemi DEVONO essere progettati secondo architetture a servizi, con orchestrazione controllata dalla PA, separazione dei componenti e possibilità di gestione centralizzata o federata. Le architetture agentiche DEVONO supportare modelli organizzativi aggregati, condivisi o cooperativi tra PA.	Il procurement DEVE sostenere modelli architetturali che rafforzino il controllo, la resilienza e l'autonomia operativa dell'amministrazione. Le procedure DEVONO favorire soluzioni riusabili, condivisibili e interoperabili tra più PA, anche attraverso accordi quadro o modelli consortili.

Nel loro insieme, le estensioni riportate nella tabella rendono i principi di adozione attuabili, verificabili anche attraverso processi di audit, rafforzando la capacità di governo delle Pubbliche Amministrazioni sui sistemi di IA adottati. L'allineamento tra principi, sviluppo e procurement favorisce soluzioni modulari, sostenibili ed evolvibili, riduce le dipendenze tecnologiche critiche e preserva l'autonomia decisionale e operativa della PA lungo tutto il ciclo di vita dei sistemi di IA, nel rispetto del quadro normativo e dell'interesse pubblico.

2.3 Principi generali della Legge 132/2025 per lo sviluppo e il procurement di sistemi IA

Nel contesto della Pubblica Amministrazione, i principi stabiliti dall'articolo 3 della Legge 132/2025 devono essere declinati nell'intero ciclo di vita dei sistemi di Intelligenza Artificiale, sia nella fase di progettazione e sviluppo dei sistemi di IA, sia nella fase di procurement. Tale previsione garantisce un utilizzo responsabile, sicuro e conforme al quadro normativo vigente.

2.3.1 Principio 1 – Tutela dei diritti fondamentali, trasparenza, proporzionalità e non discriminazione

Il principio impone che i sistemi di IA siano progettati e utilizzati nel pieno rispetto dei diritti fondamentali, evitando effetti discriminatori e garantendo un livello di trasparenza e proporzionalità adeguato al contesto d'uso.

I sistemi DEVONO essere progettati considerando gli impatti sui diritti e sulle libertà fondamentali fin dalle fasi di design e testing.

DEVONO essere effettuate valutazioni preventive come DPIA o FRIA (in caso di sistemi ad alto rischio).

Le misure adottate per rispettare i principi del presente paragrafo DEVONO essere proporzionate al contesto di utilizzo e agli obiettivi del sistema, anche tenuto conto degli esiti delle predette valutazioni preventive.

2.3.2 Principio 2 – Qualità, correttezza, attendibilità e trasparenza dei dat i e dei processi

Questo principio richiede che i dati e i processi utilizzati nei sistemi di IA siano adeguati, verificabili e trasparenti, secondo il principio di proporzionalità rispetto al settore di utilizzo.

I dataset DEVONO essere adottati processi strutturati di data governance e controllo qualità.

DEVONO essere adottati processi strutturati di data governance e controllo qualità.

I processi di addestramento, validazione e testing DEVONO essere tracciabili e verificabili.

2.3.3 Principio 3 – Centralità dell'uomo, spiegabilità e prevenzione del danno

Il principio afferma la centralità umana nei processi decisionali, la prevenzione del danno e la necessità di garantire spiegabilità, sorveglianza e intervento umano.

I sistemi DEVONO essere progettati per consentire supervisione umana e intervento umano significativo.

DEVONO essere implementate funzionalità di spiegabilità adeguate al livello di rischio.

DEVONO essere adottate misure preventive per evitare danni fisici, materiali o immateriali.

2.3.4 Principio 4 – Tutela del metodo democratico, istituzionale e sovranità

l principio tutela il **metodo democratico**, l'**autonomia delle istituzioni** e la **sovranità dello Stato**, prevenendo **interferenze illecite** nei processi pubblici.

I sistemi DEVONO prevenire usi impropri che possano alterare processi decisionali pubblici o politici.

DEVONO essere implementate misure di sicurezza contro abusi e manipolazioni informative.

DEVONO essere garantite integrità, tracciabilità e provenienza di modelli e dati.

2.3.5 Principio 5 – Coerenza con il quadro normativo europeo (AI Act)

Il principio assicura la coerenza normativa della L. 132/2025 con il Regolamento (EU) 2024/1689 (AI ACT), senza introdurre nuovi obblighi.

I sistemi DEVONO essere sviluppati in conformità ai requisiti applicabili dell'AI Act.

DEVONO essere predisposte evidenze di conformità lungo l'intero ciclo di vita.

2.3.6 Principio 6 – Cybersicurezza e resilienza lungo il ciclo di vita

La cybersicurezza è una precondizione essenziale per lo sviluppo e l'utilizzo dei sistemi di IA, socondo un approccio basato sul rischio.

Le misure di sicurezza DEVONO essere integrate by design e by default.

I sistemi DEVONO essere resilienti ad attacchi e manipolazioni.

DEVONO essere condotti test di sicurezza lungo l'intero ciclo di vita.

2.3.7 Principio 7 – Accessibilità universale e inclusione

Il principio garantisce l'accesso universale ai sistemi di IA e la piena inclusione delle persone con disabilità.

Le interfacce DEVONO essere progettate secondo i requisiti di accessibilità.

DEVONO essere effettuati test di usabilità inclusiva.

2.4 Famiglie di sistemi di IA

Si propone a seguire una possibile classificazione delle famiglie di Intelligenza Artificiale, ad oggi note e maggiormente diffuse. La classificazione si basa di fatto su un modello gerarchico di inclusione (sub-setting), per definire le relazioni tra i diversi domini della tecnologia su cui si basa l'IA.

Procedendo dal generale al particolare, la classificazione prevede:

- Intelligenza Artificiale (IA)
- IA Statistica (Data-driven)

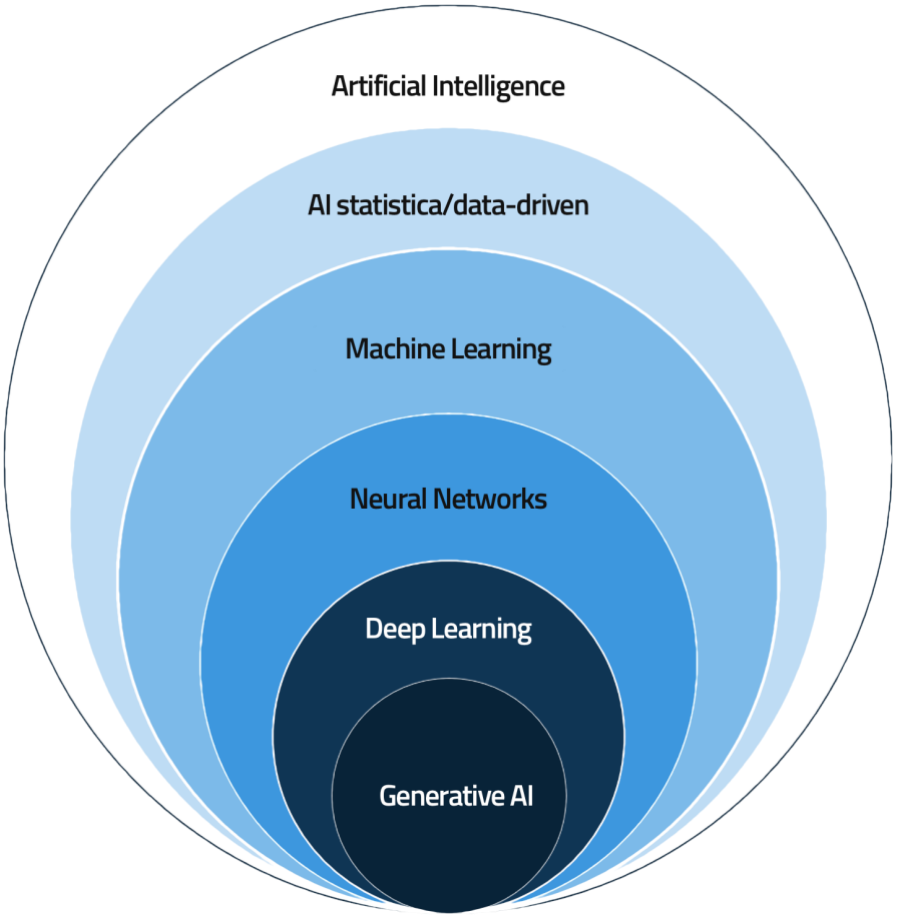
- Machine Learning (ML)
- Neural Networks (Reti Neurali)
- Deep Learning (DL)
- Generative AI (IA Generativa).

Le caratteristiche di ogni famiglia sono descritte a seguire e analogamente rappresentate nella 1.

2.4.1 Intelligenza Artificiale (IA)

Un sistema di Intelligenza Artificiale è definito dall'AI ACT come "*un sistema automatizzato progettato per funzionare con livelli di autonomia variabili e che può presentare adattabilità dopo la diffusione e che, per obiettivi espliciti o impliciti, deduce dall'input che riceve come generare output quali previsioni, contenuti, raccomandazioni o decisioni che possono influenzare ambienti fisici o virtuali*". L'IA può essere dunque rappresentato come l'insieme di tecnologie che consentono ai sistemi computazionali di simulare capacità cognitive umane, quali l'apprendimento, il ragionamento e l'adattamento. Come noto, le tecnologie di IA si dividono in due grandi categorie, quelle basate su meccanismi deduttivi che lavorano su basi di conoscenza attraverso approcci "simbolici", e quelle basati su sistemi induttivi che lavorano attraverso l'elaborazione di grandi quantità di dati. La categoria di modelli basati su meccanismi induttivi risulta essere la categoria di maggiore impatto nelle PA.

Figura 1 - Classificazione delle famiglie di IA



1

2.4.2 IA Statistica (Data-driven)

Questa categoria identifica l'approccio basato sull'elaborazione di grandi quantità di dati elaborati mediante l'esecuzione di algoritmi complessi. A differenza dei sistemi deterministici basati su regole fisse, l'IA statistica utilizza modelli matematici per apprendere capacità di classificazione o previsione osservando i dataset.

2.4.3 Machine Learning (ML)

Questa famiglia di tecniche di IA si basa su algoritmi statistici e matematici, come ad esempio algoritmi di regressione, alberi decisionali o altri modelli supervisionati, in grado di *apprendere* pattern ricorrenti

a partire dalle relazioni presenti nei dati strutturati. Nei modelli parametrici (ad esempio le regressioni) l'addestramento consiste nell'ottimizzare i *pesi* associati alle variabili, che quantificano l'influenza di ciascun input sulla previsione finale. Nei modelli non parametrici (come gli alberi decisionali), invece, l'apprendimento avviene attraverso la selezione iterativa di soglie e regole di suddivisione dei dati che massimizzano la capacità predittiva del modello.

2.4.4 Neural Networks (Reti Neurali)

Le reti neurali rappresentano una famiglia di tecnologie computazionali ispirate alla struttura dei neuroni. Le informazioni, in questo caso, vengono elaborate utilizzando reti di nodi interconnessi in grado di apprendere le rappresentazioni complesse dei dati. A differenza degli algoritmi di Machine Learning, le Reti Neurali sono particolarmente adatte per processare dati non strutturati (immagini, voce, testo) in quanto, attraverso un processo *feature engineering*, sono in grado di effettuare l'estrazione automatica delle caratteristiche *(feature extraction)*, identificando autonomamente i pattern rilevanti senza che questi debbano essere esplicitamente e preventivamente definiti.

2.4.5 Deep Learning (DL)

Questa categoria o famiglia di IA rappresenta l'estensione più complessa delle Reti Neurali ed è caratterizzata da *modelli profondi* composti da numerosi livelli che apprendono le strutture gerarchiche dei dati. La caratteristica fondamentale di questa categoria di IA consiste nell'impiego di *deep neural networks*, architetture complesse dove ogni livello della rete neurale elabora l'output del livello precedente, facendo in modo che il modello apprenda *pattern complessi e non lineari*, superando quindi i limiti dei metodi di estrazione tradizionali, anche sfruttando le cosiddette tecniche di *backpropagation*. La profondità delle reti, rappresentata dal numero di livelli o strati, è ciò che caratterizza e distingue i modelli di Deep Learning dalla famiglia di Reti Neurali meno complesse.

2.4.6 Generative AI (IA Generativa)

La IA Generativa rappresenta la famiglia di tecnologie di IA più recente, largamente diffusa ed adottata dal 2021. Questa categoria include sistemi in grado di generare nuovi contenuti, come ad esempio testi, immagini, codice sorgente, etc., sfruttando strutture e pattern appresi durante una fase di pre-training.

Questi sistemi, come i Large Language Model (LLM), utilizzano architetture profonde, in particolare transformer – ovvero architetture di deep learning basate sui meccanismi in grado di elaborare sequenze complesse sfruttando tecniche di parallelismo. I LLM sono addestrati su grandi quantità di dati non strutturati tramite tecniche di *self-supervised learning*, che consentono di costruire *rappresentazioni latenti* complesse e di produrre output coerenti con il contesto e rilevanti nel contenuto. Questo meccanismo di basa sulla capacità del modello di trasformare i dati grezzi, come ad esempio testo, immagini o audio, in una forma astratta, interna all'elaborazione, finalizzata a catturare significati, strutture e relazioni tra dati. Questa fase del processo di elaborazione prende il nome di *latent representation*.

3 Architettura logica di riferimento: orchestratore, modelli, dati e tool

La Pubblica amministrazione, negli ultimi anni, ha affrontato il processo di trasformazione digitale di sistemi, infrastrutture e servizi, introducendo strumenti e tecniche basate su Intelligenza Artificiale.

La necessità di disporre di strumenti di IA, considerata finora come un fenomeno sperimentare, si qualifica attualmente come una prassi in fase di consolidamento, come dimostrato anche dalle indagini e rilevazioni, condotte da questa Agenzia nel corso dell'anno 2025, le cui fonti consultate sono state le Pubbliche Amministrazioni centrali e locali.

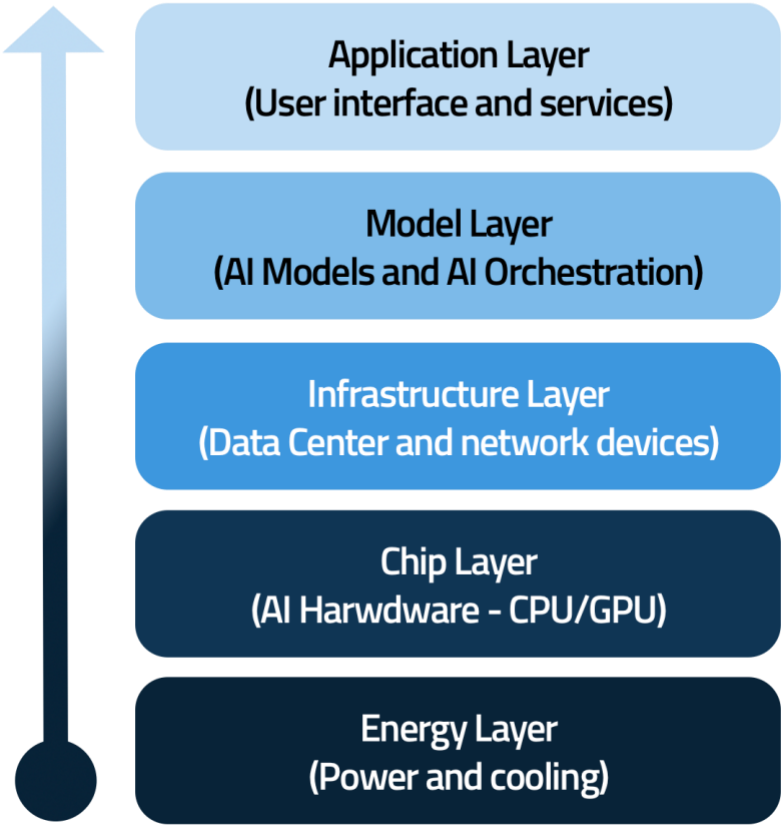
L'uso dell'intelligenza artificiale si è rapidamente trasformato da iniziative sperimentali ed esplorative verso realizzazioni essenziali e strategiche, ampiamente adottate in vari settori organizzativi delle Pubbliche Amministrazioni e delle Imprese. Le Amministrazioni fanno affidamento in misura crescente su modelli avanzati di IA, tra cui i Large Language Model (LLM), strumenti di analisi predittiva, strumenti per migliorare i processi decisionali o aumentare l'efficienza operativa. Tuttavia, man mano che questi sistemi passano da implementazioni sperimentali ad estesi e complessi ambienti di produzione di soluzioni progettuali scalabili, le Amministrazioni si trovano ad affrontare notevoli sfide nella progettazione di architetture, nel dimensionamento delle infrastrutture abilitanti e nella quantificazione dei relativi oneri economici ed energetici.

3.1 Lo Stack IA: dal livello Energy al livello Servizi e soluzioni

Lo Stack IA rappresenta una stratificazione organizzata delle componenti tecniche, hardware e software, necessarie per progettare, addestrare, distribuire ed eseguire sistemi di intelligenza artificiale. È composto da livelli funzionali interdipendenti che vanno dagli impianti in grado di assicurare la sostenibilità energetica delle infrastrutture, alle componenti computazionali (es. GPU, cloud, reti ad alte prestazioni), ai framework e *runtime* di machine learning, fino ai modelli, ai dati, agli strumenti di governance e alle applicazioni che utilizzano tali modelli.
Lo stack consente di integrare in modo coerente risorse computazionali, *pipeline* di dati, algoritmi,

modelli e meccanismi di monitoraggio, garantendo scalabilità, sicurezza, tracciabilità, riproducibilità e controllo operativo sull'intero ciclo di vita dell'IA.

Figura 2 - Stack completo della struttura IA



2Lo Stack IA, dunque, descrive l'insieme delle componenti tecnologiche e organizzative che una Pubblica Amministrazione deve presidiare e governare in caso di sviluppo end-to-end di soluzioni di intelligenza artificiale scalabili ed efficienti, garantendo il controllo e la qualità dell'intero ciclo di vita del sistema.

Lo stack IA proposto si articola su un insieme di cinque livelli, che procedendo dal basso verso l'alto sono:

- Energy Layer (Livello Energetico)
- Chip Layer (Livello Computazionale)

- Infrastructure Layer (Livello Infrastrutturale)
- AI Model Layer (Livello dei Modelli IA)
- Application Layer (Livello Applicativo).

Ognuno di questi livelli viene descritto a seguire.

3.1.1.1 Energy Layer (Livello Energetico)

L'Energy Layer costituisce la base fisica dello Stack IA: comprende la fornitura di energia elettrica, i sistemi di raffreddamento e la gestione termica necessari ad alimentare data center e dispositivi IA.

È rilevante per la PA perché un'infrastruttura energetica stabile ed efficiente è indispensabile per garantire continuità operativa. L'attuazione, inoltre, di tecniche di ottimizzazione energetica riduce costi operativi e impatto ambientale.

Componenti tipiche sono: sistemi di alimentazione ad alta efficienza; tecnologie avanzate di raffreddamento (es. raffreddamento liquido); utilizzo di fonti rinnovabili per data center.

Per la PA questo livello assume particolare importanza in relazione:

- ai costi di esercizio;
- alla sostenibilità ambientale;
- alla resilienza dei servizi pubblici digitali;
- agli impatti sulle reti di trasmissione e distribuzione, in particolare in presenza di carichi concentrati, non programmabili o localizzati in aree non originariamente dimensionate per tali utilizzi.

	Descrizione
Funzione	Costituisce la base fisica dello Stack IA: comprende la fornitura di energia elettrica, i sistemi di raffreddamento e la gestione termica necessari ad alimentare data center e dispositivi IA.

Rilevanza per la PA	Un'infrastruttura energetica stabile ed efficiente è indispensabile per garantire continuità operativa. L'attuazione, inoltre, di tecniche di ottimizzazione energetica riduce costi operativi e impatto ambientale.
Componenti tipiche	Sistemi di alimentazione ad alta efficienza; tecnologie avanzate di raffreddamento (es. raffreddamento liquido); utilizzo di fonti rinnovabili per data center.
Punti di attenzione per la PA	Questo livello assume particolare importanza in relazione: • ai costi di esercizio; • alla sostenibilità ambientale; • alla resilienza dei servizi pubblici digitali; • agli impatti sulle reti di trasmissione e distribuzione, in particolare in presenza di carichi concentrati, non programmabili o localizzati in aree non originariamente dimensionate per tali utilizzi.

