POLITECNICO DI TORINO

Corso di Laurea Magistrale in Ingegneria Gestionale

Tesi di Laurea Magistrale

Verso una gestione automatizzata della fatturazione elettronica: integrazione del machine learning nei sistemi informativi della Pubblica Amministrazione

Relatrice
Eliana Pastor
Correlatore
Maurizio Maffullo

Candidato
Emanuele Disegna

Anno Accademico 2024/2025

Indice

1 Introduzione

1.1 Contesto

Nel primo anno di implementazione della fatturazione elettronica, l'Agenzia delle Entrate ha registrato oltre 2 miliardi di documenti, di cui il 55% riguarda operazioni tra soggetti passivi (B2B - Business to Business) e il 44% verso consumatori finali (B2C - Business to Consumer) [1]. Nonostante solo l'1% delle fatture sia destinato alla Pubblica Amministrazione (B2G - Business to Government), si tratta comunque di circa 20 milioni di documenti, un numero significativo che richiede una gestione efficace.

L'azienda Leonardo S.p.a. supporta come consulente un importante Cliente, appartenente alla Pubblica Amministrazione, nella gestione, ottimizzazione e manutenzione dei processi contabili che avvengono sui loro sistemi. Il Cliente utilizza il Sistema di Contabilità Generale dello Stato, noto come SICOGE, il quale, per sfruttare i vantaggi di una gestione semplificata dei documenti contabili, interagisce con uno dei moduli dell'ERP SAP S/4HANA, chiamato SAP Vendor Invoice Management (VIM).

Nonostante i significativi passi avanti nella digitalizzazione dei sistemi e delle procedure, questi processi contabili includono ancora attività manuali di routine, caratterizzate da bassa complessità ma da un alto impatto in termini di tempo e risorse impiegate.

In questa tesi verrà proposta un'applicazione del machine learning da inserire nel processo di contabilizzazione delle fatture indirizzate al Cliente, con l'obiettivo di aumentare la produttività degli operatori attualmente impiegati nella gestione contabile, liberandoli per lo svolgimento di attività a maggior valore aggiunto. Saranno analizzati i sistemi e gli stakeholders coinvolti ed applicata la metodologia CRISP-ML per quanto riguarda l'impostazione e la descrizione del caso.

Le principali sfide da affrontare includono la gestione dei grandi volumi di dati, la protezione della privacy e la necessità di integrare soluzioni innovative con i sistemi SAP esistenti.

1.2 I sistemi informativi

La Pubblica Amministrazione (PA), nelle sue diverse declinazioni territoriali e politiche, riceve, invia e gestisce ogni giorno migliaia di documenti in formati, dimensioni e natura diversi. Questi documenti e le informazioni in essi contenute sono memorizzati ed elaborati da numerosi sistemi informativi.

Un sistema informativo (SI) è un insieme di risorse che memorizzano, elaborano e gestiscono informazioni per supportare il funzionamento di un'organizzazione. Esso facilita le decisioni, il controllo dei processi e l'esecuzione delle attività principali, come produzione o servizi. Il SI integra aspetti legati alla struttura organizzativa, con l'allocazione di compiti e responsabilità, al management, con strategie di gestione, e alla tecnologia, con strumenti che supportano le attività aziendali [2]. Si possono individuare due principali tipi di azioni effettuabili dal SI:

- Automazione di attività (altrimenti svolte manualmente): il SI applica al proprio interno logiche che permettono di eliminare parte del lavoro che prima era svolto manualmente, lasciando gli operatori svolgere più attività rispetto a prima nello stesso periodo di tempo, aumentando la produttività.

- Supporto a processi decisionali: attraverso la propria caratteristica di aggregatore di informazioni, il SI riesce a fornire una visualizzazione d'insieme agli operatori.

Molte organizzazioni, pur avendo caratteristiche uniche, condividono necessità comuni, il che rende possibile il riuso dei sistemi informativi in contesti simili. Esistono infatti famiglie di SI progettati per supportare specifici processi aziendali, dai pacchetti standard, acquistabili ed utilizzabili direttamente, ai sistemi complessi, necessariamente da personalizzare. Conoscere queste famiglie consente di valutare se convenga sviluppare un SI su misura o adottare una soluzione esistente adattabile alle esigenze dell'organizzazione. I SI possono essere classificati secondo la Piramide di Anthony, mostrata in Figura [3], che possiede due dimensioni: il *livello organizzativo* e la *funzione aziendale*. Per ogni intersezione livello-funzione è possibile includere dei processi e dei sistemi informativi che li integrano.

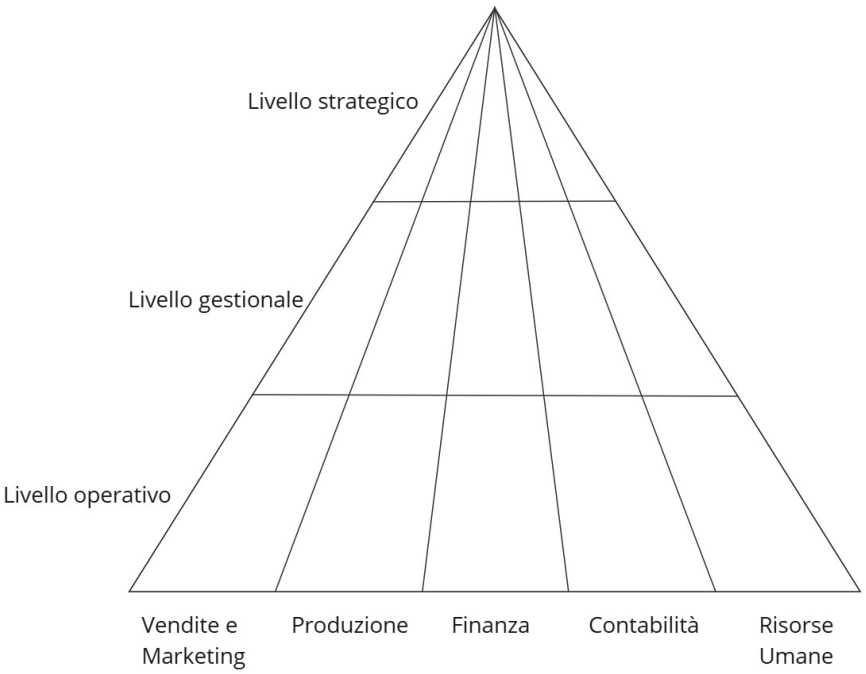

Figura 1: Piramide di Anthony, immagine tratta dal libro [2]

I livelli organizzativi sono tre:

- Livello operativo: questo livello si occupa di gestire le attività quotidiane, composte da transazioni ed eventi. Fanno parte di questo gruppo i SI di tipo:
  - Transaction Processing Systems (TPS)
- Livello gestionale: sono gestiti in questo livello i processi, che si svolgono al livello operativo. I principali SI che operano in questo livello sono:
  - Management Information Systems (MIS)
  - Decision Support Systems (DSS)
- Livello strategico: in questo livello sono visualizzati indicatori ottenuti dall'analisi di grandi volumi di dati. In generale sono analizzati i ricavi, i costi, i tempi di risposta e la qualità dei servizi o prodotti offerti. Comprendono i:
  - Executive Support Systems (ESS)

Le funzioni aziendali invece sono molteplici e variabili in base all'organizzazione che implementa il SI, alcuni esempi sono mostrati in Figura 1.

Tra i SI più diffusi, operanti a livello gestionale, si possono trovare i seguenti:

- Enterprise Resource Planning (ERP): SI che cerca di integrare l'intera gamma dei processi e delle funzioni aziendali per presentare una visione olistica dell'azienda da un'unica architettura informatica e delle informazioni [4].

- Customer Relationship Management (CRM): SI utile a garantire che l'organizzazione costruisca relazioni reciprocamente vantaggiose a lungo termine con i suoi clienti strategicamente importanti [5].

- Manufacturing Execution System (MES): SI che ottimizza, monitora e controlla in tempo reale i processi di produzione, collegando le operazioni sul campo con la gestione aziendale [6].

- Supply Chain Management (SCM): SI che permette di coordinare e ottimizzare tutte le attività legate alla creazione e distribuzione di beni o servizi. L'SCM consente di gestire il flusso di materiali, informazioni e risorse finanziarie lungo l'intera catena, dalla fornitura di materie prime fino alla consegna del prodotto finale al cliente [7].

Per rispondere in modo efficace alle esigenze dell'azienda, inoltre, i SI sono spesso divisi in moduli, ciascuno dedicato a una specifica funzione aziendale. "Un modulo rappresenta un blocco di funzionalità software che supporta una fase di un processo aziendale, omogeneo per frequenza, attore e profilo di casi d'uso" [8]. Questa divisione modulare permette di gestire i dati e i processi in modo autonomo, senza interferenze tra le diverse aree del sistema. Inoltre, facilita l'aggiornamento e la manutenzione, consentendo modifiche in un singolo modulo senza compromettere il funzionamento dell'intero sistema. Grazie alla modularità, le organizzazioni possono anche adattare e scalare il sistema, aggiungendo o rimuovendo moduli in base alle necessità, senza dover riprogettare l'intera struttura.

1.2.1 Cos'è SAP

SAP (Systems, Applications, and Products in Data Processing) [9] è una delle principali aziende di software al mondo. Si è affermata come leader globale nello sviluppo di software gestionali, offrendo soluzioni integrate per supportare le attività di aziende di qualsiasi dimensione e settore industriale. Il sistema SAP è noto soprattutto per il suo software Enterprise Resource Planning (ERP), la cui prima versione fu chiamata R/1, che consente alle organizzazioni di ottimizzare i processi operativi e gestire in modo efficace risorse come finanze, logistica, risorse umane e produzione, attraverso una struttura modulare customizzabile.

La forza di SAP risiede nella sua capacità di centralizzare le informazioni aziendali in un'unica piattaforma, permettendo a diversi reparti di condividere dati in tempo reale. Questo approccio consente una visibilità completa delle operazioni aziendali, migliorando la gestione delle risorse e il processo decisionale. Oltre ai moduli, su SAP chiamati *Add-On*, legati all'ERP, SAP offre una vasta gamma di soluzioni software che coprono molte delle funzioni aziendali esistenti e citate in precedenza.

Il prodotto ERP attuale di SAP è S/4HANA (acronimo di Business Suite 4 SAP HANA), pacchetto software lanciato nel 2015 e basato su un'*architettura monolitica*. SAP S/4HANA adotta una divisione logica a tre livelli: presentazione, applicazione e database. Il livello presentazione è responsabile dell'interfaccia utente e dell'interazione con gli utenti e, tramite SAP Fiori, offre una grafica gradevole e nel contempo funzionale alle esigenze di business. Il livello applicazione gestisce la logica aziendale, ma grazie all'architettura di S/4HANA, parte dell'elaborazione dei dati è delegata al database. Infatti, il livello database si basa su HANA, un database in-memory[1] (proprietario di SAP). Questo approccio riduce il carico di lavoro sul livello applicazione, poiché i dati vengono processati e filtrati già nel database, migliorando notevolmente la velocità e l'efficienza complessiva del sistema.

Inoltre, per rispondere alle crescenti esigenze di digitalizzazione, SAP sta evolvendo verso un approccio più moderno con il Business Technology Platform (BTP), che a differenza di S/4HANA, adotta un'*architettura a microservizi*, garantendo maggiore modularità, scalabilità e facilità di integrazione. Il BTP è progettato per lavorare in ambienti cloud, on-premise o ibridi e utilizza SAP S/4HANA come nucleo di calcolo, offrendo un'infrastruttura flessibile e moderna per sostenere le aziende nel loro percorso di innovazione e digitalizzazione.

1.3 Gli stakeholders coinvolti

Nell'analisi di un caso di studio si rende necessario effettuare l'identificazione degli stakeholder coinvolti, intesi come "tutti i soggetti, individui od organizzazioni, attivamente coinvolti in un'iniziativa economica (progetto, azienda), il cui interesse è negativamente o positivamente influenzato dal risultato dell'esecuzione, o dall'andamento, dell'iniziativa

[1] I database in-memory, noti anche come main-memory database o memory-resident database, memorizzano i dati nella memoria principale di un computer (RAM) anziché su dischi rigidi o unità a stato solido (SSD). Questa tecnologia consente tempi di interrogazione più rapidi, ma limita la quantità di dati gestibili. [10]

e la cui azione o reazione a sua volta influenza le fasi o il completamento di un progetto o il destino di un'organizzazione" [11].

Nel contesto di questo progetto di tesi, si possono identificare come stakeholders:

- Leonardo S.p.A., che all'interno dei suoi uffici Cyber & Security di Genova, mi ha supportato nell'implementazione del progetto e che ha in commessa la digitalizzazione di alcune componenti dei sistemi informativi della Pubblica Amministrazione. La società, erede di Finmeccanica, è una delle principali aziende globali nel settore della difesa, sicurezza informatica e aerospazio. Con oltre 70 anni di storia, rappresenta un'eccellenza tecnologica italiana, impiegando più di 50.000 persone in oltre 20 Paesi, con una forte presenza in Europa, Nord America, Sud America e Asia.

- Il Cliente, una società specializzata nello sviluppo e manutenzione di soluzioni digitali per la Pubblica Amministrazione.

1.4 Lavori correlati

Nell'ambito della gestione contabile, il machine learning sta assumendo un ruolo sempre più rilevante, grazie alla sua capacità di migliorare l'efficienza dei processi, eliminando la necessità di svolgere attività monotone e ripetitive. Come riportato in [12], uno dei primi impieghi è stato l'estrazione di informazioni da fatture cartacee, analizzate mediante lettori ottici. Inizialmente, tale processo si basava sulla posizione dei caratteri [13, 14], per poi evolvere verso approcci generalizzati tramite reti neurali [15–17].

Altri utilizzi si sono concentrati sull'identificazione di anomalie presenti nelle fatture, un problema sempre più critico a causa della crescente complessità dei processi aziendali e dell'incremento dei volumi di dati. Tali anomalie, se non corrette, possono causare gravi perdite economiche. In particolare, gli studi [18–20] propongono un approccio non supervisionato basato su reti neurali per individuare transazioni errate o potenzialmente fraudolente nei dati contabili. Parallelamente, lo studio [21] ha impiegato il machine learning per stimare la probabilità che una fattura venga pagata puntualmente, oltre a prevedere la durata di eventuali ritardi nei pagamenti.

I contributi più vicini agli obiettivi di questa tesi sono però [22, 23], che adottano rispettivamente insiemi di regole e tecniche di machine learning per predire il numero di conto su cui registrare l'importo di una fattura. In questi studi, le performance degli algoritmi di machine learning sono state confrontate con un approccio deterministico basato su regole derivate dall'esperienza professionale dei contabili. I risultati evidenziano

che, sebbene il machine learning presenti un potenziale promettente, le sue prestazioni non superano significativamente quelle del sistema basato su regole.

È importante evidenziare che il processo analizzato in questa tesi non si basa su un sistema strutturato di regole predefinite, poiché le regole applicate dagli operatori non sono esplicitamente codificate, ma derivano dalla loro esperienza e competenza personale, rendendole non prevedibili a priori. Per questo motivo, si è deciso di adottare il machine learning, con l'obiettivo di migliorare progressivamente le performance del processo e superare le limitazioni legate alla natura implicita delle regole operative.

1.5 Struttura della tesi

In questa tesi si è deciso di utilizzare un approccio classico circa la suddivisione degli argomenti. Il lavoro è stato strutturato in quattro sezioni: introduzione, materiali e metodi, risultati sperimentali e discussione, conclusioni. L'introduzione, di cui questa sezione fa parte, ha avuto il compito di fornire il contesto al lavoro di tesi, in materiali e metodi sono descritte, invece, le attività sperimentali e i relativi risultati vengono poi presentati e discussi nel capitolo successivo. Infine, sono tratte le conclusioni.

Con l'obiettivo di fornire le informazioni riguardo gli esperimenti in modo pulito ed ordinato, si è deciso di seguire una particolare strutturazione.

Il *CRoss-Industry Standard Process for Machine Learning* (CRISP-ML), come spiegato in [24], è uno standard di processo specifico per lo sviluppo di applicazioni utilizzanti il machine learning, emerso da pochi anni come successore del *CRoss-Industry Standard Process for Data Mining* (CRISP-DM). Questo ultimo standard infatti, sebbene sia riconosciuto come uno dei migliori standard per l'implementazione di processi di analisi dei dati in contesti industriali, non è uno standard specifico per applicazioni di machine learning, le quali hanno esigenze di essere valide ed operative per un lungo periodo di tempo, necessitando quindi di manutenzione.

Con l'aggiunta e la modifica di alcune fasi al CRISP-DM, il CRISP-ML si è affermato come lo standard preferito per l'implementazione del machine learning in contesti industriali. Questo comprende le seguenti fasi, elencate nella Tabella 1:

1. Business and Data Understanding: la prima fase consiste nella descrizione del caso di studio, dei dati a disposizione, degli obiettivi specifici che si vogliono raggiungere, dei criteri di successo dell'applicazione e dello studio di fattibilità;

2. Data Preparation: la seconda fase descrive le procedure di pulizia dei dati e le operazioni di preprocessamento necessarie a preparare i dati per le fasi successive;

3. Modeling: l'obiettivo generale di questa fase è costruire un modello che, rispettando tutti i vincoli e requisiti definiti in precedenza, sia perfetto alle esigenze aziendali;

4. Evaluation: questa fase comprende la valutazione delle performance del modello, utilizzando un insieme di dati separato, chiamato dataset di test;

5. Deployment: sono qui definite la strategia di implementazione e l'architettura nella quale inserire l'applicazione, tenendo in considerazione le esigenze tecniche e funzionali;

6. Monitoring and Maintenance: questa fase descrive le procedure che permettono all'applicazione di rimanere utilizzabile nel tempo.

Sebbene in linea teorica tutte le fasi siano da coprire durante un progetto di machine learning, nella nostra applicazione soltanto le prime quattro fasi saranno sviluppate nel dettaglio. Le ultime due fasi infatti richiedono un grande coinvolgimento e coordinazione da parte degli stakeholder, fattori che sono in contrasto con le esigenze e le tempistiche di un lavoro di tesi. Queste fasi sono state quindi descritte, nel capitolo di presentazione dei risultati sperimentali e discussione, mettendo in evidenza una possibile implementazione tecnica e le procedure di monitoraggio e manutenzione a corollario della soluzione.

CRISP-ML	**CRISP-DM**
Business and Data Understanding	Business Understanding
Data Preparation	Data Understanding and Data Preparation
Modeling	Modeling
Evaluation	Evaluation
Deployment	Deployment
Monitoring and Maintenance	-

Tabella 1: Le fasi di CRISP-ML e le fasi di CRISP-DM

2 Materiali e metodi

2.1 Business and data understanding

2.1.1 Descrizione del caso

Nel 2020, Leonardo ha supportato il Cliente in un grande aggiornamento dei propri sistemi gestionali basati su SAP. Una delle novità aggiunte è stata quella di una funzionalità di esecuzione massiva della contabilizzazione delle fatture elettroniche in arrivo sul modulo di SAP S/4HANA dedicato alla gestione contabile, chiamato Vendor Invoice Management (VIM). Questa funzionalità ha permesso una semplificazione delle operazioni manuali effettuate dall'operatore, il quale può da allora contabilizzare molte fatture in un'unica soluzione associando manualmente una o più fatture ad un modello di contabilizzazione, comunemente chiamato *template*. In seguito ad un utilizzo intenso del sistema e delle sue funzionalità, negli anni trascorsi da questo aggiornamento ad oggi novembre 2024, sono emersi spazi di miglioramento. Nello specifico. si è osservato che questa fase di assegnazione manuale, pur migliorando la situazione rispetto al passato, è comunque dispendiosa in termini di tempo, poiché richiede un'analisi visiva dettagliata delle singole fatture. Infatti, la velocità e la correttezza dell'associazione fattura-template sono dipendenti dall'esperienza e dalle competenze degli operatori coinvolti, i quali, sulla base di intuizioni maturate nel tempo, decidono quale modello utilizzare osservando le specifiche caratteristiche di ciascuna fattura. Questo rende il processo suscettibile ad errori ed instabilità, soprattutto in situazioni in cui il volume di documenti da elaborare è elevato, o in caso di nuova documentazione che potrebbe non rientrare in modelli precedenti.

Vediamo adesso come si articola il processo attuale, descrivendo la struttura di una fattura elettronica, la sua emissione da parte dell'operatore economico interessato, il suo arrivo e la sua gestione sui sistemi informativi del Cliente.

