* Document intelligence, an extension of OCR, automates the extraction, understanding, and storage of text data from documents.
* It addresses the need for processing large volumes of documents, such as receipts, for tasks like expense claims and project costs.
* Manual data entry is slow and error-prone, prompting the adoption of automated solutions like document intelligence.
* Azure AI Document Intelligence facilitates the analysis of documents and forms using both prebuilt and custom models.
* Specific data points like merchant names, addresses, and total values can be extracted efficiently.
* This technology enhances efficiency and accuracy in document processing workflows for various applications like expense report processing and invoice management.
* Azure AI services provide convenient access to document intelligence capabilities for organizations seeking to streamline their document processing tasks.
* Document intelligence utilizes machine learning models trained to recognize data in text, including text extraction, layout analysis, and key-value pair extraction.
* Document analysis involves identifying text locations on a page using bounding box coordinates, enabling the extraction of specific data such as addresses, names, and values.
* Machine learning models interpret document data by recognizing patterns in bounding box coordinates and text.
* The challenge in automating document analysis lies in the diverse formats of forms and documents, requiring separate training of machine learning models for different document types.
* Prebuilt machine learning models may suffice for commonly used document formats, while custom models are necessary for unique document layouts.
* Automating text reading and data recording processes can enhance operational efficiency, customer experiences, decision-making, and more.
* Azure AI services offer solutions for implementing document intelligence, providing tools to streamline document analysis workflows.

 **Azure AI Document Intelligence**: It comprises prebuilt and custom models for document analysis.

 **Prebuilt models**: These are pretrained models designed to process common document types like invoices, business cards, ID documents, etc., and extract specific fields crucial for each document type.

* **Example**: The prebuilt receipt model can match field names to values, identify tables of data, and extract specific fields like dates, totals, etc., from various receipt types.
* **Fields recognized**: Merchant details, purchase date/time, item details (name, quantity, price), total amount, tax values, etc.
* **Confidence levels**: Each recognized field has a confidence level indicating its accuracy.
* **Image requirements**: Images should be in JPEG, PNG, BMP, PDF, or TIFF format, within specified size and resolution limits.

 **Document Intelligence Studio**: It's a user interface for testing document analysis, prebuilt models, and creating custom models.

 **Azure AI Document Intelligence resource**: To utilize it, create either a Document Intelligence or Azure AI services resource in Azure subscription, preferably starting with the free tier for initial exploration.

 **Introduction to Knowledge Mining**:

* It addresses the challenge of extracting information from unstructured documents.
* Solutions like Azure AI Search facilitate automating information extraction from large volumes of data.

 **Azure AI Search**:

* Cloud search service facilitating building user-managed indexes.
* Capable of handling unstructured, typed, image-based, or hand-written documents.
* Can utilize Azure AI services' capabilities for image processing, content extraction, and natural language processing.

**Azure AI Search Overview:**

* **Infrastructure and Tools:** Provides the infrastructure and tools for creating search solutions.
* **Data Extraction:** Extracts data from various structured, semi-structured, and non-structured documents.
* **Platform as a Service (PaaS):** Managed by Microsoft, eliminating the need for dedicated hardware resources.

**Key Features:**

* **Data Source Flexibility:** Accepts data from any source in JSON format, with auto crawling support for Azure data sources.
* **Full Text Search:** Supports both simple query and full Lucene query syntax for comprehensive text search.
* **AI-Powered Search:** Utilizes Azure AI capabilities for image and text analysis from raw content.
* **Multi-Lingual Support:** Linguistic analysis for 56 languages, used by Bing and Office for natural language processing.
* **Geo-Enabled:** Supports geo-search filtering based on physical location proximity.
* **Configurable User Experience:** Features like autocomplete, autosuggest, pagination, and hit highlighting for enhanced user experience.

**Azure AI Search Solution Overview:**

* **Data Source:** Begins with a data source containing the artifacts to be searched, such as Azure Storage or databases like Azure SQL Database or Azure Cosmos DB.
* **Data Format:** Supports JSON format for indexing, regardless of the data's original source.

**Indexers for Data Ingestion:**

* **Automation:** Indexers automate data ingestion, including JSON serialization of source data.
* **Change Detection:** Supports change detection for data refresh, simplifying the process.

**AI Enrichment with Skillsets:**

* **Enhanced Search:** Skillsets attach a sequence of AI skills to enrich data, improving searchability.
* **Built-in Skills:** Utilizes built-in skills based on Azure AI services APIs for tasks like entity recognition, translation, sentiment analysis, and image captioning.
* **Knowledge Store:** Optionally sends enriched content to a knowledge store in Azure Storage for independent analysis or downstream processing.

**Index Persistence and Search:**

* **Index Persistence:** Fields containing content are persisted in an index for searching, filtering, and sorting.
* **Client Application Integration:** Client applications can search the index for relevant results to display or use further.

**AI Enrichment Skillset Overview:**

**Definition:** AI enrichment involves embedding image and natural language processing in a pipeline to extract text and information from content for full text search.

**Skillset Composition:** Defined by adding and combining skills to extract and enrich data for searchability.

**Types of Skills:**

1. **Built-in Skills:**
   * **Nature:** Based on pretrained models from Microsoft.
   * **Training:** Can't be trained with custom data.
   * **Billing:** Billed at Azure AI services pay-as-you-go price.

**Categories:**

* + **Natural Language Processing Skills:**
    - **Key Phrase Extraction:** Detects important phrases based on term placement, linguistic rules, and more.
    - **Text Translation Skill:** Translates text into various languages.
  + **Image Processing Skills:**
    - **Image Analysis Skill:** Identifies image content and generates text descriptions.
    - **Optical Character Recognition Skill (OCR):** Extracts printed or handwritten text from images and documents.