3.1.2 Chip Layer (Livello computazionale)

Il Chip Layer comprende semiconduttori e acceleratori IA, come le GPU (Graphics Processing Unit - particolarmente utili per l'addestramento e l'inferenza dei modelli di IA grazie alla loro architettura altamente parallela), le TPU (Tensor Processing Unit – nativamente progettati per utilizzo di IA per operazioni tensoriali e reti neurali), gli ASIC (Application-Specific Integrated Circuit - se progettati specificamente per compiti di IA sono in grado di ottimizzare i processi di inferenza); le memorie HBM (High Bandwidth Memory).

È rilevante per la PA in quanto le prestazioni dei modelli dipendono direttamente dalla potenza e dall'efficienza dell'hardware sottostante. La disponibilità di hardware tecnicamente avanzato abilita caratteristiche di scalabilità, produce tempi di calcolo ridotti e ottimizza i consumi energetici della computazione.

Componenti tipiche sono: GPU/TPU, acceleratori specifici per IA, CPU ad alte prestazioni, memorie, sistemi hardware ottimizzati per workload IA, hardware dedicato ad Edge IA.

Per la PA, le scelte relative a questo layer influenzano in modo significativo la portabilità dei modelli e le performance dei modelli di IA che si intende adottare.

	Descrizione
Funzione	Comprende semiconduttori e acceleratori IA, come le GPU (Graphics Processing Unit - particolarmente utili per l'addestramento e l'inferenza dei modelli di IA grazie alla loro architettura altamente parallela), le TPU (Tensor Processing Unit – nativamente progettati per utilizzo di IA per operazioni tensoriali e reti neurali), gli ASIC (Application-Specific Integrated Circuit - se progettati specificamente per compiti di IA sono in grado di ottimizzare i processi di inferenza); le memorie HBM (High Bandwidth Memory).
Rilevanza per la PA	Le prestazioni dei modelli dipendono direttamente dalla potenza e dall'efficienza dell'hardware sottostante. La disponibilità di hardware tecnicamente avanzato abilita caratteristiche di scalabilità, produce tempi di calcolo ridotti e ottimizza i consumi energetici della computazione.
Componenti tipiche	GPU/TPU, acceleratori specifici per IA, CPU ad alte prestazioni, memorie, sistemi hardware ottimizzati per workload IA, hardware dedicato ad Edge IA.
Punti di attenzione per la PA	Le scelte relative a questo layer influenzano in modo significativo la portabilità dei modelli e le performance dei modelli di IA che si intende adottare.

3.1.3 Infrastructure Layer (Livello Infrastrutturale)

L'Infrastructure Layer include data center, reti, storage, sistemi di virtualizzazione, piattaforme cloud e servizi di orchestrazione che permettono di distribuire, scalare e gestire carichi di lavoro IA.

È rilevante per la PA in quanto un'infrastruttura robusta garantisce disponibilità, sicurezza, continuità operativa, connettività ad alte prestazioni e gestione efficiente dei flussi dati necessari ai sistemi IA.

Componenti tipiche sono: data center, cloud pubblico/privato/ibrido, container orchestration (es. piattaforma di orchestrazione di container che automatizza la distribuzione, la gestione e la scalabilità delle applicazioni), sistemi di storage distribuito, reti ad alta velocità, servizi di monitoraggio. Questo Layer può inoltre comprendere: cluster di GPU; infrastrutture cloud-native; architetture di rete ad altissima velocità e bassa latenza; sistemi di storage parallelo per dataset di grandi dimensioni.

Per la PA, è il layer su cui si innestano i requisiti di qualificazione cloud (Regolamento ACN n. 21007), sovranità del dato, resilienza e interoperabilità tra sistemi.

	Descrizione
Funzione	Include data center, reti, storage, sistemi di virtualizzazione, piattaforme cloud e servizi di orchestrazione che permettono di distribuire, scalare e gestire carichi di lavoro IA.
Rilevanza per la PA	Un'infrastruttura robusta garantisce disponibilità, sicurezza, continuità operativa, connettività ad alte prestazioni e gestione efficiente dei flussi dati necessari ai sistemi IA.
Componenti tipiche	Data center, cloud pubblico/privato/ibrido, container orchestration (es. piattaforma di orchestrazione di container che automatizza la distribuzione, la gestione e la scalabilità delle applicazioni), sistemi di storage distribuito, reti ad alta velocità, servizi di monitoraggio. Può comprendere Cluster di GPU; infrastrutture cloud-native; architetture di rete ad altissima velocità e bassa latenza; sistemi di storage parallelo per dataset di grandi dimensioni.
Punti di attenzione per la PA	È il layer su cui si innestano i requisiti di qualificazione cloud (Regolamento ACN n. 21007), sovranità del dato, resilienza e interoperabilità tra sistemi

3.1.4 AI Model Layer (Livello dei Modelli IA)

L'AI Model Layer comprende modelli di IA, framework di machine learning, pipeline di addestramento, strumenti per fine-tuning, valutazione, monitoraggio, orchestrazione dei modelli e inferenza.

È rilevante per la PA perché questo livello rappresenta il "cuore logico" dell'IA: qui si sviluppano, addestrano, usano, monitorano e aggiornano i modelli che abilitano le funzionalità intelligenti delle applicazioni.

Componenti tipiche sono: Framework ML/DL, modelli fondamentali, dataset, strumenti e piattaforme di MLOps, sistemi di versioning di modelli e dati, strumenti di explainability e auditing, LLM, strumenti per addestramento distribuito.

In termini di punti di attenzione, la PA deve considerare sia modelli sviluppati internamente sia modelli forniti da terzi, inclusi modelli general purpose. Le scelte su questo layer incidono su:

- Correttezza e consistenza delle soluzioni realizzate;
- qualità delle prestazioni;
- gestione del rischio;
- trasparenza e spiegabilità;
- conformità normativa (es. AI Act e Legge 132/2025).

	Descrizione
Funzione	Comprende modelli di IA, framework di machine learning, pipeline di addestramento, strumenti per fine-tuning, valutazione, monitoraggio, orchestrazione dei modelli e inferenza.
Rilevanza per la PA	Questo livello rappresenta il "cuore logico" dell'IA: qui si sviluppano, addestrano, usano, monitorano e aggiornano i modelli che abilitano le funzionalità intelligenti delle applicazioni.

Componenti tipiche	Framework ML/DL, modelli fondamentali, dataset, strumenti e piattaforme di MLOps, sistemi di versioning di modelli e dati, strumenti di explainability e auditing, LLM, strumenti per addestramento distribuito.
Punti di attenzione per la PA	La PA deve considerare sia modelli sviluppati internamente sia modelli forniti da terzi, inclusi modelli general purpose. Le scelte su questo layer incidono su: • Correttezza e consistenza delle soluzioni realizzate; • qualità delle prestazioni; • gestione del rischio; • trasparenza e spiegabilità; • conformità normativa (es. AI Act e Legge 132/2025)

3.1.5 Application Layer (Livello Applicativo)

L'Application Layer raggruppa applicazioni, servizi, interfacce e sistemi che integrano modelli IA per fornire funzionalità intelligenti a cittadini, operatori, dipendenti pubblici e imprese.

È rilevante per la PA perché è il livello in cui il valore dell'IA diventa tangibile: abilita automazione, supporto decisionale, analisi avanzate e servizi digitali intelligenti.

Componenti tipiche sono: API, applicazioni AI-based, soluzioni verticali per la PA, assistenti intelligenti, sistemi di automazione, dashboard e strumenti analitici. Rientrano anche assistenti conversazionali, sistemi di forecasting, strumenti per analisi predittiva, servizi digitali potenziati da IA.

In termini di punti di attenzione, questo layer deve essere progettato secondo i principi di centralità dell'utente, accessibilità, trasparenza e accountability.

	Descrizione

Funzione	Raggruppa applicazioni, servizi, interfacce e sistemi che integrano modelli IA per fornire funzionalità intelligenti a cittadini, operatori, dipendenti pubblici e imprese.
Rilevanza per la PA	È il livello in cui il valore dell'IA diventa tangibile: abilita automazione, supporto decisionale, analisi avanzate e servizi digitali intelligenti.
Componenti tipiche	API, applicazioni AI-based, soluzioni verticali per la PA, assistenti intelligenti, sistemi di automazione, dashboard e strumenti analitici. Assistenti conversazionali, sistemi di forecasting, strumenti per analisi predittiva, servizi digitali potenziati da IA.
Punti di attenzione per la PA	Questo layer deve essere progettato secondo i principi di centralità dell'utente, accessibilità, trasparenza e accountability.

3.2 Modello di Architettura IA di riferimento per la PA

L'intelligenza artificiale sta evolvendo da modelli finalizzati all'espletamento di compiti passivi, come rispondere a domande, tradurre testi, generare immagini, verso una nuova classe di prodotti software, basati su *architettura agentica*.

Per *architettura agentica* si intende un framework di progettazione di sistemi di Intelligenza Artificiale in grado di trasformare modelli come gli LLM in componenti software dotati di autonomia operativa, se pur limitata e orientata a un obiettivo. Tali componenti o entità software (o hardware), denominati *agenti*, devono essere in grado di: acquisire informazioni dall'ambiente operativo; mantenere e aggiornare uno stato interno; definire o scomporre un obiettivo (task operativo) in attività elementari; pianificare ed eseguire azioni, in funzione dello stato del modello; interagire con risorse esterne (API, basi dati, servizi digitali cui sono connessi).

L'*architettura agentica* deve prevedere un ciclo iterativo di feedback continuo, basato su tre step: osservazione, decisione e azione. Tale ciclo deve essere supportato da meccanismi di controllo, verifica e feedback, tali da garantire il rispetto dei limiti di autonomia definiti, la tracciabilità delle decisioni, la coerenza degli output e la possibilità di interrompere, modificare o correggere l'esecuzione dell'agente in qualsiasi momento, nel rispetto delle soglie di sicurezza, dei vincoli funzionali e dei requisiti normativi applicabili.

Gli *agenti* di una architettura con queste caratteristiche gestiscono compiti complessi e multi-step, anche senza la supervisione umana continua, secondo una classificazione di autonomia che va da L0 a L5:

- Livello 0 - Completamente manuale;
- Livello 1 - Automazione basata su regole;
- Livello 2 - Automazione intelligente dei processi;
- Livello 3 - Workflow agentici;
- Livello 4 - Agenti semi-autonomi;
- Livello 5 - Agenti completamente autonomi.

I livelli sono descritti a seguire e analogamente sintetizzati dalla Tabella 2.

Nel Livello 0 - Completamente manuale, un agente non compie alcuna automazione e l'umano esegue ogni attività. Assumendo un'analogia con i veicoli, si consideri la guida manuale, senza alcuna assistenza. Dal punto di vista della tecnologia, si tratta di strumenti digitali di base, come ad esempio fogli di calcolo ed email, ad elaborazione manuale. Applicazioni si trovano in flussi di lavoro basati su carta/email, inserimento manuale dei dati, operazioni su fogli di calcolo.

Nel Livello 1 - Automazione basata su regole, l'automazione semplice segue regole fisse, come ad esempio Robotic Process Automation (RPA) o script. Assumendo un'analogia con i veicoli, si consideri il cruise control di base attivo, ad esempio per il mantenimento della velocità. A livello di tecnologia, si tratta di RPA, script, motori di regole. Applicazioni si trovano nell' instradamento delle email, nello StraightThrough Processing (STP) dei pagamenti con elaborazione automatica end-to-end senza intervento umano, nei motori di regole antifrode.

Nel Livello 2 - Automazione intelligente dei processi, un agente ha sia capacità di automazione sia capacità cognitive (Machine Learning - ML, Natural Language Processing - NLP, Computer Vision -

CV) con orchestrazione. In analogia con i veicoli, si consideri il sistema ADAS - Advanced Driver Assistance System, che gestisce velocità e sterzo con supervisione. Dal punto di vista della tecnologia, si tratta di ML, NLP, CV, RPA, orchestrazione di processi. Applicazioni si trovano nell'estrazione delle fatture passive, nel triage dei reclami, nell'assistenza al contact center.

Nel Livello 3 - Workflow agentici, gli agenti pianificano, ragionano, creano contenuti e si adattano all'interno di domini definiti. In analogia con i veicoli, si consideri la navigazione automatica in autostrada, in cui l'essere umano gestisce i casi limite. Dal punto di vista della tecnologia, si tratta di LLM, sistemi di memoria, uso di tool, Reinforcement Learning (RL) di base. Applicazioni si trovano in copilot (supporto/codice/marketing), analisti RAG (Retrieval-Augmented Generation), ETL (Extract, Transform, Load) automatizzati, ambienti di esercizio supervisionati, progetti pilota.

Nel Livello 4 - Agenti semi-autonomi, gli agenti agiscono in modo autonomo entro ambiti di competenza delimitati, adattano le strategie e apprendono. In analogia con i veicoli, si considerino i casi in cui la guida avviene in modalità autonoma in condizioni specifiche. Dal punto di vista della tecnologia, si tratta di attività di ragionamento e pianificazione avanzati, adattamento in tempo reale, ragionamento causale. Applicazioni si trovano nei taxi senza conducente, nella robotica di magazzino, nei droni per ispezioni, negli auto-remediation AIOps, negli ambienti di esercizio limitati in ambienti vincolati.

Nel livello 5 - Agenti completamente autonomi, l'agente opera con un apprendimento cross-dominio e un auto-adattamento senza intervento umano. In analogia con i veicoli, si considerino delle auto completamente autonome, che guidano ovunque e in tutte le condizioni. A livello di tecnologia, si tratta di sistemi di memoria sofisticati, meccanismi di apprendimento avanzati, safety automation. Al momento non è possibile indicare alcuna sperimentazione, ma solo attività di ricerca.

Tabella 2: Livelli di autonomia in una architettura agentica[1]

Livello	Cosa fa un agente	Analogia con i veicoli	Tecnologia	Applicazioni
Livello 0 completamente manuale	Nessuna automazione, l'umano esegue ogni attività	Guida manuale, nessuna assistenza	Strumenti digitali di base (fogli di calcolo, email),	Flussi di lavoro basati su carta/email, inserimento manuale dei dati, operazioni su fogli di calcolo

[1] *The Agentic State* - Rethinking Government for the Era of Agentic AI, Initiated and supported by Global Government Technology Centre Berlin and The World Bank

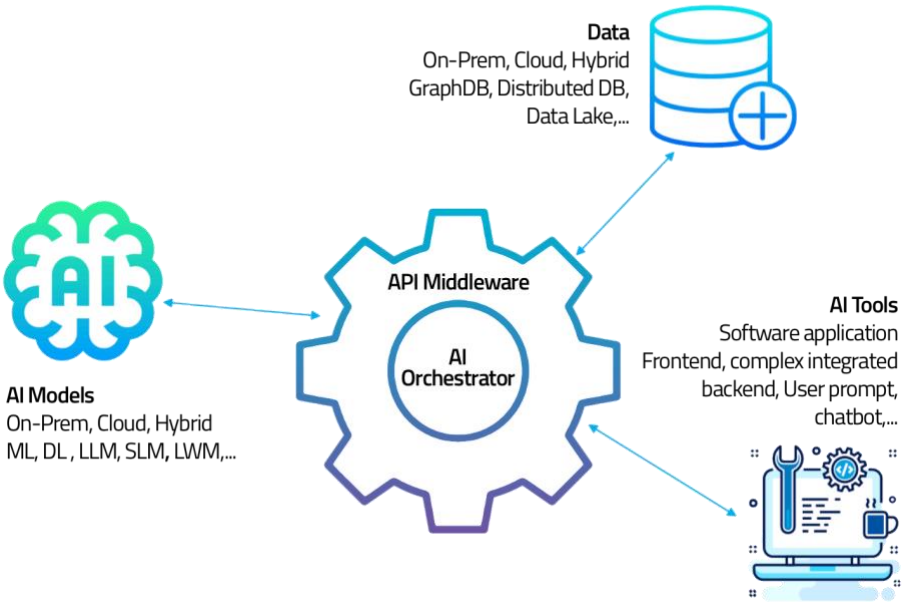
			elaborazione manuale	
Livello 1 Automazione basata su regole	L'automazione semplice segue regole fisse (Robotic Process Automation - RPA, script)	Cruise control di base attivo, es. mantenimento della velocità	RPA, script, motori di regole	Instradamento delle email, StraightThrough Processing - STP dei pagamenti con elaborazione automatica end-to-end senza intervento umano, motori di regole antifrode
Livello 2 Automazione intelligente dei processi	Automazione + capacità cognitive (Machine Learning, Natural Language Processing, Computer Vision) con orchestrazione	Il sistema ADAS (Advanced Driver Assistance System) gestisce velocità e sterzo con supervisione	ML, NLP, CV, RPA, orchestrazione di processi	Estrazione delle fatture passive, triage dei reclami, assistenza al contact center
Livello 3 Workflow agentici	Gli agenti pianificano, ragionano, creano contenuti e si adattano all'interno di domini definiti	Navigazione automatica in autostrada; l'essere umano gestisce i casi limite	LLM, sistemi di memoria, uso di tool, Reinforcement Learning (RL) di base	Copilot (supporto/codice/marketing), analisti RAG, ETL (Extract, Transform, Load) automatizzati. Ambienti di esercizio supervisionati, progetti pilota
Livello 4 Agenti semi-autonomi	Gli agenti agiscono in modo autonomo entro ambiti di competenza delimitati; adattano le strategie e apprendono	La guida avviene in modalità autonoma in condizioni specifiche	Ragionamento e pianificazione avanzati, adattamento in tempo reale, ragionamento causale	Taxi senza conducente, robotica di magazzino, droni per ispezioni, auto-remediation AIOps. Ambienti di esercizio limitati in ambienti vincolati
Livello 5 Agenti completamente autonomi	Apprendimento cross-dominio e auto-adattamento senza intervento umano	Le auto completamente autonome guidano ovunque e in tutte le condizioni	Sistemi di memoria sofisticati, meccanismi di apprendimento avanzati, safety automation	Nessuna sperimentazione, solo attività di ricerca

L'architettura di riferimento è composta da un "orchestratore IA" e da tre estensioni rappresentanti rispettivamente la molteplicità dei modelli IA, l'insieme delle fonti dati eterogenee per l'addestramento,

il tuning e le tecniche di inferenza dei modelli IA e l'insieme di tool che permettono all'orchestratore di risolvere il fabbisogno o lo *use case* specifico, ovvero il livello applicativo dello Stack IA.

Tale architettura è illustrata nella 3 e analogamente descritta a seguire.

Figura 3 - Architettura logica di riferimento



3La rappresentazione di 3 è riferita ad una architettura logica, non sono quindi indicate le caratteristiche tecniche dell'architettura fisica sottostante, che vanno altresì valutate e correttamente dimensionate, come ad esempio: la capacità computazionale, la memoria di lavoro, la latenza di trasmissione, le specifiche degli apparati di rete, etc. Il modello di riferimento che la Pubblica amministrazione DEVE prendere a riferimento per lo sviluppo di sistemi e servizi basati sull'uso di IA è rappresentato dalla 3.

L'architettura proposta si basa su un modello centrato su caratteristiche di *orchestrazione intelligente*, nel quale un componente denominato AI Orchestrator svolge il ruolo di nucleo logico di coordinamento. Tale componente è incapsulato all'interno di un API Middleware, in grado di conferire caratteristiche di *astrazione, interoperabilità e uniformità di accesso* verso tre domini eterogenei:

1. Modelli di IA (AI Models)
2. Sistemi e sorgenti dati (Data)
3. Strumenti applicativi e tool operativi (AI Tools)

Al fine di evitare il rischio di *lock-in*, l'API Middleware DEVE essere tecnicamente realizzato mediante un *abstraction layer*.

In particolare, l'API Middleware DEVE implementare un'interfaccia di inferenza basata su standard di mercato aperti e documentati. Tale scelta risponde ai requisiti di neutralità tecnologica del ModI (ovvero il Modello di Interoperabilità della Pubblica Amministrazione che individua tecnologie e standard che le Pubbliche Amministrazioni DEVONO considerare per la realizzazione dei propri sistemi informatici) e facilita l'utilizzo della molteplicità di modelli di IA, senza impatti sull'orchestratore centralizzato.