La fatturazione elettronica Si può definire una fattura elettronica, seguendo la Circolare del 19/10/2005 n. 45 emessa dall'Agenzia delle Entrate [25], come il documento informatico che è preparato in formato elettronico e con modalità tali che siano garantiti i dati contenuti e l'attribuzione univoca del documento al soggetto emittente, ottenuta tramite firma digitale autenticata. Le differenze con la fattura cartacea sono essenzialmente due:

- Una fattura elettronica deve essere compilata con l'utilizzo di un dispositivo elettronico quale smartphone, tablet o pc;

- Una fattura elettronica deve essere trasmessa al cliente tramite il Sistema di Interscambio (SdI) messo a disposizione dall'Agenzia delle Entrate. Nel paragrafo 2.1.1 sarà spiegato il suo compito.

La fatturazione elettronica in Italia è stata introdotta dal Decreto Legislativo n. 52 del 21 marzo 2014. Inizialmente fu resa obbligatoria solo per i fornitori delle Pubbliche Amministrazioni Centrali a partire dal 6 giugno 2014. Successivamente, l'obbligo è stato esteso anche alle Pubbliche Amministrazioni Locali a partire dal 31 marzo 2015, comprendendo quindi tutte le transazioni B2G (business-to-government). Questo primo passo verso la digitalizzazione è stato confermato dal Decreto Legislativo n. 127/2015, che ha mantenuto l'obbligo per le transazioni con la Pubblica Amministrazione [26].

Con la Legge di Bilancio 2018 (Legge n. 205/2017), l'obbligo di fatturazione elettronica è stato ulteriormente ampliato. Dal 1° gennaio 2019 sono state incluse tutte le transazioni tra privati, sia B2B (business-to-business) che B2C (business-to-consumer). Tuttavia, alcune categorie, come i piccoli produttori agricoli e le associazioni sportive dilettantistiche, sono state esentate. Questo ampliamento ha rappresentato un cambiamento significativo, segnando una svolta fondamentale nella digitalizzazione del sistema fiscale italiano [26].

Il passaggio al sistema di fatturazione elettronica ha portato notevoli vantaggi sia ai soggetti emittenti, ovvero i privati cittadini e le imprese, che alla Pubblica Amministrazione. Il principale vantaggio è certamente la dematerializzazione dei processi contabili, che consente alle aziende di ridurre drasticamente l'uso della carta e i relativi costi di archiviazione e gestione dei documenti. La conservazione elettronica non consiste semplicemente nel salvare il file della fattura sul proprio pc, ma richiede un processo formalizzato e regolamentato dalla legge, secondo quanto previsto dal Codice dell'Amministrazione Digitale. Attraverso questo processo di conservazione a norma, è possibile garantire la reperibilità, la leggibilità e l'integrità delle fatture nel tempo, consentendo in qualsiasi momento di recuperare l'originale della fattura o di altri documenti informatici sottoposti a conservazione. L'Agenzia delle Entrate offre gratuitamente questo servizio per tutte le fatture emesse e ricevute tramite il SdI [27].

Inoltre, la fatturazione elettronica è uno strumento efficace per combattere l'evasione fiscale, consentendo un controllo più veloce ed automatizzato delle transazioni economiche, riducendo i tempi di accertamento e migliorando la raccolta di dati utili per l'analisi economica.

La struttura Le fatture elettroniche sono create, trasmesse, gestite e ricevute sempre in file di formato XML (eXtensible Markup Language), il quale permette di avere documenti strutturati e facilmente scambiabili. Tre sono le sezioni che compongono una fattura elettronica, la *testata* (o **Header**), il *corpo* (o **Body**) e la *ds:Signature*, utilizzata per la firma elettronica, e che a differenza delle altre due, è facoltativa e di scarsa rilevanza per le analisi che seguiranno. Nell'Header e nel Body inoltre sono presenti alcune sottosezioni obbligatorie e altre facoltative.

Nella Figura 2 è possibile visualizzare lo schema generale di una fattura elettronica. Verranno poi approfondite le sue componenti. In tutte le Figure successive, sono rappresentat

Sono contenute nell'Header le informazioni fondamentali e riassuntive per identificare univocamente il documento contabile, i soggetti coinvolti e permettere al SdI di gestire, tracciare e indirizzare correttamente la fattura, assicurandosi della conformità alle normative fiscali. Tra le sottosezioni obbligatorie, sempre presenti, sono inclusi i dati di trasmissione, come il numero progressivo di fatturazione e il codice destinatario, i dati relativi al cedente/prestatore, ovvero il soggetto emittente fattura (o fornitore), tra i quali i dati anagrafici e fiscali, e i dati relativi al cessionario/committente, ovvero colui che ha richiesto il bene o servizio fatturato. Le sottosezioni concernenti il rappresentante fiscale italiano del fornitore o prestatore, nel caso la sua consulenza venga utilizzata, possono essere considerate opzionali. Ciò vale anche per le informazioni relative a un eventuale terzo intermediario o ad un soggetto emittente di fattura che agisce in sostituzione del fornitore o prestatore. Infine, qualora il soggetto emittente fosse distinto dal fornitore o prestatore, i dati inerenti a tale soggetto sono anch'essi opzionali.

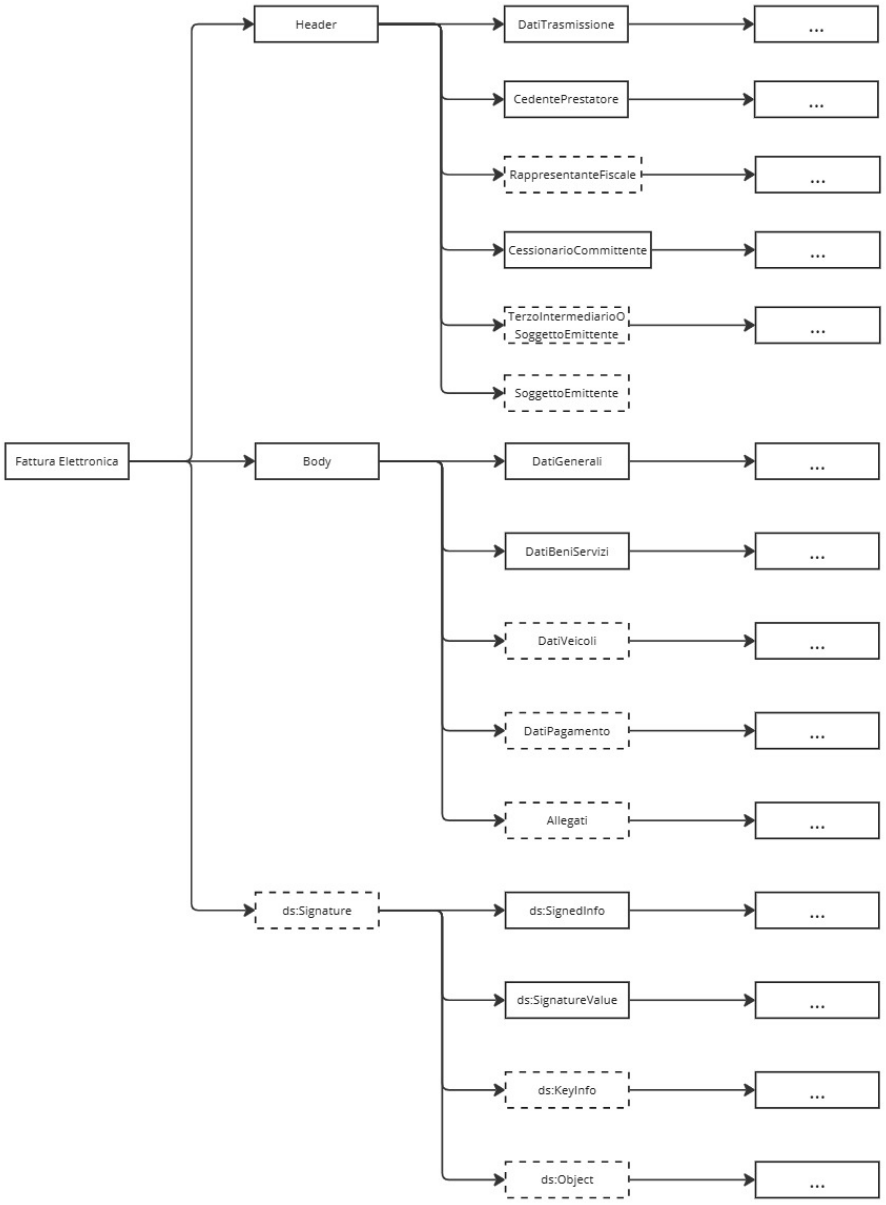

Figura 2: Schema XML di una fattura elettronica: gli elementi in linea continua sono obbligatori e quelli in linea tratteggiata sono facoltativi

Vediamo nel dettaglio i componenti dell'Header:

- Dati Trasmissione: In questa sottesezione sono presenti i dati che permettono la corretta gestione e trasmissione della fattura attraverso il SdI.

Campo	**Descrizione**
IdTrasmittente	Rappresenta l'identificativo univoco del soggetto trasmittente ed è costituito, per i soggetti residenti in Italia (sia persone fisiche sia giuridiche), dal codice fiscale preceduto dal prefisso "IT". Per i soggetti non residenti, invece, l'identificativo corrisponde al numero IVA,.
ProgressivoInvio	Numerazione progressiva stabilita dal soggetto emittente fattura.
FormatoTrasmissione	Indica il codice identificativo del tipo di trasmissione effettuata.
CodiceDestinatario	Include un codice di 7 caratteri assegnato dallo SdI ai soggetti che hanno accreditato un canale di ricezione. E' anche chiamato **Codice IPA** (Indice delle Pubbliche Amministrazioni) e rappresenta l'indice univoco di riferimento degli enti pubblici.
ContattiTrasmittente	Sono i dati relativi al soggetto trasmittente.
PecDestinatario	E' l'indirizzo di posta certificata al quale inviare la fattura.

Tabella 2: Descrizione campi Dati Trasmissione

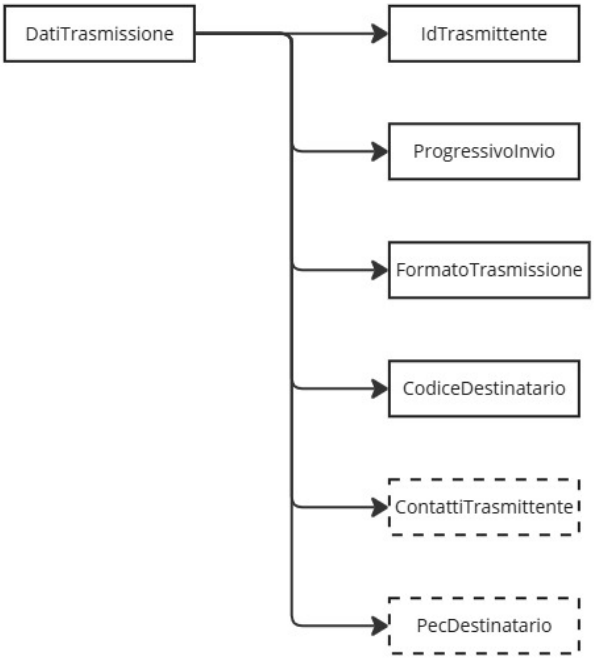

Figura 3: Schema dei Dati di Trasmissione

- Cedente / Prestatore: E' il soggetto e che vende i beni o fornisce il servizio al cliente, ovvero il cessionario / committente. Si configura quindi come il fornitore. È responsabile della creazione, della compilazione e dell'invio della fattura al SdI.

Campo	**Descrizione**
DatiAnagrafici	Sono esplicitati i dati anagrafici del soggetto tra i quali, partita IVA, codice fiscale, regime fiscale, nome, cognome, titolo, e se disponibili anche le informazioni riguardo l'albo professionale.
Sede	Sezione contenente i dati relativi alla sede del cedente / prestatore. Per le società, si riferisce alla sede legale, mentre per le ditte individuali e i lavoratori autonomi corrisponde al domicilio fiscale.
StabileOrganizzazione	Sezione facoltativa da valorizzare qualora il cedente / prestatore non è residente in Italia.
IscrizioneREA	Sezione facoltativa da valorizzare in caso di società iscritte nel registro delle imprese ai sensi dell'art. 2250 del codice civile.
Contatti	Informazioni di contatto del cedente / prestatore
RiferimentoAmministrazione	Codice di riferimento ai fini fiscali

Tabella 3: Descrizione campi Cedente / Prestatore

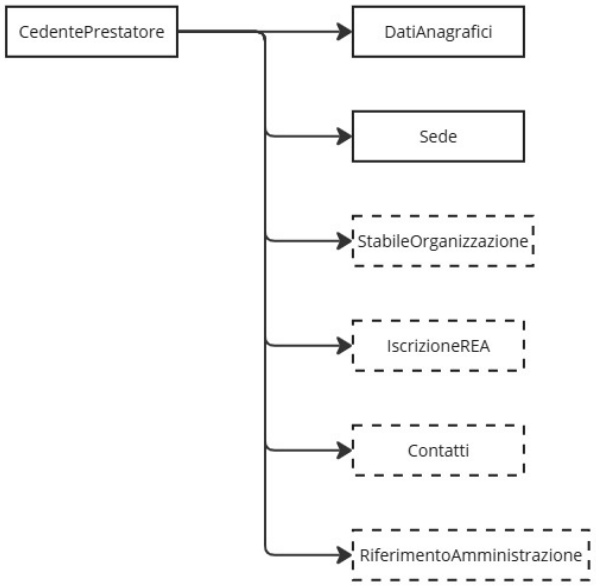

Figura 4: Schema di Cedente / Prestatore

- Rappresentante fiscale del cedente / prestatore

Campo	Descrizione
DatiAnagrafici	Sono esplicitati, se disponibili, i dati anagrafici del soggetto tra i quali, partita IVA, codice fiscale, nome, cognome, titolo.

Tabella 4: Descrizione campi Rappresentante Fiscale

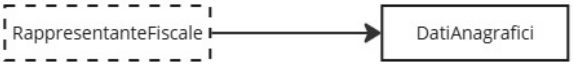

Figura 5: Schema del rappresentante fiscale del cedente / prestatore

- Cessionario / Committente: E' il soggetto che riceve il bene o servizio e che quindi deve effettuare il pagamento al cedente / prestatore. Questo può essere una società, un libero professionista, il consumatore finale oppure un ente afferente alla PA.

Campo	Descrizione
DatiAnagrafici	Sono esplicitati i dati anagrafici del soggetto tra i quali, partita IVA, codice fiscale, regime fiscale, nome, cognome, titolo, e se disponibili anche le informazioni riguardo l'albo professionale.
Sede	Sezione contenente i dati relativi alla sede del cedente / prestatore. Per le società, si riferisce alla sede legale, mentre per le ditte individuali e i lavoratori autonomi corrisponde al domicilio fiscale.
StabileOrganizzazione	Sezione facoltativa da valorizzare qualora il cessionario / committente non è residente in Italia e soltanto in caso di fatture tra privati (escluse quindi le fattura verso la PA).
RappresentanteFiscale	Sezione facoltativa da valorizzare in caso di società iscritte nel registro delle imprese ai sensi dell'art. 2250 del codice civile e soltanto in caso di fatture tra privati (escluse quindi le fattura verso la PA).

Tabella 5: Descrizione campi Cessionario / Committente

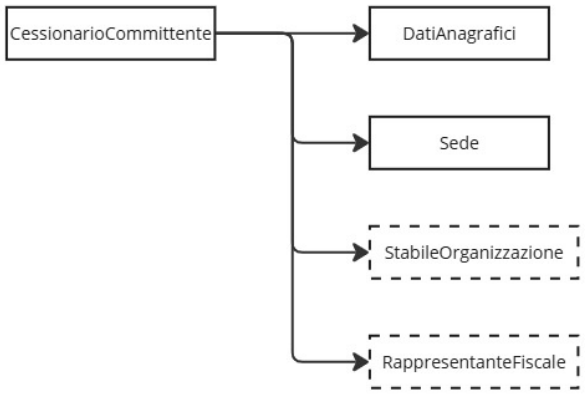

Figura 6: Schema di Cessionario / Committente

- Terzo intermediario o soggetto emittente: E' un soggetto esterno incaricato dal cedente / prestatore per gestire la creazione, l'invio o la conservazione della fattura attraverso il Sistema di Interscambio (SdI). Questo ruolo può essere ricoperto da un commercialista, una società di consulenza fiscale, o un provider specializzato di servizi digitali. L'intermediario si occupa di compilare la fattura secondo le normative vigenti, assicurando che sia correttamente formattata e contenente tutti i dati obbligatori. Inoltre, può gestire l'invio e la conservazione digitale della fattura in conformità con le disposizioni di legge, permettendo al cedente / prestatore di delegare questi adempimenti tecnici e fiscali.

Campo	**Descrizione**
DatiAnagrafici	Sono esplicitati, se disponibili, i dati anagrafici del soggetto tra i quali, partita IVA, codice fiscale, nome, cognome, titolo, denominazione.

Tabella 6: Descrizione campi terzo intermediario o soggetto emittente

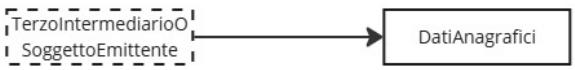

Figura 7: Schema di terzo intermediario o soggetto emittente

- Soggetto emittente: Quando un documento viene emesso da un soggetto diverso dal cedente / prestatore, è importante specificare questo elemento. Consiste in un codice che identifica se la fattura è stata emessa direttamente dal cessionario

/ committente, oppure da un soggetto terzo che agisce per conto del cedente / prestatore.

Nel Body invece sono specificati i dati sul contenuto della transazione, come i beni o servizi scambiati, i dati di pagamento e di trasporto. I contenuti obbligatori sono i dati generali, come tipo documento, divisa, importo e ritenute fiscali, i dati sui beni e servizi scambiati, dettagliati in una lista, e i dati di pagamento. E' possibile anche che siano inseriti maniera facoltativa i dati relativi al trasporto, ai veicoli coinvolti nella fattura, e anche i dati riferenti ad eventuali allegati trasmessi insieme alla fattura.

- Dati Generali: In questa sottosezione, sono riassunti ed esplicitati le informazioni di base fondamentali riguardanti il documento stesso, come la tipologia di documento, la valuta, le ritenute fiscali, il bollo da pagare, la data di emissione e i dati di trasporto.

Campo	**Descrizione**
Dati Generali Documento	E' l'unica parte che è necessario compilare relativamente ai dati generali. Contiene la tipologia di documento, la valuta, la data, il numero progressivo della fattura e altri dati di carattereste contabile e finanziario, come l'importo, il bollo e le ritenute fiscali.
DatiOrdineAcquisto	Dati relativi all'ordine di acquisto del bene o servizio ad oggetto della fattura.
DatiContratto	Dati relativi al contratto di acquisto del bene o servizio ad oggetto della fattura.
DatiConvenzione	Dati relativi alla convenzione collegata alla fattura.
DatiRicezione	Dati relativi alla ricezione del bene o servizio ad oggetto della fattura.
DatiFattureCollegate	Dati relativi alle eventuali fattura collegate al documento stesso.
DatiSAL	Questi dati sono relativi allo Stato Avanzamento Lavoro e sono presenti se richiesti dal destinatario.
DatiDDT	Sono i dati relativi al Documento Di Trasporto collegato alla fattura, se presente.
DatiTrasporto	Qui sono espressi i dati relativi al corriere e alla consegna del bene oggetto della fattura.
FatturaPrincipale	Dati inclusi nei casi di fatture relative a operazioni accessorie emesse da autotrasportatori per beneficiare delle agevolazioni sulla registrazione e sul pagamento dell'IVA.