**Purpose:** Enhances search capabilities by converting unstructured content into searchable and filterable fields, enabling more accurate and efficient searches.

**Understanding Indexes:**

**Definition:** An Azure AI Search index serves as a container for searchable documents, analogous to a table in a database where each row represents a document.

**Index Schema:**

* **Description:** Defines the structure of data within the index, specifying fields and their data types.
* **Example Schema:**

json

Copy code

{

"name": "example-index",

"fields": [

{ "name": "keyphrases", "type": "Collection(Edm.String)" },

{ "name": "imageTags", "type": "Collection(Edm.String)" }

]

}

**Index Attributes:**

* **Purpose:** Specifies behaviors and properties for each field, such as whether it's searchable or sortable.
* **Efficiency:** Optimal indexes utilize only necessary behaviors to minimize resource consumption.
* **Considerations:** Failure to set required behaviors may necessitate index rebuilding.

**Using an Indexer to Build an Index:**

**Overview:**

* **Purpose:** Export documents to JSON and load them into an index for search functionality in Azure AI Search.
* **Approaches:**
  + **Push Method:** Flexibly pushes JSON data into the search index via REST API or .NET SDK.
  + **Pull Method:** Utilizes search service indexers to extract data from Azure data sources, converting it to JSON if needed.

**Pull Method with Indexer:**

* **Definition:** Indexer acts as a crawler, extracting searchable text and metadata from external Azure data sources and populating the search index.
* **Pull Model:** Service pulls data into the index without requiring manual data addition code.
* **Field Mapping:** Indexer maps source fields to their corresponding fields in the index.

**Data Import Monitoring:**

* **Dashboard:** Search service overview page provides insights into the health of the search service, displaying document counts, index usage, and storage usage.
* **Indexer Progress:** Monitor progress by checking the associated indexer, observing document count growth.
* **Search Explorer:** Verify results and ensure expected search outcomes using the Search explorer tool.

**Making Changes to Index:**

* **Field Definitions:** Dropping and recreating indexes necessary for field definition changes, with support for adding new fields.
* **Iterative Design:** Code-based approach preferred for iterating designs due to manual schema filling in the portal.
* **Update Approach:** Create a new index with changes under a different name, utilizing the same indexer and data source, then switch the application to use the new index after importing data.

**Persisting Enriched Data in a Knowledge Store:**

**Overview:**

* **Definition:** A knowledge store serves as persistent storage for enriched content generated through AI enrichment.
* **Purpose:** Store data resulting from AI skillsets, such as image captions, for further analysis or processing.
* **Outputs of Skillsets:** Skillsets transform documents through enrichments like entity recognition or text translation, producing either a search index or projections in a knowledge store.
* **Mutually Exclusive Outputs:** While derived from the same inputs, search indexes and knowledge stores have different structures, storage methods, and applications.

**Utilizing Azure AI Search:**

* **Focus:** While Azure AI Search typically emphasizes creating searchable indexes, it offers data extraction and enrichment capabilities for persisting data in a knowledge store.
* **Advantages:** Enriched data in a knowledge store can be leveraged for in-depth analysis, visualization, or downstream processing.

**Types of Projections in a Knowledge Store:**

* **Table Projections:** Structured data stored in a relational schema, facilitating querying and visualization.
* **Object Projections:** JSON documents representing individual data entities, offering flexibility in data representation.
* **File Projections:** Used for storing extracted images in JPG format, enabling easy access and manipulation.

To create an index in the Azure portal:

1. **Prepare Your Data:**
   * Ensure your data is available in a supported data source, such as Cosmos DB, Azure SQL, or Azure Storage.
2. **Access the Import Data Wizard:**
   * Navigate to the Azure portal and locate your Azure AI Search resource.
   * Open the Import data wizard from the Azure portal, which automates the process of creating necessary objects for the search engine.
3. **Create Necessary Objects:**
   * Use the Import data wizard to create the following objects:
     + **Data Source:** Stores connection information to the source data, including credentials. Exclusive use with indexers.
     + **Index:** Physical data structure for full-text search and other queries.
     + **Indexer:** Configuration specifying a data source, target index, optional AI skillset, schedule, and error handling settings.
     + **Skillset:** Set of instructions for content manipulation, including analysis and extraction from image files.
     + **Knowledge Store:** Stores output from an AI enrichment pipeline in tables and blobs in Azure Storage.
4. **Create Azure AI Search Resource:**
   * If you haven't already, create an Azure AI Search resource in the Azure portal.
   * Manage components of your service from the resource Overview page in the portal.
5. **Building Indexes:**
   * Utilize the Azure portal to build Azure search indexes or programmatically using REST API or SDKs.

* **Query Design:**
  + Index and query design are interrelated, where the index schema dictates the supported queries.
* **Query Requests:**
  + Submit queries via HTTP or REST API requests, receiving responses in JSON format.
  + Queries define searched and returned fields, search result shape, and filtering or sorting criteria.
* **Query Syntax:**
  + Two types: Simple and Full Lucene.
  + Simple Syntax: Common scenarios, default for basic search needs.
  + Full Lucene Syntax: Advanced capabilities for complex scenarios.
* **Simple Query Requests:**
  + Consist of search terms, operators, and verbatim phrases.
  + Example: coffee (-"busy" + "wifi") searches for coffee, excluding "busy" and including "wifi".
* **Execution of Simple Query Syntax:**
  + Matches any of the terms by default.
  + Verbatim phrases match exact occurrences specified.
  + Operators like "-" exclude terms, "+" includes terms.
  + Parentheses group terms, setting precedence.