Questa struttura abilita un ecosistema *modulare*, *scalabile* e *governabile*, idoneo a supportare le Pubbliche Amministrazioni nello sviluppo di servizi e sistemi sia tramite architetture fisiche dislocate on--prem sia attraverso ambienti cloud o ibridi.

Al centro dell'architettura si colloca l'Orchestratore IA, un sistema di coordinamento che funge da punto di controllo unificato per tre componenti fondamentali dell'ecosistema IA della PA.

Questa architettura centralizzata ha dei vantaggi sia in termini di governance, sia in termini di verifica della conformità tecnica e normativa delle componenti facenti parte dell'architettura stessa. Tra i vantaggi si possono citare aspetti di efficienza operativa, interoperabilità e scalabilità delle soluzioni tecnologiche scelte per implementare soluzioni innovative.

Come rappresentato in figura, l'orchestratore è connesso a tre estensioni rappresentanti rispettivamente l'universo dei modelli IA, l'insieme delle fonti dati eterogenee per l'addestramento, il tuning e l'esercizio dei modelli e l'insieme di tool che permettono all'orchestratore di risolvere il fabbisogno o lo use case specifico. La rappresentazione seguente è riferita ad una architettura logica, non sono quindi indicate le caratteristiche dell'architettura fisica sottostante, che vanno altresì valutate e correttamente dimensionate. Ad esempio, si possono citare alcune dimensioni di analisi, come: la capacità computazionale, la memoria di lavoro, la latenza di trasmissione, etc.

3.2.1 Modelli di IA

Oltre alle categorie generali di modelli basati su Machine Learning (ML), Deep Learning (DL), utilizzato per analisi predittive e pattern recognition e Generative AI (GenAI), principalmente utilizzata per creazione di contenuti, documenti e comunicazioni, la tassonomia dei modelli di intelligenza artificiale può essere espansa per includere sia architetture specifiche che paradigmi di apprendimento specifici, come ad esempio Reinforcement Learning (RL) o Federated Learning (FL).

In generale, è utile evidenziare la differenza tra due macro-categorie: Predictive AI e Generative AI.

I modelli di Predictive AI vengono realizzati da zero (o quasi) sulla disponibilità dei dati del dominio specifico, per questo motivo le attività di Data Governance assumono un ruolo cruciale. Se i dati di training non sono ottimizzati, il modello può risultare inaffidabile. Per quanto riguarda i requisiti hardware, si deve considerare che il training avviene su grandi dataset in modalità *batch*. Se vengono utilizzate implementazioni software del metodo Random Forest, ad esempio, potrebbe essere sufficiente disporre di potenti CPU per la fase di training. Se si applicano, invece, tecniche di Deep Learning, l'approvvigionamento di GPU diventa indispensabile per accelerare i calcoli tensoriali, altrimenti troppo dispendiosi in termini temporali.

I modelli di Generative AI, come noto, sono in grado di rispondere a domande aperte o creare contenuti a partire da una base di conoscenza. La complessità computazionale ed i costi di approvvigionamento richiesti per addestrare modelli generativi, hanno condotto verso l'utilizzo dei cosiddetti *Foundational Models*, ovvero modelli pre-addestrati disponibili sul mercato. Con questa categoria di modelli, si possono fare attività di fine-tuning e inferenza: la prima (fine-tuning) non richiede di ri-addestrare tutto il modello, ma solo la sua caratterizzazione secondo lo specifico dominio applicativo, anche con tecniche PEFT (Performance-Efficient Fine-Tuning) e LORA (Low Rank Adaptation); la seconda (l'inferenza) richiede, invece, una elevata capacità computazionale (GPU) e una grande disponibilità di memoria delle GPU. Nonostante sia indispensabile anche in questo caso la disponibilità di GPU, si può optare per una capacità computazionale distribuita su più unità di processamento, ciascuna di minore capacità rispetto a quella necessaria per le attività di inferenza.

Tra le tipologie di modelli che possono essere gestite dall'orchestratore alla base dell'architettura si possono inoltre citare le tipologie open source, open weight, proprietari, oltre ovviamente agli specifici Small Language Model (SLM), per applicazioni verticali ed efficienti, modelli altamente specializzati, oppure modelli Large World Model (LWM), una classe avanzata di modelli di intelligenza artificiale

progettata per andare oltre i limiti dei tradizionali modelli linguistici di grandi dimensioni (LLM), che si concentra sulla comprensione e simulazione delle dinamiche del mondo reale.

L'aggiunta dello strato di middleware, nello scenario finora descritto, può fornire una serie di vantaggi, come ad esempio:

1. In caso di Predictive AI: Il middleware può interfacciarsi con la base di conoscenza e server on-prem e inviare all'orchestratore solo l'esito dell'elaborazione del modello;
2. In caso di Generative AI: Il middleware potrebbe gestire le chiamate verso GPU in cloud (per scalare i costi dell'inferenza) o verso server locali, in caso i dati siano da considerarsi riservati e non esponibili in cloud;

In altre parole, il middleware aggiunto attorno all'Orchestratore consente di interfacciarsi in maniera agile ed in base alle esigenze dello use case specifico, verso un modello più o meno oneroso dal punto di vista computazionale.

In una architettura estremamente dinamica per definizione, i modelli di IA possono essere usati invocando servizi cloud (massima scalabilità e accesso a modelli all'avanguardia), oppure mediante un deploy di una istanza nel perimetro infrastrutturale della PA, in modalità training, fine tuning o esercizio (sovranità dei dati e controllo totale per informazioni sensibili), oppure mediante un deploy ibrido (bilanciamento ottimale tra prestazioni, costi e sicurezza).

3.2.2 Fonti dati eterogenee (cloud, on-prem, hybrid)

I dati utilizzati nell'architettura descritta possono essere di diverse tipologie e strutture: basi di conoscenza, data lake, basi dati semantiche o graphDB, possono essere distribuite, centralizzate, con storage on-prem, in cloud o in modalità ibrida. Si possono ad esempio considerare Database distribuiti per dati transazionali e operativi, Graph Database per rappresentare relazioni complesse tra entità (cittadini, procedimenti, organizzazioni esterne), Data Lake per rappresentare archivi storici e basi di conoscenza su cui implementare tecniche di data analytics avanzate.

Per quanto riguarda il deployment, possono essere previsti Cloud storage, per flessibilità e costi variabili, on-prem, per ottenere un controllo completo e conformità massima, hybrid per

ottimizzazione tra sicurezza e accessibilità, mixed per una combinazione strategica per esigenze specifiche.

Alimentando la molteplicità dei modelli sopra citati, i dati possono essere utilizzati per attività di training di modelli propri, per il tuning di modelli utilizzati mediante servizi cloud o per l'esercizio di modelli già in produzione. L'accesso stesso ai dati, inoltre, può avvenire tramite strumenti come Retrieval-Augmented Generation (RAG), API esterne o altre fonti in cloud e on-prem.

Oltre alla disponibilità dei dati, secondo le modalità di accesso e le tecnologie sopra citate, le Pubbliche amministrazioni DEVONO avviare attività volte a creare una catena del valore del dato, attuabile mediante strumenti di Data Engineering o una pipeline del dato, finalizzata ad attuare tecniche di data quality articolate e specializzate in funzione delle caratteristiche e degli step propri delle basi di conoscenza, tra cui si possono citare le seguenti attività: Data Discovery, Normalization, Governance, Anonymization, Enrichment, Chunking, Indexing, Versioning, QA, Retrieval Tuning, Publishing, Monitoring, Human Validation.

Per **Data Discovery** si intende l'analisi preliminare per individuare struttura, qualità e caratteristiche dei dati.

Per **Normalization** si intende l'armonizzazione di formati, schemi e valori per rendere i dati confrontabili.

Per **Governance** si intende la definizione di regole, policy, metadati e controlli sul ciclo di vita del dato.

Per **Anonymization** si intende la rimozione o trasformazione di dati sensibili per tutelare privacy e compliance.

Per **Enrichment** si intende l'arricchimento semantico tramite tagging, estrazione di entità e metadati.

Per **Chunking** si intende la suddivisione dei documenti in segmenti gestibili e indicizzabili.

Per **Indexing** si intende la creazione degli indici (testuali, vettoriali, ibridi) per il retrieval efficiente.

Per **Versioning** si intende la gestione delle versioni dei dati e delle trasformazioni nel tempo.

Per **QA (Quality Assurance)** si intende la verifica automatica della qualità e consistenza dei dati processati.

Per **Retrieval Tuning** si intende l'ottimizzazione dei parametri di ricerca, ranking e combinazione degli indici.

Per **Publishing** si intende la messa a disposizione dei dati tramite API, endpoint o servizi di consultazione.

Per **Monitoring** si intende il controllo continuo di performance, drift, anomalie e utilizzo.

Per **Human Validation** si intende la supervisione umana (ove prevista) per verificare la correttezza o la rilevanza dei contenuti.

Tabella 3: Possibili fasi del Data Engineering

Fase	Descrizione
Data Discovery	Analisi preliminare per individuare struttura, qualità e caratteristiche dei dati
Normalization	Armonizzazione di formati, schemi e valori per rendere i dati confrontabili
Governance	Definizione di regole, policy, metadati e controlli sul ciclo di vita del dato
Anonymization	Rimozione o trasformazione di dati sensibili per tutelare privacy e compliance
Enrichment	Arricchimento semantico tramite tagging, estrazione di entità e metadati
Chunking	Suddivisione dei documenti in segmenti gestibili e indicizzabili
Indexing	Creazione degli indici (testuali, vettoriali, ibridi) per il retrieval efficiente
Versioning	Gestione delle versioni dei dati e delle trasformazioni nel tempo
QA (Quality Assurance)	Verifica automatica della qualità e consistenza dei dati processati
Retrieval Tuning	Ottimizzazione dei parametri di ricerca, ranking e combinazione degli indici
Publishing	Messa a disposizione dei dati tramite API, endpoint o servizi di consultazione
Monitoring	Controllo continuo di performance, drift, anomalie e utilizzo
Human Validation	Supervisione umana (ove prevista) per verificare la correttezza o la rilevanza dei contenuti

3.2.3 Tool del livello applicativo

In questa categoria sono rappresentati tutti gli strumenti software e le interfacce attraverso cui utenti finali (cittadini, imprese, operatori PA) interagiscono con i servizi sviluppati anche attraverso tecnologia IA. I tool permettono di invocare API, eseguire codice, task semplici o articolati come possono essere ad esempio quelli di trasformazione dei dati (formato, dimensione, etc.) e quelli per la raccolta automatica e intelligente di nuovi dati da fonti eterogenee. I tool possono essere soluzioni sviluppate ad hoc o soluzioni open source messe in riuso. I tool possono ricomprendere ad esempio prompt e UI per la realizzazione di frontend dedicati all'utente finale, qualora il servizio atomico sia da rendere direttamente all'utente finale, oppure essere parte di una architettura di backend più articolata qualora lo use case richieda una soluzione di interoperabilità tra più servizi applicativi.

Tra le componenti principali si possono individuare: frontend application per portali web e app mobile per cittadini, backend application come API e microservizi per integrazioni con altri sistemi, complex backend application, come sistemi di workflow e process automation e assistenti IA e chatbot per interfacce conversazionali per supporto H24 e caratteristiche di multicanalità (web, mobile, voice, chat).

3.2.4 Orchestratore

L'orchestratore è l'elemento abilitante la dinamicità dell'architettura. Attraverso questo strumento si possono creare istanze di architetture di topologie e tipologie diverse al fine di soddisfare requisiti differenti e scenari d'uso differenti.

L'AI Orchestrator può essere rappresentato come un motore decisionale centrale, responsabile della logica di coordinamento, selezione del modello opportuno nella molteplicità di quelli disponibili, orchestrando operazioni di inferenza e gestione dei flussi informativi tra modelli, dati e strumenti (Tool).

In questa architettura, il middleware API costituisce lo strato di interfaccia e integrazione, progettato per esporre API open, ovvero indipendenti dalle specificità dei sistemi sottostanti. Questo strato offre un *gateway* unificato con caratteristiche di:

- Astrazione dei protocolli, fornendo capacità di interoperabilità tra formati eterogenei;

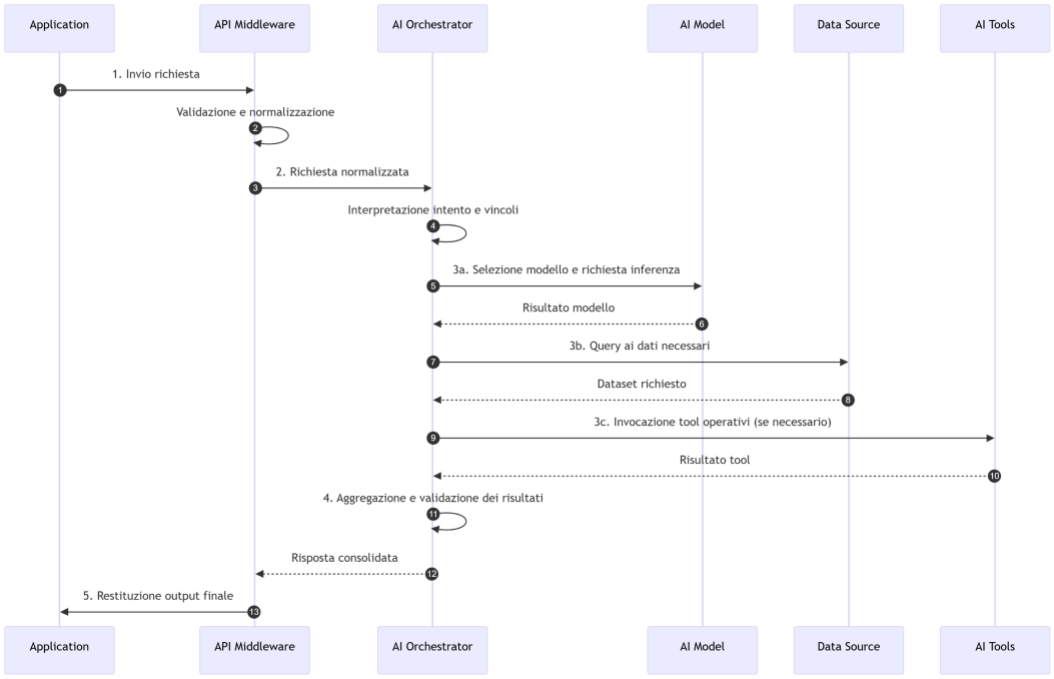
- Gestione di sicurezza e compliance, validando le richieste di attivazione di Modelli IA e garantendone caratteristiche di tracciabilità;
- Selezione dinamica delle sorgenti dati in funzione dello use-case o del tipo di interrogazione.

L'orchestratore offre la capacità di selezionare la combinazione di modelli, dati e tools in grado di individuare l'istanza di architettura fisica più sostenibile dal punto di vista economico, ambientale e sociale.

Di seguito la descrizione di un esempio di utilizzo da parte di una PA dell'architettura logica descritta:

3.2.4.1 Caso d'uso: Erogazione bonus - esempio di uso dell'architettura logica per le funzioni della chatbot per il cittadino

Figura 4 - Sequence diagram - Use case richiesta bonus



4

Il *sequence diagram* proposto in 4 descrive la sequenza di interazioni tra le componenti dell'Architettura logica di riferimento nella realizzazione di un semplice caso d'uso di esempio che preveda la richiesta di un cittadino circa la verifica del possesso dei requisiti per l'erogazione di un bonus.

L'interazione tra il cittadino e lo strato di application fa scaturire una richiesta verso l'API Middleware, il quale controlla e normalizza la richiesta per passarla successivamente all'Orchestratore.

In questa fase si avvia il processo di istanziazione della topologia di architettura fisica e di connessione tra le varie componenti, ovvero:

- L'Orchestratore, attraverso funzioni di routing intelligente, sceglie il modello IA più adatto da utilizzare per soddisfare la richiesta dell'utente;
- Sceglie quale storage di dati utilizzare e con quali ulteriori tool creare una interconnessione (se necessari).

Il routing della fase precedente consente alle componenti selezionate (modello, dati, tool) di elaborare ed effettuare le operazioni richieste.

L'Orchestratore colleziona e verifica i risultati e restituisce allo strato applicativo (chatbot cittadino) l'esito, mediante lo strato di Middleware.

3.3 Possibili dimensioni di analisi del modello di Architettura logica

La scelta della tipologia, topologia e configurazione degli strumenti tecnologici (modelli IA, dati per inferenza o training, tool applicativi, servizi di orchestrazione e middleware) è determinata da un insieme complesso di variabili progettuali e fisiche, di requisiti funzionali e non funzionali.

Tali variabili concorrono alla definizione dell'architettura target, guidando decisioni relative a deployment, governance, sicurezza, integrazione e scalabilità.

Esempi delle principali dimensioni di analisi che dovrebbero guidare le scelte architetturali di una Pubblica Amministrazione sono:

- Requisiti prestazionali;
- Topologia e connettività;
- Vincoli economici e strategia finanziaria;
- Natura dei dati;
- Sicurezza e compliance;
- Capacità computazionale.

Per ognuno di questi ambiti di impatto si dettagliano a seguire le dimensioni di analisi con riferimento alle scelte architetturali.

Requisiti prestazionali

- Latenza: il sistema è basato su interazioni realtime, near realtime, batch, modalità asincrone o ibride.
- SLA concordati in fase di design: disponibilità, throughput, time-to-market, tempo massimo di elaborazione.

- Variabilità del carico: il sistema prevede dei picchi prevedibili/non prevedibili di utenti, prevede utilizzo di meccanismi di autoscaling.

Topologia e connettività

- Topologie distribuite: dipendenza dalla banda, qualità/affidabilità del collegamento, latenza inter-dominio, presenza di segmentazioni (Demilitarized Zone - DMZ, trust zone, reti interne PA).
- Località del dato e del modello: scelta on--prem vs Edge vs cluster interni in base a vincoli di sicurezza, riservatezza, costi e prestazioni.

Vincoli economici e strategia finanziaria

- CapEx (conto capitale) vs OpEx (spesa corrente): disponibilità di budget per investimenti iniziali e/o per servizi ricorrenti.
- Analisi TCO (Total Cost of Ownership): costi di energia, hardware, storage, rete, licenze, personale, manutenzione, upgrade, costo token per inferenze.
- ROI atteso: benefici quantificabili in produttività, automazione, riduzione errori e tempi operativi.
- Costi di migrazione: costi di porting, modernizzazione applicativa, integrazione con sistemi esistenti.

Natura dei dati

- Classificazione del dato per addestramento, fine-tuning, inferenza: dati aperti, dati personali, dati sensibili, dati giudiziari o appartenenti a categorie speciali.
- Pattern e modalità di accesso ai dati: letture intensive, scritture massive, query complesse, esigenze di low-l-atency, addestramento, fine-tuning, inferenza.
- Formati e volumi: dataset strutturati, semi-strutturati, non strutturati, API streaming.

Sicurezza e compliance

- Restrizioni architetturali: limitazioni sull'esposizione dei servizi, vincoli su DMZ, necessità di modulare la topologia di rete.

- Policy PA e best practice: crittografia end-to-end, gestione di *secret* e chiavi, audit e tracciabilità completa, dove per "secret" si intende qualunque informazione sensibile utilizzata per autenticare, autorizzare o proteggere l'accesso ad un sistema informatico.

Capacità computazionale

- Disponibilità di risorse: CPU (Central Processing Unit), GPU (Graphics Processing Unit), TPU (Tensor Processing Unit), acceleratori dedicati, in funzione delle attività da svolgere (addestramento, fine-tuning, inferenza).
- Tradeoff tra modelli: valutazione di modelli più piccoli e rapidi (Edge AI, SLM) rispetto a modelli più grandi e costosi.
- Possibilità di scalabilità orizzontale: aggiunta dinamica di nodi, bilanciamento del carico, cluster IA.