Tabella 7: Descrizione campi Dati Generali

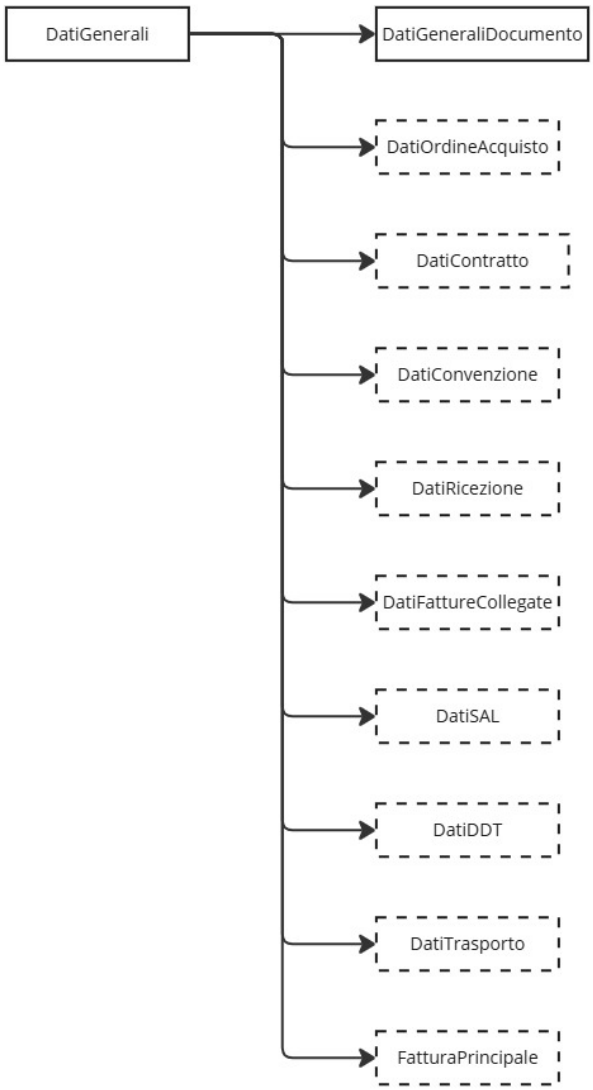

Figura 8: Schema dei Dati Generali

- Dati Beni / Servizi: Sono qui descritti i beni o servizi oggetto della fattura e i dati di riepilogo finanziari.

Campo	Descrizione
DettaglioLinee	Si tratta di un parte che descrive la natura, qualità e quantità dei beni o servizi oggetto dell'operazione. Questa parte è ripetuta per ciascuna riga di dettaglio del documento, ovvero per ciascun bene o servizio in oggetto.
DatiRiepilogo	Questa parte contiene i dati relativi all'imponibile fiscale, all'aliquota IVA, specificata per ogni bene o servizio, e la derivante imposta.

Tabella 8: Descrizione campi Dati Beni / Servizi

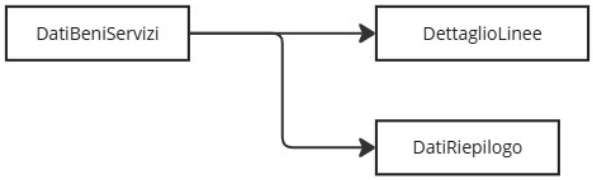

Figura 9: Schema dei Dati Beni / Servizi

- Dati Veicoli: Sottosezione presente nei casi di cessioni tra paesi, membri dell'Unione Europea, di mezzi di trasporto nuovi. .

Campo	Descrizione
Data	Data di prima immatricolazione o di iscrizione del mezzo nei pubblici registri.
TotalePercorso	Indica il totale dei chilometri percorsi, oppure totale ore navigate o volate dal mezzo di trasporto.

Tabella 9: Descrizione campi Dati Veicoli

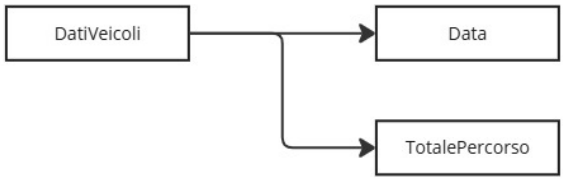

Figura 10: Schema dei Dati Veicoli

- Dati Pagamento: Sono qui esplicitate le informazioni riguardo il pagamento della fattura.

Campo	**Descrizione**
CondizioniPagamento	Indica se il pagamento è avvenuto a rate, con un anticipo o in maniera completa.
DettaglioPagamento	Contiene tutti i dettagli riguardo la transazione di denaro, tra questi sono presenti la modalità di pagamento, l'iban, l'importo e il titolare del conto.

Tabella 10: Descrizione campi Dati Pagamento

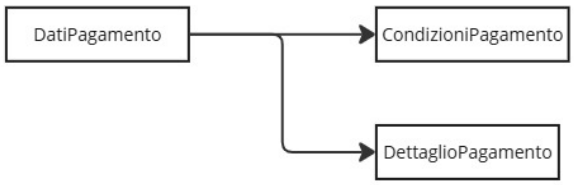

Figura 11: Schema dei Dati Pagamento

- Allegati: Sottosezione contenente gli allegati alla fattura e informazioni a riguardo.

Campo	**Descrizione**
NomeAttachment	E' il nome del file allegato alla fattura.
AlgoritmoCompressione	Indica l'algoritmo di compressione usato sul file allegato.
FormatoAttachment	Indica il formato del file allegato.
DescrizioneAttachment	E' una descrizione del file allegato.
Attachment	E' il file allegato stesso.

Tabella 11: Descrizione campi Allegati

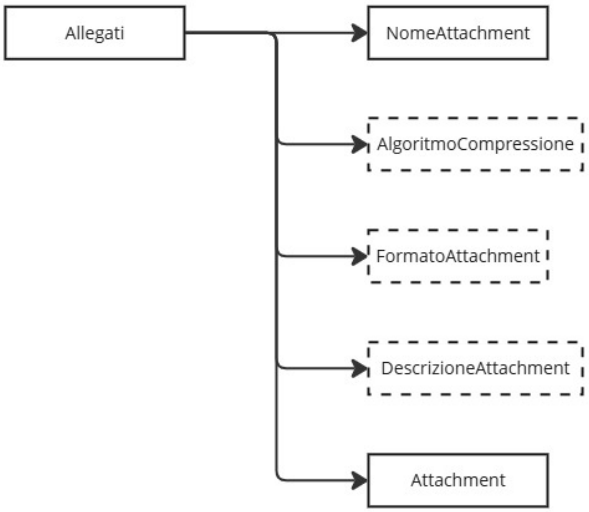

Figura 12: Schema Allegati

Emissione ed acquisizione dei documenti contabili Il processo inizia con l'emissione della fattura elettronica da parte del fornitore, indicato come si è visto con cedente/prestatore, e che avviene tramite uno dei molti sistemi gestionali presenti sul mercato che permettono la fatturazione. Il fornitore compila il documento in formato XML, secondo le regole e la struttura che si è descritta precedentemente (Figura 2). Una volta che la fattura è pronta, questa viene inviata al SdI che ha il compito di ricevere, verificare e smistare tutte le fatture elettroniche emesse in Italia. Il SdI esegue controlli preliminari su ogni fattura ricevuta per ridurre errori e prevenire contenziosi, agevolando eventuali correzioni e accelerando il ciclo di pagamento. In caso di mancato superamento dei controlli, il SdI invia al mittente una 'ricevuta di scarto'. La fattura, quindi, viene esclusa dal processo e non prosegue verso il destinatario. Lo SdI:

- Verifica che la fattura contenga tutti i dati obbligatori secondo la legge (Dpr n. 633/1972), inclusi i dati di fornitore e cliente, numero e data della fattura, descrizione dei beni o servizi, imponibile, aliquota e IVA.
- Controlla la validità della partita IVA del fornitore e del codice fiscale o partita IVA del cliente tramite l'Anagrafe Tributaria.
- Si accerta della presenza di un indirizzo telematico a cui mandare la fattura, definito dal codice destinatario e o dalla PEC destinatario.
- Effettua una verifica di coerenza tra imponibile, aliquota e IVA effettiva.
- Se presente, controlla la validità del certificato della firma digitale.

- Previene l'invio di duplicati della stessa fattura.

Se invece la fattura supera tutti i controlli allora viene inoltrata al destinatario e al suo sistema contabile di riferimento. Il mittente potrà quindi ottenere una 'ricevuta di avvenuta consegna' oppure, se il canale PEC del destinatario non fosse attivo una 'ricevuta di impossibilità di consegna'.

Nel caso in cui il destinatario sia una Pubblica Amministrazione, la fattura verrebbe indirizzata al Sistema di Fatturazione Elettronica (SIFE). Questo sistema è responsabile della separazione della fattura da eventuali documenti allegati, le cui dettagliate informazioni sono specificate nella Tabella 11. Successivamente, il SIFE ha il compito di inoltrare la fattura, senza gli allegati, al Sistema di Contabilità Generale dello Stato (SICOGE).

Il sistema SICOGE riceve la fattura e, grazie a una matrice di instradamento, identifica l'ufficio contabile di competenza basandosi sul codice IPA presente nell'Header della fattura stessa. Questa matrice associa il codice IPA a ciascun ufficio contabile, permettendo così di inoltrare automaticamente la fattura all'ufficio contabile di riferimento.

Una volta instradata, la fattura viene inviata all'interfaccia SAP Process Orchestrator (PO), che si occupa di fare da ponte tra il SICOGE e il modulo VIM all'interno di S/4HANA, consentendo la gestione e il monitoraggio dei flussi contabili all'interno del sistema SAP.

Arrivate nel sistema SAP S4/HANA VIM, le fatture sono rese disponibili agli operatori contabili.

Manipolazione della fattura in SAP S/4HANA VIM All'interno del modulo VIM, l'operatore contabile incaricato ha la possibilità di svolgere diverse attività. Una di queste è la contabilizzazione delle fatture relative ai propri uffici. Con l'ultimo aggiornamento del sistema, l'operatore ha a disposizione una funzionalità di contabilizzazione massiva. Il processo di contabilizzazione massiva consente all'operatore di utilizzare modelli di contabilizzazione (*template*) predefiniti per elaborare efficacemente un grande volume di documenti. I template, creati e salvati una sola volta, includono tutti i parametri necessari, come dati finanziari, economici e di gestione IVA, e ne potranno essere configurati diversi per uno stesso ufficio. Per avviare l'elaborazione, l'utente deve selezionare manualmente i documenti da elaborare utilizzando criteri di ricerca, scegliere la funzionalità di contabilizzazione massiva, applicare il template desiderato e avviare il processo. L'elaborazione avviene in background, permettendo all'utente di proseguire con altre attività. Al termine, è possibile controllare l'esito dell'elaborazione per verificare

i risultati. Nel diagramma BPMN in Figura 13 è possibile vedere il flusso di processo attuale.

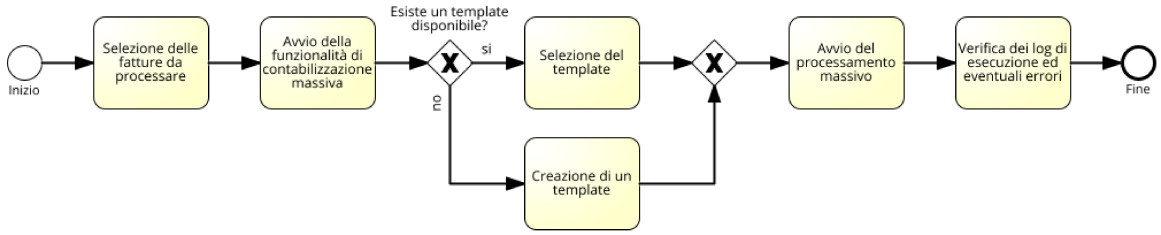

Figura 13: Diagramma BPMN del processo di contabilizzazione massiva

2.1.2 Obiettivi dell'applicazione

Emerge l'opportunità di adottare strumenti e soluzioni che possano automatizzare o supportare specificamente la fase di selezione del modello di contabilizzazione, riducendo così il margine di errore e garantendo maggiore rapidità, uniformità e coerenza nella gestione delle fatture elettroniche.

In questo contesto, il machine learning rappresenta una soluzione particolarmente efficace per ottimizzare il processo. Grazie all'uso di algoritmi di apprendimento automatico, il sistema sarebbe in grado di analizzare automaticamente le caratteristiche delle fatture e di suggerire il modello di contabilizzazione più appropriato, basandosi sempre sui template di contabilizzazione esistenti e predefiniti. Questo approccio consentirebbe di migliorare progressivamente la precisione nel tempo, man mano che aumenta il volume di dati utilizzati per l'addestramento.

Inoltre, gli operatori avranno comunque la possibilità di creare nuovi template per gestire tipologie di fatture non ancora coperte dai modelli esistenti. In questo modo, il sistema manterrà la flessibilità necessaria per adattarsi a nuovi scenari o requisiti specifici.

L'automazione della scelta del template ridurrebbe la dipendenza dall'esperienza individuale degli operatori e libererebbe questi ultimi da compiti ripetitivi e di basso livello. In pratica, l'applicazione avrebbe la capacità di osservare una fattura, individuare il modello di contabilizzazione corretto tra quelli già configurati, e avviare il processo. Agli operatori resterebbe il compito di validare la procedura, creare nuovi template dove necessario e supervisionare lo sblocco al pagamento del fornitore.

È importante sottolineare che, anche dopo l'associazione automatica tra fattura e template, il risultato deve essere sempre sottoposto a validazione da parte di un operatore umano. Questa fase di controllo è fondamentale, poiché lo sblocco del pagamento non

può essere delegato a una macchina, ma deve essere effettuato da un ufficio che si assume piena responsabilità dell'operazione.

2.1.3 Criteri di successo

Per garantire il successo di un sistema basato su machine learning, come quello proposto negli obiettivi descritti nel paragrafo precedente, è fondamentale definire in modo chiaro i criteri che permettano di valutarne l'efficacia e l'impatto. La metodologia CRISP-ML fornisce una struttura utile per stabilire questi criteri su tre distinti livelli: criteri di business, criteri tecnici e criteri economici.

Conformemente a [28], è essenziale che queste metriche siano misurabili, per consentire un confronto efficace tra il processo tradizionale senza machine learning e quello automatizzato con machine learning, e che siano anche coerenti tra loro. Questo approccio strutturato garantisce che ogni aspetto del sistema, dalla qualità delle previsioni alla loro sostenibilità economica, sia valutato in modo rigoroso e trasparente, supportando decisioni basate sui dati.

I criteri di successo di business I criteri di business, che rappresentano criteri di alto livello, definiscono gli obiettivi aziendali che l'applicazione deve raggiungere per essere considerata di successo. Questi criteri si concentrano sull'impatto tangibile che l'applicazione avrà sulle operazioni o sui risultati dell'organizzazione. Devono essere direttamente collegati alle esigenze strategiche e operative del business e tradotti in obiettivi chiari, concreti e misurabili.

L'obiettivo principale nel nostro caso è migliorare e accelerare il processo di contabilizzazione delle fatture, introducendo strumenti che permettano di gestire le operazioni in modo più rapido.Di conseguenza, gli operatori possono dedicare meno tempo ad attività di routine e concentrarsi maggiormente su compiti a più alto valore aggiunto, come la validazione dei risultati. Questo aumento di produttività non si limita a un miglior utilizzo del tempo individuale, ma si traduce anche in un miglioramento complessivo della performance aziendale.

Una maggiore produttività per operatore significa che, in media, ogni operatore riesce a contabilizzare la singola fattura in meno tempo di prima. L'esperto di dominio suggerisce che un buon obiettivo di capacità predittiva si attesta al 90% di fatture correttamente classificate. Questo vincolo è fondamentale per limitare il numero di errori, poiché ogni fattura associata a un template errato richiede un intervento manuale per lo storno, com-

portando tempi e costi aggiuntivi. Tuttavia, l'imposizione di una soglia di confidenza elevata implica che anche le fatture correttamente classificate, ma con confidenza inferiore al 90%, debbano essere gestite manualmente, accettando quindi un compromesso tra riduzione degli errori e aumento del lavoro manuale per i casi incerti.

I criteri di successo tecnici Si è compreso, quindi, che l'obiettivo concreto e misurabile su cui focalizzarsi dal punto di vista di business è l'aumento della produttività, traducibile in bontà predittiva superiore al 90%.

Questo obiettivo funzionale deve ora essere tradotto e reso operativo nel contesto tecnico. La sfida consiste nel convertire l'esigenza aziendale in criteri di progettazione e valutazione per la nostra applicazione. A tal fine, sono state selezionate sei metriche standard che costituiscono i criteri tecnici di riferimento. Questi criteri rappresentano infatti i pilastri su cui verrà basata la scelta del modello, e sarà necessario stabilire chiaramente i livelli di importanza da attribuire a ciascuna metrica riportata nella Tabella 12.

Criterio	**Definizione**
Accuratezza	Indica la capacità dell'algoritmo di predire il template corretto (qualità della predizione).
Interpretabilità	Misura il grado di comprensione del comportamento dell'algoritmo al variare di una o più circostanze.
Incrementabilità	Si riferisce alla capacità di eseguire un aggiornamento del modello in presenza di nuove classi predicibili.
Efficienza	Misura del tempo impiegato in fase di costruzione del modello e classificazione di un'istanza.
Scalabilità	Indica la capacità dell'algoritmo di scalare sui volumi di dati e sul numero di attributi.
Robustezza	Misura quanto l'algoritmo gestisca bene la presenza di rumore nei dati o dati mancanti.

Tabella 12: Tabella dei criteri tecnici con definizioni

Vediamo adesso di definire un punteggio da 1 a 5 per ciascuno di essi.

Nel caso di studio analizzato, dato come obiettivo principale quello di risparmiare del tempo ed al contempo gestire un elevato numero di fatture, i criteri di incrementabilità, efficienza e scalabilità si ergono tra tutti come prioritari. L'incrementabilità è

una caratteristica che il modello dovrà avere in quanto, per natura del problema, sono costantemente generati nuovi template da parte degli operatori, il punteggio sarà quindi 4 su 5. Il peso dell'efficienza, in particolare, può essere diviso in due, in quanto: la priorità massima è che il tempo di classificazione sia il minore possibile, quasi nullo, punteggio 5, mentre il tempo di costruzione del modello sia breve è sicuramente un nice-to-have, ma non è necessario, punteggio 3. Per questo motivo si può trovare un punteggio medio di 4 su 5. Inoltre, anche la scalabilità si configura come un elemento fondamentale, con l'obiettivo di assicurare che il modello possa adattarsi ai sistematici aumenti nei volumi dei dati, senza comprometterne le prestazioni. Il suo punteggio è anch'esso 5 su 5.

Come misura della qualità della predizione si è deciso di utilizzare l'accuratezza, definita come il rapporto tra il numero di predizioni corrette e il numero totale di predizioni effettuate. Tale scelta è motivata dal fatto che, nel nostro caso di studio, il costo associato a un falso positivo non è molto distante dal costo associato a un falso negativo, il primo associato ad uno storno della fattura erroneamente registrata e alla conseguente associazione manuale, il secondo alla sola associazione manuale al template corretto. Pertanto, metriche alternative come la precisione, il richiamo o l'F1-score, che sono spesso più indicate in situazioni con costi estremamente sbilanciati, sono state considerate meno rilevanti, anche se verranno analizzate per valutare le performance del modello nella Sezione 3.1. Come precisato nei criteri di business, la bontà della predizione, che abbiamo capito essere l'accuratezza, nel nostro caso deve essere di almeno del 90%. Il suo punteggio è 3 su 5.

Altri criteri, come la robustezza e l'interpretabilità, pur avendo un peso relativamente inferiore in questo scenario specifico, non possono essere trascurati completamente. Anche se i rischi di malfunzionamento non sono immediati, la robustezza contribuisce comunque alla stabilità complessiva del sistema. L'interpretabilità, invece, assume rilevanza nel facilitare l'adozione e la manutenzione del sistema da parte degli utenti e nel garantire la trasparenza nei processi decisionali. Gli sono assegnati rispettivamente 1 e 2 punti.

Nella Tabella 13 si ha un riassunto.

Criterio	Livello di Importanza (1-5)
Accuratezza	⋆⋆⋆
Interpretabilità	⋆⋆
Incrementabilità	⋆⋆⋆⋆
Efficienza	⋆⋆⋆⋆
Scalabilità	⋆⋆⋆⋆⋆
Robustezza	⋆

Tabella 13: Tabella dei criteri tecnici con livelli di importanza

Questi gradi di importanza sono cruciali per determinare la soluzione ottimale per il nostro caso, incluso quale modello scegliere, quanto spesso aggiornare il modello per evitare il "drift", ossia la perdita di precisione predittiva, e le modalità di implementazione del nostro progetto [29]. Il successo dell'applicazione si avrà quindi qualora siano rispettate queste esigenze tecniche.

I criteri di successo economici In aggiunta ai criteri di business e tecnici, introduciamo un KPI (Key Performance Indicator) valutabile prima e dopo l'applicazione del modello predittivo al caso di studio, che possa fornire un'informazione sintetica sull'esito della soluzione.

Un buon indicatore di successo economico nel nostro caso è il tempo medio di gestione di una fattura. Obiettivo dell'applicazione è quello di fare abbassare questo tempo.