Ambito di impatto	Dimensione di analisi per scelte architetturali
Requisiti prestazionali	Latenza: il sistema è basato su interazioni realtime, near realtime, batch, modalità asincrone o ibride SLA concordati in fase di design: disponibilità, throughput, time-to---market, tempo massimo di elaborazione Variabilità del carico: il sistema prevede dei picchi prevedibili/non prevedibili di utenti, prevede utilizzo di meccanismi di autoscaling
Topologia e connettività	Topologie distribuite: dipendenza dalla banda, qualità/affidabilità del collegamento, latenza inter-dominio, presenza di segmentazioni (Demilitarized Zone - DMZ, trust zone, reti interne PA) Località del dato e del modello: scelta on-prem vs edge vs cluster interni in base a vincoli di sicurezza, riservatezza, costi e prestazioni
Vincoli economici e	CapEx (conto capitale) vs OpEx (spesa corrente): disponibilità di budget per investimenti iniziali e/o per servizi ricorrenti

strategia finanziaria	Analisi TCO (Total Cost of Ownership): costi di energia, hardware, storage, rete, licenze, personale, manutenzione, upgrade, costo token per inferenze ROI atteso: benefici quantificabili in produttività, automazione, riduzione errori e tempi operativi Costi di migrazione: costi di porting, modernizzazione applicativa, integrazione con sistemi esistenti
Natura dei dati	Classificazione del dato per addestramento, fine-tuning, inferenza: dati aperti, dati personali, dati sensibili, dati giudiziari o appartenenti a categorie speciali. Pattern e modalità di accesso ai dati: letture intensive, scritture massive, query complesse, esigenze di low-latency, addestramento, fine-tuning, inferenza Formati e volumi: dataset strutturati, semistrutturati, non strutturati, API streaming
Sicurezza e compliance	Restrizioni architetturali: limitazioni sull'esposizione dei servizi, vincoli su DMZ, necessità di modulare la topologia di rete Policy PA e best practice: crittografia end-to-end, gestione di *secret* e chiavi, audit e tracciabilità completa, dove per *secret* si intende qualunque informazione sensibile utilizzata per autenticare, autorizzare o proteggere l'accesso ad un sistema informatico
Capacità computazionale	Disponibilità di risorse: CPU (Central Processing Unit), GPU (Graphics Processing Unit), TPU (Tensor Processing Unit), acceleratori dedicati, in funzione delle attività da svolgere (addestramento, fine-tuning, inferenza) Tradeoff tra modelli: valutazione di modelli più piccoli e rapidi (Edge AI, SLM) vs modelli più grandi e costosi. Possibilità di scalabilità orizzontale: aggiunta dinamica di nodi, bilanciamento del carico, cluster IA

3.4 Ruolo del dato nello sviluppo dei sistemi di IA

Al fine di supportare le Pubbliche Amministrazioni nell'individuazione delle istanze architetturali da progettare, in coerenza con l'architettura logica di riferimento, risulta essenziale distinguere il ruolo che i dati assumono nelle diverse fasi del ciclo di vita di un sistema di Intelligenza Artificiale. I dati, infatti, non costituiscono un elemento omogeneo: la loro natura, le finalità d'uso e le relative

implicazioni tecniche e organizzative variano in modo significativo in relazione alla fase in cui intervengono.

È opportuno, in particolare, differenziare tra:

- **Training data**, ossia i dati utilizzati per l'addestramento o il fine-tuning dei modelli di IA. Essi sono impiegati nella fase di sviluppo del sistema e richiedono specifici requisiti di qualità, rappresentatività, gestione del bias, tracciabilità delle trasformazioni e conformità alle norme in materia di protezione dei dati e proprietà intellettuale.
- **Operational data**, ossia i dati trattati durante l'esercizio del modello di IA, nella fase di runtime del sistema (es. inferenza). Questi includono sia gli input forniti dagli utenti o dai sistemi esterni sia i dati generati dal modello durante l'esecuzione. La loro gestione impatta direttamente su aspetti di sicurezza, monitoraggio delle prestazioni, auditabilità, logging e gestione degli incidenti.

La distinzione tra *training data* e *operational data* consente di definire con chiarezza le responsabilità, gli elementi architetturali necessari e le misure di governance richieste, nonché di garantire l'aderenza ai requisiti normativi applicabili lungo l'intero ciclo di vita del sistema di IA.

3.4.1 Dati di Addestramento (Training)

I dati di addestramento sono utilizzati per creare, addestrare e validare i modelli di IA prima che questi vengano messi in esercizio. La fase di training, nel ciclo di vita di un sistema di IA, avviene tipicamente una volta ma può anche essere ripetuta in fase di aggiornamento del modello stesso. I dati utilizzati per il training godono di caratteristiche particolari e generalmente appartengono a determinate categorie, ad esempio:

- Appartengono a Dataset pubblici e standardizzati;
- Sono dati storici della PA (se necessario pseudo-anonimizzati o anonimizzati se personali);
- Appartengono a Dataset sintetici, realizzati appositamente per l'addestramento di uno specifico modello di IA;
- Provengono da banche dati acquisite da fornitori esterni.

Queste categorie di dati possono essere qualificate secondo determinati requisiti:

- Volume: Grande quantità di dati (miliardi/trilioni di token);
- Varietà: Massima diversificazione per garantire una sufficiente generalizzazione del modello;
- Qualità: Dati etichettati, verificati, bilanciati;
- Rappresentatività: Dati che coprono il maggior numero di casi d'uso previsti, riproducendo la distribuzione dei valori delle caratteristiche di interesse osservata nell'universo di riferimento

Rispetto all'architettura di riferimento, la Pubblica Amministrazione che si appresta a sviluppare soluzioni basate su IA, potrebbe decidere, per la fase di training del modello, di configurare l'istanza architetturale secondo tre opzioni di classificazione, che sono di seguito descritte e analogamente riportate in Tabella 4.

- **opzione 1 – Cloud (dati utilizzabili in cloud):** questa tipologia di deploy prevede infrastrutture scalabili per training intensivo, GPU/TPU in cluster per calcolo parallelo e storage ottimizzato per grandi volumi. Un esempio può essere rappresentato dal training di un chatbot generico su servizi PA;
- **opzione 2 - On-prem (per dati sensibili):** questa tipologia di deploy prevede controllo e sovranità dei dati; conformità by design al GDPR e un investimento in hardware dedicato. Un esempio può essere rappresentato dal training di un modello per l'analisi documenti riservati;
- **opzione 3 – Hybrid:** questa tipologia di deploy prevede un approccio ibrido, con dati non sensibili in cloud e dati sensibili on-prem. Il modello è quello del Federated Learning, cioè di un training distribuito senza condivisione dati; un esempio può essere rappresentato da un modello che impara da più enti senza centralizzare dati personali.

Tabella 4: Tipologie di deploy per il Training di Modelli di IA

Opzione 1 – Cloud (dati utilizzabili in cloud)	**Opzione 2 - On-prem (per dati sensibili)**	**Opzione 3 - Hybrid**
Infrastrutture scalabili per training intensivo	Controllo e sovranità dei dati Conformità GDPR by design	Dati non sensibili in cloud, sensibili on-prem

GPU/TPU cluster per calcolo parallelo Storage ottimizzato per grandi volumi Esempio: Training di un chatbot generico su servizi PA	Investimento in hardware dedicato **Esempio**: Training di modello per analisi documenti riservati	Federated Learning: training distribuito senza condivisione dati **Esempio**: Modello che impara da più enti senza centralizzare dati personali

Un possibile esempio di attività necessarie per addestrare un modello di IA prevede quattro fasi principali (5):

1. Raccolta e preparazione dei dati;
2. Processing dei dati;
3. Training del modello;
4. Produzione del modello.

La fase di **Raccolta e preparazione dei dati** è dedicata ad attività come:

1.1 Identificazione delle fonti dati (dataset interni della PA, open data, data provider certificati);
1.2 Raccolta ed integrazione di dati eterogenei;
1.3 Verifica dei requisiti legali (copyright, licenze, GDPR, Data Governance Act);
1.4 Produzione del *dataset di training* in formato strutturato.

La fase di **Processing dei dati** è dedicata ad attività come:

2.1 Data cleaning;
2.2 Normalizzazione e trasformazioni numeriche e/o testuali;
2.3 Feature engineering e selezione delle variabili rilevanti;
2.4 Data augmentation, ove applicabile (vision, audio, NLP);
2.5 Controlli di qualità e misure di mitigazione del bias;
2.6 Generazione dei dataset: training, validation, test.

La fase di **Training del modello** è dedicata ad attività come:

3.1 Selezione dell'architettura del modello ML, DL, LLM, SLM, GenAI);

3.2 Configurazione degli iperparametri;

3.3 Esecuzione iterativa del processo di addestramento;

3.4 Validazione su dataset separato (monitoraggio metriche, tuning, early stopping);

3.5 Valutazione finale su test set indipendente;

3.6 Documentazione dei risultati.

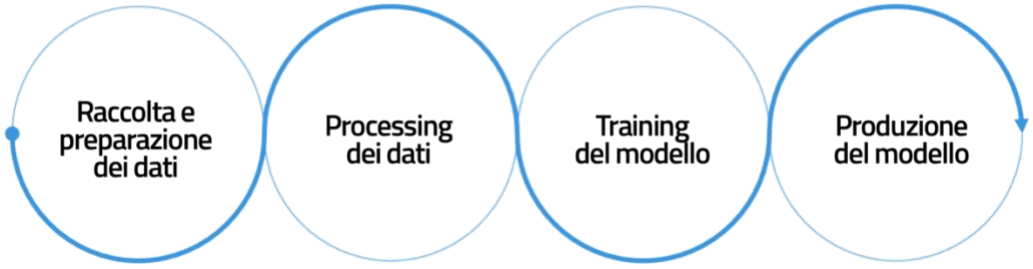
La fase di **Produzione del modello addestrato** è dedicata ad attività come:

4.1 Esportazione del modello in un formato distribuibile;

4.2 Applicazione di eventuali tecniche di ottimizzazione;

4.3 Preparazione per il deployment in ambienti On-prem, Cloud o ibridi.

Figura 5 - Esempio di processo di Training di un modello di IA



5

Per quanto concerne la necessità di capacità computazionali per la fase di training di un modello di IA, considerando l'architettura logica di riferimento (3), è possibile fare considerazioni a seguire (Tabella 5).

La fase di training richiede capacità computazionali significativamente superiori rispetto all'esercizio, poiché deve elaborare grandi volumi di dati e ottimizzare iterativamente i parametri del modello. In questo contesto le GPU – o, per modelli di grandi dimensioni, acceleratori specializzati come TPU o cluster multi-GPU – risultano essenziali per garantire tempi di addestramento sostenibili, grazie alla loro capacità di eseguire operazioni matriciali e tensoriale in parallelo.

Le CPU entrano invece in gioco per attività complementari come pre-processing, data loading, orchestrazione dei job e gestione delle pipeline distribuite. Nell'Architettura di riferimento, ciò implica la necessità di integrare ambienti di training scalabili (cloud o ibridi), in grado di supportare workload intensivi, distribuire il calcolo tra più nodi e garantire throughput elevato nel trasferimento dei dati tra lo strato Data e il componente AI Model durante le fasi iterative di addestramento.

Tabella 5: Capacità computazionale per le fasi di training di un modello IA

Training	Attività complementari
La fase di training richiede capacità computazionali significativamente superiori rispetto all'esercizio, poiché deve elaborare grandi volumi di dati e ottimizzare iterativamente i parametri del modello. In questo contesto le GPU – o, per modelli di grandi dimensioni, acceleratori specializzati come TPU o cluster multi-GPU – risultano essenziali per garantire tempi di addestramento sostenibili, grazie alla loro capacità di eseguire operazioni matriciali e tensoriale in parallelo.	Le CPU entrano invece in gioco per attività complementari come pre-processing, data loading, orchestrazione dei job e gestione delle pipeline distribuite. Nell'Architettura di riferimento, ciò implica la necessità di integrare ambienti di training scalabili (cloud o ibridi), in grado di supportare workload intensivi, distribuire il calcolo tra più nodi e garantire throughput elevato nel trasferimento dei dati tra lo strato Data e il componente AI Model durante le fasi iterative di addestramento.

3.4.2 Dati in Esercizio (Operational Data)

I dati operativi sono i dati trattati dal sistema di Intelligenza Artificiale nella fase di esercizio, durante il funzionamento quotidiano del modello già addestrato. Essi costituiscono gli input e i contesti informativi necessari affinché il sistema possa erogare servizi, supportare processi decisionali o interagire con cittadini, imprese ed operatori delle Pubbliche Amministrazioni.

A differenza dei dati di training, i dati operativi sono caratterizzati da maggiore dinamicità, variabilità e da requisiti stringenti in termini di sicurezza, tempestività e conformità normativa, in quanto direttamente coinvolti nell'erogazione di servizi pubblici.

I dati di esercizio di un modello di IA possono provenire da una pluralità di sorgenti eterogenee:

- Interazioni dirette con gli utenti: dati immessi da cittadini o operatori, come ad esempio: form online, chatbot, upload di documenti, ticketing, sportelli digitali;
- Sistemi gestionali e basi dati della PA: anagrafiche, registri, basi dati documentali, sistemi di protocollo, sistemi di gestione delle pratiche;
- Flussi provenienti da piattaforme storiche, servizi SOAP/REST, API interne;
- Fonti real-time, come ad esempio: sensori Internet of Thighs (IoT), dispositivi di monitoraggio territoriale.

L'utilizzo dei dati operativi impone requisiti tecnici e organizzativi generalmente più stringenti rispetto ai dati di training.

I requisiti di tempestività, continuità del servizio e capacità di processare flussi eterogenei di dati operativi hanno implicazioni dirette anche sulla configurazione delle risorse computazionali necessarie per sostenere l'esecuzione del modello di IA, come di seguito dettagliato (Tabella 6).

Nei contesti in cui l'inferenza deve avvenire in tempo reale o near-real-time, come ad esempio nei sistemi di assistenza automatica, analisi documentale istantanea o flussi IoT- in fase di training diventa necessario adottare infrastrutture dotate di GPU o acceleratori dedicati (TPU), particolarmente efficaci nell'elaborazione parallelizzata tipica dei modelli di deep learning e dei LLM.

Rispetto alle attività complementari, nei casi in cui i carichi di inferenza siano meno intensivi, più distribuiti nel tempo o basati su modelli più compatti (es. SLM), l'uso di CPU ad alte prestazioni risulta spesso sufficiente, soprattutto se integrate in architetture on-prem o ibride che devono garantire autonomia operativa, isolamento dei dati sensibili e controllo diretto delle risorse.

Tabella 6: Capacità computazionale per le fasi di esercizio di un modello IA

Training	Attività complementari
Nei contesti in cui l'inferenza deve avvenire in tempo reale o near-real-time, come ad esempio nei sistemi di assistenza automatica, analisi documentale istantanea o flussi	Nei casi in cui i carichi di inferenza siano meno intensivi, più distribuiti nel tempo o basati su modelli più compatti (es. SLM), l'uso di CPU ad alte prestazioni

IoT—diventa necessario adottare infrastrutture dotate di GPU o acceleratori dedicati (TPU), particolarmente efficaci nell'elaborazione parallelizzata tipica dei modelli di deep learning e dei LLM	risulta spesso sufficiente, soprattutto se integrate in architetture on-prem o ibride che devono garantire autonomia operativa, isolamento dei dati sensibili e controllo diretto delle risorse

All'interno dell'Architettura logica di riferimento (3), la scelta tra CPU, GPU o acceleratori incide su più componenti:

- **AI Model**: il tipo di hardware disponibile condiziona la latenza e la scalabilità dell'inferenza, influenzando la possibilità di adottare modelli complessi o versioni ottimizzate mediante modelli *quantized* e *distilled*. I *quantized model* possono essere descritti come modelli ottimizzati mediante *quantizzazione*, che riduce la precisione dei pesi (es. da 32-bit a 8-bit), diminuendo memoria e latenza senza degradare troppo le prestazioni. I *distilled model*, invece, sono modelli ottenuti tramite *distillazione*, in cui un modello più ampio e complesso trasferisce conoscenza a uno più snello e veloce, mantenendo però prestazioni comparabili e comunque accettabili.
- **API Middleware**: deve supportare il bilanciamento delle richieste verso nodi con differenti capacità computazionali, gestendo routing intelligente tra CPU e GPU a seconda dei carichi.
- **AI Orchestrator**: coordina il provisioning delle risorse, monitora lo stato degli accelerator node e applica politiche di scaling automatico per garantire continuità anche in presenza di picchi di traffico.
- **Data Layer**: requisiti di throughput elevato richiedono storage veloci e pipeline di ingestione ottimizzate per non rendere la CPU/GPU un collo di bottiglia.

Di seguito vengono illustrati alcuni scenari possibili con esempi di deploy nella fase di esercizio, differenziando per tipologia di dato e scenario d'utilizzo (

Tabella 7).

Scenario A - Dati Non Sensibili (Cloud)

- Tipologie di dati: dati pubblici, richieste informative, log anonimi, statistiche aggregate;
- Topologia/residenza: cloud pubblico/ibrido senza restrizioni particolari;
- Esempi di utilizzo nella PA: chatbot informativi, FAQ automatiche, servizi di assistenza.

Scenario B – Dati Personali

- Tipologie di dati: anagrafiche, documenti identificativi, dati amministrativi di base;
- Topologia/residenza: On-prem per dati personali; cloud per elaborazioni non sensibili;
- Esempi di utilizzo nella PA: Richiesta certificati, con dati personali on-prem e generazione di documenti in cloud.

Scenario C – Dati Sensibili (On-Premise)

- Tipologie di dati: dati sanitari, giudiziari, vulnerabilità sociali, informazioni protette da norme specialistiche;
- Topologia/residenza: interamente on-prem o su cloud privato PA con segregazione forte;
- Esempi di utilizzo nella PA: IA di supporto a pratiche sanitarie, valutazioni giudiziarie o istruttorie sensibili.

Scenario D – Data Lake Ibrido (Storico + Corrente)

- Tipologie di dati: dati storici aggregati, dataset anonimizzati, dati operativi giornalieri;
- Topologia / residenza: transazionale on-prem, con data lake in cloud per analytics;
- Esempi di utilizzo nella PA: analisi predittive pluriennali, pianificazione strategica basata su dati di lungo periodo.

Tabella 7: Tipologie di deploy per l'esercizio di Modelli di IA

Scenario	Tipologie di Dati	Topologia / Residenza	Esempi di Utilizzo nella PA
A – Dati Non Sensibili (Cloud)	Dati pubblici, richieste informative, log anonimi, statistiche aggregate.	Cloud pubblico/ibrido senza restrizioni particolari.	Chatbot informativi, FAQ automatiche, servizi di assistenza
B – Dati Personali Comuni (Hybrid)	Anagrafiche, documenti identificativi, dati amministrativi di base.	On-prem per dati personali; Cloud per elaborazioni non sensibili.	Richiesta certificati: dati personali on-prem; generazione documenti in cloud
C – Dati Sensibili (On-Premise)	Dati sanitari, giudiziari, vulnerabilità sociali, informazioni protette da norme specialistiche.	Interamente on-prem o su cloud privato PA con segregazione forte.	IA di supporto a pratiche sanitarie, valutazioni giudiziarie o istruttorie sensibili
D – Data Lake Ibrido (Storico + Corrente)	Storici aggregati, dataset anonimizzati, dati operativi giornalieri.	Transazionale on-prem + Data Lake cloud per analytics.	Analisi predittive pluriennali, pianificazione strategica basata su dati di lungo periodo

Un possibile esempio di attività di base per l'esercizio di modelli di IA prevede (6):

1. Input dell'utente o di sistema;
2. Validazione e processing;
3. Inferenza del modello già addestrato.

Durante la fase di **Input dell'utente o di sistema** le attività principali sono:

1.1 Ricezione dei dati operativi provenienti da utenti o da sistemi integrati. Gli input possono assumere forme differenti (testo, documenti, immagini) e vengono acquisiti attraverso interfacce applicative o API interne;

1.2 Esecuzione di controlli preliminari di integrità, formato e disponibilità del dato.

Le attività tipiche della fase di **Validazione e processing** sono:

2.1 Attività di verifica e trasformazioni necessarie per rendere i dati utilizzabili dal modello di IA;
2.2 Estrazione di feature rilevanti;
2.3 Filtraggio di eventuali elementi errati, duplicati o potenzialmente dannosi;
2.4 Applicazione di regole di sicurezza (ad esempio rimozione elementi non ammessi).

Nella fase di **Inferenza del modello già addestrato** il modello di IA elabora l'input validato applicando il proprio meccanismo inferenziale. A seconda del caso d'uso, l'inferenza può comportare la classificazione o la estrazione di informazioni; la generazione di testo o contenuti, supporto decisionale. In ogni caso, è necessario garantire tempi di risposta compatibili con i requisiti operativi, bilanciando carichi tra CPU/GPU e gestendo eventuali picchi tramite scaling automatico.

Figura 6 - Esempio di processo di Esercizio di un modello di IA generativa

3.5 6Classificazione dei ruoli: pattern per individuare il livello di autonomia delle Pubbliche Amministrazioni

Alla luce dell'architettura logica evidenziata nel Cap.3 (3), si rende necessario descrivere i diversi scenari di integrazione delle tecnologie di IA nei sistemi della PA. A tal fine, considerando i diversi livelli di competenze e risorse disponibili, si possono individuare quattro ruoli, definendo così un pattern che di fatto individua anche il livello di autonomia e controllo ottenibile*: Operatore base, Operatore avanzato, Operatore esperto, Operatore controllore* (7).

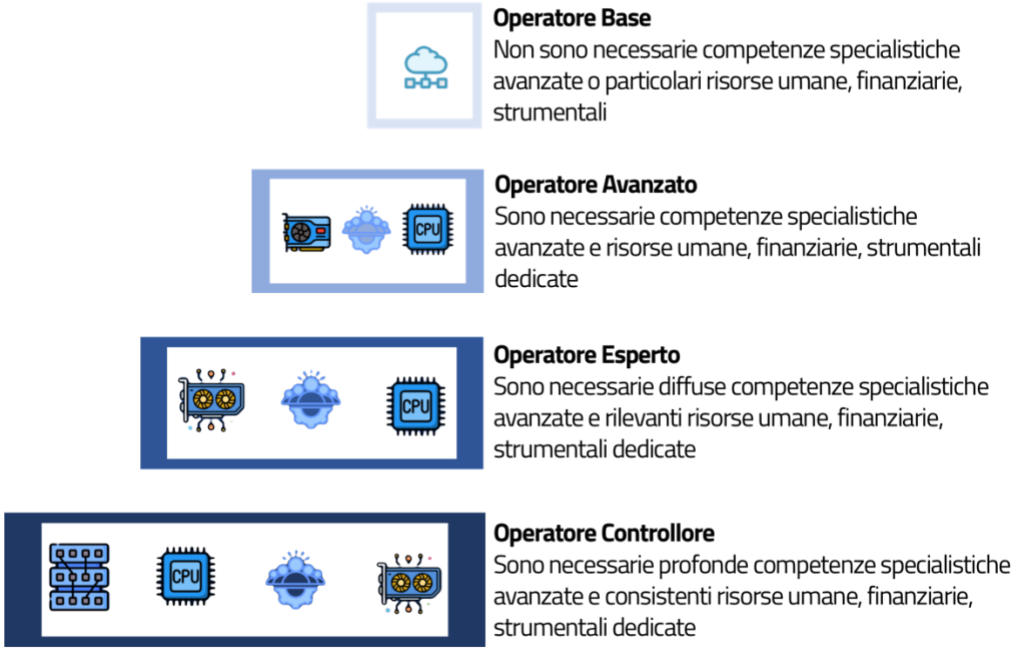
Per un **Operatore base** non sono necessarie competenze specialistiche avanzate o particolari risorse umane, finanziarie, strumentali;

Per un **Operatore avanzato** sono necessarie competenze specialistiche avanzate e risorse umane, finanziarie, strumentali dedicate;

Per un **Operatore esperto** sono necessarie diffuse competenze specialistiche avanzate e rilevanti risorse umane, finanziarie, strumentali dedicate;

Per un **Operatore controllore** sono necessarie competenze specialistiche avanzate e consistenti risorse umane, finanziarie, strumentali dedicate**.**

Figura 7 - Categorie di Operatori della Pubblica Amministrazione



7I ruoli non individuano necessariamente una Pubblica Amministrazione e, anzi, ogni singola amministrazione o organizzazione, in funzione del proprio caso d'uso, potrebbe ricoprire diversi profili: parliamo quindi di famiglie di utilizzatori.

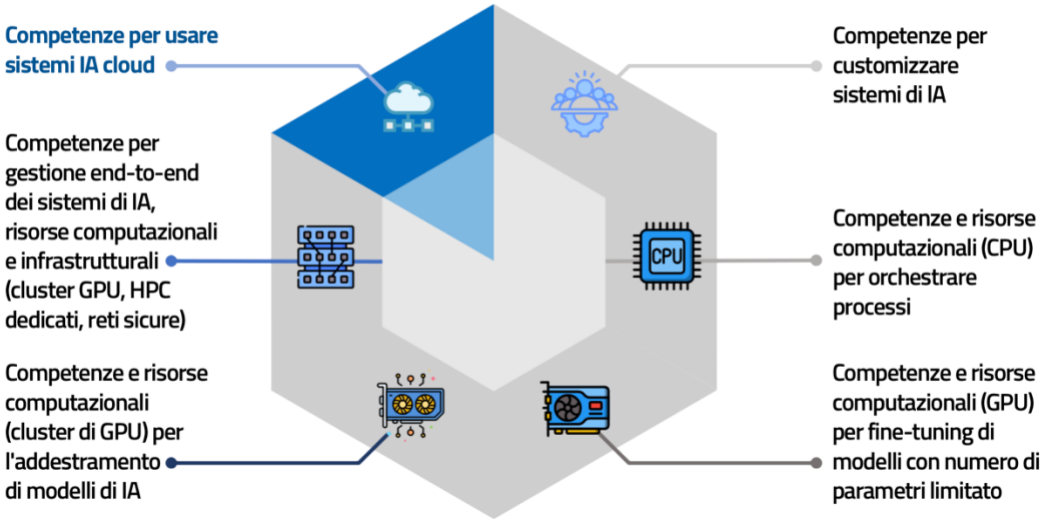
3.5.1 Operatore base

Gli *Operatori base* comprendono Pubbliche Amministrazioni o organizzazioni private che utilizzano sistemi di Intelligenza Artificiale senza effettuare attività di sviluppo, addestramento o personalizzazione degli stessi. Si tratta dell'utenza che impiega soluzioni di IA "as-a-service",

limitandosi a integrarle nelle proprie attività operative, senza intervenire profondamente sui componenti delle architetture sottostanti. Questa categoria rappresenta la maggioranza delle amministrazioni, includendo enti centrali, regionali e locali, agenzie, istituti scolastici, università, aziende sanitarie e, più in generale, tutte le organizzazioni che adottano sistemi di IA per migliorare l'erogazione dei servizi pubblici o ottimizzare processi interni, senza poter contare su competenze specialistiche avanzate e su particolari risorse umane, finanziarie, strumentali.

Gli *Operatori base* (8) possiedono competenze per usare i sistemi IA in cloud e fanno ad esempio uso di General Purpose AI (GPAI), ossia modelli di Intelligenza Artificiale per finalità generali, addestrati su grandi quantità di dati e capaci di svolgere un'ampia gamma di compiti, tra cui la generazione di testo, immagini, audio o video in risposta a una richiesta (prompt) da parte dell'utente. Per questa tipologia di utilizzatori, sono ovviamente richieste competenze su tutto lo schema architetturale di 3, con limitato impiego di risorse umane, strumentali e finanziarie. Per questa categoria di utilizzatori, quindi, il controllo e l'autonomia della PA sullo stack IA (2) è limitato, poiché i sistemi utilizzati sono forniti da terze parti e non vengono modificati nelle parti essenziali come il modello, i dati di addestramento o le pipeline di bilanciamento.

Figura 8 - Operatore Base



8Nell'ambito dell'architettura di riferimento per le Pubbliche Amministrazioni (3), l'istanza adottata da questa categoria assume una forma semplificata, basata sull'utilizzo di un orchestratore che fa leva su componenti e configurazioni quali:

- Invocazione di una molteplicità di modelli di IA mediante API esposte in cloud;
- Utilizzo di dati memorizzati in ambienti cloud, on-prem o in configurazioni ibride, da utilizzare eventualmente nella fase inferenziale nel rispetto del grado di riservatezza dei dati stessi;
- Predisposizione di strumenti e interfacce utente (ad esempio chatbot o prompt) attraverso i quali è possibile generare inferenze, interagendo con i modelli di IA.

3.5.2 Operatore avanzato

Gli *Operatori avanzati* (9) rappresentano la categoria di Pubbliche Amministrazioni o organizzazioni private che non sviluppano modelli di Intelligenza Artificiale da zero, ma effettuano attività di adattamento (ad esempio integrando RAG in contesti specifici, senza necessariamente modificare i pesi del modello) o fine-tuning- di modelli esistenti, con particolare riferimento ai modelli di Intelligenza Artificiale per finalità generali (GPAI).

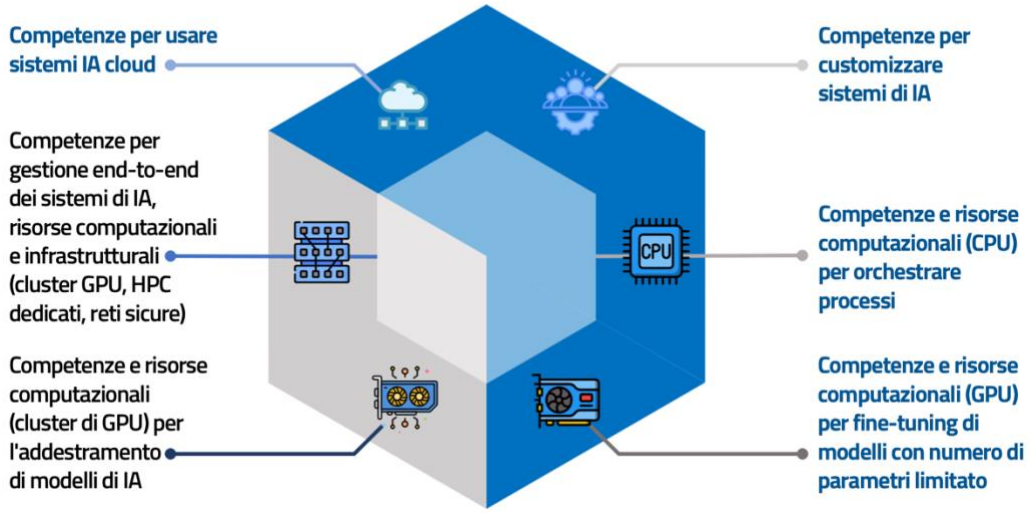
L'obiettivo della PA in questo caso potrebbe essere quello di rispondere a esigenze specifiche di innovazione e creare soluzioni verticali personalizzate.

Queste amministrazioni operano quindi a un livello intermedio della catena del valore dell'IA: non sono semplici utilizzatori finali, ma neppure creatori integrali di sistemi. Si collocano invece nella fascia degli adattatori, capaci di personalizzare e ottimizzare l'IA in funzione di specifici casi d'uso.

Le amministrazioni che adottano questo modello generalmente intendono perseguire uno dei seguenti obiettivi:

- Riduzione dei costi e dei tempi di sviluppo rispetto alla creazione di un modello proprietario;
- Sperimentazione di modelli verticali, ad esempio nei settori sanità, giustizia, finanza pubblica, ambiente, trasporti, dove il contesto normativo e tecnico richiede specializzazioni ad hoc.

Figura 9 - Operatore Avanzato



9

Le Pubbliche Amministrazioni classificate come Operatori avanzati presentano capacità tecnico-organizzative più avanzate rispetto agli Operatore base. Per effettuare attività di fine-tuning e specializzazione dei modelli, infatti, le PA devono dotarsi di competenze interne quali, ad esempio:

- Competenze di ML, necessarie a condurre attività di fine-tuning, valutazione delle performance e integrazione di processi operativi interni;
- Competenze di Data Governance, accesso a dataset amministrativi, essenziali per l'addestramento e la specializzazione del modello;
- Competenze di MLOps, comprendenti pipeline di addestramento, validazione, versioning, rilascio controllato e monitoraggio continuo.

Questa categoria di PA, quindi, ha un maggiore controllo dello stack IA, proprio perché DEVE agire direttamente sul Layer dei modelli di IA, DEVE mettere a disposizione la capacità computazionale necessaria (*Chip Layer*) e DEVE tenere in considerazione gli aspetti di sostenibilità energetica per il funzionamento degli strati di computazione.

Sono necessarie in questo caso competenze specialistiche avanzate e risorse umane, finanziarie, strumentali dedicate.

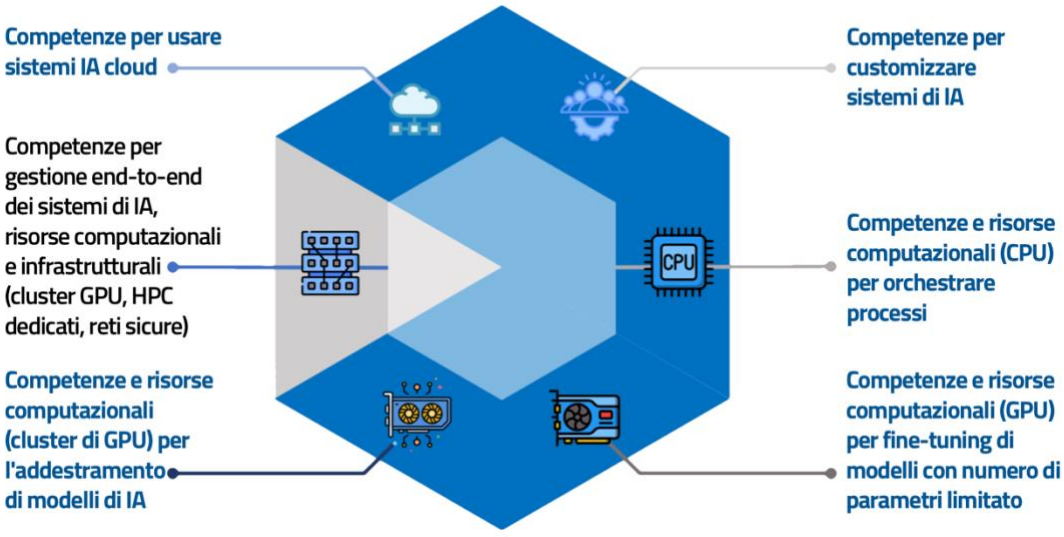
3.5.3 Operatore esperto

Gli *Operatori esperti* (10) rappresentano la categoria di Pubbliche Amministrazioni o organizzazioni pubbliche dotate di una forte capacità e competenza tecnologica interna e di una missione istituzionale che include la progettazione, lo sviluppo e la gestione di sistemi di Intelligenza Artificiale, dataset strutturati e capacità computazionale.

A differenza degli Operatori avanzati, che si concentrano principalmente sull'adattamento e specializzazione di modelli esistenti, gli Operatori esperti operano a un livello più profondo della catena del valore dell'IA, contribuendo alla creazione di componenti fondamentali dell'ecosistema digitale del Paese.

Gli Operatori esperti possono assumere un ruolo importante nel processo di diffusione e adozione dell'IA nel settore pubblico, diventando di fatto attori abilitanti perché sono in grado di fornire tecnologie, asset dati e infrastrutture che possono essere *adottate*, *utilizzate* e *riutilizzate* da altre Amministrazioni, imprese e centri di ricerca che non dispongano del know-how necessario per sviluppare in autonomia, come discusso nel cap.6.

Figura 10 - Operatore Esperto



10

Le Pubbliche Amministrazioni classificate come Operatori esperti, infatti, perseguono i seguenti obiettivi:

- Garantire autonomia tecnologica nazionale, riducendo dipendenze strategiche da pochi fornitori globali;
- Creare e mantenere modelli fondazionali pubblici, dataset strutturati e infrastrutture AI condivise;
- Promuovere standard, interoperabilità e riuso, sostenendo l'allineamento tecnologico dell'intero settore pubblico;
- Sostenere la ricerca, la sperimentazione e la definizione di linee guida tecniche e normative.

Per operare a questo livello, le amministrazioni devono possedere competenze e capacità quali:

- Competenze interne avanzate di ML/AI, comprendenti sviluppo di modelli, monitoraggio performance, gestione del ciclo di vita e ottimizzazione di architetture su infrastrutture complesse;
- Infrastrutture dedicate di calcolo (cluster di GPU, ambienti cloud specializzati), necessarie all'addestramento di modelli fondazionali e alla gestione di dataset di grandi dimensioni;
- Capacità di Data Governance avanzata, con responsabilità diretta su dataset nazionali o settoriali, definiti e gestiti con standard di qualità, sicurezza e accessibilità;
- Collaborazione strutturata con università, centri di ricerca e industria, per la co-creazione di tecnologie e metodologie.

Sono necessarie in questo caso diffuse competenze specialistiche avanzate e rilevanti risorse umane, finanziarie, strumentali dedicate.

3.5.4 L'Operatore controllore

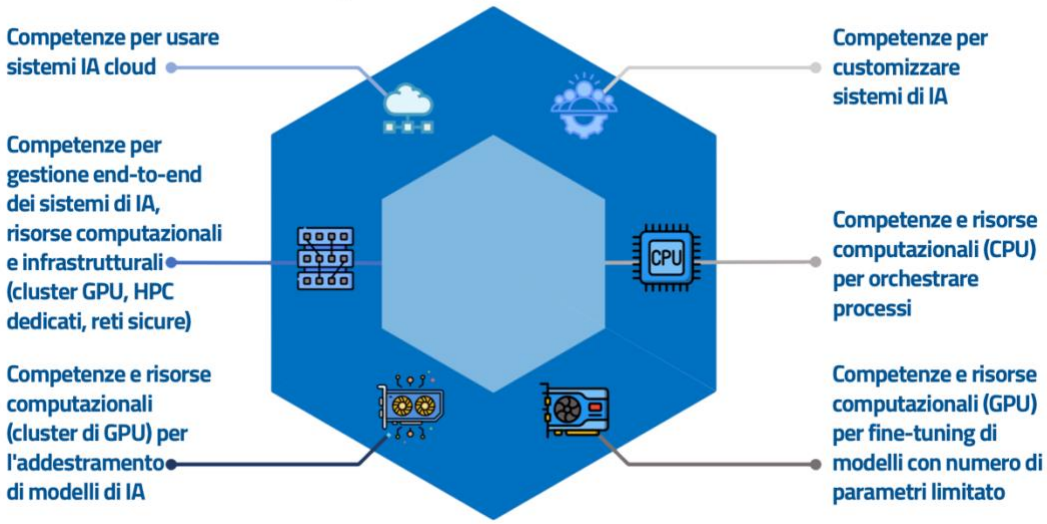
Gli *Operatori controllori* (11) rappresentano le Pubbliche Amministrazioni con mandato strategico, responsabilità istituzionali sensibili o esigenze operative tali da richiedere un controllo completo sull'intero ciclo di vita dei sistemi e dei servizi che facciano uso di Intelligenza Artificiale.

A differenza degli Operatori avanzati ed esperti, che si collocano lungo la catena del valore come adattatori o creatori, l'Operatore controllore mira a eliminare ogni dipendenza critica da fornitori

esterni, esercitando pieno governo tecnologico, infrastrutturale e operativo su ogni componente del sistema AI.

Queste Amministrazioni devono garantire che i modelli, i dataset, le pipeline dei dati, le infrastrutture e i sistemi di supporto siano nel pieno controllo senza condizioni esterne che possano limitarne l'autonomia o generare rischi strategici.

Figura 11 - Operatore Controllore



11

Gli Operatori controllori agiscono quindi sull'intera filiera AI, assicurandosi che ogni elemento, dal dato alla componente computazionale, dal modello al deployment, sia sotto il controllo istituzionale e diretto della Pubblica Amministrazione stessa.