La situazione attuale presenta un tempo medio di gestione di una fattura uguale al tempo medio di associazione di una fattura ad un template.

$$\bar{t}_{\text{gestione\_fattura\_0}} = \bar{t}_{\text{associazione}}$$

Dove:

- $\bar{t}_{\text{gestione\_fattura\_0}}$ è il tempo medio di gestione di una fattura al tempo zero, ovvero al momento attuale;
- $\bar{t}_{\text{associazione}}$ è il tempo medio di associazione di una fattura ad un template.

Una volta implementato l'automatismo proposto, il tempo medio di gestione di una fattura sarà calcolabile sommando alcuni fattori. Il primo fattore è quello che indica il tempo medio che sarà dedicato a gestire gli storni, ovvero la quota di fatture associate erroneamente ma con una confidenza superiore almeno al 90% moltiplicate per il tempo

medio di gestione di uno storno. Il secondo fattore sarà quello dedicato alle fatture che saranno da classificare manualmente, dato dalla quota di fatture non associate perché aventi confidenza inferiore al 90% unite alla quota di fatture stornate, moltiplicate per il tempo medio di associazione di una fattura. Infine il terzo fattore, indica il tempo medio che sarà necessario ad effettuare gli sblocchi al pagamento delle fatture, dato dal prodotto tra la quota di fatture correttamente associate e il tempo medio di sblocco al pagamento.

$$\bar{t}_{\text{gestione\_fattura\_1}} = \%_{\text{storni}} \cdot \bar{t}_{\text{storno}} + (\%_{\text{storni}} + \%_{<90\%}) \cdot \bar{t}_{\text{associazione}} + \%_{\geq 90\%} \cdot \bar{t}_{\text{sblocco}}$$

Dove:

- $\bar{t}_{\text{gestione\_fattura\_1}}$ è il tempo medio di gestione di una fattura al tempo uno, ovvero dopo l'inserimento dell'automatismo;
- $\bar{t}_{\text{associazione}}$ è il tempo medio di associazione di una fattura ad un template;
- $\bar{t}_{\text{sblocco}}$ è il tempo medio di sblocco al pagamento di una fattura;
- $\bar{t}_{\text{storno}}$ è il tempo medio di gestione di una fattura stornata;
- $\%_{\text{storni}}$ è la quota di fatture stornate dal sistema;
- $\%_{<90\%}$ è la quota di fatture non associate perchè aventi confidenza inferiore al 90%;
- $\%_{\geq 90\%}$ è la quota di fatture correttamente associate con confidenza di almeno il 90%.

Compito dell'automatismo sarà fare sì che:

$$\bar{t}_{\text{gestione\_fattura\_1}} < \bar{t}_{\text{gestione\_fattura\_0}}$$

Vediamo adesso quali degli elementi descritti poc'anzi sono a disposizione prima dell'applicazione e quali invece andranno ricavati una volta terminati gli esperimenti. L'esperto di dominio suggerisce che:

- $\bar{t}_{\text{associazione}} = 2$ minuti;
- $\bar{t}_{\text{storno}} = 3$ minuti;
- $\bar{t}_{\text{sblocco}} = 0.5$ minuti.

Pertanto:

$$\bar{t}_{\text{gestione\_fattura\_0}} = 2\ minuti$$

Mentre gli elementi da ricavare per trovare il tempo medio di gestione della fattura dopo l'applicazione sono:

- $\%_{\text{storni}}$ è la quota di fatture stornate dal sistema;
- $\%_{<90\%}$ è la quota di fatture non associate perchè aventi confidenza inferiore al 90%;
- $\%_{\geq 90\%}$ è la quota di fatture correttamente associate con confidenza di almeno il 90%.

Nella Sezione 3.1.1 è mostrato il tempo di gestione medio dopo l'applicazione.

2.1.4 Studio di fattibilità

Come spiegato in [30], è importante nel progetto verificarne la fattibilità, per evitare di procedere con il lavoro fino a un punto morto, oltre il quale non si può andare perché, ad esempio, mancano i dati per addestrare il modello oppure alcuni criteri di business o tecnici non sono perseguibili. Si esamineranno adesso alcune fonti di incertezza da verificare, identificate da [24], e di come eventualmente queste possano venire mitigate.

- *Disponibilità, quantità e qualità dei dati*: Il bacino di dati utilizzabile è particolarmente ampio, poiché tutte le fatture già registrate nel sistema attuale possono essere scaricate e utilizzate per addestrare il modello. Si parla di un ordine di grandezza di decine di migliaia.

  Inoltre, la qualità e la consistenza dei dati è assicurata dai numerosi controlli effettuati dai sistemi (SdI, SICOGE), aggiornati ai più moderni standard di sicurezza informatica, sui quali transitano i documenti prima di arrivare al sistema informativo del Cliente.

- *Applicabilità del machine learning al dominio di interesse*: Nel capitolo 1.4 sui lavori correlati, è stata già effettuata un'analisi approfondita della letteratura che ha identificato applicazioni comparabili nello stesso dominio. Questo studio preliminare ha contribuito a definire lo stato dell'arte, dimostrando la possibilità di utilizzare algoritmi di machine learning nel contesto considerato.

- *Vincoli legali e di privacy*: In merito alle questioni di privacy e rispetto del GDPR[2], è importante sottolineare due punti:

    1. Il nome del Cliente non è mai esplicitato;
    2. I dati sensibili contenuti nelle fatture sono stati interamente mascherati durante ogni fase del lavoro.

2.1.5 Raccolta dati

La raccolta dati è avvenuta scaricando dalla piattaforma di sviluppo del modulo SAP VIM, in gestione a Leonardo, circa 5200 fatture in formato XML, unitamente ad un documento Excel che abbina gli XML ai relativi template, la cui struttura è mostrata nella Tabella 14.

Queste fatture contengono dati sensibili, come informazioni anagrafiche e dati di pagamento, che, in conformità al GDPR, non possono essere utilizzati senza il consenso esplicito dei soggetti coinvolti. Per proteggere queste informazioni, è stato deciso di mascherarle utilizzando un algoritmo di hash.

Un algoritmo di hash è una funzione matematica che converte un input di lunghezza arbitraria, nel nostro caso ad esempio il codice fiscale e la partita iva, in un output di lunghezza fissa, chiamato *digest*. Una delle proprietà fondamentali di questo processo è la sua natura deterministica: due input identici genereranno sempre lo stesso hash. Allo stesso tempo, l'algoritmo garantisce l'irreversibilità, rendendo impossibile risalire al dato originale a partire dal digest. Un'altra caratteristica importante è la sensibilità ai cambiamenti: anche una minima variazione dell'input produce un hash completamente diverso.

Grazie a queste caratteristiche, l'hash consente di proteggere i dati sensibili presenti nelle fatture pur permettendo di mantenere le relazioni tra i dati di fondamentale importanza per le nostre analisi.

000007939827.xml	9009837
000012345678.xml	8502741
000098765432.xml	9103948
000001112233.xml	8890473

Tabella 14: Struttura di esempio del file Excel

[2]Garante per la Protezione dei Dati Personali

2.2 Data preparation

Ottenuti i documenti XML mascherati e il file Excel di mappatura, si hanno tutti gli elementi per costruire il dataset, esplorarlo ad alto livello e preprocessarlo.

L'obiettivo di questa fase preparatoria è quello di fornire un'insieme di dati nella forma migliore possibile per la successiva fase di costruzione e valutazione del modello di classificazione. Di fondamentale importanza, specialmente quando si trattano grandi dataset in termini di dimensioni e volume, è quindi la selezione degli attributi, ovvero l'individuazione delle caratteristiche rilevanti da fornire al modello e su cui basare la predizione.

Come evidenziato da [31], i metodi di selezione degli attributi possono essere generalmente suddivisi in due categorie principali: metodi filtro (filter methods) [32–35] e metodi wrapper (wrapper methods) [36, 37]. I metodi filtro selezionano un sottoinsieme di attributi in base a determinati criteri, prima della costruzione del modello, ignorando quindi l'algoritmo utilizzato nella fase successiva. Poiché non richiedono una fase di valutazione sul modello, risultano poco onerosi dal punto di vista computazionale e facilmente adattabili a qualsiasi tipo di algoritmo di machine learning. Questo li rende un passaggio iniziale efficace per ridurre la dimensione del dataset, concentrandosi solo sugli attributi più rilevanti e predittivi. I metodi wrapper, invece, sono più accurati ma computazionalmente più onerosi. Tali metodi ricercano iterativamente il miglior sottoinsieme di caratteristiche, valutato in base ad un criterio scelto (accuratezza, f1-score,...), per l'algoritmo di classificazione. Ciò li rende particolarmente costosi nei casi in cui il dataset contenga un numero elevato di attributi, poiché il loro costo aumenta in modo esponenziale con l'incremento degli attributi.

L'approccio adottato nel nostro caso di studio è di tipo misto, in quanto si sono prima applicati metodi di filtro, nella fase di costruzione del dataset, e poi successivamente metodi di wrapper in fase di costruzione e valutazione del modello.

Le operazioni di costruzione ed esplorazione del dataset, descritte di seguito, sono state realizzate utilizzando Python e le sue principali librerie.

2.2.1 Selezione degli attributi e costruzione del dataset

La principale difficoltà nella trasformazione di una fattura elettronica in formato XML in un formato tabellare risiede nella gestione dei campi con molteplicità 1:N, a causa delle differenze strutturali tra i due formati. L'XML è basato su una struttura gerarchica, in cui un elemento "padre" può contenere molteplici elementi "figlio", come nel caso di un

ordine che include diversi articoli. Al contrario, una tabella relazionale ha una struttura piatta, dove ogni riga rappresenta un'unità uniforme di dati.

Tuttavia, nel caso specifico, questa difficoltà è stata superata grazie a un lungo e approfondito dialogo con l'esperto di domino, il quale ha suggerito come i campi da estrarre avessero solamente molteplicità 1:1 o 0:1.

In particolare, di tutti i dati presenti in una fattura, solamente alcune informazioni contenute nella testata (Header) sono state prese in considerazione. Nella Tabella 15 si possono notare i nomi degli attributi[3] estratti.

Header/DatiTrasmissione/CodiceDestinatario
Header/CedentePrestatore/DatiAnagrafici/IdFiscaleIVA/IdCodice
Header/CedentePrestatore/DatiAnagrafici/CodiceFiscale
Header/CedentePrestatore/DatiAnagrafici/Anagrafica/Denominazione
Header/CedentePrestatore/DatiAnagrafici/RegimeFiscale
Header/CessionarioCommittente/DatiAnagrafici/IdFiscaleIVA/IdCodice
Header/CessionarioCommittente/DatiAnagrafici/CodiceFiscale
Header/CessionarioCommittente/DatiAnagrafici/Anagrafica/Denominazione

Tabella 15: Nomi degli attributi estratti dalle fatture in formato XML

I dati sono stati quindi utilizzati per popolare una tabella, con il nome del file XML come chiave primaria, appiattendo le informazioni della fattura su una sola riga. Successivamente, è stata eseguita un'operazione di join, mostrata in Figura 14 tra la tabella contenente i dati delle fatture e quella delle associazioni fatture-template, utilizzando come chiave primaria il campo NomeFile. Questo ha permesso di ottenere una prima versione del dataset.

[3]Per facilitarne la leggibilità, a questi nomi è stato rimosso il valore del namespace: "http://ivaservizi.agenziaentrate.gov.it/docs/xsd/fatture/v1.2FatturaElettronica/FatturaElettronica"

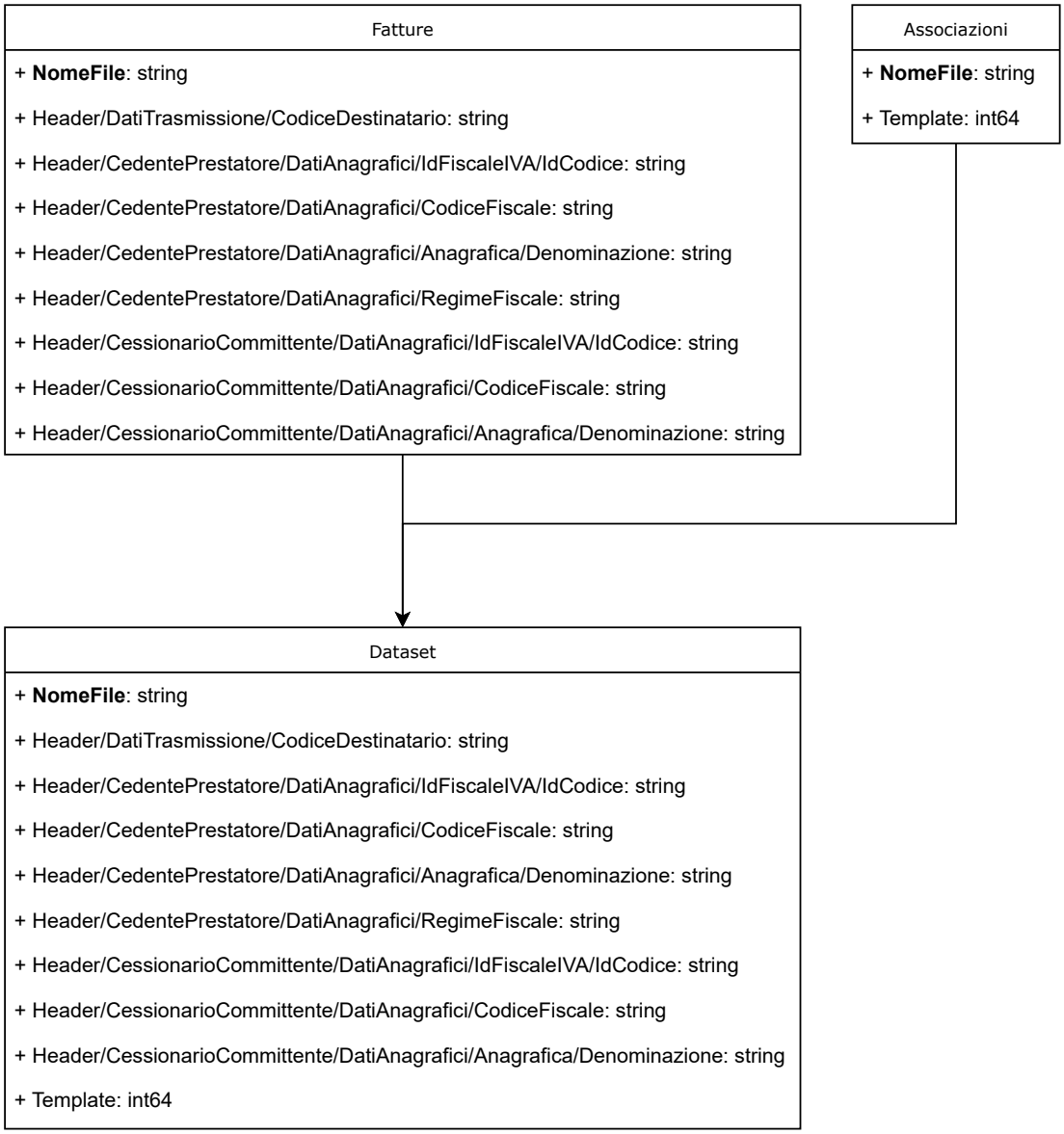

Figura 14: Operazione di join tra i dati delle fatture e i dati degli accoppiamenti

Inoltre, in questo momento è anche eliminata la colonna contenente il nome del file xml per evitare che il modello apprenda informazioni non rilevanti o errate. I nomi dei file spesso non forniscono dati significativi in relazione al problema che si sta cercando di risolvere, e includerli potrebbe portare il modello a fare previsioni basate su correlazioni casuali piuttosto che sulle caratteristiche reali e pertinenti, generando overfitting[4].

[4]L'overfitting si verifica quando un modello di machine learning impara troppo bene i dettagli e il rumore dei dati di addestramento, al punto da non riuscire a generalizzare su dati nuovi. Questo porta a performance elevate sui dati di addestramento ma scarse su quelli di test. È un segno che il modello ha "memorizzato" piuttosto che "compreso" i dati.

La Tabella 16 mostra per ogni attributo del dataset il relativo tipo di dato su Pyhton.

Nome Colonna	**Tipo di dato**
Header/DatiTrasmissione/CodiceDestinatario	string
Header/CedentePrestatore/DatiAnagrafici/IdFiscaleIVA/IdCodice	string
Header/CedentePrestatore/DatiAnagrafici/CodiceFiscale	string
Header/CedentePrestatore/DatiAnagrafici/Anagrafica/Denominazione	string
Header/CedentePrestatore/DatiAnagrafici/RegimeFiscale	string
Header/CessionarioCommittente/DatiAnagrafici/IdFiscaleIVA/IdCodice	string
Header/CessionarioCommittente/DatiAnagrafici/CodiceFiscale	string
Header/CessionarioCommittente/DatiAnagrafici/Anagrafica/Denominazione	string
Template	int64

Tabella 16: Nome delle colonne e il relativo tipo di dato

2.2.2 Esplorazione e pulizia dei dati

Il dataset, nel formato ottenuto, risulta facilmente esplorabile per verificare alcuni elementi sulla qualità dei dati come il tipo di dato estratto, i valori mancanti in ogni colonna, e anche per capire la distribuzione delle classi di predizione.

Nome Colonna	**% di valori mancanti**
Header/DatiTrasmissione/CodiceDestinatario	0.00
Header/CedentePrestatore/DatiAnagrafici/IdFiscaleIVA/IdCodice	0.00
Header/CedentePrestatore/DatiAnagrafici/CodiceFiscale	0.00
Header/CedentePrestatore/DatiAnagrafici/Anagrafica/Denominazione	0.02
Header/CedentePrestatore/DatiAnagrafici/RegimeFiscale	0.00
Header/CessionarioCommittente/DatiAnagrafici/IdFiscaleIVA/IdCodice	15.17
Header/CessionarioCommittente/DatiAnagrafici/CodiceFiscale	0.00
Header/CessionarioCommittente/DatiAnagrafici/Anagrafica/Denominazione	0.00
Template	0.00

Tabella 17: Nome delle colonne e la % di dati mancanti in ciascuna

La percentuale di valori mancanti per ogni colonna Nell'ambito dell'analisi dei dati mancanti, si distinguono tre tipi di dati mancanti possibili: **MCAR** (Missing Com-

pletely at Random), **MAR** (Missing at Random), e **MNAR** (Missing Not at Random). I dati si considerano MCAR quando la loro assenza avviene completamente a caso, senza dipendere da alcuna variabile osservata o latente; in tal caso, l'assenza non introduce bias ed è possibile decidere di rimuovere completamente la riga o la colonna del dataset. Si parla di MAR quando la mancanza è correlata sistematicamente ad altre variabili osservabili nel dataset, consentendo di modificare il dataset con strategie di imputazione ad hoc. Infine, i dati sono MNAR quando la loro assenza è direttamente correlata al valore della variabile interessata, rendendo il problema più complesso e non ignorabile. Nel nostro contesto, osservando la Tabella 17, la colonna 'Header/CessionarioCommittente/DatiAnagrafici/IdFiscaleIVA/IdCodice' appare essere MNAR, poiché le mancanze possono derivare dalla situazione in cui certi soggetti non possiedono un identificativo fiscale specifico, un'informazione direttamente legata alla variabile in questione che si vuole predire.

In questa fase. si è scelto di mantenere la colonna contenente i dati mancanti, *gestendone la presenza durante la successiva fase di selezione dell'algoritmo di classificazione*, poiché alcuni algoritmi sono in grado di trattare questi dati in modo nativo. Ad esempio, gli algoritmi basati sugli alberi decisionali sono in grado di funzionare efficacemente anche in presenza di dati mancanti, poiché possono gestirli durante il processo di suddivisione dei dati in base alle caratteristiche disponibili. Questi algoritmi, infatti, possono prendere decisioni anche con input incompleti, senza compromettere significativamente le prestazioni del modello. Invece, algoritmi come le Support Vector Machines (SVM) e le reti neurali richiedono una gestione esplicita dei dati mancanti, poiché non tollerano valori assenti durante l'addestramento del modello. Per questi approcci, è necessario ricorrere a tecniche di imputazione (come la sostituzione dei valori mancanti con la media, la mediana, o altri metodi) o alla rimozione dei dati mancanti, eliminando righe o colonne incomplete. In alternativa, alcuni algoritmi possono essere adattati per trattare i dati mancanti mediante l'uso di tecniche di apprendimento parziale o modelli che imputano automaticamente i valori durante il processo di ottimizzazione.