Obiettivi

- Garantire piena autonomia strategica, riducendo dipendenze tecnologiche da fornitori esterni e minimizzando i rischi di dinamiche di costi esterne non governabili;
- Gestire in modalità end-to-end l'intera catena del valore dell'IA, dal dato alla messa in produzione, dalla creazione del modello, al suo addestramento ed esercizio, alle piattaforme ospitanti i modelli e le pipeline ed infine la governance del prodotto finale.

Sono necessarie in questo caso profonde competenze specialistiche avanzate e consistenti risorse umane, finanziarie, strumentali dedicate.

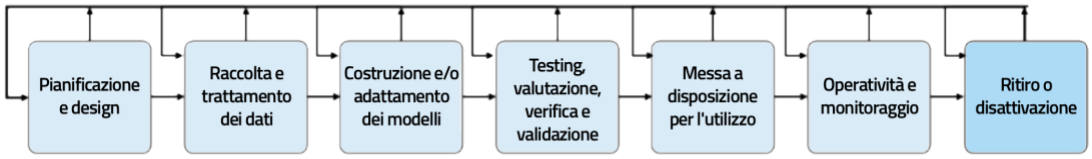
4 Ciclo di vita dei sistemi di Intelligenza Artificiale e azioni per lo sviluppo e il procurement

Le presenti Linee guida adottano come modello di riferimento il ciclo di vita di un sistema di Intelligenza Artificiale definito al paragrafo 3.1 delle Linee guida per l'adozione dell'IA, che copre l'intero arco temporale del sistema, dalla pianificazione iniziale fino alla dismissione.

In particolare, in ciclo di vita di un sistema di IA prevede (12) le fasi di:

1. Pianificazione e design;
2. Raccolta e trattamento dei dati;
3. Costruzione e/o adattamento dei modelli;
4. Testing, valutazione, verifica e validazione;
5. Messa a disposizione per l'utilizzo;
6. Operatività e monitoraggio;
7. Ritiro / disattivazione.

Figura 12 - Ciclo di vita di un sistema di IA



12Tale modello costituisce un riferimento comune per le Linee guida relative all'adozione, al procurement e allo sviluppo dei sistemi di IA nella Pubblica Amministrazione, al fine di garantire coerenza metodologica, continuità decisionale e controllabilità delle scelte tecnologiche.

Nei paragrafi successivi sono sinteticamente descritte le azioni che le Pubbliche Amministrazioni DEVONO adottare nelle fasi di sviluppo e acquisto dei sistemi di IA, secondo le cinque dimensioni tecnologiche introdotte nel capitolo 3: Energy, Chip, Infrastruttura, Modelli di IA e Applicazioni.

Tale impostazione consente di integrare le valutazioni tecniche, organizzative e contrattuali lungo l'intero ciclo di vita del sistema di IA, riducendo i rischi di lock-in tecnologico e favorendo soluzioni sostenibili, interoperabili e conformi al quadro normativo.

4.1 Pianificazione e design

La fase di pianificazione e design è determinante per orientare correttamente l'intero ciclo di vita del sistema di IA. In questa fase, le Pubbliche Amministrazioni DEVONO adottare scelte architetturali e contrattuali che tengano conto fin dall'inizio dei profili energetici, infrastrutturali, di rischio e di sostenibilità, evitando soluzioni rigide o difficilmente evolvibili nelle fasi successive.

Le Pubbliche Amministrazioni DEVONO valutare, fin dalla fase di pianificazione e design, l'adozione di architetture agentiche e di orchestrazione dei servizi di IA, al fine di rafforzare la modularità, la sostituibilità dei componenti, la gestione multi-fornitore e l'autonomia operativa dei sistemi nel tempo.

In termini di sviluppo, questo significa:

- per il Layer Energia, stimare ex ante l'impatto energetico del sistema di IA, includendo training e inference e tenendo conto anche delle implicazioni sulla rete elettrica locale e della necessità di coordinamento con i soggetti gestori;
- per il Layer Chip, progettare architetture hardware-agnostic, evitando dipendenze da specifici acceleratori;
- per il Layer Infrastruttura disegnare architetture modulari e portabili, basate su container, microservizi e API standard;
- per il Layer Modelli IA, definire criteri di selezione dei modelli coerenti con la classificazione del rischio e con la natura dei dati trattati;
- per il Layer Applicazioni, separare chiaramente logica applicativa, modelli di IA e infrastruttura, secondo principi di modularità e disaccoppiamento.

Layer	Sviluppo
Energia	Stimare ex ante l'impatto energetico del sistema di IA, includendo training e inference e tenendo conto anche delle implicazioni sulla rete elettrica locale e della necessità di coordinamento con i soggetti gestori.
Chip	Progettare architetture hardware-agnostic, evitando dipendenze da specifici acceleratori.
Infrastruttura	Disegnare architetture modulari e portabili, basate su container, microservizi e API standard.
Modelli IA	Definire criteri di selezione dei modelli coerenti con la classificazione del rischio e con la natura dei dati trattati.
Applicazioni	Separare chiaramente logica applicativa, modelli di IA e infrastruttura, secondo principi di modularità e disaccoppiamento.

4.2 Raccolta e trattamento dei dati

La fase di raccolta e trattamento dei dati è centrale per garantire la qualità, l'affidabilità e la conformità normativa dei sistemi di IA. In questa fase, le Pubbliche Amministrazioni DEVONO assicurare che le scelte tecniche e contrattuali consentano un uso proporzionato, controllabile e sostenibile dei dati, riducendo sprechi computazionali, rischi di lock-in tecnologico e dipendenze non necessarie da modelli o infrastrutture ad alta intensità di risorse.

In termini di sviluppo, questo significa:

- per il Layer Energia, ottimizzare le pipeline di raccolta e trattamento dei dati per ridurre elaborazioni inutili e duplicazioni, nonché minimizzare i cicli di training e preprocessing ad alto consumo energetico;
- per il Layer Chip, progettare pipeline dati compatibili con elaborazioni distribuite e scalabili su hardware eterogeneo, nonché progettare modelli e pipeline eseguibili anche in assenza di GPU, utilizzando CPU e acceleratori generici;

- per il Layer Infrastruttura, prediligere soluzioni che consentano data locality, controllo dei flussi di dati e riduzione dei trasferimenti non necessari, e allo stesso tempo implementare meccanismi di governance dei flussi dati coerenti con la classificazione dei dati della PA;
- per il Layer Modelli IA, preparare dataset adeguati anche a modelli di dimensioni ridotte, evitando fenomeni di over-fitting e "big model bias", e allo stesso tempo applicare tecniche di data minimization e data quality management;
- per il Layer Applicazioni, integrare controlli di qualità dei dati, versioning e tracciabilità lungo l'intero ciclo di trattamento, nonché esporre interfacce applicative che consentano la gestione controllata dei flussi informativi.

Layer	Sviluppo
Energia	Ottimizzare le pipeline di raccolta e trattamento dei dati per ridurre elaborazioni inutili e duplicazioni. Minimizzare i cicli di training e pre-processing ad alto consumo energetico.
Chip	Progettare pipeline dati compatibili con elaborazioni distribuite e scalabili su hardware eterogeneo. Progettare modelli e pipeline eseguibili anche in assenza di GPU, utilizzando CPU e acceleratori generici
Infrastruttura	Prediligere soluzioni che consentano data locality, controllo dei flussi di dati e riduzione dei trasferimenti non necessari. Implementare meccanismi di governance dei flussi dati coerenti con la classificazione dei dati della PA.
Modelli IA	Preparare dataset adeguati anche a modelli di dimensioni ridotte, evitando fenomeni di over-fitting e "big model bias". Applicare tecniche di data minimization e data quality management.
Applicazioni	Integrare controlli di qualità dei dati, versioning e tracciabilità lungo l'intero ciclo di trattamento. Esporre interfacce applicative che consentano la gestione controllata dei flussi informativi.

4.3 Costruzione e/o addestramento dei modelli

La fase di costruzione e/o addestramento dei modelli di IA incide in modo significativo sui costi energetici, sulla scalabilità, sulla gestione del rischio e sulle dipendenze tecnologiche del sistema. In questa fase, le Pubbliche Amministrazioni DEVONO privilegiare approcci proporzionati al contesto d'uso, evitando soluzioni sovradimensionate e vincoli tecnologici che possano compromettere l'evoluzione del sistema nel tempo.

In termini di sviluppo, questo significa:

- per il Layer Energia, privilegiare approcci di fine-tuning e trasferimento dell'apprendimento, riducendo il ricorso a training completi ad alta intensità energetica, quando necessario, nonché ottimizzare i processi di addestramento per contenere i consumi energetici complessivi;
- per il Layer Chip, adottare tecnologie di decoupling tra modello e processore e progettare modelli eseguibili anche in assenza di GPU, utilizzando CPU e acceleratori generici;
- per il Layer Infrastruttura, utilizzare ambienti di training replicabili e portabili, coerenti con architetture cloud, on-prem o ibride, nonché documentare le dipendenze infrastrutturali del processo di training;
- per il Layer Modelli IA, selezionare i modelli in base al livello di rischio, alla sensibilità dei dati e al contesto operativo, nonché privilegiare modelli più piccoli o specializzati quando adeguati allo scopo;
- per il Layer Applicazioni, isolare il modello di IA come componente sostituibile, separandolo dalla logica applicativa.

Layer	Sviluppo
Energia	Privilegiare approcci di fine-tuning e trasferimento dell'apprendimento, riducendo il ricorso a training completi ad alta intensità energetica, quando necessario. Ottimizzare i processi di addestramento per contenere i consumi energetici complessivi.
Chip	Adottare tecnologie di decoupling tra modello e processore. Progettare modelli eseguibili anche in assenza di GPU, utilizzando CPU e acceleratori generici.
Infrastruttura	Utilizzare ambienti di training replicabili e portabili, coerenti con architetture cloud, on-prem o ibride. Documentare le dipendenze infrastrutturali del processo di training
Modelli IA	Selezionare i modelli in base al livello di rischio, alla sensibilità dei dati e al contesto operativo. Privilegiare modelli più piccoli o specializzati quando adeguati allo scopo
Applicazioni	Isolare il modello di IA come componente sostituibile, separandolo dalla logica applicativa.

4.4 Testing, valutazione, verifica e validazione

La fase di testing, valutazione, verifica e validazione è finalizzata a garantire che i sistemi di IA siano affidabili, accurati, robusti, sicuri e idonei all'impiego nei contesti amministrativi reali. Le attività di sviluppo e di procurement DEVONO essere condotte in modo coordinato, considerando l'intero stack tecnologico dell'IA e adottando criteri verificabili e soggetti ad audit.

In termini di sviluppo, questo significa:

- per il Layer Energia, monitorare e misurare i consumi energetici del sistema durante le fasi di test, validazione e inferenza, nonché valutare l'efficienza energetica in relazione ai carichi di lavoro previsti in esercizio;

- per il Layer Chip, verificare il corretto funzionamento del sistema su hardware eterogeneo, inclusi acceleratori diversi e CPU, nonché testare scenari di fallback e degradazione controllata delle prestazioni;
- per il Layer Infrastruttura, testare la portabilità del sistema tra ambienti infrastrutturali diversi (cloud, on-prem, ibrido) e verificare la replicabilità degli ambienti di test e produzione;
- Layer Modelli IA, valutare performance, robustezza, affidabilità e bias dei modelli in relazione al livello di rischio del sistema di IA, nonché effettuare test su dataset rappresentativi e scenari di uso reale;
- Per il Layer Applicazioni, testare l'integrazione delle funzionalità di IA nei processi amministrativi reali e nei flussi operativi esistenti, e verificare la continuità funzionale in caso di sostituzione di componenti dello stack.

Layer	Sviluppo
Energia	Monitorare e misurare i consumi energetici del sistema durante le fasi di test, validazione e inferenza. Valutare l'efficienza energetica in relazione ai carichi di lavoro previsti in esercizio.
Chip	Verificare il corretto funzionamento del sistema su hardware eterogeneo, inclusi acceleratori diversi e CPU. Testare scenari di fallback e degradazione controllata delle prestazioni.
Infrastruttura	Testare la portabilità del sistema tra ambienti infrastrutturali diversi (cloud, on-prem, ibrido). Verificare la replicabilità degli ambienti di test e produzione
Modelli IA	Valutare performance, robustezza, affidabilità e bias dei modelli in relazione al livello di rischio del sistema di IA. Effettuare test su dataset rappresentativi e scenari di uso reale.
Applicazioni	Testare l'integrazione delle funzionalità di IA nei processi amministrativi reali e nei flussi operativi esistenti. Verificare la continuità funzionale in caso di sostituzione di componenti dello stack.

4.5 Messa a disposizione per l'esercizio

La fase di messa a disposizione per l'esercizio segna il passaggio del sistema di IA dall'ambiente di sviluppo a quello operativo. In questa fase, le Pubbliche Amministrazioni DEVONO garantire che il sistema sia stabile, sicuro, sostenibile e controllabile, assicurando al contempo la continuità del servizio pubblico e la reversibilità delle scelte tecnologiche effettuate nelle fasi precedenti.

In termini di sviluppo, questo significa:

- per il Layer Energia, ottimizzare i processi di inferenza e la configurazione iniziale del sistema per ridurre carichi energetici e consumi operativi, nonché verificare l'efficienza dei sistemi di alimentazione e raffreddamento dell'infrastruttura utilizzata;
- per il Layer Chip, configurare modalità di esecuzione distribuite dei modelli, nonché abilitare meccanismi di fallback su CPU o architetture hardware alternative;
- per il Layer Infrastruttura, abilitare orchestratori indipendenti dal fornitore per la gestione dei workload e configurare sia ambienti scalabili sia una chiara separazione tra produzione e test di stabilizzazione;
- per il Layer Modelli IA, effettuare la validazione finale dei modelli prima dell'entrata in produzione e attivare meccanismi iniziali di monitoraggio delle prestazioni e della qualità dei dati;
- per il Layer Applicazioni, verificare la corretta integrazione dei servizi di IA nei processi applicativi esistenti e garantire la continuità funzionale del servizio nella fase di go-live.

Layer	Sviluppo
Energia	Ottimizzare i processi di inferenza e la configurazione iniziale del sistema per ridurre carichi energetici e consumi operativi. Verificare l'efficienza dei sistemi di alimentazione e raffreddamento dell'infrastruttura utilizzata.
Chip	Configurare modalità di esecuzione distribuite dei modelli. Abilitare meccanismi di fallback su CPU o architetture hardware alternative.
Infrastruttura	Abilitare orchestratori indipendenti dal fornitore per la gestione dei workload. Configurare ambienti scalabili e una chiara separazione tra produzione e test di stabilizzazione.
Modelli IA	Effettuare la validazione finale dei modelli prima dell'entrata in produzione. Attivare meccanismi iniziali di monitoraggio delle prestazioni e della qualità dei dati.
Applicazioni	Verificare la corretta integrazione dei servizi di IA nei processi applicativi esistenti. Garantire la continuità funzionale del servizio nella fase di go-live.

4.6 Operatività e monitoraggio

La fase di operatività e monitoraggio riguarda l'esercizio continuativo del sistema di IA nel tempo. In questa fase, le Pubbliche Amministrazioni DEVONO garantire che il sistema rimanga stabile, sicuro, sostenibile e controllabile, assicurando la continuità del servizio pubblico, il monitoraggio delle prestazioni e dei rischi e la possibilità di evoluzione o sostituzione delle componenti tecnologiche senza effetti negativi sui servizi erogati. Tali prescrizioni DEVONO essere garantite mediante specifiche attività di governance e conduzione operativa di sistemi e servizi, attraverso la previsione di audit interni con periodicità dipendente dalla complessità dei sistemi, delle architetture e dalla criticità dei servizi realizzati.

In termini di sviluppo, questo significa:

- per il Layer Energia, monitorare in modo continuativo i consumi energetici del sistema in esercizio e ottimizzare dinamicamente i carichi di inferenza per ridurre sprechi energetici e picchi di consumo;

- per il Layer Chip, monitorare l'utilizzo degli acceleratori di IA e delle architetture hardware impiegate, nonché garantire la continuità operativa attraverso la riallocazione dei carichi e l'uso di hardware alternativi;
- per il Layer Infrastruttura, monitorare prestazioni, disponibilità e resilienza dell'infrastruttura, e gestire il sistema tramite orchestratori indipendenti dal fornitore per facilitare scalabilità e migrazione;
- per il Layer Modelli IA, monitorare fenomeni di data drift (cambiamento nella distribuzione dei dati in input rispetto a quelli usati per addestrare il modello), concept drift (cambiamento nella relazione tra input e output che il modello deve prevedere rendendo le predizioni progressivamente meno accurate) e degradazione delle performance dei modelli. Allo stesso modo, attivare processi di ri-addestramento, fine-tuning o sostituzione dei modelli;
- per il Layer Applicazioni, monitorare la qualità del servizio e l'esperienza utente, nonché gestire gli aggiornamenti applicativi garantendo la continuità del servizio pubblico.

Layer	Sviluppo
Energia	Monitorare in modo continuativo i consumi energetici del sistema in esercizio. Ottimizzare dinamicamente i carichi di inferenza per ridurre sprechi energetici e picchi di consumo.
Chip	Monitorare l'utilizzo degli acceleratori di IA e delle architetture hardware impiegate. Garantire la continuità operativa attraverso la riallocazione dei carichi e l'uso di hardware alternativi.
Infrastruttura	Monitorare prestazioni, disponibilità e resilienza dell'infrastruttura. Gestire il sistema tramite orchestratori indipendenti dal fornitore per facilitare scalabilità e migrazione.
Modelli IA	Monitorare fenomeni di data drift (cambiamento nella distribuzione dei dati in input rispetto a quelli usati per addestrare il modello), concept drift (cambiamento nella relazione tra input e output che il modello deve prevedere rendendo le predizioni progressivamente meno accurate) e degradazione delle performance dei modelli. Attivare processi di ri-addestramento, fine-tuning o sostituzione dei modelli
Applicazioni	Monitorare la qualità del servizio e l'esperienza utente. Gestire gli aggiornamenti applicativi garantendo la continuità del servizio pubblico.

4.7 Ritiro e/o disattivazione

La fase di ritiro e/o disattivazione del sistema di IA è strategica per prevenire il lock-in tecnologico, tutelare l'investimento pubblico e garantire la piena reversibilità delle scelte tecnologiche effettuate lungo l'intero ciclo di vita del sistema. In tale fase, le Pubbliche Amministrazioni DEVONO assicurare una dismissione ordinata, documentata e controllabile delle componenti del sistema, preservando la continuità dei servizi essenziali e la riutilizzabilità delle risorse ove possibile.

In termini di sviluppo, questo significa:

- per il Layer Energia, pianificare la disattivazione ordinata delle risorse energetiche dedicate al sistema, nonché ridurre e cessare progressivamente i consumi energetici residui associati all'infrastruttura di IA;
- per il Layer Chip, gestire la disattivazione o riallocazione dell'hardware di IA utilizzato e preparare il sistema alla sostituzione modulare delle componenti hardware;
- per il Layer Infrastruttura, documentare l'architettura infrastrutturale e le dipendenze tecniche del sistema, nonché preparare le procedure di spegnimento, migrazione o smantellamento dell'infrastruttura;
- per il Layer Modelli IA, documentare modelli, dataset, pipeline di addestramento e configurazioni, nonché preparare i modelli alla sostituzione o all'archiviazione controllata;
- per il Layer Applicazioni, documentare funzionalità applicative, interfacce e integrazioni con altri sistemi, nonché preparare le applicazioni alla sostituzione graduale o alla dismissione controllata.

Layer	Sviluppo
Energia	Pianificare la disattivazione ordinata delle risorse energetiche dedicate al sistema. Ridurre e cessare progressivamente i consumi energetici residui associati all'infrastruttura di IA.
Chip	Gestire la disattivazione o riallocazione dell'hardware di IA utilizzato. Preparare il sistema alla sostituzione modulare delle componenti hardware
Infrastruttura	Documentare l'architettura infrastrutturale e le dipendenze tecniche del sistema. Preparare le procedure di spegnimento, migrazione o smantellamento dell'infrastruttura.
Modelli IA	Documentare modelli, dataset, pipeline di addestramento e configurazioni. Preparare i modelli alla sostituzione o all'archiviazione controllata
Applicazioni	Documentare funzionalità applicative, interfacce e integrazioni con altri sistemi. Preparare le applicazioni alla sostituzione graduale o alla dismissione controllata

4.8 Considerazioni conclusive

Il presente capitolo ha definito un modello operativo per lo sviluppo dei sistemi di Intelligenza Artificiale nella Pubblica Amministrazione lungo l'intero ciclo di vita, dalla pianificazione alla dismissione. Tale capitolo sarà anche citato nella Linea Guida Procurement, la cui descrizione è basata sulla medesima Architettura logica.