La distribuzione dei template In ogni progetto di machine learning, è fondamentale analizzare la distribuzione della classe che si desidera predire, con l'obiettivo di capire se esistono valori, o insiemi di valori, che si presentano con maggiore frequenza rispetto ad altri, oppure se, al contrario, ogni valore appare con una probabilità che riflette in modo uniforme la sua frequenza.

Quando si parla di **dataset bilanciato**, si fa riferimento a una situazione in cui tutte le classi della colonna target hanno una frequenza simile o uguale. Ad esempio, se il target è una variabile categorica con tre classi, un dataset bilanciato avrà un numero simile di esempi per ciascuna classe. In questo scenario, il modello può apprendere da tutte le classi in modo equo, senza che una classe prevalga sulle altre. I modelli addestrati su dataset bilanciati tendono a generalizzare meglio e sono più sensibili alle differenze tra le classi.

Al contrario, un **dataset sbilanciato** si verifica quando una o più classi sono significativamente più rappresentate di altre. Per esempio, se una classe rappresenta il 90% dei dati e le altre classi sono rappresentate solo dal 10%, il modello rischia di imparare a predire solo la classe maggioritaria, ignorando le classi minoritarie. Questo fenomeno può portare a una scarsa capacità di generalizzazione, soprattutto sulle classi meno frequenti.

Nel nostro caso, come si può osservare dalla Figura 15, la distribuzione dei valori della colonna 'Template' è molto lontana dall'essere uniforme, ma anzi si avvicina molto ad una distribuzione paretiana. *Siamo dunque davanti ad un caso di dataset sbilanciato*, nel quale circa il 20% dei template più frequenti coprono circa l'80% delle osservazioni.

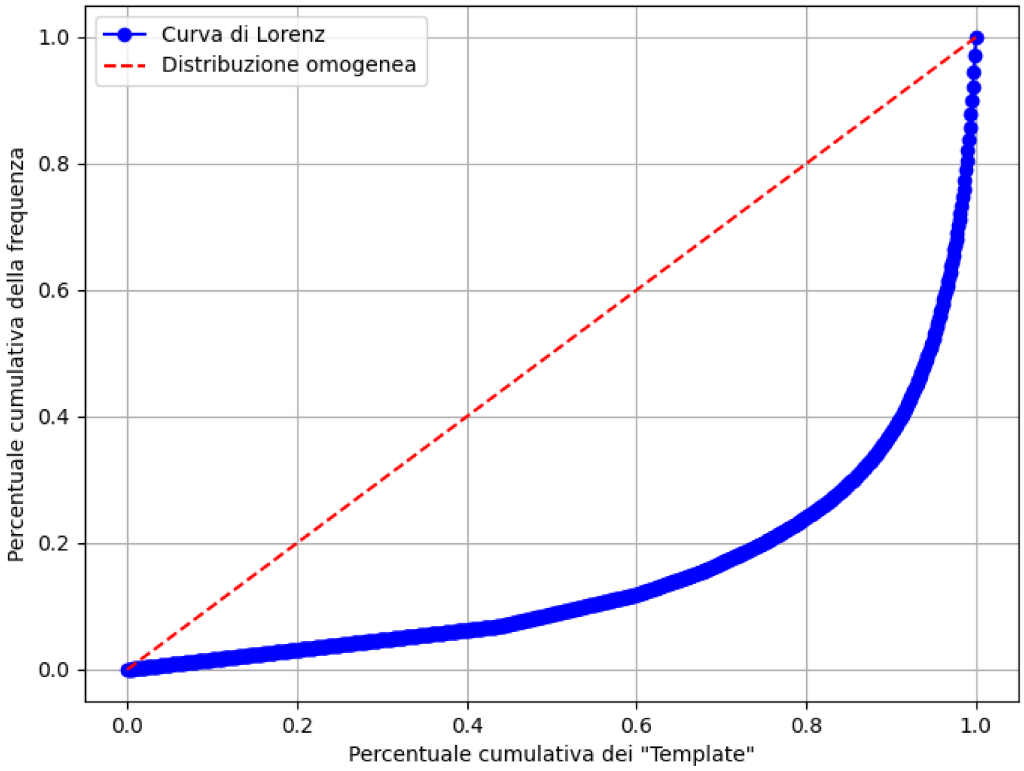

Figura 15: In blu la Curva di Lorenz, relativa alla distribuzione dei template, in rosso invece il caso ipotetico di classi perfettamente bilanciate

Nel nostro caso, una bassa frequenza di un particolare template significa che questo non è stato utilizzato a sufficienza, e i motivi possibili sono due:

1. Il template è troppo recente;

2. Il template è poco utile.

Solo il primo caso è problematico, in quanto riguarda template che potrebbero diventare più rilevanti con il tempo. Tuttavia, questo può essere facilmente risolto con degli aggiornamenti periodici del modello, tramite riaddestramento, per garantire che le classi più recenti vengano considerate e sfruttate. Il secondo caso, invece, rappresenta una situazione in cui il template è poco utilizzato dagli operatori contabili, e pertanto eliminarlo non è solo accettabile, ma necessario per ridurre il rumore nei dati. Ulteriori discussioni in merito sono state fatte nelle Sezioni 3.1.1, 3.1.2, 3.3.

Si è quindi deciso, unitamente all'esperto di dominio, di eliminare le classi aventi bassa frequenza (meno di 10, escluso), concentrando la capacità predittiva verso classi più comuni e significative.

Analisi di correlazione Un ulteriore elemento che compone questa analisi esplorativa è il calcolo e la visualizzazione delle correlazioni tra gli attributi presenti nel dataset. Per rilevare queste correlazioni, si utilizza spesso una matrice delle correlazioni, che rappresenta in forma tabellare i coefficienti di correlazione di tutte le coppie di variabili. Tali coefficienti si possono calcolare in vario modo sulla base del tipo di osservazioni ad oggetto del calcolo. In particolare, come già descritto in [38, 39], sono più comunemente usati i seguenti indici:

- L'indice di Pearson, quando si hanno coppie di variabili continue;
- L'indice di Cramér, quando si hanno coppie di variabili categoriche.

Nel nostro caso siamo in presenza di variabili categoriche, come mostrato dalla Tabella 16, e quindi si è utilizzato l'indice di correlazione di Cramér [40].

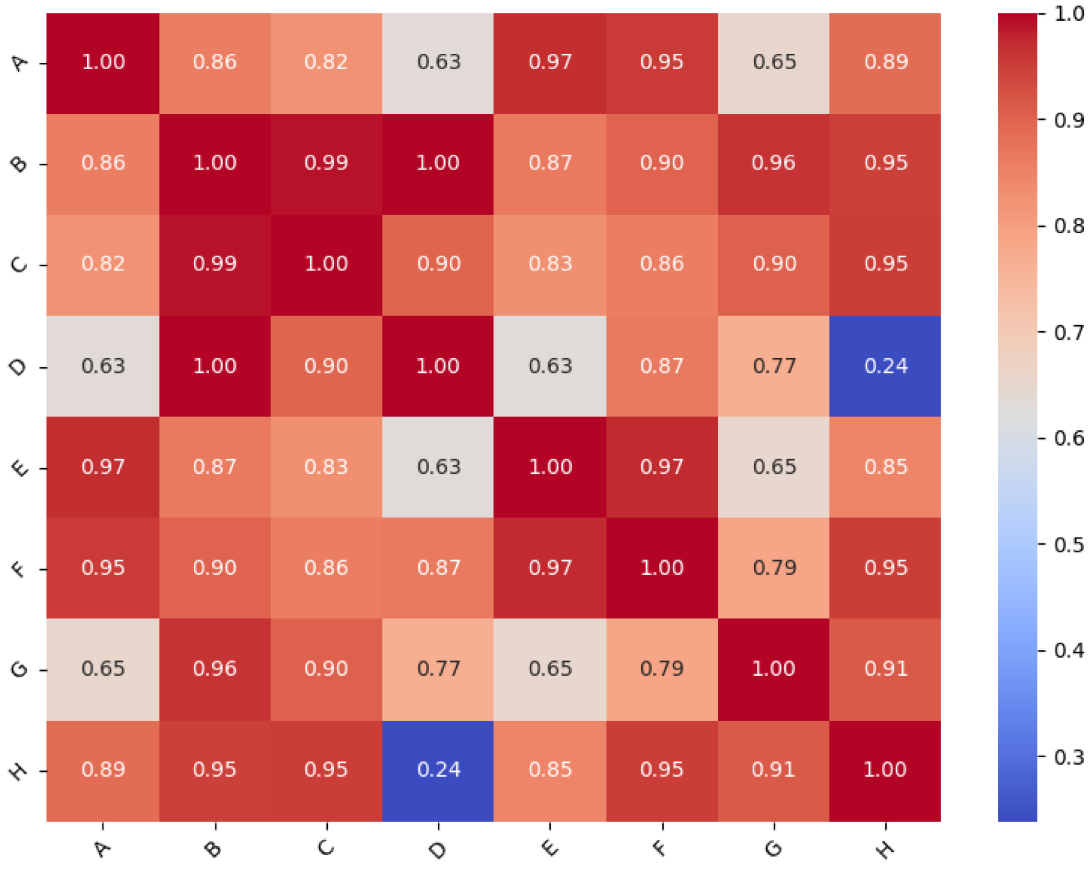

Figura 16: Matrice di Cramér con 0 correlazione minima e 1 correlazione massima

Nome variabile	Identificativo
Header/DatiTrasmissione/CodiceDestinatario	A
Header/CedentePrestatore/DatiAnagrafici/IdFiscaleIVA/IdCodice	B
Header/CedentePrestatore/DatiAnagrafici/CodiceFiscale	C
Header/CedentePrestatore/DatiAnagrafici/RegimeFiscale	D
Header/CessionarioCommittente/DatiAnagrafici/CodiceFiscale	E
Header/CessionarioCommittente/DatiAnagrafici/Anagrafica/Denominazione	F
Header/CedentePrestatore/DatiAnagrafici/Anagrafica/Denominazione	G
Header/CessionarioCommittente/DatiAnagrafici/IdFiscaleIVA/IdCodice	H

Tabella 18: Legenda della matrice di Cramér

Come si può osservare dalla Figura 16 e dalla legenda in Tabella 18 sono presenti molte coppie di attributi che presentano una correlazione elevata, che possiamo identificare in un valore superiore a 0.90. Questo significa che l'insieme di attributi a nostra disposizione presenta delle ridondanze, le quali possono influire negativamente sull'efficienza di costruzione del modello, aumentandone la sua complessità. Infatti, come già spiegato da [41], un buon insieme di attributi contiene variabili che sono altamente correlate con la classe di riferimento, ma poco correlate tra di loro. Per questi motivi, si è deciso di mantenere solamente gli attributi che presentano una correlazione non elevata, quindi inferiore a 0.90, ovvero:

- Header/DatiTrasmissione/CodiceDestinatario
- Header/CedentePrestatore/DatiAnagrafici/RegimeFiscale
- Header/CedentePrestatore/DatiAnagrafici/Anagrafica/Denominazione

2.2.3 Preprocessamento

Gli ultimi passaggi di questa fase preparativa sono quelli di codifica degli attributi categorici e il partizionamento dei dati.

Codifica degli attributi categorici La codifica (o encoding) è un'operazione che prevede la trasformazione di un attributo categorico, e quindi discreto e non numerico, in uno o più attributi di tipo numerico. Questa operazione di preprocessamento è necessaria se vogliono essere utilizzati determinati tipi di algoritmi di machine learning che effettuano confronti di tipo numerico tra gli attributi per ottenere il valore della classe di predizione.

Esistono diversi tipi di codifica:

- La codifica ad etichette trasforma i valori categorici in valori numerici, sostituendo ad ogni valore distinto un numero intero.
- La codifica One-Hot trasforma una colonna, associata ad un attributo categorico, in una serie di colonne binarie (composte da 0 e 1), di cardinalità pari al numero di valori distinti presenti nella colonna. Ogni valore distintivo dell'attributo categorico originale viene rappresentato da una colonna separata, e per ogni riga del dataset viene assegnato un "1" nella colonna corrispondente al valore presente, mentre nelle altre colonne si assegna "0". Questa codifica ha di vantaggioso che non inserisce alcuna informazione ordinale tra i valori dell'attributo con il contro di incrementare molto la dimensionalità del dataset.

Entrambe le codifiche hanno dei pro e dei contro per essere utilizzate. La codifica ad etichette ha il vantaggio di non aggiungere dimensionalità al dataset creando nuove colonne, come invece fa la codifica One-Hot. Quest'ultima però ha il fondamentale vantaggio di non inserire alcuna informazione ordinale nel dataset, che potrebbe confondere gli algoritmi di machine learning, come invece fa la codifica ad etichette.

Per questi motivi si è scelto di usare la codifica One-Hot che garantisce una rappresentazione più chiara del nostro caso di studio.

Partizionamento dei dati Il partizionamento dei dati nei progetti di machine learning consiste nella suddivisione dei dati in tre insiemi principali: addestramento, validazione e test. L'insieme di addestramento viene utilizzato nella fase di costruzione e apprendimento del modello. L'insieme di validazione, invece, è usato nella fase di valutazione e ottimizzazione degli iperparametri (anche detta "tuning"), aiutando a confrontare diversi modelli o configurazioni al fine di migliorare le loro prestazioni. Infine, l'insieme di test viene impiegato nella valutazione finale del modello su dati completamente nuovi, non utilizzati né in fase di addestramento né in fase di validazione, fornendo una stima oggettiva della sua capacità di generalizzazione.

È possibile effettuare questa divisione in diversi modi. Una delle tecniche è l'*holdout*. Con l'holdout si selezionano tre quote diverse del dataset: una quota per l'addestramento, in genere tra il 60% e l'80%, una quota per la validazione, in genere tra il 10% e il 20%, e una quota per il test, in genere tra il 10% e il 20%. Questa divisione è molto semplice da effettuare, ma fornisce una stima dell'accuratezza che dipende molto dalla divisione specifica in addestramento, validazione e test.

La *cross-validation* (o convalida incrociata) è un metodo che permette di ottenere una stima più accurata delle prestazioni di un modello, superando il problema descritto prima, suddividendo il dataset, a cui è stato precedentemente sottratto un insieme di test, in K sottoinsiemi chiamati "folds". Il modello viene addestrato su K-1 fold e testato sull'insieme rimanente. Questo processo viene ripetuto K volte, ogni volta utilizzando un fold diverso come insieme di validazione e gli altri K-1 fold uniti come insieme di addestramento. Alla fine, i risultati ottenuti da ciascuna iterazione vengono mediati per ottenere una stima complessiva delle prestazioni del modello, riducendo il rischio che la valutazione dipenda eccessivamente da una particolare divisione dei dati.

Per questi motivi è stata scelta la cross-validation per fase di tuning dei modelli, con 5 folds, mentre per l'insieme di dati di test sono state riservate circa il 10% delle fatture totali.

2.3 Modeling

Effettuate le operazioni di costruzione, pulizia e preprocessamento del dataset, si vuole ora individuare, dato un insieme di algoritmi, quello che permette di costruire un modello di classificazione più vicino alle esigenze funzionali e tecniche del nostro problema.

Gli algoritmi di machine learning possono essere classificati in diverse categorie in base al tipo di risultato che si vuole ottenere, come spiegato da [42]. Nell'*apprendimento supervisionato* l'algoritmo impara a classificare gli input utilizzando un insieme di dati già etichettati, dove le risposte corrette sono note a priori, come nel nostro caso abbiamo la colonna 'Template'. Al contrario, l'*apprendimento non supervisionato* opera su dati privi di etichette, cercando di individuare relazioni o insiemi nei dati stessi. Un approccio ancora diverso è quello dell'*apprendimento per rinforzo*, in cui l'algoritmo sviluppa una strategia basata sulle osservazioni e ottiene feedback dall'ambiente per migliorare le proprie decisioni.

Ognuna di queste categorie ha i suoi algoritmi, specializzati e ottimizzati per le caratteristiche e necessità di un problema, e in questa fase, verranno descritti e valutati cinque tra i più comuni e semplici algoritmi di classificazione per l'apprendimento supervisionato secondo [43–45]:

- Decision Tree;
- Random Forest;
- k-Nearest Neighbour;

- Naive Bayes;
- Support Vector Machine.

Di ognuno di loro sarà descritto il comportamento in fase di costruzione del modello e in fase di classificazione, focalizzandosi maggiormente sulle caratteristiche di accuratezza, interpretabilità, incrementabilità, efficienza, scalabilità e robustezza, ovvero i criteri di valutazione tecnici descritti in precedenza nella tabella 12. L'obiettivo di questa analisi è capire quale algoritmo si adatta meglio alle esigenze del nostro caso e per questo verrà assegnato un punteggio di valutazione, con minimo 1 e massimo 5, per ognuna di queste caratteristiche, basandosi su [42–46]. Infine, si procederà al confronto tra gli algoritmi calcolando il punteggio finale di ognuno di essi, moltiplicando il punteggio di valutazione di ogni caratteristica di ogni algoritmo, con la relativa importanza, mostrata nella tabella 13. L'algoritmo con il punteggio finale maggiore sarà quindi il più adatto alla risoluzione del nostro problema e sarà utilizzato nelle fasi successive.

2.3.1 Descrizione e valutazione degli algoritmi

Decision Tree Gli alberi decisionali (o Decision Trees) sono tra i più consolidati e importanti algoritmi di machine learning utilizzati attualmente [46]. Un albero decisionale modella le logiche decisionali in una struttura ad albero, dove ogni nodo radice e intermedio rappresenta una condizione di test su un attributo e ogni nodo foglia rappresenta il risultato della classificazione. Partendo dal nodo radice, l'algoritmo si muove da un nodo all'altro passando dai rami, fino ad arrivare al nodo foglia. Gli alberi decisionali sono apprezzati per la loro semplicità interpretativa e la rapidità di apprendimento e classificazione, anche se forniscono in genere predizioni caratterizzate da una bassa accuratezza e da un'alta varianza. La Tabella 19 mostra i punteggi ottenuti dall'algoritmo Decision Tree.

Decision Tree					
Accuratezza	Interpretabilità	Incrementabilità	Efficienza	Scalabilità	Robustezza
⋆⋆	⋆⋆⋆⋆⋆	⋆⋆	⋆⋆⋆⋆⋆	⋆⋆⋆⋆	⋆⋆

Tabella 19: Punteggi di valutazione delle caratteristiche dell'algoritmo Decision Tree

Random Forest Un algoritmo Random Forest funziona seguendo l'approccio di *ensemble learning* [45], generando un insieme numeroso di alberi decisionali così come una

foresta è composta da molte piante [46]. Questo algoritmo vuole migliorare le caratteristiche degli alberi decisionali, che spesso tendono a generare overfitting con i dati di addestramento, causando una significativa variabilità nei risultati di classificazione anche in presenza di lievi variazioni nei dati di input. Questa sensibilità intrinseca degli alberi decisionali al loro specifico dataset di addestramento li rende più inclini agli errori quando vengono applicati a nuovi dati di test.

Nella Random Forest, i diversi alberi decisionali vengono addestrati utilizzando sottoinsiemi casuali di dati e attributi distinti del dataset di addestramento, in modo tale che ogni albero si possa specializzare su un certo aspetto del problema. Quando si deve classificare una nuova istanza, ogni albero decisionale presente nella foresta genera la sua predizione. Successivamente, per compiti di classificazione discreta, i risultati di ciascun albero sono combinati e viene effettuata la predizione finale per votazione. Per compiti di classificazione numerica, invece, viene calcolata la media dei risultati forniti da tutti gli alberi.

Il principale vantaggio dell'algoritmo Random Forest risiede nella sua capacità di aumentare la capacità predittiva e ridurre la varianza che si avrebbe utilizzando un singolo albero decisionale. Questo è reso possibile dall'integrazione dei risultati provenienti da una varietà di alberi decisionali, aumentando la robustezza del modello complessivo a scapito di una minore interpretabilità ed efficienza, questa minore soprattutto in fase di costruzione del modello. In Tabella 20 sono mostrati i punteggi di questo algoritmo.

Random Forest					
Accuratezza	Interpretabilità	Incrementabilità	Efficienza	Scalabilità	Robustezza
⋆⋆⋆⋆	⋆⋆	⋆⋆	⋆⋆⋆⋆	⋆⋆⋆⋆	⋆⋆⋆⋆

Tabella 20: Punteggi di valutazione delle caratteristiche dell'algoritmo Random Forest

k-Nearest Neighbour L'algoritmo K-Nearest Neighbour (KNN) è uno dei metodi di classificazione più semplici e appartiene alla famiglia dei classificatori *istance based* [46, 47]. Infatti, KNN non costruisce un modello di classificazione a cui poi sottopone una nuova istanza da classificare, ma bensì utilizza sempre tutto il dataset di addestramento per eseguire il suo compito. L'algoritmo, infatti, prima calcola la distanza tra l'istanza da classificare e tutti gli elementi del dataset di addestramento, utilizzando una metrica precedentemente definita, e assegna la classe più comune tra i "K" valori più vicini ad essa. La lettera "K" in KNN quindi indica il numero di vicini più prossimi considerati per determinare il voto sulla classificazione di un'istanza e, per questo motivo, deve essere

scelta con cura, in quanto per diversi valori di "K" possono esistere diversi risultati di classificazione.

La sua caratteristica principale quindi è che tutta l'elaborazione è spostata nella fase di classificazione, non essendo necessario costruire un modello anteriormente, il che, per i nostri scopi, è sicuramente un punto a sfavore, in quanto insieme alla fattura da classificare sarebbe necessario avere sempre a disposizione anche il dataset di addestramento, già spiegato essere di grandi dimensioni. Inoltre, KNN è un algoritmo poco robusto e scalabile, poiché ha necessità di avere dati scalati o normalizzati ed il suo risultato è molto dipendente dalla metrica scelta per il calcolo della distanza tra istanze [45].

In Tabella 21 sono presenti i punteggi di KNN.

k-Nearest Neighbour					
Accuratezza	Interpretabilità	Incrementabilità	Efficienza	Scalabilità	Robustezza
⋆⋆	⋆⋆⋆	⋆⋆⋆⋆	⋆⋆	⋆⋆	⋆⋆

Tabella 21: Punteggi di valutazione delle caratteristiche dell'algoritmo k-Nearest Neighbour

Naive Bayes L'algoritmo di Naive Bayes è un metodo di classificazione semplice e basato sulla probabilità che calcola un insieme di probabilità contando la frequenza e le combinazioni di valori in un determinato dataset. L'algoritmo calcola la probabilità che un'istanza appartenga a una certa classe moltiplicando le probabilità individuali delle caratteristiche date quella classe e scegliendo la classe con probabilità massima. Perchè questo sia effettuabile senza considerare le probabilità condizionate però è necessaria l'assunzione che tutti gli attributi siano indipendenti dato il valore della variabile di classe. Questa ipotesi di indipendenza condizionale è raramente vera nelle applicazioni del mondo reale, da cui il termine "naive" (ingenuo) [46]. Sebbene la teoria abbia una forte congettura, l'algoritmo tende a funzionare bene e ad apprendere rapidamente in vari problemi di classificazione supervisionata. Nel nostro caso, avendo rimosso gli attributi maggiormente correlati, come spiegato nella Sezione 2.2.2, si può dire che ci siamo avvicinati anche nella pratica alla situazione di indipendenza condizionale ideale.

Questa "ingenuità" consente all'algoritmo di costruire facilmente regole di classificazione su dataset di grandi dimensioni senza dover ricorrere a complessi schemi iterativi di stima dei parametri, risultando in un'ottima incrementabilità, scalabilità, efficienza ed interpretabilità. L'accuratezza, invece, proprio per l'estrema semplicità del classificatore, non è il suo punto di forza, così come la robustezza, limitata dalla sensibilità alla

correlazione tra attributi [45]. In Tabella 22 sono riportati i punteggi ottenuti da Naive Bayes.

Naive Bayes					
Accuratezza	Interpretabilità	Incrementabilità	Efficienza	Scalabilità	Robustezza
⋆	⋆⋆⋆	⋆⋆⋆⋆⋆	⋆⋆⋆⋆⋆	⋆⋆⋆	⋆⋆

Tabella 22: Punteggi di valutazione delle caratteristiche dell'algoritmo Naive Bayes

Support Vector Machine Il funzionamento dell'algoritmo Support Vector Machine è il seguente. Durante la fase di addestramento, l'algoritmo dispone tutte le istanze in uno spazio di n-dimensioni, con n pari al numero di attributi presenti nel dataset. Quindi calcola un iperpiano, anch'esso in n-dimensioni, che divide lo spazio in tante sezioni quante classi presenti, cercando di massimizzare il "margine", ovvero la distanza tra le istanze e l'iperpiano. In fase di classificazione, l'algoritmo andrà a posizionare nello stesso spazio l'istanza da classificare e in base alla sua posizione questa verrà dichiarata di una o di un'altra classe.

Le performance di classificazione sono sicuramente tra le migliori in assoluto e anche l'efficienza, in fase di classificazione, è molto elevata. Tuttavia, il modello è difficilmente interpretabile, non è incrementale e richiede un tempo di addestramento e impostazione dei parametri non banale. La scalabilità però è buona, così come la robustezza [45]. La Tabella 23 valuta l'algoritmo SVM.

Support Vector Machine					
Accuratezza	Interpretabilità	Incrementabilità	Efficienza	Scalabilità	Robustezza
⋆⋆⋆⋆⋆	⋆	⋆⋆	⋆⋆	⋆⋆⋆	⋆⋆⋆

Tabella 23: Punteggi di valutazione delle caratteristiche dell'algoritmo Support Vector Machine

Confronto finale Si vogliono adesso confrontare i vari punteggi ottenuti dagli algoritmi. Per farlo, effettuiamo il prodotto matriciale tra il vettore delle importanze dei criteri tecnici, definito in Tabella 13, con la Tabella 24 riassuntiva dei punteggi dei singoli algoritmi, ottenendo la Tabella 25 con i punteggi finali e la relativa posizione in classifica. Ecco che si può capire come l'algoritmo che, date le sue caratteristiche, risulta essere più adatto al nostro problema sia quello di Naive Bayes, subito seguito da Decision Tree. Questo non ci stupisce perché sappiamo che i criteri di incrementabilità, efficienza

e scalabilità sono molto importanti, favorendo algoritmi più semplici. Al contrario, algoritmi più pesanti, anche se più accurati, come Random Forest, k-Nearest Neighbour e soprattutto Support Vector Machine, hanno ottenuto un punteggio molto basso. Questo ci dimostra come, nei sistemi complessi, non sempre la soluzione migliore sia quella che produce la predizione migliore.

Caratteristica	**Decision Tree**	**Random Forest**	**k-Nearest Neighbour**	**Naive Bayes**	**Support Vector Machine**
Accuratezza	⋆⋆	⋆⋆⋆	⋆⋆	⋆	⋆⋆⋆⋆⋆
Interpretabilità	⋆⋆⋆⋆⋆	⋆⋆	⋆⋆⋆	⋆⋆⋆	⋆
Incrementabilità	⋆⋆	⋆⋆	⋆⋆⋆	⋆⋆⋆⋆⋆	⋆⋆
Efficienza	⋆⋆⋆⋆⋆	⋆⋆⋆	⋆⋆	⋆⋆⋆⋆⋆	⋆⋆
Scalabilità	⋆⋆⋆	⋆⋆⋆	⋆⋆	⋆⋆⋆	⋆⋆⋆
Robustezza	⋆⋆	⋆⋆⋆	⋆⋆	⋆⋆	⋆⋆⋆

Tabella 24: Punteggi di valutazione delle caratteristiche degli algoritmi

Algoritmo	**Punteggio finale ottenuto**	**Posizione**
Decision Tree	66	2°
Random Forest	61	3°
k-Nearest Neighbour	48	5°
Naive Bayes	73	1°
Support Vector Machine	51	4°

Tabella 25: Punteggi finali ottenuti dagli algoritmi e posizione in classifica

2.3.2 Tuning

La fase di tuning fa riferimento al punto, nel quale, scelto un tipo di algoritmo, si cerca di ottimizzare gli iperparametri utilizzando un insieme di validazione. L'obiettivo del tuning è quindi quello di selezionare la configurazione di iperparametri, specifica per ogni tipo di algoritmo, che generalizza meglio su dati mai visti, massimizzando metriche di valutazione come accuratezza, F1-score o altre metriche utilizzate nel problema. Per raggiungere questo obiettivo si possono utilizzare diversi approcci, come la ricerca esaustiva tramite GridSearch o la ricerca non esaustiva tramite RandomizedSearch. La GridSearch è una tecnica che consente di valutare ogni possibile combinazione degli iperparametri specificati come input per un determinato algoritmo, garantendo un'esplorazione completa di tutte le configurazioni potenziali. Un significativo svantaggio risiede nell'elevato costo

computazionale e nella mancanza di efficienza di questa tecnica, specialmente quando ci si confronta con ampi spazi di ricerca. Questa inefficienza diventa particolarmente evidente in contesti in cui il numero di parametri da ottimizzare è elevato. La RandomizedSearch, invece, esegue un'esplorazione non esaustiva di uno spazio di iperparametri, selezionando in modo casuale ad ogni iterazione un insieme di valori da un insieme di iperparametri che sono stati inizialmente forniti come input. Questa seconda tecnica riduce significativamente i tempi di calcolo e risulta più efficiente per spazi di ricerca grandi, ma non garantisce di individuare la configurazione ottimale di iperparametri, testando solo combinazioni casuali.

Nel nostro caso si è deciso di provare tutte e cinque i diversi algoritmi di classificazione basati su Naive Bayes offerti dalla libreria ScikitLearn di Python [48], ovvero:

- Gaussian Naive Bayes;
- Multinomial Naive Bayes;
- Complement Naive Bayes;
- Bernoulli Naive Bayes;
- Categorical Naive Bayes.

Tra questi, Gaussian Naive Bayes ha come unico iperparametro *var_smoothing*, che aggiunge una piccola quantità alla varianza per migliorare la stabilità numerica del modello, mentre gli altri quattro hanno l'iperparametro *alpha*, utilizzato per la regolarizzazione Laplace, che evita di avere probabilità zero aggiungendo un valore costante al conteggio degli attributi. Le liste con i valori possibili degli iperparametri contengono il valore di default e poi i suoi tre ordini di grandezza successivi e precedenti. Quindi *var_smooth*: $\{10^{-12}, 10^{-11}, 10^{-10}, 10^{-9}, 10^{-8}, 10^{-7}, 10^{-6}\}$ con 10^{-9} valore di default. Mentre *alpha*: {0.001, 0.01, 0.1, 1, 10, 100, 1000} con 1 valore di default. Oltre ad indagare su una lista di iperparametri però, come detto nel Paragrafo 2.2.3, si è deciso di utilizzare la cross-validation per la validazione dei risultati su più folds, con lo scopo di ottenere stime delle performance più precise. ScikitLearn implementa delle versioni di GridSearch e RandomizedSearch che utilizzano la cross-validation, chiamate GridSearchCV e RandomizedSearchCV. Per la sua maggiore completezza, si è deciso di utilizzare la GridSearchCV con la metrica di accuratezza.

Il numero di operazioni di fit che andranno effettuate sarà quindi pari al prodotto tra il numero di algoritmi che si vogliono testare $n_{\text{algoritmi}} = 5$, per il numero di iperparametri

per algoritmo $n_{\text{iperparametri}} = 7$, per il numero di folds per ogni iterazione della GridSearch k=5, risultando in 175 operazioni.

$$N_{\text{fit}} = n_{\text{algoritmi}} \cdot n_{\text{iperparametri}} \cdot k$$

$$N_{\text{fit}} = 5 \cdot 7 \cdot 5 = 175$$

La combinazione ottima di tipo di algoritmo e iperparametri è stata individuata nella coppia *Multinomial Naive Bayes* e *alpha* = 0.1.

3 Risultati sperimentali e discussione

3.1 Evaluation

La fase di valutazione dei risultati, secondo [24], rappresenta il momento in cui i risultati del modello predittivo vengono analizzati e confrontati con i criteri di successo definiti nella sezione 2.1.3, che includono aspetti di business, tecnici ed economici. Se il confronto ha esito positivo, significa che il modello soddisfa i requisiti richiesti ed è pronto per l'implementazione. In caso contrario, un esito negativo del confronto indicherebbe che il modello presenta delle imperfezioni e necessita di ulteriori interventi.

3.1.1 Risultati ottenuti

I risultati di un modello di classificazione si ottengono applicando il modello ad un insieme di dati "nuovo", cioè diverso dai dati di addestramento e validazione, e calcolando alcune metriche standard basandosi su quante istanze sono correttamente predette e quante no. Questo insieme di dati e chiamato dataset di test ed è fondamentale sia sottoposto alle stesse operazioni di preprocessamento applicate al dataset di addestramento, in modo da garantire coerenza tra i due insieme di dati. Nel nostro caso, ciò riguarda principalmente la codifica delle variabili categoriche, che deve seguire le stesse regole o mappature definite durante l'addestramento. In caso contrario, il modello non sarebbe in grado di interpretare correttamente le nuove istanze, diventando inutilizzabile.

Come descritto nel Paragrafo sul partizionamento dei dati 2.2.3, il dataset di test corrisponde al 10% dei dati iniziali, includendo circa 520 fatture su un totale di 5200. È importante notare che il dataset di test non subisce la rimozione delle istanze associate a template poco frequenti, operazione invece effettuata sui dataset di addestramento e validazione, descritta nel Paragrafo 2.2.2. Questa scelta mira a rendere il dataset di test il

più rappresentativo possibile dei dati reali, includendo anche fatture associate a template rari.

Per ogni istanza del dataset di test, il modello fornisce la sua predizione e anche la relativa confidenza.

Nella Tabella 26 sono mostrati i risultati di classificazione aggregati e nelle Figure 17, 18, 19 sono mostrate le distribuzioni dei valori di precisione, richiamo ed F1-score.

	Precisione	**Richiamo**	**F1-score**
Media aritmetica	0.25	0.32	0.27
Media ponderata	0.46	0.60	0.51
Accuratezza complessiva	0.60		

Tabella 26: Risultati di classificazione

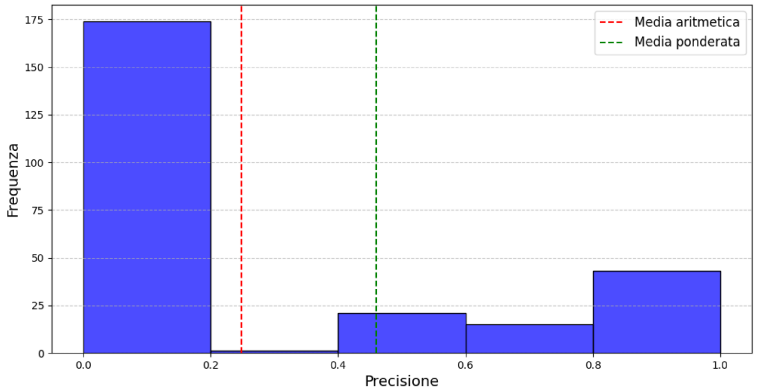

Figura 17: Distribuzione dei valori di *precisione* su 5 bins

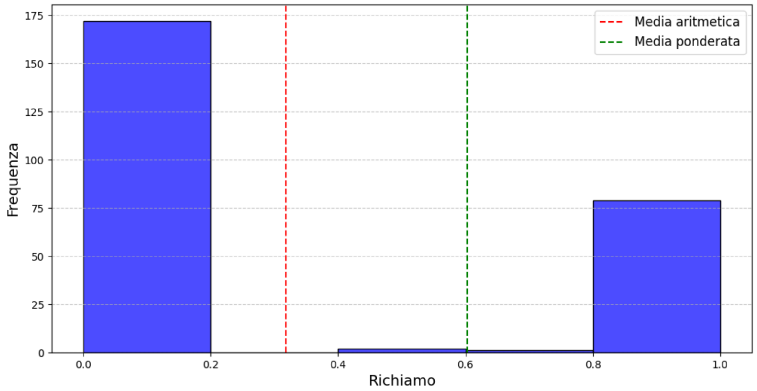

Figura 18: Distribuzione dei valori di *richiamo* su 5 bins

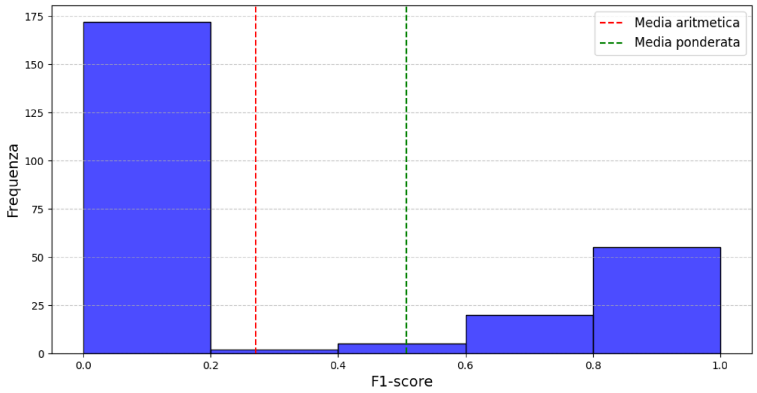

Figura 19: Distribuzione dei valori di *F1-score* su 5 bins

Si nota come questi siano, in generale, abbastanza bassi, e decisamente lontani rispetto all'obiettivo iniziale di una capacità predittiva, tradotta dalla metrica dell'accuratezza, superiore al 90%. I grafici, inoltre, mostrano distribuzioni non solo simili, ma caratterizzate da una forte concentrazione dei valori agli estremi, ossia 0 e 1. In particolare, circa la metà delle istanze presenta valori di precisione, richiamo e F1-score prossimi a 0, mentre l'altra metà tende verso 1. Questo indica che il modello si comporta in modo binario: o è in grado di prevedere una classe con elevata precisione, richiamo e F1-score, oppure non riesce a effettuare alcuna previsione significativa. In aggiunta, si vede nella Tabella 26 che i valori delle medie ponderate delle metriche sono circa il doppio dei valori delle medie aritmetiche. La media aritmetica calcola il valore medio delle metriche su tutte le classi, senza considerare la distribuzione delle istanze tra di esse. La media ponderata, invece, tiene conto del peso specifico di ciascuna classe, riflettendo la reale distribuzione delle istanze nel dataset.

Questi indizi conducono alla conclusione di come i risultati siano fortemente influenzati da classi poco numerose e che queste rappresentino circa la metà del dataset.

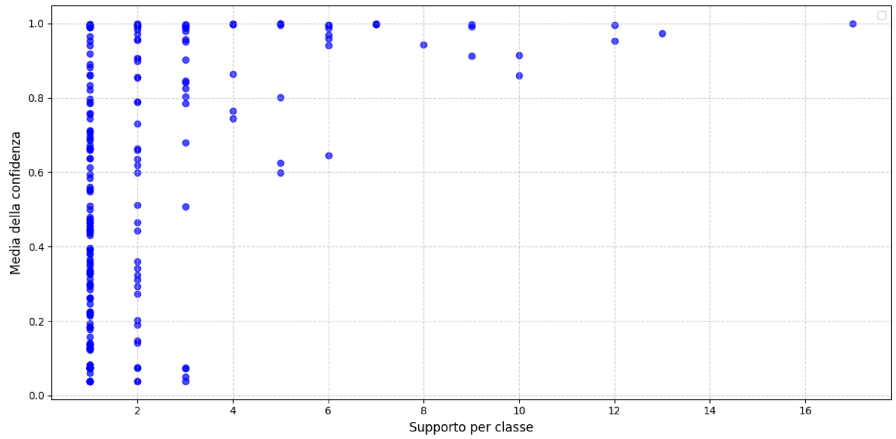

Figura 20: Valore medio della confidenza per supporto della classe

La confidenza media della predizione inoltre aumenta sensibilmente con l'aumentare del supporto della classe che, come evidenziato in Figura 20, rimane superiore al 90% se il supporto è di almeno 7.

Questo valore di confidenza, come spiegato nel Paragrafo 2.1.3, è il valore minimo di confidenza che permette di accettare la predizione e passare alla successive fasi di elaborazione della fattura. Tuttavia, in questo modo si escludono, dall'insieme delle istanze corrette, le istanze correttamente predette ma con una confidenza minore del 90%. Questo vincolo impone un ricalcolo dell'accuratezza complessiva, adottando una formula leggermente diversa e più esclusiva. L'accuratezza complessiva reale corrisponde quindi al rapporto tra il numero di istanze correttamente predette con una confidenza pari o superiore al 90% e il numero totale di istanze classificate. La Tabella 27 presenta la distribuzione delle istanze correttamente e non correttamente predette, suddivise in base alla confidenza della predizione ($\geq 90\%$ o $< 90\%$). Di conseguenza, l'accuratezza complessiva reale si attesta intorno a 0.48.