L'approccio basato sullo stack tecnologico dell'IA – Energy, Chip, Infrastruttura, Modelli di IA e Applicazioni – consente alle Pubbliche Amministrazioni di effettuare scelte consapevoli e controllabili, rafforzando la capacità di governo delle soluzioni adottate, riducendo dipendenze tecnologiche critiche e preservando l'autonomia decisionale e operativa nel tempo.

Le azioni individuate per ciascuna fase del ciclo di vita sono verificabili e auditabili e coerenti con un approccio basato sul rischio, con gli standard internazionali di riferimento e con il quadro normativo europeo.

Il modello proposto supporta l'adozione di sistemi di IA sostenibili, interoperabili ed evolvibili, in grado di garantire la continuità dei servizi pubblici e la tutela dell'interesse pubblico non solo nel breve ma anche nel medio e lungo periodo.

5 Sicurezza cibernetica

La sicurezza rappresenta un requisito fondamentale per l'adozione, l'acquisizione e lo sviluppo dell'intelligenza artificiale dei sistemi IA nella Pubblica Amministrazione. Nonostante l'intelligenza artificiale appaia come una risorsa preziosa per poter affrontare la crescente necessità di migliorare l'efficienza e l''efficacia nella gestione e nella fornitura dei servizi pubblici, questa tecnologia introduce nuovi rischi, minacce e vulnerabilità che devono essere presi in considerazione nella sua implementazione.

La sicurezza di un sistema di IA può essere definita[2] come l'insieme di strumenti, strategie e processi implementati per identificare e prevenire le minacce e gli attacchi che potrebbero compromettere la riservatezza, l'integrità o la disponibilità di un modello di IA o di un sistema abilitato all'IA.

In considerazione della loro natura inerentemente socio-tecnica – in cui elementi sociali (l'influenza delle dinamiche sociali e l'impatto sulle persone che li usano o che ne sono condizionati) e tecnici (quali dataset, algoritmi, modelli) risultano strettamente intrecciati tra loro – i sistemi di IA sono caratterizzati da specifiche peculiarità nell'ambito della sicurezza in particolare con riferimento ai *rischi* ai quali sono soggetti e agli *attacchi* dei quali sono vittima.

In questo capitolo sono discusse ed esaminate tali peculiarità: la successiva sezione introduce, ai fini dell'inquadramento del contesto, un modello del ciclo di vita dei sistemi, la seconda e la terza trattano rispettivamente la gestione del rischio e le tassonomie di attacco, l'ultima sezione elenca una serie di obiettivi di sicurezza specifici per l'ambito di applicazione delle presenti linee guida.

5.1 Ciclo di vita

Ai fini dell'individuazione dei rischi e degli attacchi associati ai sistemi di intelligenza artificiale, la Pubblica Amministrazione DEVE partire da una comprensione del ciclo di vita di un sistema di IA, così come dettagliato nei paragrafi successivi.

[2] La definizione è tratta dall'articolo pubblicato sul sito di MITRE ATLAS all'indirizzo https://atlas.mitre.org/resources/ai-security-101. MITRE ATLAS è una knowledge base delle tattiche e tecniche avversarie relative ai sistemi di IA.

Il ciclo di vita di un sistema di IA può essere suddiviso nelle seguenti fasi[3]:

- **definizione dell'obiettivo**: è individuato l'obiettivo che il sistema di intelligenza artificiale si prefigge di raggiungere. A tal fine devono essere compresi il contesto nel quale il sistema dovrà operare e i dati richiesti;
- **raccolta dei dati**: sono acquisiti i dati (in forma strutturata e non) dalle varie sorgenti e i corrispondenti metadati. L'acquisizione deve essere conforme alle politiche sulla gestione dei dati dell'organizzazione e di eventuali fornitori (dei dati) e alle normative riguardanti la protezione dei dati personali;
- **esplorazione dei dati**: sono analizzati i dati acquisiti ed è verificato se si adattano a distribuzioni statistiche al fine di stimarne i parametri corrispondenti;
- **pre-elaborazione dei dati**: i dati sono ripuliti (correzione di errori, rimozione del rumore, anonimizzazione/pseudo-anonimizzazione), integrati dalle varie fonti e trasformati secondo le esigenze (ad esempio, convertendoli in formato numerico);
- **selezione delle caratteristiche**: le caratteristiche più rilevanti dell'insieme di dati (*dataset*) sono identificate e selezionate, scartando quelle non di interesse con l'obiettivo di ridurre le dimensioni;
- **selezione del modello**: è scelto il tipo di modello di IA[4] che maggiormente si adatta al sistema sulla base dell'obiettivo e dei dati. Scegliere il modello implica anche la scelta della tipologia di algoritmo di apprendimento[5] con il quale verrà addestrato;
- **addestramento del modello**: il modello di IA scelto è addestrato sulla base dei dati e dell'algoritmo di apprendimento individuato.
- ***tuning* del modello**: il modello viene messo a punto regolando gli iperparametri[6] sulla base di un *dataset* di convalida, questa fase tipicamente si sovrappone alla precedente.
- **trasferimento dell'apprendimento**: l'organizzazione acquisisce un modello di IA già addestrato e configurato che utilizza come punto di partenza per l'ulteriore addestramento. Questa fase può avvenire quando non si hanno dati sufficienti per l'addestramento.

[3] European Union Agency for Cybersecurity (ENISA), AI Cybersecurity Challenges, 2020, https://www.enisa.europa.eu/publications/artificial-intelligence-cybersecurity-challenges.

[4] Con il termine *modello di IA* si indica un programma che è stato addestrato a partire da un insieme di dati con lo scopo di individuare schemi e relazioni, effettuare predizioni o creare nuovi contenuti.

[5] Le tre categorie più comuni di algoritmi sono l'apprendimento supervisionato, non supervisionato e con rinforzo.

[6] Parametri che consentono di controllare il processo di addestramento del modello.

- **distribuzione del modello**: il modello addestrato viene reso disponibile agli utenti.
- **manutenzione del modello**: il modello e i dati in ingresso sono continuamente monitorati e manutenuti per gestire i cambiamenti, riaddestrando il modello ove necessario.
- **valutazione del raggiungimento dell'obiettivo**: viene verificato il livello di raggiungimento dell'obiettivo prefissato del sistema di I.A.

La sicurezza dei sistemi di intelligenza artificiale DEVE essere mantenuta lungo tutto il ciclo di vita.

5.2 Gestione del rischio

Come già osservato, i sistemi di intelligenza artificiale sono sistemi socio-tecnici caratterizzati da specifici rischi la cui gestione rappresenta un elemento imprescindibile per lo sviluppo e l'utilizzo responsabile dell'intelligenza artificiale.

Tale esigenza è chiaramente emersa anche a livello regolamentare unionale. L'*AI Act* stabilisce infatti regole armonizzate per l'immissione sul mercato UE di sistemi di IA e, sulla base dei possibili rischi e del loro livello d'impatto, requisiti specifici per i sistemi di IA valutati ad alto rischio. In particolare, l'articolo 15 prevede che i sistemi di IA ad alto rischio siano progettati e sviluppati in modo tale da conseguire un adeguato livello di accuratezza, robustezza e cibersicurezza e operare in modo coerente con tali aspetti durante tutto il loro ciclo di vita.

Un'efficace strutturazione di un processo di gestione dei rischi di un sistema di IA deve necessariamente considerare le caratteristiche distintive di tali sistemi. A tale scopo è possibile fare riferimento a framework e standard specializzati, quale, ad esempio, il Risk Management Framework per l'intelligenza artificiale (AI RMF) sviluppato dal *National Institute of Standards and Technology (NIST)*.

Il framework è disegnato per assistere organizzazioni e individui, definiti come *AI Actors*[7], e fornisce un approccio strutturato per identificare, valutare e mitigare i rischi associati ai sistemi di IA. L'obiettivo è quello di minimizzare i potenziali impatti negativi e massimizzare quelli positivi in modo da avere sistemi di IA affidabili.

[7] L'organizzazione per la cooperazione e lo sviluppo economico (OCSE) definisce gli *AI Actors* come coloro i quali svolgono un ruolo attivo nel ciclo di vita dei sistemi di IA, compresi le organizzazioni e gli individui che distribuiscono o gestiscono l'intelligenza artificiale.

Potenziali impatti negativi derivanti dall'uso dei sistemi di IA sono classificati a seconda della tipologia di attore coinvolto:

- *impatti sugli individui*, come ad esempio gli impatti sulle libertà civili, sulla sicurezza fisica/psicologica o sulla sfera economica di un individuo;
- *impatti sulle organizzazioni*, come ad esempio gli impatti sulle attività commerciali, sulla reputazione o derivanti dalla compromissione di un'organizzazione;
- *impatti sull'ecosistema*, come ad esempio gli impatti su elementi e risorse interconnessi e interdipendenti, sul sistema finanziario, sulla catena di approvvigionamento o sulle risorse naturali e l'ambiente.

Il cosiddetto *Core* del framework individua le seguenti quattro funzioni (a loro volta suddivise in categorie e sottocategorie) per supportare le organizzazioni nella gestione dei rischi posti dai sistemi di IA:

- GOVERN: promuovere una cultura di gestione del rischio all'interno delle organizzazioni che progettano, sviluppano, acquisiscono e adottano sistemi di IA;
- MAP: stabilire il contesto nel quale inquadrare i rischi di un sistema di IA, identificare i rischi e i relativi fattori di rischio;
- MEASURE: analizzare, valutare, confrontare e monitorare il rischio e i relativi impatti. Utilizza le informazioni acquisite dalla precedente funzione e fornisce indicazioni a quella successiva;
- MANAGE: definire le priorità e ad agire sui rischi identificati. Il trattamento del rischio comprende piani di risposta, recupero e comunicazione di incidenti o eventi.

La funzione GOVERN è trasversale e si applica a tutte le fasi del processo di gestione del rischio. Le funzioni MAP, MEASURE e MANAGE comprendono componenti della funzione GOVERN (in particolare quelle riguardanti la conformità o la valutazione) e possono essere utilizzate in contesti specifici e in determinate fasi del ciclo di vita dei sistemi di intelligenza artificiale.

Per supportare le organizzazioni nell'uso del Framework, il NIST ha sviluppato un playbook[8] allineato a ciascuna sottocategoria delle quattro funzioni.

[8] https://airc.nist.gov/AI_RMF_Knowledge_Base/Playbook

L'individuazione dei rischi avviene tipicamente a partire dalle minacce cui può essere esposto un sistema e dai suoi asset, sui quali tali minacce tentano di sfruttarne le vulnerabilità. In ragione di ciò, nei seguenti paragrafi sono elencati, sulla base del modello del ciclo di vita discusso nella precedente sezione, le categorie di asset e minacce relative ai sistemi di intelligenza artificiale.

5.3 Asset

Un sistema di IA è costituito da un insieme di *asset*. Per asset si intende tutto ciò che ha un valore per un individuo o un'organizzazione e che quindi deve essere protetto. In aggiunta agli asset caratteristici dell'IA (come, ad esempio, i modelli e gli iperparametri) sono qui considerati anche gli asset dell'infrastruttura ICT (come, ad esempio, le reti di comunicazione e i sistemi operativi)

Gli asset di un sistema di IA possono essere categorizzati in[9] (tra parentesi è riportata la corrispondente fase del ciclo di vita dell'asset):

- *dati*, come ad esempio: dati grezzi (acquisizione dei dati), dati di valutazione (tuning del modello), dati etichettati (pre-elaborazione dei dati), dati di test (addestramento del modello);
- *modelli*, come ad esempio: algoritmi (addestramento del modello), iperparametri (tuning del modello), algoritmi di addestramento (selezione del modello);
- *attori*, come ad esempio: fornitori di servizi cloud (raccolta dei dati, addestramento del modello, tuning del modello), proprietari dei dati (definizione dell'obiettivo, raccolta dei dati, esplorazione dei dati), fornitori dei dati (raccolta dei dati);
- *processi*, come ad esempio: acquisizione dei dati (raccolta dei dati), etichettamento dei dati (pre-elaborazione dei dati), comprensione dei dati (esplorazione e convalida dei dati);
- *strumenti*, come ad esempio: reti di comunicazione (raccolta dei dati), database (raccolta dei dati), sistemi operativi (distribuzione del modello, mantenimento del modello);
- *artefatti*, come ad esempio: politiche di gestione dei dati (raccolta dei dati), architettura del modello (selezione del modello, distribuzione del modello), casi d'uso (comprensione del business).

[9] Per la tassonomia completa si faccia riferimento all'annesso A del documento "AI Cybersecurity Challenges" pubblicato dall'ENISA (European Union Agency for Cybersecurity), https://www.enisa.europa.eu/publications/artificial-intelligence-cybersecurity-challenges.

5.4 Minacce

Ogni fase del ciclo di vita è caratterizzata da una o più minacce. In generale, possono essere individuate le seguenti categorie di minacce[10]:

- *attività malevole/abuso*: azioni intenzionali che prendono di mira sistemi, infrastrutture e reti ICT mediante atti malevoli con l'obiettivo di sottrarre, alterare o distruggere un obiettivo specifico (ad esempio[11] data poisoning e backdoors nel modello);
- *eavesdropping/ intercettazioni/ hijacking*: azioni volte a intercettare, interrompere o prendere il controllo di una comunicazione di terzi senza consenso (ad esempio divulgazione del modello e furto di dati);
- *attacchi fisici*: azioni che mirano a distruggere, esporre, alterare, disattivare, rubare o ottenere un accesso non autorizzato a risorse fisiche come infrastrutture, hardware o interconnessioni consenso (ad esempio sabotaggio del modello e DDOS);
- *danno non intenzionale*: azioni non intenzionali che causano distruzione, danni a proprietà o persone e che si traducono in un guasto o in una riduzione dell'utilità (ad esempio riduzione dell'accuratezza dei dati e compromissione della selezione delle caratteristiche);
- *guasti o malfunzionamenti*: funzionamento parziale o totalmente insufficiente di un asset hardware o software (ad esempio scarsità di dati e degradazione delle performance del modello);
- *interruzioni*: interruzioni impreviste del servizio o diminuzione della qualità al di sotto di un livello richiesto (ad esempio interruzione dell'infrastruttura/sistemi, interruzione delle reti di telecomunicazione);
- *disastro*: incidente improvviso o catastrofe naturale che causa ingenti danni (ad esempio disastri naturali e fenomeni di cambiamento climatico);
- *legale*: azioni legali di terzi (ad esempio a causa di divulgazione di informazioni personali e profilazione degli utenti).

[10] Negli annessi B e D del documento "AI Cybersecurity Challenges" di ENISA sono riportati, rispettivamente, la tassonomia completa delle minacce e la mappatura con le fasi del ciclo di vita. Per ogni categoria di minacce sono altresì individuati le specifiche minacce, le dimensioni di sicurezza potenzialmente impattate e gli asset impattati.
[11] Il paragrafo "Tassonomie di attacco" tratta le varie categorie di attacchi associati a questo tipo di minacce.

Oltre alle categorie di minaccia sopra elencate, va richiamata l'attenzione su rischi specifici relativi agli agenti IA. Gli agenti ampliano la superficie d'attacco perché combinano capacità decisionali, persistenza di stato e privilegi operativi; per questo motivo possono essere soggetti a tipologie di compromissione non pienamente coperti dalle tassonomie tradizionali.

Tra gli esempi più significativi si segnalano:

- *Escalation di privilegi* e *movimento laterale*: se un agente viene compromesso può utilizzare credenziali o API per propagarsi all'interno dell'infrastruttura.
- *Prompt injection* e *manipolazione delle istruzioni*: input malevoli o manipolazioni possono indurre l'agente a cambiare comportamento e compiere azioni non previste (per esempio divulgare informazioni o eseguire comandi non autorizzati).
- *Persistenza* e *backdoor comportamentali*: l'agente può conservare nello "stato" informazioni o istruzioni corrotte che, in momenti successivi, attivano comportamenti dannosi o difficili da rilevare.
- *Collusione* e *comportamenti emergenti*: quando più agenti interagiscono, possono coordinarsi - volontariamente o perché manipolati - per realizzare azioni dannose o sviluppare comportamenti inaspettati e non desiderati.
- *Esfiltrazione* e *abuse of capabilities*: un agente che ha accesso a dati sensibili o a canali esterni può trasferire informazioni fuori dall'organizzazione o usarle impropriamente.
- *Uso improprio di risorse*: un agente può consumare in modo eccessivo risorse di calcolo o di rete per scopi malevoli (per esempio attacchi DDoS o criptomining), degradando le prestazioni o causando interruzioni.

5.5 Tassonomie di attacco

In aggiunta ai tradizionali attacchi cyber[12] all'infrastruttura ICT ospitante, i sistemi di intelligenza artificiale sono soggetti ad attacchi a componenti specifiche dell'IA quali, ad esempio, i modelli e i dati di addestramento. La conoscenza delle varie tassonomie di attacco permette di porre in essere azioni di protezione e contenimento mirate e contestualizzate.

[12] Attacchi nei quali l'attore malevolo fa uso di TTP (tattiche, tecniche e procedure che caratterizzano il comportamento di un attaccante per raggiungere il proprio obiettivo) tipiche del dominio cyber.

Come tassonomia di riferimento per gli attacchi ai sistemi di IA può essere usata quella sviluppata dal NIST[13] che prevede le seguenti macro-categorie di attacchi:

- *evasion attacks*;
- *poisoning attacks*;
- *privacy attacks*;
- *abuse attacks*;

Le prime tre categorie riguardano sia i modelli di IA predittiva che quelli di IA generativa, mentre l'ultima solo i modelli di IA generativa.

Nei successivi paragrafi sono discussi brevemente queste categorie e possibili strategie di mitigazione.

5.5.1 Evasion attacks

Questa categoria di attacchi ha come obiettivo quello di generare un errore nella classificazione del modello introducendo perturbazioni (spesso impercettibili per l'uomo) negli *input* del modello, denominate *adversarial examples* (esempi avversari).

Questi attacchi sono progettati per sfruttare le vulnerabilità nel processo decisionale del modello. Possono indurre il modello a prevedere un valore desiderato dall'attaccante o determinare una riduzione dell'accuratezza del modello[17].

Come possibili strategie di mitigazione, è possibile applicare l'adversarial training, riaddestrando il modello con *adversarial examples* etichettati correttamente, il randomized smoothing e il formal verification, con i quali si cerca di rendere invariante il modello all'eventuale rumore introdotto da un avversario aumentandone la robustezza.

5.5.2 Poisoning attacks

Questa categoria di attacchi ha come obiettivo degradare le prestazioni di un modello o fargli generare uno specifico risultato alterando (*avvelenando*) i dati di addestramento del modello. Un esempio di

[13] Adversarial Machine Learning - A Taxonomy and Terminology of Attacks and Mitigations: https://nvlpubs.nist.gov/nistpubs/ai/NIST.AI.100-2e2023.pdf

attacco consiste nel cosiddetto *label flipping* nel quale l'attaccante cambia l'etichetta dei dati di addestramento con l'obiettivo di addestrare il modello sulla base dell'etichetta da lui scelta[18].