Confidenza	**Numero di istanze correttamente predette**	**Numero di istanze non correttamente predette**	**Totale**	**Accuratezza**
$\geq 90\%$	253	6	259	0.98
$< 90\%$	64	203	267	0.23
Totale	317	209	526	

Tabella 27: Distribuzione delle istanze corrette e non corrette in base alla confidenza predittiva e accuratezza relativa

$$\text{Accuratezza complessiva reale} = \frac{\text{n}_{\text{istanze corrette con confidenza } \geq 90\%}}{\text{n}_{\text{istanze totali}}}$$

$$\text{Accuratezza complessiva reale} = \frac{253}{526} = 48.10\%$$

Come era prevedibile, i risultati ottenuti evidenziano che il modello gestisce meglio le classi più frequenti. In particolare, si è riscontrato che un supporto di 7 rappresenta il minimo necessario per raggiungere una confidenza media predittiva di almeno il 90%. All'interno del sottoinsieme di classi con tale livello di supporto, l'accuratezza raggiunge circa il 98%.

Tuttavia, considerando che il caso di studio prevede un supporto minimo di 10 per includere una classe nel dataset di addestramento, come spiegato nel Paragrafo 2.2.2, è evidente che la principale limitazione deriva attualmente dalla scarsità di dati disponibili.

Pertanto, per affrontare efficacemente questo problema, sarebbe necessario acquisire ulteriori dati sia per la fase di addestramento sia per quella di test. In questo modo, si potrebbe migliorare la copertura delle classi meno rappresentate e ottimizzare le performance generali del modello.

Dalla Tabella 27 si possono inoltre ricavare i valori di $\%_{\text{storni}}$, $\%_{<90\%}$ e $\%_{\geq\ 90\%}$, necessari al calcolo del tempo medio di gestione di una fattura al tempo uno, ovvero dopo l'inserimento dell'automatismo. Quindi:

$$\%_{\text{storni}} = \frac{6}{526} = 1.14\%$$

$$\%_{<90\%} = \frac{267}{526} = 50.76\%$$

$$\%_{\geq\ 90\%} = \frac{267}{526} = 48.10\%$$

Utilizzando la formula espressa nel Paragrafo 2.1.3 si ottiene che il tempo medio di gestione di una fattura al tempo uno è pari a .

$$\bar{t}_{\text{gestione\_fattura\_1}} = \%_{\text{storni}} \cdot \bar{t}_{\text{storno}} + (\%_{\text{storni}} + \%_{<90\%}) \cdot \bar{t}_{\text{associazione}} + \%_{\geq\ 90\%} \cdot \bar{t}_{\text{sblocco}}$$

$$\bar{t}_{\text{gestione\_fattura\_1}} = 1.14\%\cdot 3\,\text{minuti} + (1.14\% + 50.76\%)\cdot 2\,\text{minuti} + 48.10\%\cdot 0.5\,\text{minuti} = 1.31\,\text{minuti}$$

3.1.2 Confronto con i criteri di successo

Nel paragrafo precedente è stato evidenziato che l'accuratezza complessiva reale, calcolata sui dati di test, è di circa il 48%. Questo risultato indica che, in media, il modello è in grado di classificare correttamente circa una fattura su due. Tale valore è significativamente distante dal principale criterio di business, descritto nel Paragrafo 2.1.3, che richiede un'accuratezza minima del 90%. Tuttavia, ci sono diverse considerazioni da tenere in conto.

In primo luogo, nella traduzione dei criteri di business in criteri tecnici, sono stati introdotti numerosi altri parametri di successo oltre all'accuratezza, come l'interpretabilità, l'incrementabilità, l'efficienza, la scalabilità e la robustezza. Come spiegato nel Paragrafo 2.3.1, l'algoritmo Naive Bayes è stato scelto poiché si adattava meglio al problema in questione. Pertanto, si può affermare che i criteri tecnici, ad eccezione dell'accuratezza, sono stati soddisfatti per costruzione.

In secondo luogo, il tempo medio di gestione di una fattura è sensibilmente diminuito tra i due periodi considerati, di pre e post implementazione. Infatti, questo è passato da essere pari a 2 minuti per fattura a 1.31 minuti per fattura. Percentualmente, questa variazione rappresenta una riduzione del 34.5% sui tempi. Tale riduzione, seppure di modesta entità, al momento, significa che l'applicazione ha raggiunto il criterio di successo economico individuato, riducendo il carico di lavoro degli operatori.

Per le considerazioni effettuate, quindi, si ritiene che il modello abbia tutti i requisiti per essere implementato.

3.2 Deployment

Come anticipato, le prestazioni del modello sono state ritenute accettabili e di conseguenza questo può venire implementato. Per ragioni legate ad aspetti puramente aziendali, la soluzione tecnica proposta in questo capitolo non verrà realmente implementata sui sistemi gestionali del Cliente di Leonardo, ma rimarrà tra le pagine di questa tesi, per il momento. Ciononostante, si vuole comunque cercare di trovare una soluzione che risponda ai requisiti tecnici e funzionali esposti nei Capitoli 2.1.2 e 2.1.3.

Entrando nel dettaglio, si è pensato di dividere il processo generale in quattro parti, riassunde schematicamente nella Figura 22.

Attività propedeutiche ed addestramento modello Questa parte del processo è fisicamente effettuata su tre macchine diverse, la prima è rappresentata dalla macchina dell'operatore contabile su cui è installato il sistema gestionale SAP S/4HANA VIM, la seconda è una macchina di laboratorio Leonardo S/4HANA (F4S) e la terza è un supercomputer connesso alla rete di F4S.

La soluzione proposta prevede la realizzazione di un cruscotto avanzato nel sistema gestionale SAP S/4 HANA VIM del Cliente, installato sulla prima macchina, in Figura 21. L'operatore, nella sezione di riferimento 'Addestramento Modello', avrà la possibilità di avviare lo svolgimento delle due attività propedeutiche, completamente automatiche, sotto riportate.

- Selezione delle fatture già contabilizzate a disposizione e relativi template di contabilizzazione: l'automatizzazione del processo di reperimento e inoltro dati può essere ottenuta utilizzando processi background (JOB SAP) oppure interazioni con l'utente tramite interfaccia grafica (SAP GUI / WEB GUI / UX FIORI).

- Mascherazione delle fatture: la mascherazione delle fatture tramite hash garantisce la protezione dei dati sensibili in esse contenuti.

Le fatture mascherate e i template relativi vengono quindi trasferiti, mediante tecnologia web service (SOAP/REST), alla macchina di laboratorio F4S. Durante la fase di addestramento, il numero di fatture potrebbe risultare elevato; pertanto, le chiamate al web service saranno opportunamente suddivise al fine di evitare il sovraccarico del servizio. Ogni chiamata, processata sequenzialmente, aggiornerà il repository dedicato presente nella macchina F4S. Questa macchina riceve i dati e costruisce un file Singularity[5], che invia al supercomputer disponibile in rete, il quale sarà incaricato del pesante lavoro di addestrare il modello.

Il file Singularity viene creato a partire da un file di definizione (.def), che specifica tutti gli step necessari, come il download delle librerie richieste (tra cui Scikit-Learn per l'algoritmo di classificazione, CUDA per l'utilizzo di GPU), la copia o il binding di file

[5]Singularity rappresenta un'implementazione di container runtime specificamente ottimizzata per ambienti High Performance Computing (HPC), possiede infatti la capacità di accedere direttamente alle risorse hardware specializzate, come GPU e interconnessioni ad alta velocità, senza compromettere le performance e lo rende significativamente più efficiente rispetto alle alternative containerizzate tradizionali come ad esempio Docker.

e directory, e soprattutto i parametri relativi alle risorse hardware da utilizzare nell'ambiente HPC, come le GPU Nvidia. Una volta completata la build, si ottiene un file unico in formato .sif, che viene trasferito al supercomputer, progettato per l'elaborazione rapida di grandi dataset. Questo sistema genererà ed eseguirà un container, avviando così l'addestramento del modello. Al termine dell'addestramento, si produrrà un file unico contenente il modello addestrato, pronto per essere utilizzato nelle successive fasi, e un file txt contenente l'esito dell'addestramento. Il file contenente il modello, potenzialmente di grandi dimensioni, viene salvato sulla macchina F4S di laboratorio, mentre l'esito dell'addestramento viene notificato, attraverso un gateway di comunicazione alla macchina del Cliente e all'operatore. Il cruscotto di monitoraggio permetterà di visualizzare gli esiti dell'addestramento nella sezione 'Addestramento Modello'.

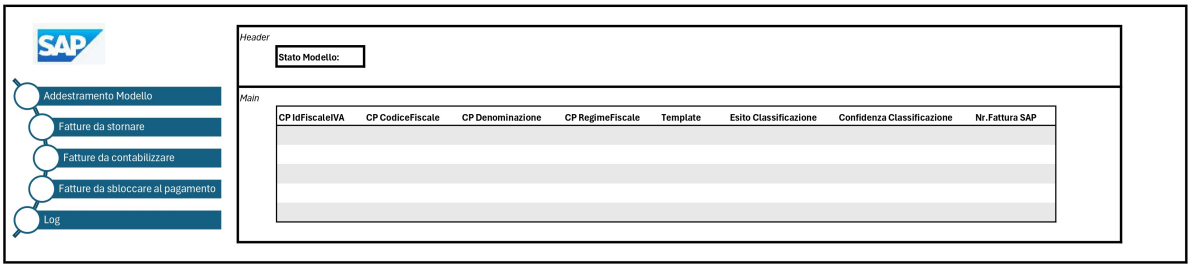

Figura 21: Cruscotto sul sistema SAP S/4HANA VIM proposto

Classificazione e contabilizzazione fattura La seconda parte del processo descrive le attività che sono effettuate automaticamente al momento di ingresso di una fattura sul sistema gestionale. Il processo si attiva con l'inoltro di una fattura su SAP S/4HANA VIM da parte di SAP PO. Subito il sistema procede con la sua mascherazione e, una volta eseguita questa operazione, con le stesse modalità di trasferimento descritte prima, la fattura è inviata alla macchina F4S, dove viene chiamato il modello ed effettuata la predizione. L'esito dell'attività, ovvero il template predetto e la confidenza relativa, sono inoltrati alla macchina S/4HANA VIM del Cliente. Se la confidenza della predizione è inferiore al 90% allora questa viene considerata non valida e viene salvata tra le 'fatture da contabilizzare'. Quando invece la predizione ha confidenza di almeno il 90%, il sistema procede automaticamente alla contabilizzazione tentando di usare il template indicato. Se questo template non è accettato dal sistema allora la fattura viene inserita tra le 'fatture da stornare', operazione che deve essere gestita da un operatore in un secondo momento. Se invece la confidenza della predizione è almeno 90% e il template viene accettato dal sistema, allora la fattura è inserita nell'insieme delle 'fatture da sbloccare al pagamento'.

Storno fatture L'operatore avrà la possibilità di controllare l'insieme delle fatture da stornare nell'apposita sezione 'Fatture da stornare' mostrata nel cruscotto in Figura 21. Per ogni fattura quindi ci sarà da annullare l'operazione di contabilizzazione e associare la fattura a un template esistente o a uno nuovo. Le fatture associate saranno salvate tra le 'fatture da sbloccare al pagamento'.

Contabilizzazione fatture Dalla sezione 'Fatture da contabilizzare' l'operatore potrà visionare le fatture di cui il modello non ha prodotto una predizione con una sufficiente confidenza. Qui, avrà la possibilità di associare la fattura a un template esistente o a uno nuovo. Il sistema salverà le fatture associate tra le 'fatture da sbloccare al pagamento'.

Sblocco al pagamento fatture Infine tutte le fatture confluiscono nella sezione 'Fatture da sbloccare al pagamento' in cui l'operatore può appunto sbloccare (o no) il trasferimento di denaro verso l'intestatario.

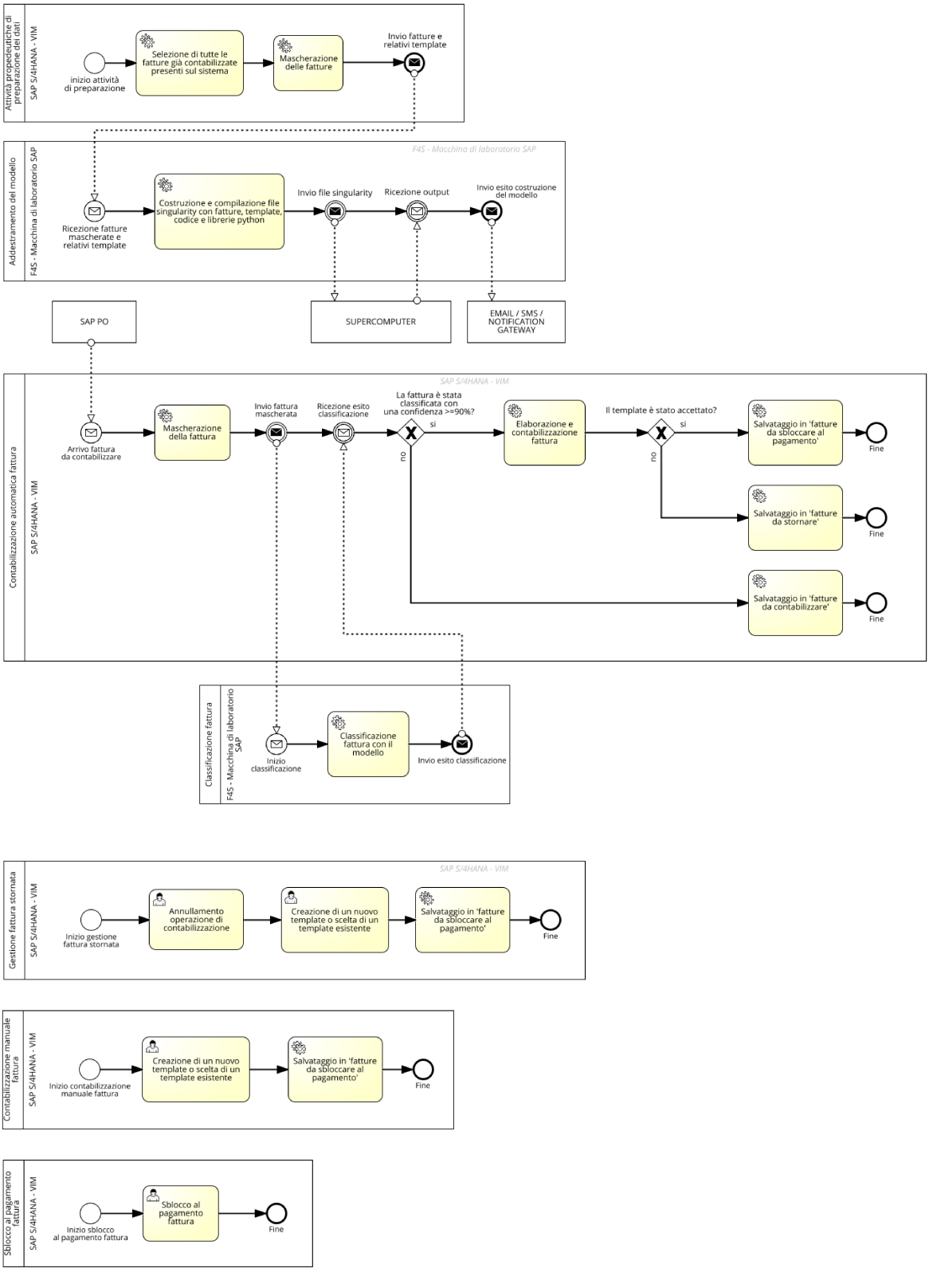

Figura 22: Diagramma BPMN della soluzione tecnica proposta

3.3 Monitoring and maintenance

L'ultima fase proposta dalla metodologia CRISP-ML riguarda la progettazione e l'implementazione di sistemi di monitoraggio e manutenzione per l'applicativo basato su machine learning, fondamentali per garantire la stabilità e l'efficacia delle prestazioni del modello nel tempo. Il concetto di "drift" assume un ruolo centrale in questa fase e si riferisce alla progressiva perdita di accuratezza o affidabilità del modello predittivo a causa di variazioni nei dati di input o nelle condizioni operative, che possono essere causate da un cambiamento nella relazione tra le variabili indipendenti e la variabile dipendente (ad esempio in presenza di nuove classi e quindi template) [29]. Per affrontare il drift, è essenziale implementare un sistema di monitoraggio continuo delle prestazioni del modello e dei dati, valutando costantemente le metriche di performance scelte per la propria applicazione, come l'accuratezza complessiva reale, confrontando le distribuzioni dei nuovi dati con i dati di addestramento attraverso test statistici, rilevando anomalie e pianificando il riaddestramento del modello con nuovi dati raccolti per garantire l'allineamento con il contesto operativo attuale. Inoltre, la manutenzione del modello prevede l'automazione delle fasi di riaddestramento e validazione, la gestione delle versioni del modello per garantire tracciabilità e rollback, e l'integrazione di sistemi di allerta per segnalare anomalie o cali di prestazioni. L'adozione di un approccio proattivo alla gestione del drift e alla manutenzione del modello non solo migliora le prestazioni complessive dell'applicativo di machine learning, ma aumenta anche la fiducia degli utenti e delle parti interessate nel sistema implementato.

Nel nostro caso, il monitoraggio delle performance del modello avverrà attraverso la sezione dedicata nel cruscotto mostrato in Figura 21. Qui l'operatore potrà visualizzare il KPI di riferimento, $\bar{t}_{\text{gestione\_fattura}}$, e le sue componenti. Qualora qualcuno di questi elementi dovesse raggiungere una soglia, considerata critica, il sistema potrebbe inviare una notifica all'operatore. Quest'ultimo avrebbe quindi la possibilità di effettuare un riaddestramento del modello seguendo le attività descritte in precedenza nel Paragrafo 3.2. Un'altra opzione possibile è automatizzare completamente questa procedura, in questo caso il sistema si accorge da solo che il KPI è fuori dai limiti e quindi manda in riaddestramento il modello.

4 Conclusioni

Questo lavoro di tesi ha esplorato l'applicazione del machine learning in un ambito ancora fortemente caratterizzato da operazioni manuali e ripetitive: le registrazioni contabili.

Dopo un'introduzione ai sistemi e agli attori coinvolti nel processo, è stata condotta un'accurata ricerca bibliografica per individuare eventuali studi precedenti sul tema e valutarne i risultati. L'analisi ha confermato come il machine learning sia già stato applicato con successo in ambiti contabili, tra cui l'estrazione di informazioni da documenti, la rilevazione di transazioni fraudolente e l'assegnazione automatica dei conti di registrazione delle fatture.

Dopo averne verificato l'applicabilità, l'attenzione si è focalizzata su un caso concreto segnalato dai tecnici di Leonardo. Collaborando con un importante Cliente della Pubblica Amministrazione, è stato individuato un processo contabile con un forte potenziale di ottimizzazione. In particolare, la registrazione delle fatture prevedeva una fase di associazione con un modello di contabilizzazione, un'attività svolta manualmente da operatori non specializzati e spesso caratterizzata da tempi lunghi. Per rendere il processo più efficiente, il progetto di tesi si è concentrato sulla ricerca di un sistema automatizzato in grado di gestire questa associazione.

La prima fase del lavoro ha previsto uno studio approfondito del processo, dei dati disponibili e degli obiettivi aziendali, con la definizione di criteri di successo a livello aziendale, tecnico ed economico. Si è poi passati alla fase sperimentale, che ha incluso l'estrazione, la preparazione e il preprocessamento dei dati. Sono stati affrontati problemi quali la conversione dei dati da formato XML a tabulare, la selezione delle variabili indipendenti e della variabile dipendente (il modello di contabilizzazione) e l'organizzazione del dataset in un formato facilmente gestibile con Python.

Successivamente, è stato condotto un confronto tra diversi algoritmi di classificazione, valutandone punti di forza e debolezze. In base ai criteri di successo definiti, l'algoritmo Naive Bayes è risultato il più adatto alle esigenze del caso. Il modello è stato quindi ottimizzato attraverso una fase di tuning, mirata alla scelta dei parametri più efficaci.

Una volta applicato l'algoritmo al dataset costruito, ne sono state analizzate le prestazioni nel contesto specifico. I risultati sono stati incoraggianti: sebbene l'accuratezza complessiva reale si sia attestata al 48%, l'analisi successiva ha evidenziato una caratteristica strutturale del dataset che ha influenzato le prestazioni. Nello specifico, molte classi erano rappresentate da pochissimi esempi, il che ha reso complessivamente meno efficiente la classificazione. Tuttavia, per le classi con almeno sette occorrenze, l'algoritmo ha

raggiunto un'accuratezza del 98%, dimostrando eccellenti capacità predittive in presenza di un numero adeguato di dati.

Alla luce di questi risultati, l'applicazione è stata considerata un successo. Gli esperti del settore hanno infatti confermato che i modelli di contabilizzazione vengono utilizzati principalmente su volumi elevati di fatture, ovvero in situazioni in cui il supporto dei dati è consistente e dove il modello ha dimostrato di funzionare in modo ottimale. Questo suggerisce che un aumento dei dati disponibili per l'addestramento potrebbe migliorare ulteriormente le prestazioni del classificatore, rendendolo ancora più affidabile in contesti reali.

Una volta verificata la validità della soluzione proposta, si è iniziato a progettare un processo e un'architettura per la sua integrazione nei sistemi esistenti. Questa fase ha richiesto un costante dialogo con i tecnici esperti dei sistemi informativi coinvolti. Il risultato è stato un processo che soddisfaceva tutti i requisiti funzionali e tecnici, con l'obiettivo principale di ridurre il carico di lavoro degli operatori. Tuttavia, si è compreso che i tempi e le modalità di implementazione sui sistemi del Cliente non erano compatibili con le tempistiche di un lavoro di tesi. Per questo motivo, la soluzione è stata teorizzata e descritta, senza essere implementata operativamente.

Infine, il lavoro si è concentrato sulla manutenibilità della soluzione nel tempo. Sono stati descritti concetti chiave come il drift dei dati e l'importanza di strumenti di monitoraggio delle prestazioni, essenziali per garantire l'efficacia del modello anche in futuro. In base ai dati raccolti nel tempo, i gestori del sistema potranno aggiornare periodicamente il classificatore, utilizzando un dataset sempre più rappresentativo della realtà operativa.

I risultati ottenuti dimostrano che esistono margini di miglioramento significativi nel campo delle registrazioni contabili e che sono affrontabili con tecniche di machine learning. Un possibile sviluppo futuro potrebbe consistere nello spostare il problema dall'apprendimento supervisionato a quello non supervisionato. In questo scenario, un algoritmo potrebbe identificare autonomamente gruppi di fatture simili, creare modelli di contabilizzazione senza intervento umano e procedere direttamente alla registrazione contabile.

Riferimenti bibliografici

[1] Agenzia per l'Italia Digitale. *Fatturazione elettronica: oltre due miliardi le fatture emesse in un anno.* URL: `https://www.agid.gov.it/it/agenzia/stampa-e-`

comunicazione/notizie/2020/04/24/fatturazione-elettronica-oltre-due-miliardi-fatture-emesse-anno.

[2] Marco Torchiano Luca Ardito Fulvio Corno. *Sistemi Informativi Aziendali*. Politecnico di Torino – Dipartimento di Automatica e Informatica, 2022.

[3] Robert Newton Anthony. *Planning and Control Systems: A Framework for Analysis*. Division of Research, Graduate School of Business Administration, Harvard University, 1965.

[4] Guy G. Gable Helmut Klaus Michael Rosemann. "What is ERP?" In: *Kluwer Academic Publishers* (2000).

[5] Francis Buttle. "The CRM Value Chain". In: *Macquarie University* (2000).

[6] Jürgen Kletti. "Manufacturing Execution Systems – MES". In: *Springer* (2007).

[7] Oracle. *Cos'è l'SCM (Supply Chain Management)?* URL: https://www.oracle.com/it/scm/what-is-supply-chain-management/.

[8] Matteo Golfarelli. *Classificazione dei Sistemi Informativi*. URL: http://bias.csr.unibo.it/golfarelli/SISPEC/dispense/ClassificazioneSI.pdf.

[9] SAP. *Che cos'è SAP*. URL: https://www.sap.com/italy/about/what-is-sap.html.

[10] SAP. *Che cos'è SAP HANA*. URL: https://www.sap.com/italy/products/technology-platform/hana/what-is-sap-hana.html.

[11] Treccani. *Stakeholder definizione*. URL: https://www.treccani.it/enciclopedia/stakeholder/.

[12] Chiara Bardelli et al. "Automatic Electronic Invoice Classification Using Machine Learning Models". In: *Machine Learning and Knowledge Extraction* 2.4 (2020), pp. 617–629. ISSN: 2504-4990. DOI: 10.3390/make2040033. URL: https://www.mdpi.com/2504-4990/2/4/33.

[13] Y.Y. Tang et al. "Financial document processing based on staff line and description language". In: *IEEE Transactions on Systems, Man, and Cybernetics* 25.5 (1995), pp. 738–754. DOI: 10.1109/21.376488.

[14] F. Cesarini et al. "INFORMys: a flexible invoice-like form-reader system". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 20.7 (1998), pp. 730–745. DOI: 10.1109/34.689303.

[15] Xavier Holt e Andrew Chisholm. “Extracting structured data from invoices”. In: *Proceedings of the Australasian Language Technology Association Workshop 2018*. 2018, pp. 53–59.

[16] Yu Wang et al. “Deep learning for optical character recognition and its application to VAT invoice recognition”. In: *Communications, Signal Processing, and Systems: Proceedings of the 2018 CSPS Volume III: Systems 7th*. Springer. 2020, pp. 87–95.

[17] Ahmad S. Tarawneh et al. “Invoice Classification Using Deep Features and Machine Learning Techniques”. In: *2019 IEEE Jordan International Joint Conference on Electrical Engineering and Information Technology (JEEIT)*. 2019, pp. 855–859. DOI: 10.1109/JEEIT.2019.8717504.

[18] Marco Schreyer et al. “Detection of anomalies in large scale accounting data using deep autoencoder networks”. In: *arXiv preprint arXiv:1709.05254* (2017).

[19] Mario Zupan, Svjetlana Letinic e Verica Budimir. “Accounting Journal Reconstruction with Variational Autoencoders and Long Short-term Memory Architecture.” In: *SEBD*. 2020, pp. 88–99.

[20] Martin Schultz e Marina Tropmann-Frick. “Autoencoder neural networks versus external auditors: Detecting unusual journal entries in financial statement audits”. In: (2020).

[21] Hu Peiguang. “Predicting and improving invoice-to-cash collection through machine learning”. In: *Massachusetts Institute of Technology* 92 (2015).

[22] Johan Bergdorf. *Machine learning and rule induction in invoice processing: Comparing machine learning methods in their ability to assign account codes in the bookkeeping process*. 2018.

[23] Hampus Bengtsson e Johannes Jansson. “Using classification algorithms for smart suggestions in accounting systems”. In: (2015).

[24] Stefan Studer et al. “Towards CRISP-ML(Q): A Machine Learning Process Model with Quality Assurance Methodology”. In: *Machine Learning and Knowledge Extraction* 3.2 (2021), pp. 392–413. ISSN: 2504-4990. DOI: 10.3390/make3020020. URL: https://www.mdpi.com/2504-4990/3/2/20.

[25] Agenzia delle Entrate - Direzione Centrale Normativa e Contenzioso. *Circolare del 19/10/2005 n. 45*. URL: https://www.agenziaentrate.gov.it/portale/documents/20143/392503/Circolare+45+del+19_10_2005_circolare_45_2005.pdf/5bb34e52-3558-8ef6-a6d9-fa6e91468370.

[26] Agenda Digitale. *Fatturazione elettronica obbligatoria: cos'è, come funziona, come fare, esoneri e normativa (tra privati, PA e B2B)*. URL: https://www.agendadigitale.eu/documenti/fatturazione-elettronica/fatturazione-elettronica-tra-privati-quale-futuro-ci-attende/#:~:text=L'obbligo%20riguarda%20inizialmente%20la,esteso%20anche%20alle%20PA%20locali.

[27] Agenzia dell'Entrate. *Come si conservano le fatture elettroniche*. URL: https://www.agenziaentrate.gov.it/portale/web/guest/aree-tematiche/fatturazione-elettronica/guida-fatturazione-elettronica/come-predisporre-inviare-ricevere-fe/come-si-conservano-fe.

[28] David J Schultz et al. "IEEE standard for developing software life cycle processes". In: *IEEE Std* (1997), pp. 1074–1997.

[29] Jan Zenisek, Florian Holzinger e Michael Affenzeller. "Machine learning based concept drift detection for predictive maintenance". In: *Computers & Industrial Engineering* 137 (2019), p. 106031.

[30] Yasuhiro Watanabe et al. "Preliminary literature review of machine learning system development practices". In: *2021 IEEE 45th Annual Computers, Software, and Applications Conference (COMPSAC)*. IEEE. 2021, pp. 1407–1408.

[31] Sheng-yi Jiang e Lian-xi Wang. "Efficient feature selection based on correlation measure between continuous and discrete features". In: *Information Processing Letters* 116.2 (2016), pp. 203–215. ISSN: 0020-0190. DOI: https://doi.org/10.1016/j.ipl.2015.07.005. URL: https://www.sciencedirect.com/science/article/pii/S0020019015001271.

[32] Mark Last, Abraham Kandel e Oded Maimon. "Information-theoretic algorithm for feature selection". In: *Pattern Recognition Letters* 22.6 (2001), pp. 799–811. ISSN: 0167-8655. DOI: https://doi.org/10.1016/S0167-8655(01)00019-8. URL: https://www.sciencedirect.com/science/article/pii/S0167865501000198.

[33] Feng Jiang, Yuefei Sui e Lin Zhou. “A relative decision entropy-based feature selection approach”. In: *Pattern Recognition* 48.7 (2015), pp. 2151–2163. ISSN: 0031-3203. DOI: https://doi.org/10.1016/j.patcog.2015.01.023. URL: https://www.sciencedirect.com/science/article/pii/S0031320315000424.

[34] Yumin Chen, Duoqian Miao e Ruizhi Wang. “A rough set approach to feature selection based on ant colony optimization”. In: *Pattern Recognition Letters* 31.3 (2010), pp. 226–233. ISSN: 0167-8655. DOI: https://doi.org/10.1016/j.patrec.2009.10.013. URL: https://www.sciencedirect.com/science/article/pii/S0167865509002888.

[35] Yuanhong Li, Ming Dong e Jing Hua. “Localized feature selection for clustering”. In: *Pattern Recognition Letters* 29.1 (2008), pp. 10–18. ISSN: 0167-8655. DOI: https://doi.org/10.1016/j.patrec.2007.08.012. URL: https://www.sciencedirect.com/science/article/pii/S0167865507002632.

[36] Ron Kohavi e George H. John. “Wrappers for feature subset selection”. In: *Artificial Intelligence* 97.1 (1997). Relevance, pp. 273–324. ISSN: 0004-3702. DOI: https://doi.org/10.1016/S0004-3702(97)00043-X. URL: https://www.sciencedirect.com/science/article/pii/S000437029700043X.

[37] Sebastián Maldonado e Richard Weber. “A wrapper method for feature selection using Support Vector Machines”. In: *Information Sciences* 179.13 (2009). Special Section on High Order Fuzzy Sets, pp. 2208–2217. ISSN: 0020-0255. DOI: https://doi.org/10.1016/j.ins.2009.02.014. URL: https://www.sciencedirect.com/science/article/pii/S0020025509000917.

[38] Ewa Skotarczak, Anita Dobek e Krzysztof Moliński. “Comparison of some correlation measures for continuous and categorical data”. In: *Biometrical Letters* 56.2 (2019), pp. 253–261.

[39] Zeyneb Kurt Yavuz, Nizamettin Aydin e Gökmen Altay. “Comprehensive review of association estimators for the inference of gene networks”. In: *Turkish Journal of Electrical Engineering and Computer Sciences* 24.3 (2016), pp. 695–718.

[40] H Cramer. “Mathematical methods of statistics, Princeton, 1946”. In: *Math Rev (Math-SciNet) MR16588 Zentralblatt MATH* 63 (1946), p. 300.

[41] Mark A Hall. “Correlation-based feature selection for machine learning”. Tesi di dott. The University of Waikato, 1999.

[42] FY Osisanwo et al. “Supervised machine learning algorithms: classification and comparison”. In: *International Journal of Computer Trends and Technology (IJCTT)* 48.3 (2017), pp. 128–138.

[43] Taiwo Oladipupo Ayodele. “Types of Machine Learning Algorithms”. In: *New Advances in Machine Learning*. A cura di Yagang Zhang. Rijeka: IntechOpen, 2010. Cap. 3. DOI: 10.5772/9385. URL: https://doi.org/10.5772/9385.

[44] Sotiris B Kotsiantis, Ioannis Zaharakis, P Pintelas et al. “Supervised machine learning: A review of classification techniques”. In: *Emerging artificial intelligence applications in computer engineering* 160.1 (2007), pp. 3–24.

[45] Elena Baralis. *Slides del corso Business Intelligence per Big Data*. Politecnico di Torino, 2024.

[46] Shahadat Uddin et al. “Comparing different supervised machine learning algorithms for disease prediction”. In: *BMC medical informatics and decision making* 19.1 (2019), pp. 1–16.

[47] David W Aha, Dennis Kibler e Marc K Albert. “Instance-based learning algorithms”. In: *Machine learning* 6 (1991), pp. 37–66.

[48] URL: https://scikit-learn.org/1.5/modules/naive_bayes.html.

[49] FatturaPA. URL: https://www.fatturapa.gov.it/export/documenti/fatturapa/v1.2/IT01234567890_FPA02.xml.

Appendices

Fattura elettronica B2G di esempio, con una sola linea di dettaglio

```
1 <p:FatturaElettronica
    xmlns:ds="http://www.w3.org/2000/09/xmldsig#"
    xmlns:p="http://ivaservizi.agenziaentrate.gov.it/doc/xsd/fatture/v1.2"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    versione="FPA12"
    xsi:schemaLocation="http://ivaservizi.agenziaentrate.gov.it/doc/xsd/fattu
    http://www.fatturapa.gov.it/export/fatturazione/sdi/fatturapa/v1.2/Schema
2     <script/>
3         <FatturaElettronicaHeader>
4             <DatiTrasmissione>
```

```
5               <IdTrasmittente>
6                   <IdPaese>IT</IdPaese>
7                   <IdCodice>01234567890</IdCodice>
8               </IdTrasmittente>
9               <ProgressivoInvio>00001</ProgressivoInvio>
10                  <FormatoTrasmissione>FPA12</FormatoTrasmissione>
11                  <CodiceDestinatario>AAAAAA</CodiceDestinatario>
12          </DatiTrasmissione>
13          <CedentePrestatore>
14              <DatiAnagrafici>
15                  <IdFiscaleIVA>
16                      <IdPaese>IT</IdPaese>
17                      <IdCodice>01234567890</IdCodice>
18                  </IdFiscaleIVA>
19                  <Anagrafica>
20                      <Denominazione>ALPHA SRL</Denominazione>
21                  </Anagrafica>
22                  <RegimeFiscale>RF19</RegimeFiscale>
23              </DatiAnagrafici>
24              <Sede>
25                  <Indirizzo>VIALE ROMA 543</Indirizzo>
26                  <CAP>07100</CAP>
27                  <Comune>SASSARI</Comune>
28                  <Provincia>SS</Provincia>
29                  <Nazione>IT</Nazione>
30              </Sede>
31          </CedentePrestatore>
32          <CessionarioCommittente>
33              <DatiAnagrafici>
34                  <CodiceFiscale>09876543210</CodiceFiscale>
35                  <Anagrafica>
36                      <Denominazione>AMMINISTRAZIONE
                           BETA</Denominazione>
37                  </Anagrafica>
38              </DatiAnagrafici>
39              <Sede>
40                  <Indirizzo>VIA TORINO 38-B</Indirizzo>
```

```
                <CAP>00145</CAP>
                <Comune>ROMA</Comune>
                <Provincia>RM</Provincia>
                <Nazione>IT</Nazione>
            </Sede>
        </CessionarioCommittente>
    </FatturaElettronicaHeader>
    <FatturaElettronicaBody>
        <DatiGenerali>
        <DatiGeneraliDocumento>
            <TipoDocumento>TD01</TipoDocumento>
            <Divisa>EUR</Divisa>
            <Data>2017-01-18</Data>
            <Numero>123</Numero>
            <Causale>LA FATTURA FA RIFERIMENTO AD UNA
               OPERAZIONE AAAA BBBBBBBBBBBBBBBBB CCC
               DDDDDDDDDDDDDD E FFFFFFFFFFFFFFFFFFF
               GGGGGGGGGG HHHHHHH II LLLLLLLLLLLLLLL MMM
               NNNNN OO PPPPPPPPPPP QQQQ RRRR
               SSSSSSSSSSSSS</Causale>
            <Causale>SEGUE DESCRIZIONE CAUSALE NEL CASO IN
               CUI NON SIANO STATI SUFFICIENTI 200 CARATTERI
               AAAAAAAAAAA BBBBBBBBBBBBBBBB</Causale>
        </DatiGeneraliDocumento>
        <DatiOrdineAcquisto>
            <RiferimentoNumeroLinea>1</RiferimentoNumeroLinea>
            <IdDocumento>66685</IdDocumento>
            <NumItem>1</NumItem>
            <CodiceCUP>123abc</CodiceCUP>
            <CodiceCIG>456def</CodiceCIG>
        </DatiOrdineAcquisto>
        <DatiContratto>
            <RiferimentoNumeroLinea>1</RiferimentoNumeroLinea>
            <IdDocumento>123</IdDocumento>
            <Data>2016-09-01</Data>
            <NumItem>5</NumItem>
            <CodiceCUP>123abc</CodiceCUP>
```

```
71              <CodiceCIG>456def</CodiceCIG>
72          </DatiContratto>
73              <DatiConvenzione>
74              <RiferimentoNumeroLinea>1</RiferimentoNumeroLinea>
75              <IdDocumento>456</IdDocumento>
76              <NumItem>5</NumItem>
77              <CodiceCUP>123abc</CodiceCUP>
78              <CodiceCIG>456def</CodiceCIG>
79          </DatiConvenzione>
80              <DatiRicezione>
81              <RiferimentoNumeroLinea>1</RiferimentoNumeroLinea>
82              <IdDocumento>789</IdDocumento>
83              <NumItem>5</NumItem>
84              <CodiceCUP>123abc</CodiceCUP>
85              <CodiceCIG>456def</CodiceCIG>
86          </DatiRicezione>
87              <DatiTrasporto>
88                  <DatiAnagraficiVettore>
89                  <IdFiscaleIVA>
90                      <IdPaese>IT</IdPaese>
91                      <IdCodice>24681012141</IdCodice>
92                  </IdFiscaleIVA>
93                  <Anagrafica>
94                      <Denominazione>Trasporto
                           spa</Denominazione>
95                  </Anagrafica>
96              </DatiAnagraficiVettore>
97              <DataOraConsegna>2017-01-10T16:46:12.000+02:00</DataOraConse
98              </DatiTrasporto>
99          </DatiGenerali>
100         <DatiBeniServizi>
101             <DettaglioLinee>
102                 <NumeroLinea>1</NumeroLinea>
103                 <Descrizione>DESCRIZIONE DELLA
                       FORNITURA</Descrizione>
104                 <Quantita>5.00</Quantita>
105                 <PrezzoUnitario>1.00</PrezzoUnitario>
```

```
106                     <PrezzoTotale>5.00</PrezzoTotale>
107                     <AliquotaIVA>22.00</AliquotaIVA>
108                 </DettaglioLinee>
109                 <DatiRiepilogo>
110                     <AliquotaIVA>22.00</AliquotaIVA>
111                     <ImponibileImporto>5.00</ImponibileImporto>
112                     <Imposta>1.10</Imposta>
113                     <EsigibilitaIVA>I</EsigibilitaIVA>
114                 </DatiRiepilogo>
115             </DatiBeniServizi>
116             <DatiPagamento>
117                 <CondizioniPagamento>TP01</CondizioniPagamento>
118             <DettaglioPagamento>
119                 <ModalitaPagamento>MP01</ModalitaPagamento>
120                 <DataScadenzaPagamento>2017-02-18</DataScadenzaPagamento>
121                 <ImportoPagamento>6.10</ImportoPagamento>
122             </DettaglioPagamento>
123         </DatiPagamento>
124     </FatturaElettronicaBody>
125 </p:FatturaElettronica>
```

Listing 1: Fattura in formato XML di esempio, presa da [49]