Questi attacchi possono essere distinti in:

- *availability poisoning*: determinano una violazione della disponibilità del modello tramite una degradazione delle sue prestazioni su tutti i *sample* di dati. Sono rilevabili monitorando le prestazioni del modello, possibili strategie di mitigazione prevedono la sanitizzazione dei dati di addestramento e l'apprendimento robusto (possono, ad esempio, essere addestrati molteplici modelli);
- *targeted poisoning*: determinano una violazione dell'integrità del modello alterandone la previsione su un numero ridotto di *sample* mirati. Possibili strategie di mitigazione prevedono l'implementazione di controlli di sicurezza sull'origine e sull'integrità dei dati;
- *backdoor poisoning*: analogamente agli attacchi *targeted poisoning* determinano una violazione dell'integrità del modello, in questo caso tuttavia l'obiettivo è indurre in errore il modello in risposta a uno specifico *sample* di dati (denominato *trigger*). Possibili strategie di mitigazione prevedono la sanitizzazione dei dati di addestramento, la ricostruzione del *trigger*, l'ispezione e la sanitizzazione del modello;
- *model poisoning*: modificano direttamente il modello addestrato iniettandogli funzionalità malevole. Possono determinare una violazione sia dell'integrità che della disponibilità e avvengono generalmente nell'ambito del cosiddetto apprendimento federato in cui sistemi *client* inviano aggiornamenti locali del modello a un *server* che li aggrega in un modello globale. Sono inoltre possibili in scenari di *supply chain* nei quali sono acquisiti modelli o relativi componenti che sono stati *avvelenati*. Possibili strategie di mitigazione prevedono l'individuazione ed esclusione degli aggiornamenti malevoli o (nel caso di avvelenamenti del modello tramite *backdoor*) l'ispezione e la sanitizzazione del modello.

5.5.3 Privacy attacks

Questa categoria di attacchi ha come obiettivo compromettere le informazioni degli utenti *ricostruendole* a partire dai dati di addestramento. Possono essere distinti in:

- *data reconstruction*, ricostruiscono le informazioni a partire dalle informazioni aggregate;

- *membership inference*, determinano se un particolare *record* è stato incluso nel dataset utilizzato per l'addestramento di un modello, compromettendo le informazioni dell'utente;
- *model extraction*, ottengono informazioni estraendo informazioni sul particolare modello utilizzato, come la sua architettura o i suoi parametri.
- *property inference*, accedono a informazioni globali sulla distribuzione dei dati di addestramento interagendo con il modello.

Per la mitigazione degli attacchi di ricostruzione è stato proposto di far uso di tecniche per migliorare la privacy come la *privacy differenziale*, che – attraverso opportune manipolazioni dei dati – permette di fissare un limite su quanto un attaccante, con accesso ai risultati dell'algoritmo, può inferire su ogni singolo *record* del dataset.

Possibili strategie di mitigazione prevedono inoltre limitazioni al numero di interrogazioni dell'utente al modello, e di rilevamento delle interrogazioni sospette al modello.

5.5.4 Abuse attacks

Questa categoria di attacchi ha come obiettivo alterare il comportamento di un sistema di IA generativa per adattarlo ai propri scopi come, ad esempio, perpetrare frodi, distribuire malware e manipolare informazioni.

Possibili strategie di mitigazione prevedono l'uso di metodi come l'apprendimento rinforzato con *feedback* umano, il filtraggio degli input o il rilevamento di valori anomali di output (*outliers*).

5.6 Obiettivi di sicurezza

In questa sezione sono enunciati gli obiettivi di sicurezza che devono guidare lo sviluppo dell'intelligenza artificiale da parte della Pubblica Amministrazione.

Gli obiettivi sono mutuati dalle *Linee guida per uno sviluppo sicuro dell'Intelligenza Artificiale*[14] promosse dal *National Cyber Security Centre del Regno Unito (NCSC)* e alle quali hanno aderito 35 Agenzie di 18 Paesi, tra le quali l'Agenzia per la Cybersicurezza Nazionale.

Le raccomandazioni fornite per il raggiungimento dei già citati obiettivi sono da intendersi come pratiche di base, le cui implicazioni devono essere comprese e valutate attentamente in fase di realizzazione. Resta in capo a ciascuna Amministrazione la valutazione, in esito alla differente esposizione alle minacce e alla propria analisi del rischio, in merito all'individuazione e conseguente raggiungimento di ulteriori obiettivi di sicurezza per il rafforzamento della sicurezza cibernetica dei propri sistemi di intelligenza artificiale.

5.6.1 Modellare le minacce ai sistemi

La modellazione delle minacce (*threat modeling*) è un processo strutturato che identifica potenziali minacce alla sicurezza, valutandone il rischio e fornendo le necessarie contromisure. Per tale attività è possibile far riferimento alle *Linee guida per la modellazione delle minacce e individuazione delle azioni di mitigazione conformi ai principi del Secure/Privacy by Design* emanate dall'*Agenzia per l'Italia Digitale (AGID)*[15].

Nel modellare le minacce ai sistemi di intelligenza artificiale devono essere individuate le minacce specifiche a questa tipologia di sistemi e compresi i potenziali impatti su utenti, sulle organizzazioni e sulla società della compromissione di un elemento del sistema di IA.

Nel condurre tale esercizio deve essere considerato che anche la tipologia e la sensibilità dei dati utilizzati nei sistemi di IA determinano l'interesse di un attaccante nei confronti del sistema.

Nella modellazione delle minacce occorre inoltre considerare l'intelligenza artificiale stessa quale ulteriore strumento per creare nuovi e più sofisticati vettori di attacco automatizzati[16].

5.6.2 Analisi statica e dinamica del codice

L'adozione di test di sicurezza regolari, come l'analisi statica del codice (SAST) e l'analisi dinamica del codice (DAST), è fondamentale per garantire la sicurezza delle applicazioni di Intelligenza Artificiale

[14] National Cyber Security Centre (NCSC), Guidelines for secure AI system development https://www.ncsc.gov.uk/collection/guidelines-secure-ai-system-development. Le linee guida sono pubblicate anche sul sito di ACN all'indirizzo https://www.acn.gov.it/portale/linee-guida-ia.

[15] https://www.agid.gov.it/it/sicurezza/cert-pa/linee-guida-sviluppo-del-software-sicuro.

[16] L'intelligenza artificiale può essere usata a sua volta per condurre attacchi malevoli con lo scopo di migliorare l'efficacia degli attacchi (ad esempio attraverso malware generato dall'IA, ingegneria sociale, falsi account sui social media alimentati dall'IA, attacchi DDoS potenziati dall'IA, deep-fake, attacchi alle password supportati dall'IA).

prima del loro rilascio. Questi test aiutano a identificare e mitigare le vulnerabilità e contribuiscono a costruire un software più robusto e affidabile riducendo di conseguenza il rischio di attacchi informatici.

- L'analisi statica del codice (SAST) consente di esaminare il codice sorgente senza eseguirlo, analizzando la struttura e la logica del codice per identificare vulnerabilità comuni, come SQL injection, buffer overflow e problemi di gestione delle credenziali. SAST è particolarmente utile nelle fasi iniziali dello sviluppo poiché permette di correggere le problematiche di sicurezza prima che il codice venga integrato nel prodotto finale.
- L'analisi dinamica del codice (DAST) analizza l'applicazione in esecuzione, simulando attacchi reali per identificare vulnerabilità che potrebbero non emergere nell'analisi statica, come problemi di configurazione, interazioni con il database o esecuzione arbitraria di codice che possono manifestarsi solo quando l'applicazione è attiva e interagisce con altri sistemi.

5.6.3 Considerare vantaggi e compromessi in termini di sicurezza nella scelta del modello

La scelta del modello (architettura, configurazione, dati e algoritmo di addestramento, iperparametri) dovrebbe tenere conto dei modelli di minaccia identificati ed essere periodicamente rivalutata sulla base degli sviluppi della ricerca nel campo della sicurezza dell'intelligenza artificiale e della comprensione dell'evoluzione della minaccia.

La scelta del modello dovrebbe prendere in considerazione almeno i seguenti elementi:

- complessità del modello utilizzato in termini di architettura e numero di parametri (che influenzeranno la quantità di dati di addestramento richiesti e la robustezza del modello a variazioni dei dati di ingresso durante l'uso);
- capacità di allineare, interpretare e spiegare i risultati del modello (ad esempio per il debugging, l'audit o la conformità alle normative). Sotto questo profilo, può risultare preferibile utilizzare modelli più semplici e trasparenti rispetto a modelli più complessi ma più difficili da interpretare;
- caratteristiche dei dataset di addestramento, tra cui dimensioni, integrità, qualità, sensibilità, periodo temporale, rilevanza e diversità;

- applicabilità di tecniche di *hardening* al modello (come l'*adversarial training*), di regolarizzazione e/o di rafforzamento della privacy;
- provenienza e catena di approvvigionamento dei componenti dei sistemi di IA, tra i quali il modello, i dati di addestramento e gli strumenti associati.

5.6.4 Progettare in funzione della sicurezza, della funzionalità e delle prestazioni

Una volta definito l'obiettivo del sistema, le scelte di progetto devono essere effettuate ponderando opportunamente le esigenze derivanti dal modello di minaccia e dalle connesse mitigazioni di sicurezza, con quelle derivanti da altri elementi quali funzionalità, esperienza utente, ambiente di implementazione, prestazioni, garanzia, supervisione, requisiti etici e legali.

A seguire sono riportati degli scenari di esempio e gli elementi da tenere in considerazione nelle scelte progettuali:

- *supply chain*: nello scegliere se sviluppare *in house* o acquisire dall'esterno un componente di IA deve essere valutata la sicurezza della catena di approvvigionamento in termini, ad esempio, di adeguatezza alle esigenze, *due diligence* di sicurezza sul fornitore, controlli di sicurezza nell'importazione di modelli di terze parti, sanitizzazione dei dati e degli input;
- *sviluppo sicuro del codice*: lo sviluppo del *software* dei sistemi di IA deve seguire le *best practice* esistenti[17], tutti gli elementi del sistema di IA devono essere sviluppati in ambienti idonei utilizzando pratiche e linguaggi di programmazione che riducono o eliminano le classi di vulnerabilità note ove possibile.
- *Integrazione con altri sistemi*: se i componenti del sistema di IA attivano azioni, come, ad esempio, modificare file o reindirizzare l'output a sistemi esterni, dovrebbero essere applicate restrizioni appropriate includendo, se necessario, meccanismi esterni di *fail-safe*;
- *interazione con gli utenti*: le scelte che riguardano l'interazione del sistema con gli utenti dovrebbero tenere in considerazione i rischi specifici dell'IA, in considerazione dei quali, dovranno essere previste specifiche quali ad esempio l'*output* del sistema non rivela livelli di

[17] *L'Agenzia per l'Italia digitale (AGID)* ha pubblicato le Linee guida per lo sviluppo del software sicuro pubblicate all'indirizzo https://www.agid.gov.it/it/sicurezza/cert-pa/linee-guida-sviluppo-del-software-sicuro.

dettaglio non necessari a un potenziale aggressore, principio del privilegio minimo per limitare l'accesso alle funzionalità del sistema, integrazione delle impostazioni più sicure *by default*, spiegazione agli utenti delle funzionalità più rischiose con richiesta di acconsentirne all'utilizzo (*opt-in*), comunicazione dei casi d'uso non consentiti e, se possibile, informazione di eventuali soluzioni alternative.

5.6.5 Documentare i dati, i modelli e le richieste

La creazione, il funzionamento e la gestione del ciclo di vita di tutti i modelli, i dataset, gli *input* e le richieste al sistema, devono essere documentati.

La documentazione deve includere informazioni rilevanti per la sicurezza quali, ad esempio, l'origine dei dati di addestramento (compresi i dati di *tuning* e il *feedback* umano o di altri operatori), l'ambito e le limitazioni del sistema, gli hash e/o le firme crittografiche, il periodo di conservazione dei dati e le modalità note con le quali il sistema potrebbe generare errori.

5.7 Buone pratiche per la PA

In generale, le procedure e i requisiti di sicurezza di sistemi tecnologico-informatici tradizionali si applicano anche a sistemi IA: ciò implica che le Pubbliche Amministrazioni, prima di integrare soluzioni IA all"interno dei propri sistemi e infrastrutture, DEVONO assicurarsi che l"ambiente preesistente rispetti solidi principi di sicurezza, come ad esempio un"architettura ben strutturata, configurazioni sicure e responsabilità definite chiaramente. In particolare, si faccia riferimento alle Linee guida per lo sviluppo del software sicuro emanate da AgID.

Si riportano di seguito alcune pratiche RACCOMANDATE per le fasi di sviluppo, deployment e manutenzione di un sistema IA.

5.7.1 Sviluppo sicuro

- Adottare le indicazioni delle Linee guida per lo sviluppo del software sicuro di AgID per garantire la sicurezza e l'affidabilità dei sistemi.
- Salvare codice sorgente, eseguibili, file di configurazione, documentazione, nonché modelli, dati, test in un sistema di controllo delle versioni con adeguati controlli di accesso, al fine di garantire che tutte le modifiche siano tracciate, e che venga utilizzato solo codice di fonte nota.

- Se si utilizzano librerie esterne, valutare la loro affidabilità prima di integrarle nel sistema, verificando che non introducano vulnerabilità e che non permettano di usare modelli non attendibili.
- Adottare soluzioni innovative, inclusi metodi crittografici, firme digitali e *checksum* per garantire l'autenticità e l'integrità di ogni artefatto, nonché la riservatezza di informazioni sensibili.
- Se i componenti IA effettuano azioni, come ad esempio modificare file o inoltrare informazioni ad altri sistemi, consentire solo le azioni desiderate, applicando le opportune restrizioni.
- Se si utilizzano API di servizi terzi, effettuare gli opportuni controlli sulle informazioni trasmesse, limitando la diffusione di dati sensibili all'esterno.
- Proteggere eventuali API esposte tramite meccanismi di autenticazione e autorizzazione, e utilizzare protocolli sicuri (e.g. HTTPS).
- Validare e sanitizzare tutti gli input per ridurre il rischio che input malevoli, sospetti o non desiderati vengano passati al sistema (ad esempio, attacchi di tipo *prompt injection*).
- Testare accuratamente il sistema IA per verificare che le proprietà desiderate rimangano inalterate a seguito di aggiornamenti del modello o modifiche sostanziali, applicando le tecniche e le metriche consolidate.
- Adottare sistemi di ripristino manuali e automatici, al fine aumentare l'affidabilità dei sistemi e consentire di ridurre l'impatto sugli utenti nel caso in cui un nuovo modello o un aggiornamento dovesse causare problemi o effetti non desiderati.

5.7.2 Deployment sicuro

- Adottare le *best practice* di sicurezza nell'ambiente di deployment, come l'esecuzione di applicazioni e servizi all'interno di *sandbox* utilizzando container o macchine virtuali (VM), monitorare il traffico di rete e limitarlo tramite firewall opportunamente configurati.
- Proteggere le informazioni sensibili crittografando i dati su disco, utilizzando se possibile moduli di sicurezza hardware (HSM).

- Mantenere *hash* di ogni versione del software, modelli e dati utilizzati in produzione e di relative copie di backup.
- Implementare robusti meccanismi di autenticazione, controlli di accesso e protocolli di comunicazione sicuri (e.g. TLS) per crittografare tutti i dati in transito, anche in rete locale.
- Distinguere utenti con permessi standard e utenti con permessi da amministratore e richiedere l"uso di autenticazione a più fattori (MFA) da parte di tutti gli utenti per consentire l"accesso a servizi.
- Adottare tecniche di controllo di accesso, come *attribute-based access control* (ABAC) o *role-based access control* (RBAC), per limitare l'accesso al solo personale autorizzato.

5.7.3 Gestione e manutenzione sicura

- Seguire durante l"intero ciclo di vita le indicazioni di fornitori di hardware e software, applicando aggiornamenti e *patch* di sicurezza per limitare le vulnerabilità.
- Utilizzare soluzioni di *threath intelligence* e sistemi di rilevamento e prevenzione delle intrusioni (IDS e IPS) per identificare e bloccare tentativi di accesso non autorizzati o sospetti.
- Raccogliere log su comportamento, input, output ed errori del sistema, nel rispetto delle normative sul trattamento automatizzato dei dati personali, e configurare avvisi automatici al fine di rilevare il prima possibile eventi critici o anomali, prestando particolare attenzione a input ripetitivi o molto frequenti.

Effettuare audit di sicurezza, *penetration test* e *vulnerability assessment*. Ingaggiare esperti di sicurezza esterni può aiutare a identificare vulnerabilità o punti critici che potrebbero essere stati sfuggiti a un controllo interno.

6 Neutralità hardware, acceleratori e portabilità dei sistemi di IA

I sistemi di intelligenza artificiale, in particolare quelli basati su modelli di apprendimento automatico e deep learning, possono richiedere elevate capacità computazionali per le fasi di addestramento e di inferenza. Per rispondere a tali esigenze, il mercato ha sviluppato nel tempo acceleratori hardware specializzati (ad es. GPU, TPU e dispositivi analoghi) e ambienti software ottimizzati per il calcolo parallelo.

Tuttavia, molte di queste soluzioni presentano un elevato grado di integrazione verticale tra hardware, software e servizi, che può tradursi in una dipendenza strutturale da specifiche tecnologie o fornitori. Nel lungo periodo, tale dipendenza può limitare la capacità delle amministrazioni di:

- migrare i sistemi verso infrastrutture diverse;
- adattare le soluzioni a contesti operativi eterogenei;
- valorizzare infrastrutture locali, consortili o a controllo pubblico.

6.1 Architetture hardware-agnostic

Per "architetture hardware-agnostic" si intendono sistemi progettati in modo da separare il modello di IA e la logica applicativa dal livello hardware sottostante, consentendo l'esecuzione su infrastrutture differenti senza modifiche sostanziali al codice o al funzionamento del sistema.

Questo approccio non esclude l'uso di acceleratori dedicati, ma evita che essi diventino un requisito imprescindibile per l'operatività del sistema.

In tali architetture, l'eventuale assenza di acceleratori può comportare una riduzione delle prestazioni, ma non l'impossibilità di utilizzo del sistema.

6.2 Ottimizzazione dei modelli e uso delle CPU

L'evoluzione delle tecniche di ottimizzazione dei modelli consente oggi di ridurre significativamente il fabbisogno computazionale dei sistemi di IA. Tra queste tecniche rientrano, a titolo esemplificativo:

- la riduzione delle dimensioni del modello;
- l'uso di rappresentazioni numeriche a precisione ridotta;
- il trasferimento delle capacità di modelli complessi verso modelli più piccoli e specializzati.

L'adozione di tali tecniche permette, in molti casi, l'esecuzione dei modelli anche su infrastrutture basate su CPU general-purpose, ampliando le possibilità di distribuzione dei sistemi di IA e riducendo i costi energetici e operativi.

6.3 Implicazioni per la Pubblica Amministrazione

Nel contesto della Pubblica Amministrazione, la portabilità dei sistemi di IA e la possibilità di esecuzione su hardware eterogeneo assumono una valenza strategica.

Esse contribuiscono infatti a:

- ridurre il rischio di lock-in tecnologico;
- aumentare la resilienza rispetto a vincoli di mercato o di approvvigionamento;
- favorire la nascita e il consolidamento di una filiera locale e nazionale dell'IA;
- migliorare la sostenibilità ambientale dei sistemi attraverso un uso più efficiente delle risorse computazionali.

Per tali ragioni, la neutralità hardware e l'efficienza dei modelli rappresentano fattori abilitanti, più che meri aspetti tecnici, nella progettazione e nella gestione dei sistemi di intelligenza artificiale nella PA.

6.4 Esempi di clausole contrattuali

Di seguito un elenco non esaustivo di esempi di clausole contrattuali, trattate con maggior dettaglio nelle Linee Guida Procurement di IA.

6.4.1 Neutralità hardware

Il sistema di intelligenza artificiale deve essere progettato secondo principi di neutralità hardware, garantendo la possibilità di esecuzione su infrastrutture computazionali eterogenee e evitando dipendenze strutturali da specifiche architetture, acceleratori o tecnologie proprietarie.

6.4.2 Portabilità e reversibilità

Il sistema di intelligenza artificiale deve essere progettato secondo principi di neutralità hardware, garantendo la possibilità di esecuzione su infrastrutture computazionali eterogenee e evitando dipendenze strutturali da specifiche architetture, acceleratori o tecnologie proprietarie.

6.4.3 Efficienza e fallback CPU

Il sistema deve prevedere modalità di esecuzione alternative, basate su tecniche di ottimizzazione dei modelli, che consentano l'utilizzo anche su infrastrutture CPU-only, garantendo continuità operativa e livelli di servizio proporzionati al contesto di utilizzo.

Strumenti

Strumento A – Termini e definizioni

Strumento B – Training, validazione, fine-tuning di Modelli di IA e RAG